

Konvolúciós megoldások hangtérszimulációs feladatokhoz

Convolution: methods for soundfield rendering

WERSÉNYI GYÖRGY

Széchenyi István Egyetem, Távközlési Tanszék, 9026, Győr, Egyetem tér 1.

Absztrakt:

A hangtérszimulációs eljárások egyik legfontosabb része, amikor a nyers hanganyagot a szükséges térinformációval „ellátjuk”. Ennek során manapság digitális jelfeldolgozási eljárásokkal hozunk létre irányinformációt, térérzetet, utószó (auralizáció). A konvolúciót megvalósíthatjuk a frekvenciatartományban és a mai számítástechnika segítségével az időtartományban is. Ebben a cikkben ismeretterjesztő jelleggel néhány lehetőséget mutatunk be: ismert hangfeldolgozó- és teremakusztikai programokat, másrészt MATLAB alatt megvalósítható és futtatható környezetben is létező megoldást. Továbbá bemutatunk egy binaurális felvevőrendszert, amely a fenti algoritmusokat kiszolgálhatja a hangtérszimulációs feladatok során.

Abstract:

The most important part of the sound field reproduction is the simulation of spatial and directional information (auralization). Digital signal processing methods help us to add this directional information, the feeling of room size or reverberation. This can be realized in the frequency domain and using state-of-the-art computer technique, also in the time domain. This paper presents some methods and software applications to render fast convolution of input data and impulse response. Some commercially available software solutions and a MATLAB application will be introduced, as well as a binaural recording system that can assist such sound field simulations.

1. Bevezetés

Hangtér-szimulációnak nevezünk minden olyan eljárást, megoldást, amikor hangmintákat „ruházunk fel” olyan információval, ami eredetileg nem volt része. Ebbe a tág értelmezésbe sok minden belefér, de elsősorban teremakusztikában és irányhallás vizsgálatokban találkozunk vele. Teremakusztikában nyers, utőzengéssel nem rendelkező stúdiófelvételeket használunk bemenő jelként. Ezt követően a terem valamilyen rendszerleíró függvényét felhasználva szűrünk, és a végtermék úgy fog szólni, mintha abban az adott teremben, a felvétel helyén hallgatnánk. Ezt nevezzük auralizációnak is. Ennek során a lejátszás körülményei is fontosak. Az előálló időfüggvény tartalmazza a bemenő jel és a környezet hatását, de még mentes a lejátszórendszer tulajdonságaitól, a hangsugárzók vagy éppen a lehallgatási szoba paramétereitől, melyeket esetleg utólag figyelembe kell venni.

Írány- és térhallás vizsgálatoknál jórészt fejhallgatót használunk, és a fenti bemenő jeleket ez esetben az emberi fül átviteli függvényeivel (HRTF) szűrjük [1-3]. Ilyenkor nem csupán a függvényeket kell a szűrés során bevetni, hanem a fejhallgató kiegyenlítést is meg kell valósítani [4, 5]. Ezek a folyamatok egyben is leírhatók, illetve egymás után kapcsolt szűréssel valósíthatók meg.

A szűrésről rögtön a frekvenciatartomány és a szorzás egyszerű művelete jut eszünkbe. Villamosságtani ismereteinket elővéve emlékezhetünk, hogy a feladat megoldása létezik az időtartományban is, melyet konvolúciónak nevezünk. A probléma forrása az, hogy a konvolúciós integrál (illetve digitális esetben a diszkrét szummázás) megoldása bonyolult és nagyon időigényes. A számításigénye ennek lényegesen meghaladja a szorzás műveletét, így korábban a számítógépeknek más lehetőségük nem volt, mint a bemenő jel Fourier-transzformáltjának és a rendszer (terem) átviteli függvényének szorzása, majd az eredmény visszatranszformálása. Erre kettő hatványának megfelelő DFT módszert ill. annak gyors implementációit (FFT algoritmusokat) használtunk, a szorzás pedig komplex módon valósult meg.

A mai számítástechnika már lehetővé teszi az időtartománybeli digitális konvolúciót. Ennek számítási igénye és ideje még mindig meghaladja a frekvencia tartománybeli módszereket, de a valós idejű megoldás megcélozható. A továbbiakban egy rövid áttekintő után bemutatunk néhány ismert vagy kevésbé elterjedt szoftveralkalmazást a feladat megoldására, valamint egy saját fejlesztésű MATLAB implementációt, mely a végén bemutatott binaurális mérőrendszerhez készült, de alkalmas más forrásból származó wave fájlok kezelésére is.

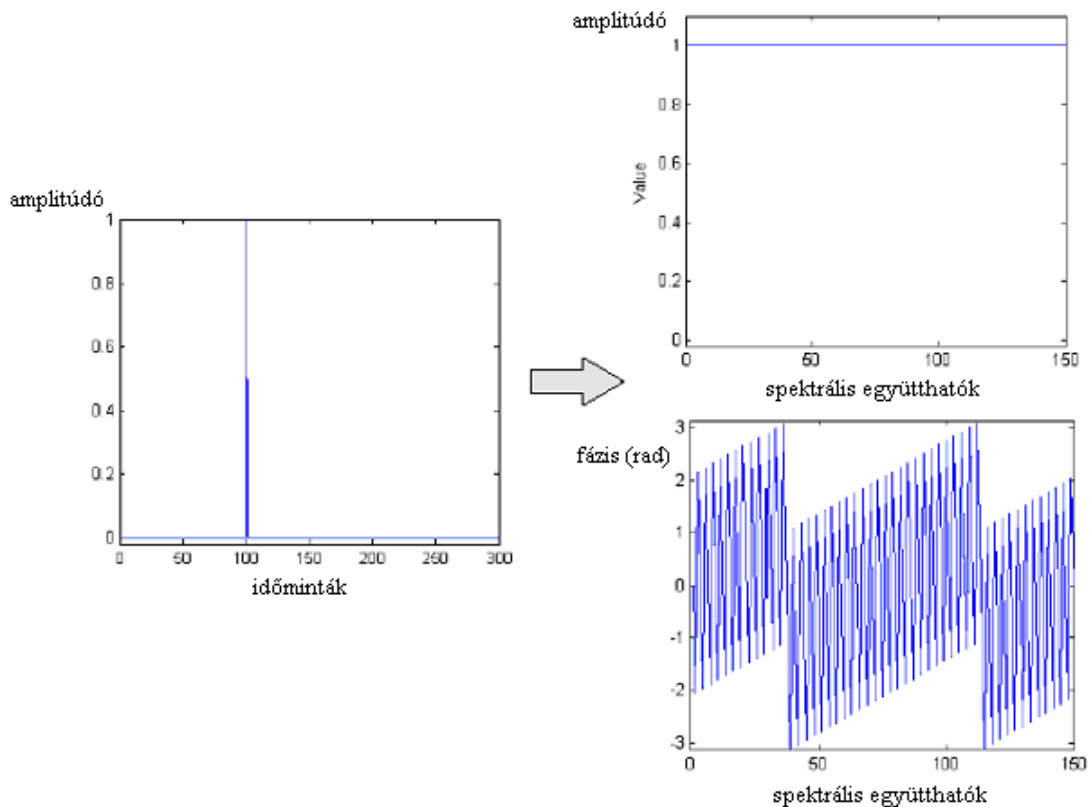
2. A rendszerleíró függvények

A frekvenciatartományban dolgozva még mindig kisebb számítási idő szükséges a kimeneti hangfájl előállításához, mint az időtartományban. A későbbiekben bemutatott MATLAB-os program bizonyította, hogy a konvolúciós algoritmusok futási ideje még mindig többszöröse a frekvenciatartománybeli szorzásos megoldásoknak.

A szorzást a transzformáció után komplex módon kell elvégezni. Valós időfüggvény transzformáltja komplex spektrumot ad: az amplitúdó karakterisztika az abszolút értéket jelenti, míg a fázisspektrum az időbeli eltolódások információját hordozza. Bizonyos programok a valós-képzetes leírást használják (C++ algoritmusok, MATLAB beépített függvényei), számunkra szerencsésebb az ebből könnyedén kiszámítható abszolút érték-fázis leírási mód. Hasonlóan, a rendszer átviteli függvénye is komplex (bárhogy is határoztuk azt meg), a végeredmény pedig a válaszjel komplex spektruma lesz. Ezt visszatranszformálva valós időfüggvényt kapunk, helyes időzítésekkel.

Felmerül a kérdés, mi a helyzet akkor, ha pld. a rendszer átviteli karakterisztikája valós, csak az amplitúdó karakterisztikával rendelkezünk. Valós spektrumnak a transzformáltja komplex időt adna, melynek értelmezése nem lehetséges. Ilyen esettel akkor találkozhatunk, ha egy adott rendszerről készült mérés nem teljes és/vagy a fázisinformáció – bár nem azonosan nulla a valóságban – hiányzik. Ilyen esetben megoldás lehet, ha a bemenő jelet sem komplex alakban használjuk, hanem csak az amplitúdókkal dolgozunk és visszatranszformálás után mesterségesen adjuk hozzá az időbeni eltolást a jelekhez. Sajnos, ez a módszer nem mindig lehetséges, csak akkor, ha a futásidőknek nincs nagy jelentősége (pld. azonos távolságú hangforrások esetén), vagy ha ismerjük a felvétel helyén a távolságokat. Ez esetben is csak közelítőleg tudjuk utólag eltolni az időfüggvényeket a becsült hangsebesség ismeretében. Egy másik lehetséges eljárás a fázisspektrum mesterséges feltöltése a vissz irányú transzformáció előtt, pld. lineáris fázismenettel. Ilyen gondok akkor lépnek fel, ha véletlen fázisú, pld. fehérzajú méréseket végeztünk a rendszeren. Ekkor a fázisinformáció felhasználása erősen kérdéses [6].

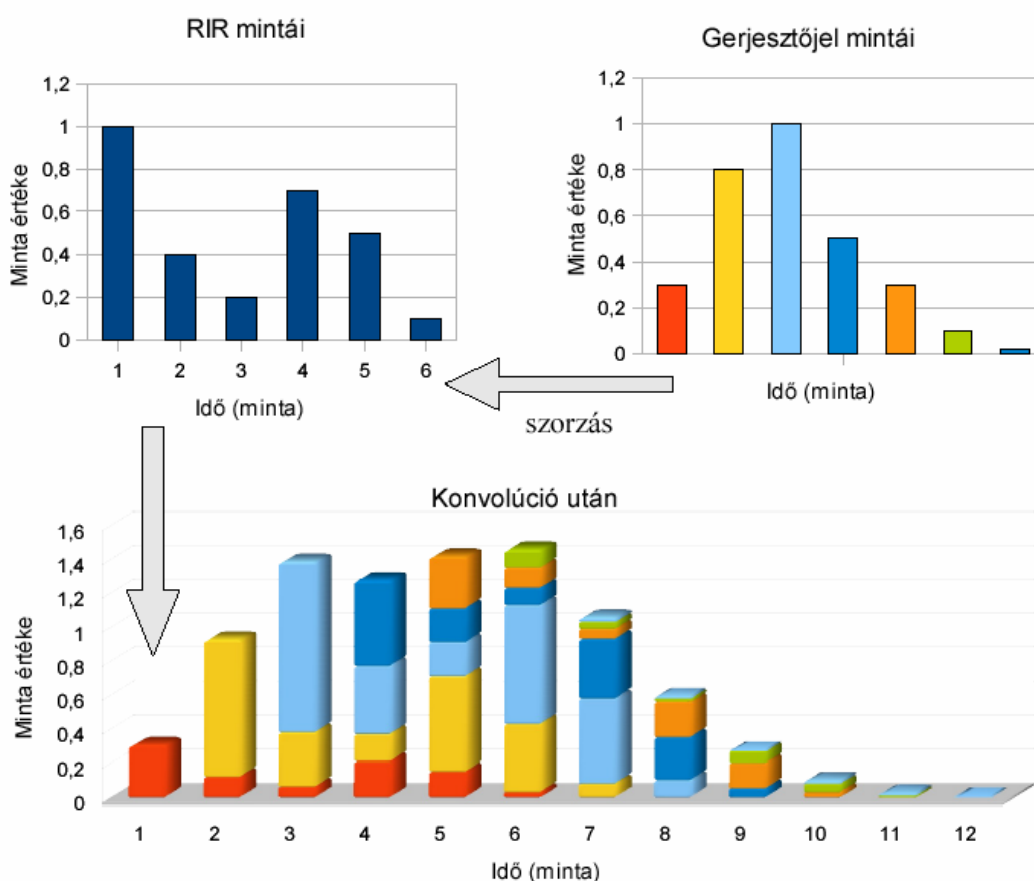
Ezeket a gondokat jól orvosolja – de cserében nagy számítási igénnyel fizetünk – az időtartománybeli megoldás. Az időfüggvények ugyanis közvetlenül, egyszerű és gyors méréssel előállnak. Az impulzusválasz egyben hordozza az amplitúdó- és késleltetés viszonyokat is, mindezt szemmel is jól látható módon. (1. ábra). A feladat, hogy a bemenő jel időfüggvényének mintáit a rendszer impulzusválaszának mintáival egy megfelelő konvolúciós algoritmussal „társítsuk”, és máris kész a kimeneti hangfájl. Egy szóval: wave fájlokat kell konvolválni.



1. ábra. MATLAB által előállított Dirac-impulzus az időtartományban (balra) és az abból számított amplitúdó és fázisspektrum.

Fig. 1. Time domain (left) and frequency domain representation of a Dirac impulse created by MATLAB.

A konvolúció tulajdonképpen két jel egymásra hatása. Az impulzusválasz elméletileg a Dirac-impulzusra adott válasz, amely teljes egészében hordozza a lineáris rendszerünk átviteli tulajdonságait. Ennek Fourier-transzformáltja a (komplex) átviteli karakterisztika ill. Laplace világban ez az átviteli függvény [7]. Ha a rendszer végtelen sávszélességű, mint a Dirac-impulzus, akkor a kimenő jele is torzítatlan Dirac-impulzus lesz, valamekkora késleltetési idővel. A valóságban a véges átviteli sávszélesség a Dirac-impulzust eltorzítja, elkeni, szélesíti. Továbbá – az akusztika világában maradván – ez az impulzus a visszaverődésekkel átlapolódik: megjelennek az eredeti jel időben eltolts és csillapított változatai is. Ez az impulzusválasz aztán súlyozza a bemenő jeleket úgy, hogy azok nem „egy pillanatban” hanem több lépésen keresztül is kifejti hatásukat. Minél hosszabb az impulzusválasz, annál tovább lesz hatása a bemenő jelre. A kimenő jel lineáris konvolúció esetén időben hosszabb lesz (nagyobb méretű wave fájl eredményez), mint a bemenő jel vagy az impulzusválasz. A rendszer egy adott pillanatban kifejtett hatása nem csak az adott pillanatban beérkező bemeneti jelkomponensre lesz hatással, hanem az utána következőkre is (2. ábra). Ennek időtartamát éppen az impulzusválasz időbeni hossza mutatja meg számunkra. Ha pedig a bemeneti jelnek vége, a kimeneti jel nem szűnik meg azonnal, hanem még lecseng. A valóságban, ha hangtechnikáról beszélünk, ez ténylegesen utózengésként jelentkezhet.



2. ábra: Diszkrét konvolúció: a rendszer linearitása és időinvarianciája miatt a válaszok a gerjesztések megfelelő együtthatóival szorozódnak és időben eltolódnak, majd összegződnek [8].

Fig. 2. Representation of the digital convolution. The input samples are multiplied, delayed and summed using the samples of the impulse response [8].

Lineáris rendszerben a szuperpozíció fennáll: a kimeneten a válaszfüggvény az egyes impulzusokra adott válaszok összegével egyenlő. A konvolúció egyenlete:

$$x_{ki}(t) \approx \sum_{i=1}^t w(t - \tau_i) x_{be}(\tau_i) \Delta \tau \quad (1)$$

Határértékben, ha $\Delta \tau \rightarrow 0$, integrálegyenlettel számolunk. Legtöbbször 0-tól t -ig, véges tartományon belül számolunk, hiszen a jelünk valamikor bekapcsol és a vizsgálatot valamikor el kell kezdenünk, a rendszer továbbá kauzális:

$$f * g = \int_0^t f(t - \tau) g(\tau) d(\tau). \quad (2)$$

A t - τ kifejezést gyakran új változóra cserélik, így alakítva ki a konvolúciós integrál alakját. Digitális esetben ismét a szummázás kerül előtérbe, ahogy a 3. ábra is mutatja. A konvolúciós tétel mondja ki számunkra, hogy az időtartománybeli művelet a frekvenciatartományban (Fourier vagy Laplace) szorzássá egyszerűsödik. Ez egyébként visszafelé is igaz: az időtartománybeli szorzás spektrális konvolúciót eredményez. Ilyesmi gyakran előfordul ablakoló függvények esetén, vagy éppen burkológörbe módosításoknál (amplitúdómodulációk). Akárhányszor átfarmáljuk egy hanghullám burkológörbét, egyben konvolváljuk a burkológörbe spektrumát az újraformált hang burkológörbéjének spektrumával. Diszkrét esetben adott két jelsorozat: $a=a[n]$ és $b=b[n]$. A sorozatok hossza rendre N és M . Ekkor:

$$a[n] * b[n] = kimenet[n] = \sum_{m=0}^N a[m] b[n - m]. \quad (3)$$

Valójában az a minden mintája a b egy-egy eltolt másolatának súlyozó tényezője lesz. A keletkező súlyozott minták mind összeadódnak, és így áll elő az eredmény. Ez látható a 3. ábrán. A szokásos módja a megoldásnak az úgynevezett direkt konvolúció, azaz minden n -re kiszámítják a szumma értékét. Az eredményt általában normalizálják a végén. A konvolúció kommutatív, jellemzően egy viszonylag hosszú mintasorozatot konvolválunk egy meglehetősen rövid sorozattal. A konvolúció után előálló hangfájl hangerőssége általában nő, túlvezérlés előfordulhat, így előzetes dinamika beállításokkal ezt szabályoznunk kell. A konvolúció FIR szűrő segítségével megvalósítható.

Sok megoldás létezik már a frekvencia tartománybeli algoritmusok optimalizálására, gyorsítására. A leggyakoribb ötlet, hogy az impulzusválaszt kisebb részekre osztjuk fel, és ezt a szűrőkészletet vetjük be a bemenő jel szűrésekor. Az így részleteiben megszárt bemenet kimeneti összegzése azt az eredményt adja, mintha egyben végeznék a műveletet [9-12]. Ez a módszer spórolhat az idővel, a memóriával, sőt csökkentheti kvantálási zajt, input-output késleltetéseket [13, 14]. Érdekes, hogy a particionálás során a számítási igény (idő) áttevődik az FFT algoritmusról a szorzás műveletére. Minél több részre osztjuk fel az impulzusválaszt, annál több szorzás és annál kevesebb FFT történik. Mivel az FFT rövidül, egyre kevesebb problémával találkozunk, mely a hosszú FFT esetén előfordulhat, legfőképpen annak optimalizálási nehézségei. A minimális késleltetés érdekében érdemes a particionálást egyre hosszabb blokkokra felosztva végezni.

3. Konvolúciós megoldások

3.1 Hagyományos, korai zengetők

A konvolúciós megoldások elnevezést általánosan használva, beleérthetjük a korai zengetőket, melyek valójában nem konvolúciót valósítanak meg, hanem analóg módon, mechanikai eszközökkel dúsítják a bemeneti hangot [15, 16]. Ezek a hagyományos zengetők visszacsatolással működnek, mely különösen hosszabb impulzusválaszok esetén számít. Céljuk nem elsősorban a terem valóságghű akusztikai reprodukciója, sokkal inkább stúdiótechnikai alkalmazás. Az első ilyen alkalmazások mechanikai elvűek voltak, mint pld. a rugós zengető. Ebben megtalálható egy (dinamikus) bemeneti elektromechanikai átalakító, mely egy lágy gumi felfüggesztéshez kapcsolódik. Különböző mechanikai keménységű fémrugókon át a kimeneten található piezzo átalakítóhoz csatolva érhető el a zengetés. A rugókon végigfutó rezgések egymással csatolásban hozzák létre a kívánt hatást. Zenéhez ez megfelelő is lehet, de csak a szint szabályozására van lehetőség, így alkalmazása korlátozott. Hasonló elven működik a lemezes zengető, ahol egy rugalmasan felfüggesztett és mozgatható csillapítással ellátott lemezen találhatók az átalakítók. Ezek a mechanikailag nagy, kevés szabályozás és beállítási lehetőségekkel rendelkező eszközök a modern teremakusztika számára nem használhatók.

3.2 Visszacsatolt elektronikus zengetők

A modernebb hardveres zengetők visszacsatolt késleltetők szövedékéből állnak. A Schröder-féle zengető előre és hátra irányba is tartalmaz csatolást, késleltetők és műveleti erősítők összekapcsolásával, hangzásszínező és színezetnélküli ismétlődéseket is előállítva. A Sony korai zengetője hasonló építőelemekből építkezve külön állítja elő az első visszaverődéseket, majd a hangzásszínező ismétlődéseket, végül ezeket aperiodikusan sűrítve éri el a hatást. Ezek a megoldások már alkalmasak lehetnek teremérzet előállítására, széles határok között állíthatók a paraméterek, de nagy gyakorlatot és sok időt igényel használatuk. A számítógépes technika fejlődése lehetővé tette először az offline szoftveres megoldások alkalmazását, majd a valós idejű frekvenciatartományi implementációt. A hangsúly áttevődött a megfelelően optimalizált és így felgyorsított FFT algoritmusok és műveletek irányába. A gépek gyorsulásával előtérbe került az időtartománybeli megvalósítás is, így a mért impulzusválasz és a bemeneti jel közvetlen konvolúciója megoldhatónak tűnt. Megfigyeléseink szerint (MATLAB-os implementáció), ez még mindig nagyságrendileg elmarad a frekvenciatartománybeli megoldás(ok)hoz képest, és amennyiben a valós idejű megoldás elvárás, még mindig nem veszi fel a versenyt a szokványos vagy különösen a particionált konvolúciós megoldásokkal.

A létező megoldásokat sokszor szabadalom védi. Ez vonatkozik a hardveres megoldásokra, processzorokra, de a mai világban a szoftveres megoldásokra, plug-in-ekre, algoritmusokra is. Az ilyen eljárásokat gyakran cégek liszenszelik és megvásárolt programcsomagokban kapjuk meg (Steinberg, Lake RT, Pro Tools stb.).

3.3 Szoftveres megoldások

A nagy kérdés, honnan szerezzünk ehhez a feladathoz szoftvert? Ebben a fejezetben erre adunk ötleteket. Először is, fizethetünk érte nagyon sok pénzt. A piacon több ezer dolláros értékrendben találhatóak erre programcsomagok. Ilyen pld. az *Altiverb* és a *Waves IR-1* [17,

18]. Ez sajátos fájlformátumot használ az impulzusválaszok tárolására, de külön pénzért az internetről is rendelhető, letölthető a világ rengeteg jó akusztikájú termék, stadionjának rögzített impulzusválasza. Az *Altiverb* saját, optimalizált eljárást használ a konvolúcióhoz. Mindkét programhoz találhatunk ingyenesen is elérhető mintafájlokat [19]. Kedvünkre konvolválhatjuk ezekkel bementi fájljainkat.

Fizetős szoftver a *WizooVerb* programcsalád professzionális teremakusztikai, zengető szoftver, elsősorban sokcsatornás (5.1 alkalmazásokhoz) [20]. A program kezeli a saját formátumú nagyfelbontású HDIR (High-Definition Impulse Response) impulzusválasz fájlokat és engedi más programok által rögzített, más formátumú impulzusválaszok importálását. Képes az impulzusválaszok előtti csend és akár a direkt hullám kiszűrésére is. Ezen túl algoritmikus zengetésre is van lehetőség, ahol parametrikusan állíthatjuk a terem méretét, tulajdonságait stb.

A *VibeStudio Designer* kimondottan „3D audio gyorsítónak” hirdetve magát próbál a piacon érvényesülni [21]. Az angolban olyan elterjedt 'rendering' kifejezést meglovagolva dinamikus, valószerű zengetést és szűrést kínál meglehetősen borsos áron. Lehetőség van nagy számú hangforrás elhelyezésére, teremmodellezésre, felületek anyaggal való bevonására. Érdekessége, hogy a hangutak számítása és megjelenítése is beépített funkció, 2.1-7.1 között rögzíthető sokcsatornás kimenet és beépített valamint rögzített HRTF adatbázis felhasználva binaurális felvételek előállítás is lehetséges.

A *Trillium Lane Space* család és alkalmazásai elsősorban a profi világban használatosak, pld. Pro Tools használóknak, de létezik platformfüggetlen verzió is, meglehetősen drága megoldás [22].

Az olcsóbb, gazdaságosabb megoldások közül tallózva, mindjárt érdemes megemlíteni a legelterjedtebb wave editor alkalmazást. A korábban a Syntrillium Software által gyártott és forgalmazott *CoolEdit* wave editor program és annak utódja is rendelkezik beépített funkcióval. Az Adobe cég felvásárolta, majd a Premiere-programcsalád részévé tette ennek utódját az *Adobe Audition* programot (jelenleg a 3.0 verzió érhető el). Érdemes megjegyezni itt, hogy a wave fájlok formátuma szabványos kell legyen. Az Audition a mentéskor felajánlja a „save non-audio extra information” elmentését is, melyet a wave fájl fejlécébe ment bele, és ez még a saját konvolúciós algoritmusának sem felel meg. Az Effects menü/Special/Convolution alkalmazást megnyitva készíthetjük el kimeneti fájljainkat. Léteznek már beállított saját impulzusválaszok is, de magunk is előállíthatjuk felvett wave fájlból, amennyiben azt előtte a munkaablakban kijelöltük. Az Audition több beépített funkciót is tartalmaz, mint a zengetés, előre beépített termék függvényei, késleltetések, echo. Ezek részletes tárgyalását itt kihagyjuk.

A *Voxengo Pristine Space* sztereótól akár 7.1-ig képes zengetésre, a szokványos piaci árú programoknál jóval olcsóbban [23].

A *Giga Pulse VST* modul is a gazdaságosabb megoldások közé tartozik [24]. Különlegessége a mikrofonmodellezésnek nevezett eljárás. Lehetővé teszi különféle mikrofonok virtuális kiválasztását, elsősorban a beépített adatbázisból. Ezek iránykarakterisztikáját módosíthatjuk is, amelyet a valóságban csak a mikrofon elmozdításával érhetünk el. A teremben való elhelyezés és kísérletezés így megoldott.

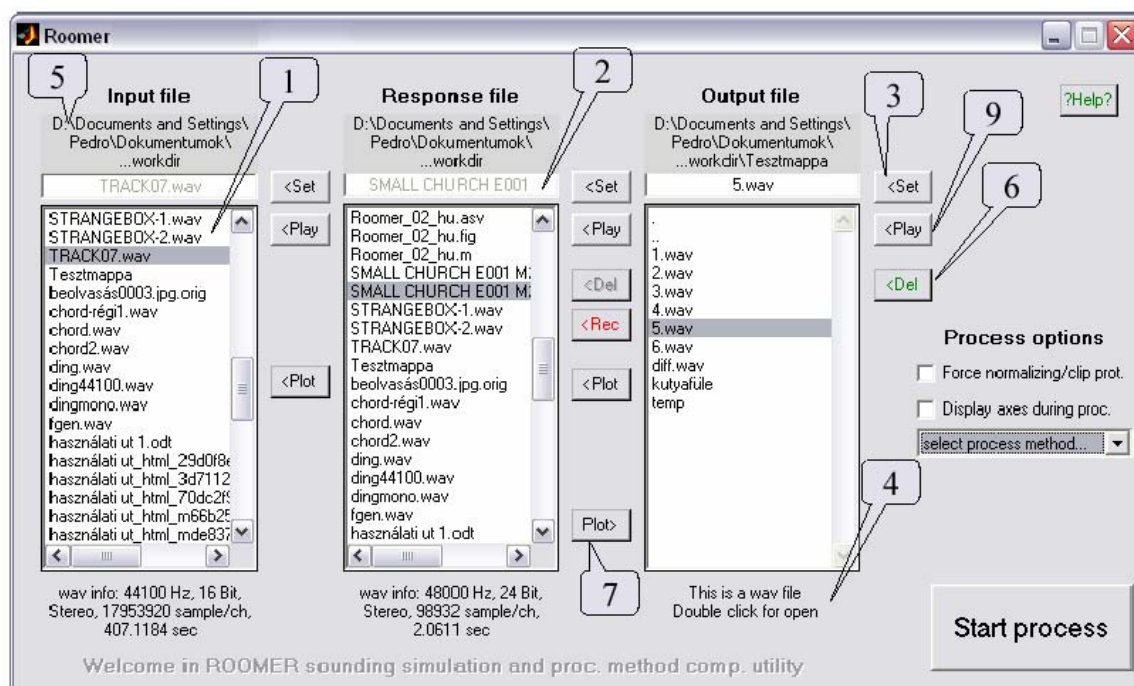
Létezik ingyenes, egyszerűbb tervező program (mint pld. a COPE szoftverek [25]). Ennek hátránya, hogy korlátozott a wave fájlok hossza, melyet használhatunk, de bemutató jelleggel és próbákhoz megfelelő lehet.

A *BruteFIR* szoftver segítségével hosszú FIR szűrést (10000 lépésnél hosszabb) tudunk megvalósítani valós időben is, elsősorban sokcsatornás hangtechnika számára [26]. Parancssorból vezérelhető és nagy előnye, hogy moduláris felépítésű, nyílt forráskódú, elsődleges platformja a Linux. A szűrő algoritmus valós idejű FFT, de a program „csak” egy FIR szűrőmotor, mely nem tartalmaz együtthatókat és nem is nyújt szűrőtervező alkalmazást.

A viszonylag hosszú input-output (hangkártyakezelés) késleltetés miatt nem a legjobb megoldás direkt alkalmazásokban.

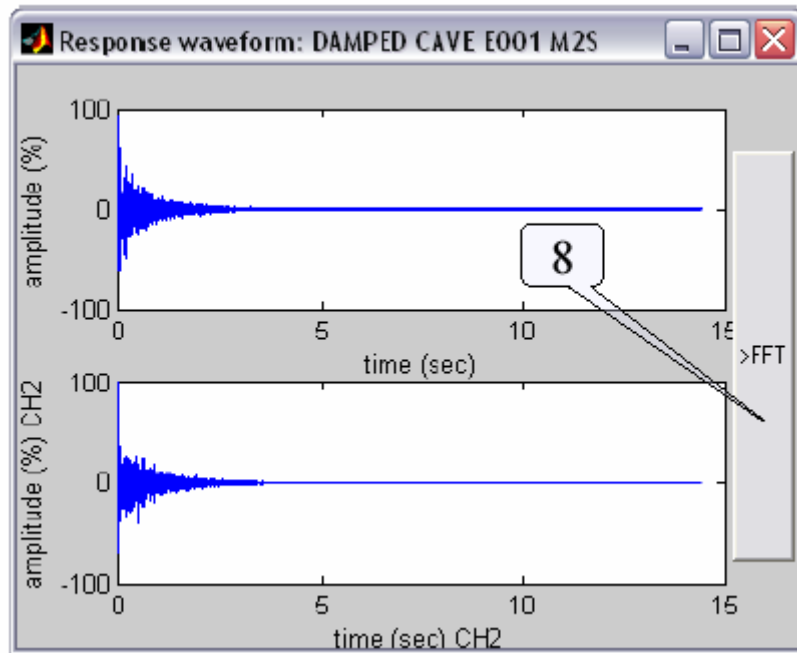
Az interneten a programozók találhatnak kész C++, Basic és egyéb forráskódokat konvolúció megvalósítására, melyeket külső függvényhívással, tömbkezeléssel használhatnak saját C programjukban. Ezek hátránya, hogy nehezen fellelhetők, kérdéses működésük, általában fix tömböket használnak és közvetlenül wave fájlokat nem használhatunk. Ilyen esetben a wave fájlok fejlécét le kell választani, és a mintákat külön tömbben tárolni a további felhasználáshoz. Ehhez a wave fájlok struktúrájának ismerete és külön programok megírása is szükséges.

A további megoldások egyike a MATLAB programcsomag használata. Ez nem kimondottan zengető, beépített függvényeivel és programozói felületével azonban egyénre és a feladathoz szabható, néha többet is nyújthat társainál. Binaurális szimulációknál, HRTF függvények felhasználáshoz, fejhallgató-kiegyenlítéshez, időfüggvények ablakolásához nagyon gyakran használják. A szabványos, egy- és kétsatornás wave fájlokat kezeli. A MATLAB-nak létezik ide vonatkozó alapfüggvénye (a *conv* utasítás), de rengeteg extrával lehet ezt „felturbózni”. A particionált konvolúció implementációja nem megoldott, a beépített FFT műveletek tesztjeinkben gyorsabbak voltak, mint az időtartománybeli megoldás. A szoftver természetesen könnyen, gyorsan meg is jeleníti nekünk a wave fájlokat és azok spektrumát. Nem csak lejátszani, de felvételt készíteni is lehet, megfelelő lappal és mikrofonokkal felszerelve a MATLAB-on belül rögzíthetjük egy terem impulzusválaszát. A 3. ábra diplomamunka keretében készült konvolváló program grafikus felületét mutatja. Balra a bementi fájl tölthető be, középen az impulzusválasz wave fájlja. A kimenetet szintén wave formátumban menthetjük el. Lehetőség van mindhárom időfüggvény és annak spektrumának is megjelenítésére (4. ábra). A REC gombbal rögzíthetünk felvételeket, a PLAY gombbal lejátszhatjuk azokat. Összesen három módszer (két időtartományi konvolúció és egy FFT utáni szorzómódszer) választható és hasonlítható össze. A program képes két különböző módszer után előálló wave fájlok összehasonlítására is, valamint azok normalizálására (amplitúdó).



3. ábra. A Roomer nevű MATLAB program grafikus kezelőfelülete.

Fig. 3. GUI of a MATLAB application for convolution.



4. ábra. A *plot* funkció használata kétsatornás impulzusválasz esetén. Az FFT gombra kattintva megtekinthetjük a komplex spektrumot is.

Fig. 4. The *plot* function of MATLAB showing the time function of a two-channel impulse response.

Az auralizáció során alapvető kérdés, hogy az ehhez szükséges konvolúció valós idejű-e ill. előre elkészített (offline); valamint, hogy ennek megvalósítása hardveres vagy szoftveres módon történik-e. A feladatot tehát elsősorban a számításigény és a valós idejűség szempontjából vizsgálva lényegében arra keressük a választ, melyik a jobb megoldás: időtartománybeli konvolúció az impulzusválasz segítségével, vagy a frekvenciatartománybeli szorzás(ok) megvalósítása? A számításigény ebben az értelemben rokon fogalom a számításához szükséges idővel, ugyanakkor ezek objektív, pontos összehasonlítása különböző programok esetén igencsak nehézkes, gyakran nincs is rá lehetőség vagy éppen lényegtelen az eltérés. Nagyságrendileg azonban érdemes vizsgálatot végezni.

Az általunk készített MATLAB GUI program tesztfuttatásai során megállapítottuk, hogy ugyanazon gépen futtatva egy 10 mp hosszú bementi sztereó hangfájlt és a vele társítandó 2 mp-es impulzusválasz fájlt, a beépített FFT-n alapuló szorozómódszer a valós időt megközelítve, néha annál gyorsabban is teljesítette a műveletet. Ugyanakkor az időtartománybeli konvolúció akár beépített, akár saját algoritmust használva akár két nagyságrenddel lassabban dolgozott, így a valós idejű megoldásnak nem felel meg (a felhasználási területeink jó része offline). Összehasonlítva a három módszer kimeneti időfüggvényét (mintánként kivonva egymásból) szinte teljes egyezést kapunk, így semmi nem szól jelenleg az időtartomány megoldás mellett.

3.4 A futtatási környezet

A futtatási környezetet is számításba kell vennünk, ha valóban gyors alkalmazásokra van szükségünk. Többségünk Windows-t használ, esetleg Linux-ot, érdemes lehet mélyebbre ásni az operációs rendszer meghajtói és hangkezelési módszeribe. Felhívjuk a figyelmet a VST

betűszóra, amely a Virtual Studio Technology rövidítése. Ez a Steinberg cég által kifejlesztett interfész, mely egyesíti az audio szintézert, az effekteket az audio editorokkal, elsősorban merevlemezalapú munkaállomások számára. Sok fizetős és szabad letöltésű plug-in érhető el, melyek ezt a motort használják. Ezek a kiegészítők beépülnek az editor alkalmazásba, kiegészítő funkcióval látva el azt. Ezek leggyakrabban hangszerek, hangeszközök (MIDI szemplerek) vagy effektek (tipikusan szűrők, zengetők), melyeket hardveres eszközök nélkül érhetünk el az emuláció során. Megfelelő hardver és meghajtó esetén valós idejű alkalmazásokra is lehetőség van, de ennek gyakran előfeltétele az ASIO (Audio Stream Input/Output) driver. Utóbbi kiiktatja a Windows lassú hangfeldolgozó eszközét (Direct Sound, Windows Driver Model), a szoftvert közvetben köti össze a hangkártyával, így kisebb késleltetéssel dolgozhatunk. Ez különösen sokcsatornás hangfeldolgozásban és valós-idejű alkalmazásoknál lehet fontos. Az ASIO driver is a Steinberg cég szabadalma.

(A cikk folytatása következik)

Irodalom

- [1] J. Blauert, *Spatial Hearing*. The MIT Press, MA, 1983.
- [2] C. I. Cheng, G. H. Wakefield, *Introduction to Head-Related Transfer Functions (HRTFs): Representations of HRTFs in Time, Frequency, and Space*. J. Audio Eng. Soc., vol. 49, 2001, pp. 231-249.
- [3] H. Møller, D. Hammershøi, C. B. Jensen, M. F. Sørensen, *Transfer Characteristics of Headphones Measured on Human Ears*. J. Audio Eng. Soc., vol. 43(4), 1995, pp. 203-216.
- [4] H. Møller, *Fundamentals of binaural technology*. Applied Acoustics, vol. 36, 1992, pp. 171-218.
- [5] Gy. Wersényi, *Virtuális hangtérszimuláció és a binaurális technológia*. Híradástechnika, Vol. LXII, Nr.2, 2007 február. pp 25-32.
- [6] P. Berényi, Gy. Wersényi, *A külső fül fejre vonatkoztatott átviteli függvényeinek vizsgálata*. Akusztikai Szemle, IV.évf., 1.-4., 1999, Budapest. pp. 35-41.
- [7] http://e-oktat.pmmf.hu/jelek_ii_fej_1
- [8] Cs. Huszty, F. Augusztinovicz, *Virtuális akusztikai valóság és auralizáció*. 12. Televízió és Hangtechnikai konferencia és kiállítás, Budapest, 2007, pp. 1-24. http://vibac.hit.bme.hu/documents/84tv2007_hcs.pdf
- [9] T. G. Stockham Jr., *High-speed convolution and correlation*. AFIPS Proc. 1966 Spring Joint Computer Conf., vol. 28, Spartan Books, 1966, pp. 229-233.
- [10] J. W. Cooley, J. W. Tukey, *An Algorithm for the Machine Computation of the Complex Fourier Series*. Mathematics of Computation, vol. 19, April 1965, pp. 297-301.
- [11] P. C. W. Sommen, *Partitioned frequency domain adaptive filters*. Proc Asilomar Conf. Signals, Systems and Computers, 1989, pp. 676-681.
- [12] J. M. P. Borrallo, M. G. Otero, *On the implementation of a partitioned block frequency domain adaptive filter (PBFDAF) for long acoustic echo cancellation*. Elsevier Signal Processing, vol. 27(3), 1992, pp. 301-315.
- [13] J. S. Soo, K. K. Pang, *A new structure for block FIR adaptive digital filters*. Proc. IRECON, vol. 38, 1987, pp. 364-367.
- [14] J. S. Soo, K. K. Pang, *Multidelay block frequency adaptive filter*, IEEE Trans. Acoust. Speech Signal Process., vol. ASSP-38, No. 2, February 1990, pp. 373-376.
- [15] F. Takács, *Hangstúdiótechnika*, Műegyetemi kiadó, 2004.
- [16] M. Warstat, T. Görne T, *Studiotechnik – Hintergrund und Praxiswissen*. Elektro-Verlag, Aachen, 2001.
- [17] <http://waves.com/>
- [18] <http://www.audioease.com/>
- [19] http://www.cksde.com/p_6_250.htm
- [20] <http://www.musiciansfriend.com/product/Wizoo-Wizooverb-W5-5.1-Surround-Sound-Room-Emulator?sku=704568&src=3SOSWXXA>
- [21] www.vrsonic.com
- [22] <http://www.digidesign.com>
- [23] www.voxengo.com/pspace
- [24] www.gigapulse.co.jp
- [25] http://irc.nrc-cnrc.gc.ca/ie/cope/07_e.html