Kandidatarbete i Medieteknik, 30 hp
Vårtermin 2013

I can’t believe it’s not surround sound

En studie i realtidsgenererat binauralt ljud

Magnus Fredrikson
Pontus Svensson

Handledare: Pirjo Elovaara & Sebastian Hastrup
Examinator: Peter Ekdahl
Blekinge Tekniska Högskola, Sektionen för planering och mediedesign

Tack till:

Vi vill här ge tack till alla de som bidragit till detta projekt, i stor eller liten utsträckning.

Thank you Robert Hamilton for all your help and the graciousness to let us use your software

Tack till Jacob Westberg som var med under impuls-inspelningarna

Tack till alla de som kom på testningen.

Tack till Andreas Rossholm.

Tack till våra handledare Pirjo Elovaara och Sebastian Hastrup

Kontaktuppgifter:

Magnus Fredrikson

krummelur@gmail.com

+46 (0)762176128

Pontus Svensson

polle.svensson@gmail.com

+46 (0)708690243

Abstrakt

Många av dagens dator och TV-spelsljud är anpassade för högtalaruppspelning, i många fall även flerkanalssystem som 5.1 surround och liknande. Få spel är anpassade för uppspelning genom hörlurar och för att få en bra ljudlokalisering i hörlurar behövs ofta någon form av surroundhörlurar. Vi tror att man kan skapa trovärdig (Riktnings)återgivning av ljud i hörlurar genom bara två ljudkanaler genom så kallat binauralt ljud. Vi kommer i detta arbete att beskriva de faktorer som spelar roll i vår stereohörsel, varför de fungerar samt vilka svagheter som finns i detta system. Vi kommer att skapa ett system för att behandla realtidsgenererat binauralt ljud för spel och beskriva de verktyg vi valt och den generella arbetsprocessen. Genom att göra detta hoppas vi visa att man genom snabb fouriertransform och andra signalbehandlingsmetoder kan skapa en ljudåtergivning som lämpar sig till hörlurar och håller en nivå för realism som är högre än den traditionella stereoåtergivningen i hörlurar.

Nyckelord: Binauralt ljud, Ljudmotor, Spelljud, Max/MSP, UDK, Digital ljudproduktion.

Abstract

Many of today's TV and computer games are designed for loudspeaker audio playback, in many cases even multichannel sound such as 5.1 surround. Few games are designed for playback through earphones, and to attain good sound localization through headphones. Some kind of surround headphone is often required. We believe that good audio localization can be attained in headphones through the use of 'binaural sound'. In this essay we will describe those factors that matter in to human stereo hearing and how they work. We will create a system for processing real-time binaural audio for games. We hope that through the use of fast fourier transform and other audio processing tools create headphone audio that holds a higher standard for sound localization than traditional stereo audio.

Keywords: Binaural sound, Sound engine, Game sound, Max/MSP, UDK, Digital sound production.

Innehållsförteckning

1.Inledning

De vanligaste sätten att idag återge ljud från spel är med ett stereosystem, en uppsättning med två högtalare där ljud från höger och vänster skiljs, eller ett flerkanalssystem där många högtalare placeras runt lyssnaren. Ett sådant här system ger en mer definierad ljudbild än ett stereosystem eftersom det kan återge ljud från fler infallsvinklar men det har vissa brister. För att ljudet skall kunna presenteras från många olika vinklar behövs flera högtalare, exempelvis i den vanliga konfigurationen 5.1 surround behövs 5 högtalare; två främre, två för sidan samt en framför lyssnaren i centrum för att kunna få en surround effekt. Trots att det i detta system finns högtalare från många vinklar i det horisontella planet så blir "sweetspotten", d.v.s. det område där högtalarnas ljud samverkar optimalt, liten (Lee & Lee, 2010, s.835). För att kunna återge ljud från fler infallsvinklar måste fler högtalare läggas till, exempelvis finns det system där ljud även kan lokaliseras ovanifrån (11.x surround).

Binauralt ljud är inspelat på ett sådant sätt att det skall åstadkomma trovärdig återgivning av ljud från alla möjliga infallsvinklar med enbart två ljudkällor, i de flesta fall hörlurar (Rumsey & McCormick, 2003, s.374). Skillnaden mellan denna återgivning och normalt mixat ljud, d.v.s. ljud anpassat för uppspelning med stereohögtalare eller hörlurar, är att den binaurala mixningen skall utnyttja örats naturliga filtrering och förstärkning av ljuden.

Örat är specifikt utformat för att påverka det inkommande ljudet så att vi skall kunna avgöra var det kommer ifrån. Det faktum att vi har två öron, placerade på motstående sidor av huvudet förenklar hörseln i horisontellt plan framförallt då ljudkällan är placerad rakt ut från huvudet sett. Den mänskliga hörseln använder ett antal olika metoder för att lokalisera ljudkällor, dels volymskillnaden mellan öronen, tidsfördröjningen mellan öronen, samt den frekvenspåverkan som örat, skallen och resten av kroppen ger ljudet.

2. Problemområde

2.1 Bakgrund

Anledningen till att vi valt detta ämne är att möjligheter som för några år sedan inte fanns har öppnat sig i och med att datortekniken utvecklats. Dagens snabba processorer klarar av att hantera ljudberäkningar som för tidigare generationer processorer var otänkbara. Binauralt inspelat ljud är ljud inspelat med ett artificiellt huvud och artificiella öron i form av mikrofoner så att den effekt kroppen har på ljudet återskapas. Vid lyssning i hörlurar skall det det binauralt inpelade ljudet upplevas precis som om du befann dig vid inspelningen (Rumsey & McCormick, 2003, s374).

2011 gjordes en undersökning på KTH fokuserad på binauralt ljud i spel (Brieger & Göthner, 2011, s.24-33). Undersökningen gick ut på ett speltest i spelet *Unreal Tournament 3*(Epic Games, 2009) där binauralt ljud testades av ett antal frivilliga som sedan bedömde tekniken i olika kategorier, från förmåga att rikningsbestämma och avståndsbestämma ljud till spelglädje. Reaktionerna på tekniken var enligt undersökningen mycket positiva i riktning och avståndsurskiljning; alla de tillfrågade svarade att de skulle använda tekniken om den fanns som menyval. Vi ser detta som ett grund för att tekniken uppskattas av dagens spelare och en anledning att fortsätta forska inom ämnet.

Trots att tekniken idag finns att göra spel med ljud som bygger på binaural princip så är det svårt att hitta spel som produceras med detta i åtanke. Av de här anledningarna tycker vi att just nu är en bra tidpunkt att utforska ämnet.

2.2 Syfte

Syftet med kandidatarbetet är att undersöka hur man kan generera binauralt ljud i realtid och se om de det går att använda i en spelproduktion och genom detta bidra till utvecklingen av nya ljudläggningstekniker.

2.3 Problemformulering

Vårt kandidatarbete kommer att handla om hur denna teknik kan appliceras på digitala spelmiljöer. Vår huvudsakliga problemformulering är:

Hur utvecklar man en ljudmotor som fungerar efter binaural princip?

2.4 Tidigare forskning

I detta kapitel kommer vi gå igenom tidigare forskning kring mänsklig hörsel för att ge en överblick av hur mänsklig stereohörsel fungerar samt ge definiera begrepp som behöver etableras för att förstå vårt probelmområde

2.4.1 Ljudmotor

För att vi skall kunna diskutera detta ämne behövs en gemensam definition av begreppet ljudmotor. Med ljudmotor menar vi den del av ett färdigt spel som sköter signalbehandlingen, d.v.s. den del av spelmotorn som sköter om att spela och manipulera ljud så att de placeras rätt i förhållande till lyssnaren. Vi utesluter från begreppet den del av spelet som sköter logiken för hur ljuden skall spelas, exempelvis den del av spellogiken som beräknar spelarens avstånd till ljudet.

Det system som sköter om logiken för ljuden, är förstås ändå en viktig del av att ljudet fungerar, men vi anser att denna logik ser ungefär likadan ut oavsett vilket signalbehandlingsverktyg som används, medan signalbehandlingen i sig skiljer sig dramatiskt.

2.4.2 Örats funktioner

I detta kapitel kommer vi att gå igenom hur örat och hjärnan tar in och tolkar ljudsignaler från omgivningen ur ett lokaliseringsperspektiv. Då vi syftar till att presentera information för användning i skapandet av digitala ljudmiljöer kommer informationen begränsas till vad som är relevant för detta, alltså kommer mer invecklade psykoakustiska fenomen[1] samt neurologiska perspektiv inte tas upp.

[1] vetenskapen om människans subjektiva uppfattning av ljud.

2.4.2.1 Tidsskillnad

Då ljudets hastighet är relativt långsam (c:a 340 m/s) kan vi uppfatta en viss tidsskillnad mellan den tid den inkommande signalen når vänster och höger öra (se Fig 1) Denna tidsförskjutning brukar kallas *Interaural Time Difference*, vilket härefter kommer hänvisas till som ITD.

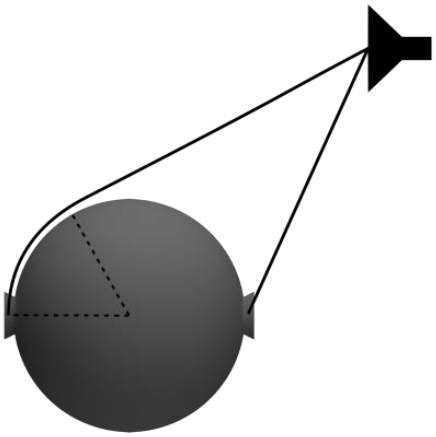


Fig. 1. ITD med hänsyn till en 3 dimensionell form (författares illustration)

Människors öron är placerade på varsin sida om huvudet i samma höjdposition, vilket gör att vi genom ITD har svårt att avgöra ett ljuds höjdposition eller snarare vi kan avgöra vilken vinkel ljudet kommer från relativt huvudet men inte om ljudkällan befinner sig ovanför eller under huvudet. Andra djur, exempelvis ugglor har öronen placerade på olika höjd, vilket ger en klar fördel i ITD-urskiljning jämfört med människor då människor har samma ITD oavsett om man skulle invertera ljudkällans avstånd i framåt-bakåt ledd eller upp-ner ledd i alla kombinationer (se *Cone of Confusion* s.7).

ITDn är väldigt liten; normal maximal fördröjning, d.v.s då en signal är positionerad rakt till höger eller vänster om en persons huvud har uppmätts till drygt 600µs (Blauert, 1997, s76). Denna uppfattning fungerar huvudsakligen på ljud under 1000 Hz då fas-skillnaden för ljud över denna frekvens är svår att urskilja, d.v.s. skillnaden mellan var i signalens period man befinner sig när den träffar öronen blir för svår att urskilja p.g.a. den snabba perioden (höga frekvensen).

Trots detta kan örat höra ljudets ansats och slut och på så sätt även tolka ITD på ljud som ligger ovanför 1 kHz om de har en ojämn natur, t.e.x. ljudet från en EKG maskin.

I ett fritt fält, d.v.s. ett hypotetiskt rum där inga reflektioner förekommer (Rumsey & Mccormick, 2002, s.22) fungerar denna funktion som bäst. Kvaliteten på ITD-lokalisering sjunker då det omgivande rummet stör den inkommande signalen genom att reflektera ljudet, vid reflektionerna diffrakteras ljudet och örat får in fler signaler än den ursprungliga (torra) signalen, vilket leder till förvirring då hjärnan skall tolka dessa. Experiment har dock visat att det trots dessa reflektioner är möjligt att lokalisera ljud (Wallach, Newman & Rosenweig, 1949, s.315) och att dessa reflektioner kan till och med hjälpa oss att lokalisera ett ljud.

Hjärnan använder ett antal olika tekniker för att uppfatta ett ljuds position, denna teknik har som nämnts brister och det finns även flera scenarier där den är oduglig.

I de omständigheter då ljudkällan befinner sig på symmetriplanet(se Fig.2.), d.v.s. då den befinner sig framför, bakom, över eller under personen kommer ljudet att träffa både höger och vänster trumhinna vid samma tidpunkt.

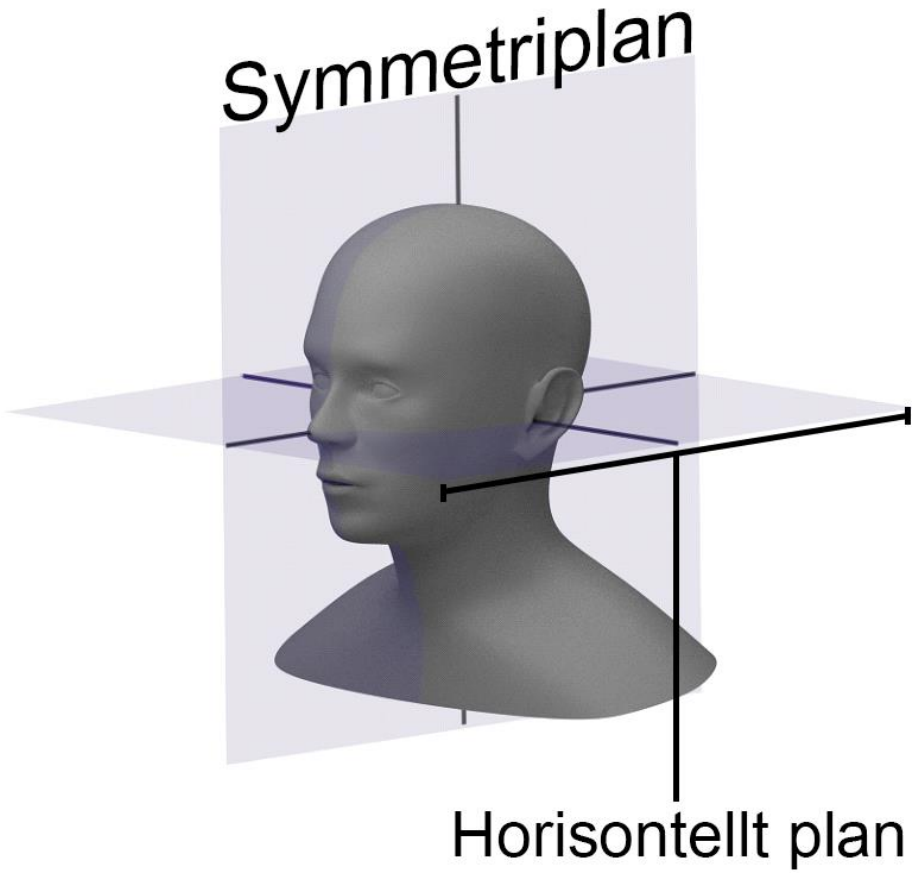


Fig. 2. Planindelning av huvudet (författares illustration)

Ett liknande fenomen uppstår även när en ljudkälla befinner sig i det så kallade *cone of confusion*(se Fig.3.). Med cone of confusion menas alla punkter som kan nås genom en kon med toppunkt i ena örat som går rakt ut, eller in genom huvudet. För varje möjlig ljudposition finns alltså ett antal punkter i vilka ITD för höger repektive vänster öra är identiska, detta är annorlunda från punkter på symmetriplanet där ITD för båda öronen är densamma.

Konen för en given ljudkälla bildar alltid en lodrät cirkel. I båda ovan nämnda scenarier kan vi istället använda andra signaler, exempelvis ytterörats effekt på ljudet för att bättre fastställa en ljudkälla. Ett vanligt sätt att försöka lokalisera ljud då de befinner sig utanför vårt synfält eller i cone of confusion är att vrida på huvudet så att det hamnar framför oss (Blauert, 1997, s.179), denna vridning sker mer eller mindre reflexmässigt, vi vill ha ljudet framför oss eftersom det är då vår hörsel fungerar som bäst.

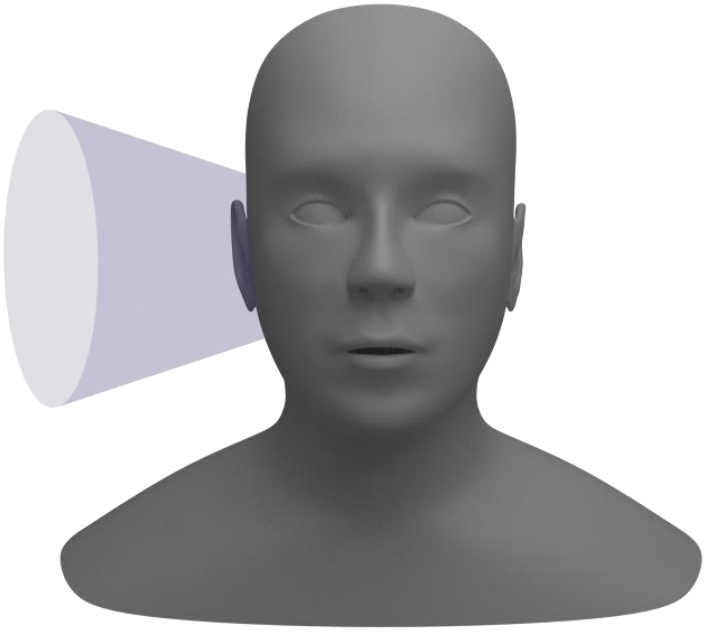


Fig. 3. Cone of Confusion (författares illustration)

Det enklaste sättet att beräkna ITD för en specifik ljudposition är att anta att ljudet passerar genom huvudet, i detta fall kan man enkelt anta en rätvinklig triangel för att fastställa en rak linje mellan ljudkälla och öra. En annan modell som används är att beräkna den kortaste sträcka som ljudet kan nå varje öra utan att passera igenom huvudet (Blauert, 1997, s.76). Även i enkla fall som när man antar huvudets form till en sfär blir denna metod något mer komplicerad än att beräkna en rät linje. För att beräkna ITD med hänsyn till en komplexare form som ett mänskligt huvud blir det uppenbarligen mer komplicerat.

Sanningen är att båda dessa modeller är i verkan, ljud kan passera igenom olika material i mån av dess frekvens. I regel har lågfrekventa ljud lättare att ta sig igenom fasta material, exempelvis väggar eller ett mänskligt huvud, medan ljud med högre frekvens absorberas mer. Detta är dock

en generalisering. Andra faktorer t.ex. föremålets egna resonanta frekvens[2] kan påverka dess inverkan.

2.4.2.2 Inner och Ytteröra

Ytterörat har flera funktioner, huvudsakligen syftar det till att lättare låta oss höra de frekvenser som ingår i mänskligt tal och förstärker dem med c:a 5db (Everest & Pohlman, 2009, s40).

Utformningen varierar från individ till individ och är en av de faktorer som är mest individuell vid ljudlokalisering. Ytterörat agerar på det inkommande ljudet på många sätt och fungerar som ett filter som försvagar och förstärker olika frekvenser. Beroende på vinkeln till ljudkällan varierar frekvenserna. Varje individs öra har en unik utformning vilket leder till en unik utformad skuggning och förstärkning av frekvenser.

Även innerörat påverkar ljudet och har en egen frekvenspåverkan. Denna frekvenspåverkan beror likväl som ytterörats på det inkommande ljudets vinkel. Man har länge vetat om dessa effekter men har inte kunnat mäta dessa p.g.a. mikrofonkvalitet samt att det krävs stor beräkningskraft för att sammanställa resultaten (Blauert, 1997, s.290).

Örat är extremt komplext och ytterörat är bara en liten del av allt som spelar in i hur vi upplever ljud.

Innerörat påverkar ljudet, bland annat genom hörselgången, vars utformning till största delen är till för att förstärka de inkommande ljuden för att de lättare skall uppfattas. Dess utformning är den av ett slutet rör, slutet vid den sidan som har förbindelse med trumhinnan. De akustiska egenskaperna för ett slutet rör är alltså relevanta då man vill ta reda på vilka frekvenser som kommer att förstärkas som mest i hörselgången.

Everest och Pohlman (2009, s41) skriver att den normala längden för hörselgången är 2,5 cm, och att detta resulterar i en maximal förstärkning av 12db vid 3000Hz, övertoner tillkommer till denna förstärkning.

[2] De flesta objekt har en inneboende resonant frekvens, om ett objekt blir utsatt för denna frekvens kommer det i medvibration.

Eftersom denna effekt kvarstår även i lyssning i hörlurar är det överflödigt att återskapa den vid lyssning. Det är dock inte helt otänkbart då denna effekt också beror på ljudets riktning, problem uppstår dock då effekten skulle bli dubbel eftersom ljudet alltid kommer att vara placerat rakt utåt från vardera öra. En lösning på detta skulle kunna vara att väga upp för den 'verkliga' effekt som uppstår i överförandet av ljud från hörlur till trumhinna genom att använda filter som motverkar den.

En bieffekt av den roll ytterörat spelar för vår ljuduppfattning är den att vi lätt kan misstolka ljud då de har en klangfärg som innefattar naturliga eller syntetiskt framställda bandspärrfilter, d.v.s. ett filter som spärrar ett specifikt frekvensband. Vår hjärna är byggd för att tolka sådana klangfärgsegenskaper som ett signalement för var ljudet befinner sig snarare än en naturlig egenskap hos ljudet och vi har ett specifikt område i hjärnan som reagerar på dessa ljud (Davis. 2005, s294). Ett enkelt experiment som illustrerar det här går ut på att en person får lyssna på vitt brus[3] i ena örat med endast en hörlur och sedan fått anpassa ett bandspärrfilter visar att en sänka vid 7.2kHz gett en subjektiv uppfattning att ljudkällan befann sig i öronhöjd. Experimentet visar att en sänka vid 8kHz däremot ger intrycket att ljudkällan flyttat sig högre än öronhöjd, medan en filtersänka vid 6.3kHz ger intrycket att ljudet befinner sig lägre än ögonhöjd (Everest & Pohlman, 2009, s.41).

2.4.2.3 Nivåskillnad

Ett ljuds nivåskillnad mellan höger och vänster öra, även kallad *Interaural Level Difference* (ILD) ger oss möjlighet att urskilja ett ljuds riktning. Denna nivåskillnad kan dock inte ge direkt information om vilket avstånd källan ligger. Precis som vid ITD, är det en relativ skillnad mellan de två öronen. Relationen mellan nivåerna i öronen säger alltså ingenting om avståndet till källan. Relationen mellan det inkommande ljudet och den första reflektionen säger däremot mycket om ljudets avstånd, ju högre andel reflektion desto längre avstånd till källan, exempelvis, en viskning på nära avstånd ger ett i stort sett helt torrt ljud. (Rumsey & McCormick, 2002, s35).

[3] En ljudsignal som innehåller alla möjliga frekvenser och har en likvärdig energi för alla frekvensband.

ILD kan illustreras genom ett hypotetiskt försök med två punkter representerande öron och en punkt representerande ljudkälla i ett fritt fält. Ljudtrycket från en oriktad[4] ljudkälla färdas utåt i en sfärisk form. Trycket i en given punkt på sfärens yta kommer därför att försvagas enligt:

$I = W/4\pi r^2$ [5]

Där *r* är avståndet till ljudkällan.

Örat jämför därför de inkommande signalerna och härleder positionen genom skillnaden. Enligt Blauert är det svårare att avgöra en persons position då talaren inte är tidigare känd av lyssnaren (9° felmarginal för känd talare respektive 17° för okänd talare i horisontellt plan)(Blauert, 1997, s.44).

Det bör dock noteras att de två undersökningar Blauert baserar detta påstående på har olika förhållanden, den ena är utförd med en talarvolym på 35 phon[6] medan den andra använder 65 phon. Det noteras även att vitt brus har lägre felmarginal än båda dessa (4° vid 60 phon). Detta kan bero på att vitt brus skiljer sig från en människas röst i och med att dess bandbredd är större, vilket skulle kunna vara anledningen till att den är mer lättlokaliserad; vitt brus skall ha en likvärdig energi för alla frekvenser och hypotetiskt sträcker sig till alla möjliga frekvenser vilket innebär att det träffar alla mänskligt hörbara frekvenser.

Blauert nämner även att det är lättare att lokalisera ett ljud med smal bandbredd då signalens frekvens är högre än 5 kHz (Blauert, 1997, s.311).

I de fall personen har en tidigare referens till ljudet som upplevs borde även ljudets avstånd kunna identifieras.

De medel vi har beskrivit som hjärnan kan använda för att avgöra ljudets position är alltså den inkommande signalens tidsskillnad, dess nivåskillnad mellan öronen samt det sätt som ytter och innerörat påverkar ljudet. De olika metoderna skiljer sig vilt i deras komplexitet samt hur enkelt de kan överföras till digital form. Digitalisering av ITD samt ljudnivåjustering är tämligen simpla

[4] Ljudkälla som ger lika ljudtryck i alla riktningar.

[5] I = Intensitet vid given punkt, W = Effekten vid ljudkällan

[6] skala för ljudintensitet baserad på mänsklig ljuduppfattning

beräkningar för en dator medan ytterörats effekt inte är genomförbart utan någon typ av frekvensfiltrering, något som har tämligen hög beräkningskostnad[7] att utföra.

2.4.3 Binauralt ljud

Detta kapitel kommer att behandla binauralt ljud och kommer att gå igenom olika tekniker för inspelning av det. Kapitlet kommer även att förklara skillnaden mellan binauralt inspelat ljud och syntetiserat binauralt ljud samt hur de här teknikerna lämpar sig för interaktivt användande.

Med binauralt ljud menas ljud som spelas upp på ett sådant sätt att det återskapar de akustiska förhållande som örat och resten av kroppen utgör. Binauralt ljud kan uppnås genom inspelning av ljud genom en simulerad lyssnare som påverkar ljudet som en människa skulle ha gjort. Vanligtvis sker det här genom en speciell mikrofonuppsättning inbyggd i en modell av ett mänskligt huvud eller en mänsklig kropp, de här modellerna brukar kallas för *"dummyheads"*. (Rumsey & McCormick, 2003, s374). Man kan även uppnå detta genom speciella mikrofoner som kan placeras i en människas ytteröra, exempelvis *Soundman OMK II*. Även innerörsmirofoner (Probe microphones) har använts (Blauert, 1997, s.32), de här mikrofonerna är annorlunda i och med att de når ännu längre in i hörselgången än mikrofoner av *OMK II* typen och därför kan mäta trycket inne i hörselgången.

Denna mikrofonpositionering återskapar örats influens på det inkommande ljudet, vilket underlättar lyssnarens förmåga att lokalisera ljudets ursprungsposition.

Det finns olika typer av *"dummyheads"*, de som har mikrofoner vid öronöppningen och de som har mikrofoner längre in i huvudet.

De sistnämnda är utformade mestadels för ljudmätningar och inte för produktionssammanhang (innehåller även ibland simulering av örats inre delar) medan de andra är designade för inspelning.

Båda de här modellerna kan användas i produktionssammanhang men vilken modell som bör väljas beror på den typ av lyssning man avser. För lyssning med utanpåsittande hörlurar, d.v.s.

[7] En beräkningsinstensiv process för en dator.

öronkåpor med små högtalare i bör man inte använda "*dummyheads"* med simulering av örats inre delar då detta leder till en dubblering av innerörats effekt på ljudet. De här inspelningarna kan dock vara bra om man lyssnar med innerörslurar d.v.s. små hörlurar som sticks in i öronen, då ljudkällan kommer längre in i örat. Detta innebär att filtreringen sker under inspelningen i stället för i örat vid lyssning.

Lyssnar man med utanpåsittande hörlurar så är det en inspelning gjord med ett "*dummyhead"* med mikrofoner vid öronöppningen bäst lämpat, eftersom innerörat själv påverkar ljudet(Rumsey & McCormick, 2003, s.375).

2.4.3.1 Binaural Syntes

Binauralt inspelat ljud är alltså fast och därför inget som är optimalt för spel i förstapersonsvy eftersom ljudets riktning behöver ändras väldigt ofta helt beroende på hur spelaren rör sig.

För att komma runt detta men ändå uppnå en binaural effekt kan man använda sig av något som heter *Head Related Transfer Function*, som är en syntetisk avbild av en persons individuella öronsignatur. Denna funktion kommer härefter hänvisas till som HRTF.

En personlig HRTF skapas genom att spela in många impulssvar från olika infallsvinklar med mikrofoner placerade i personens öron som sedan faltas med originalsignalen. Faltning (se bilaga 1.) innebär att man tar in signalerna i frekvensdomän för att multiplicera dem. För att kunna återskapa en persons HRTF behöver man mätningar som visar hur ljudet av en impuls förändras beroende på hur ljudkällan är placerad gentemot åhöraren. Liknande mätningar kan göras med ett ett *"dummyhead"* eller ibland även med en artificiell kropp. Ett *"dummyhead"* har som tidigare nämnt fördelen att mikrofonerna kan byggas in i modellen och även simulera innerörat och torsons effekt på ljudet.

Vad man är ute efter när man gör mätningar av detta slag är kroppens effekt på ljudet vid olika vinklar.

Mätningarna sammanställs för att man skall få fram ett register där vinklarna representeras.

Det finns olika sätt att praktiskt genomföra mätningarna, två praktiska metoder är att antingen använda en rigg med högtalare som kan roteras medan en testperson eller ett *"dummyhead"* sitter statiskt i centrum, den andra metoden är att rotera en åhörare i rummet.

För att en impulsmätning skall vara effektiv behövs en så kort impuls som möjligt, optimal tid anses vara 4.5ms (Algazy, 2001, s.1). Digitalt sett innebär detta 200 sampel vid 44.1kHz. impulsen skall även vara av hög amplitud, vilket kan leda till problem för både mänsklig hörsel och mätutrustning; är signalen högre än mikrofonen klarar av att ta upp kommer signalen att klippas[8] i de här fallen kan man använda en serie impulser (Blauert, 1997, s.24).

Varje individ är unik, en HRTF mätning utförd för en person kommer inte nödvändigtvis att fungera för en annan.

Några fenomen återfinns hos alla individers öron, t.ex. hur ljudet skuggas av öronvingen då infallsvinkeln närmar sig baksidan av huvudet, även hur örat förstärker vissa frekvenser.

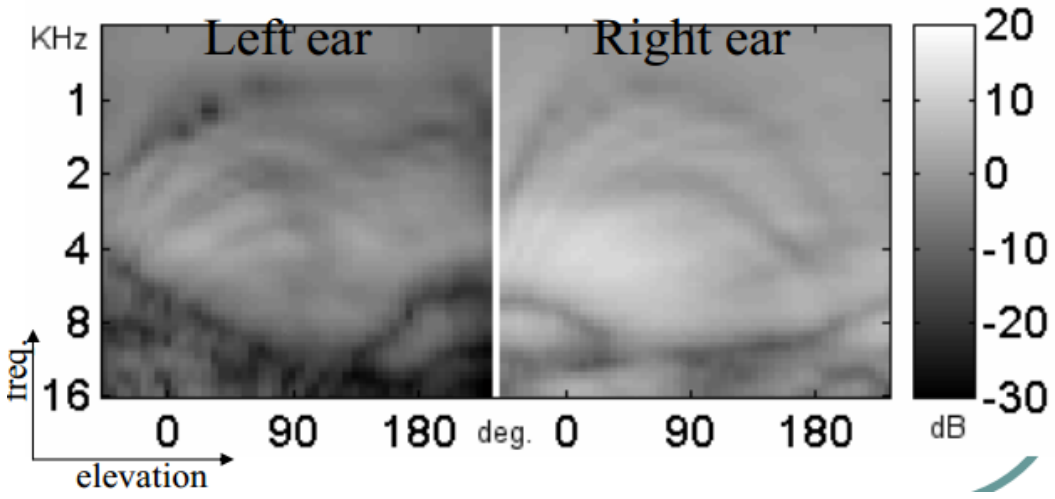


Fig 4. Frekvensrespons (Duraiswami s.34)

Bilden ovan är en visualisering av en impulsinspelning från olika vinklar i höjdled vid 45 grader horisontell vinkel för ett öronpar. De olika infallsvinklarna ger förstärkningar och försvagningar

[8] Klippning uppstår när en signal är starkare än ett medium kan återge. Vid klippning uppstår övertoner eftersom sinustonens rundade form inte får plats. Detta leder till ofta oönskade övertoner, så kallad distortion.

av responsen hos de olika frekvenserna. De mörka regionerna på frekvenserna över 4kHz är filtrering från ytterörat (Duraiswami s.34)

Vi kan se att de här försvagningarna och förstärkningarna är sammanhängande och att de med vinkeln flyttar sig i spektrumet. Som nämnt tidigare finns de här mönstren hos alla olika individer men skiljer sig i sitt yttrande beroende på örats och huvudets utformning. Detta mönster kan ses som ett individuellt öras signatur och med hjälp av detta kan man förstå sin ljudomgivning, lokalisera ljud etc. Problemet med denna teknik blir därför att när individen tolkar sin omgivning använder sitt eget öras signatur för att lokalisera ljuden, en signatur som är inlärd genom lång tids exponering för den.

Undersökningar har gjorts där en individ använder en artificiell modifikation till sitt ytteröra för att förändra sitt öras unika signatur. Det visade sig att lokalisering till en början var dålig men att förmågan efter viss tid återhämtade sig då personen lärde sig sin nya signatur (Hofman, Van Riswickand & Van Opstal, 1998). Det bör dock noteras att trots att lokaliseringsförmågan försämrades vid modifieringen så behölls dock viss lokaliseringsförmåga och att den försämrade lokaliseringsförmågan nästan helt och hållet handlade om ljud i höjdled medan lokaliseringen i horisontell vinkel(denna vinkel utgår rakt framifrån, vilket innebär att 0◦ befinner sig rakt framför personen och sedan räknas åt höger) förblev i stort sett oförändrad (Hofman, 1998, s.418). Förvirring kan alltså uppstå om en miljö presenteras genom en annan individs signatur. HTRF upplevs från den inspelade signaturens perspektiv.

Ett av problemen med en digitalisering av en persons öronsignatur är den generalisering som måste ske om resultaten inte skall behöva beräknas per individ, en av de tänkbara lösningarna på detta är vad företaget *Blue Ripple Sound* har gjort för sin mjukvara *Rapture 3D*[9]: att låta användaren välja mellan 5 olika generella typer.

En annan tekniskt möjlig lösning kan vara att låta användaren anpassa HRTF-logaritmen genom att själv ställa in värden som längd på hörselgång och utformning av ytteröra genom att ställa in parametrar. Detta är förstås inte information som man kan anta att en lyssnare har omedelbar

[9] mjukvara byggd för att simulera 3d ljud för redan existerande spel genom att agera som ett tillägg till dessa.

tillgång till. Blauert använder begreppet ‘det typiska örat’ (the typical ear) och hänvisar till ett generaliserat ytteröra som ger upphov till en viss frekvenspåverkan.

Med HRTF teknik tillkommer även linjär interpolering[10] av ljudets vinkel eftersom mätningar inte praktiskt kan göras vid alla möjliga vinklar; antalet mätpunkter är begränsade. Interpoleringen innebär att de olika inspelningarna skall blandas på ett sådant sätt att ljudet upplevs flytta på sig när spelaren gör det för att placeras i rummet. Detta antal skulle också bli avsevärt mycket större om man även utför mätningar vid olika avstånd.

Vid användande av binauralt ljud i en digital 3d miljö där spelaren själv får navigera omgivningen fritt, kan antingen denna tekniken eller binaural inspelning tillämpas. Skillnaden mellan de två teknikerna i detta sammanhang är enorm, den ena, riktig binaural inspelning är mer statisk då ljud måste spelas in från alla de tänkbara vinklarna. Båda teknikerna måste tillämpa interpolering, men vid den binaurala inspelningen måste interpolering mellan inspelade ljudfiler ske snarare än en frekvenskurva.

De praktiska skillnaderna mellan HRTF Syntes och binauralt inspelat ljud är stora.

En inspelning för digital miljö gjord med binaural mikrofonuppsättning blir otroligt komplicerad när spelaren får rum att fritt röra sig i miljön. För att inspelning av detta slaget skall bli övertygande måste inspelningar göras för ett tillfredsställande antal inspelningar. Det antal som behövs för bestäms av ljuddesignerns krav på kvalitet, fler inspelningar ger ett mer exakt resultat. I det fall man skulle nöja sig med korrekt lokalisering i vågrätt plan skulle man t.ex. kunna använda sig av 10 inspelningspositioner som omger spelaren.

Inspelningarna måste dessutom fånga exakt samma ljud varje gång, för att interpolering skall fungera. För att klargöra vad vi menar med detta stycke kan vi använda ett exempel av hur inspelningsmetoden skulle kunna användas vid ett spelprojekt. Först och främst skulle en inspelningsrigg behövas, d.v.s. en rigg där både inspelningsmekanism, mänskligt huvud med mikrofoner eller ett “*dummyhead*”, samt ljudkällor; högtalare finns. För ljud som skulle vara placerade statiskt gentemot spelaren t.e.x. fotsteg skulle man kunna placera ljudkällan på rätt plats

[10] Att anta ett värde för en punkt genom att utgå från andra kända värden. Linjär interpolering innebär att mellan varje känd punkt dra en rak linje

och sedan spela in det. För ljudkällor som skall vara dynamiskt placerade relativt lyssnaren skulle man behöva spela in ljudet från alla vinklar man vill ha med i spelet.

Vi nämner att alla inspelningar skulle behöva vara likadana, anledningen till detta är så att spelaren när den vrider sig skall höra samma ljud, fast från den nya positionen. Vi föreslår att för att uppnå detta använda en rigg där man flyttar en högtalare och sedan spelar in varje individuellt ljud.

Båda de här teknikerna kräver dock en ljudnivåjustering som motsvarar objektets avstånd från spelaren. Denna justering måste stämma överens med hur ljud avtar i verkligheten, som vi beskrivit tidigare.

Det finns alltså olika alternativ att välja då man skall uppnå binauralt ljud. Binauralt ljud ter sig som ett utmärkt val av ljudteknik då man skall uppnå virtuella miljöer där användaren skall höra från ett förstapersonsperspektiv. Vi har beskrivit nackdelar och fördelar med olika tekniker och kommit fram till att det rimligaste alternativet för binauralt ljud i sammanhang som skall vara dynamiska är binaural syntes snarare än binaurala inspelningar gjorda med de speciella stereomikrofonuppsättningar vi nämnt.

3. Tillvägagångssätt

Vårt produktionsmål för denna delen av arbetet var att överföra den forskning vi gjort under föregående delar till en användbar ljudmotor, detta eftersom vår frågeställning var:

Hur utvecklar man en ljudmotor som fungerar efter binaural princip?

Med ljudmotor menar vi en enhet som hanterar data från ett spel, och spelar upp ljud därefter, med hänsyn till variabler som avstånd, vinkel etc. Även funktionalitet som inte ingår i den tidigare forskningen ingår i begreppet ljudmotor som vi definierade i början i arbetet (se kapitel 2.4.1.). För att vi skulle anse vår ljudmotor vara färdig för användning skulle den kunna spela upp ljud på ett sådant sätt att ljuden uppfattas som rätt placerade i spelmiljön, men även kunna utföra funktioner som att stänga av och slå på ljud, hantera både repeterande ljud och så kallade one shots (ljud som bara hörs en gång och sedan tystnar) samt hantera ambiensljud (skiljer sig från andra ljudeffekter eftersom de inte spatialiseras[11] utan ger ett ljud som omger spelaren från alla vinklar). Vi ville hålla oss till en hög standard för ljudet i vår produkt, detta inebar att vi även implementerade ett system för att hantera efterklang, för att ge en känsla av rumslighet.

Vi döpte vår ljudmotor till *BaSE* (*Binaural Sound Engine*) och kommer att använda denna förkortning när vi skriver om ljudmotorn.

3.1 Produktion

I denna del kommer vi att gå igenom den process som ledde till resultatet. Vi kommer att gå igenom verktygsval, samt hur arbetsprocessen och testningen gick till.

Vår process skedde med ett antal delmål som ledde processen:

[11] Att rumsligt placera ljudeffekten.

3.1.1 Fungerande kommunikation/verktygsval.

Med fungerande kommunikation menar vi att hitta en lösning för att få användbara data från en spelmotor till *BaSE*. Därför var vi tvungna att välja verktyg (programvara), samt en kommunikationsmetod som var kompatibel med de program vi vill jobba i. Att välja ut programvara blev det första steget vi genomförde eftersom resten av produktionen berodde på de verktygsval vi gjorde.

3.1.2 Implementering.

Vi var tvungna att bestämma oss för vilka data vi kommer att behöva och hur vi processar dessa i de program vi väljer. Detta steg innefattar även att återskapa de effekter vi beskrivit i forskningen.

De här effekterna är:

ITD (se kap.2.4.2.1)

Örats och resten av kroppens frekvenspåverkan av ljudet(se s8)

Ljudets avtagning beroende på avståndet (se kap.2.4.2.3)

Även effekter som inte nämns i problemområde kommer att återskapas:

Doppler effekt[12]

Ljudskymning (ljud som skyms av geometri i miljön)

3.1.3 Optimering/Testning

Dessa två steg sker överlappande eftersom optimeringen skedde under hela slutfasen av arbetsprocessen och behövde göras allteftersom ändringar i programmet gjordes. Vi genomförde ett konsumenttest, där vi lät personer testa *BaSE* och en annan ljudmotor utan binaural funktion.

[12] Dopplereffekt (inom ljud) är ett fenomen som uppstår då en ljudkälla närmar eller avlägsnar sig från en åhörare. Dopplereffekten påverkar ljudkällans tonhöjd; om ljudkällan närmar sig blir den högre men om källan avlägsnar sig blir den lägre.

3.1.4 Metoder

Vår utvecklingsmetod var väldigt iterativ; vi skapade modulerna, testade dem och satte de sedan i sitt sammanhang. I varje steg av den här processen har vi testat och kunnat gå tillbaka och redigera, eller helt slopa delar av ljudmotorn. Processen ansågs som färdig när vi gått tillbaka till testningsteget utan att hitta några fel eller någonting som behövde åtgärdas.

Denna karta illustrerar den iterativa processen:

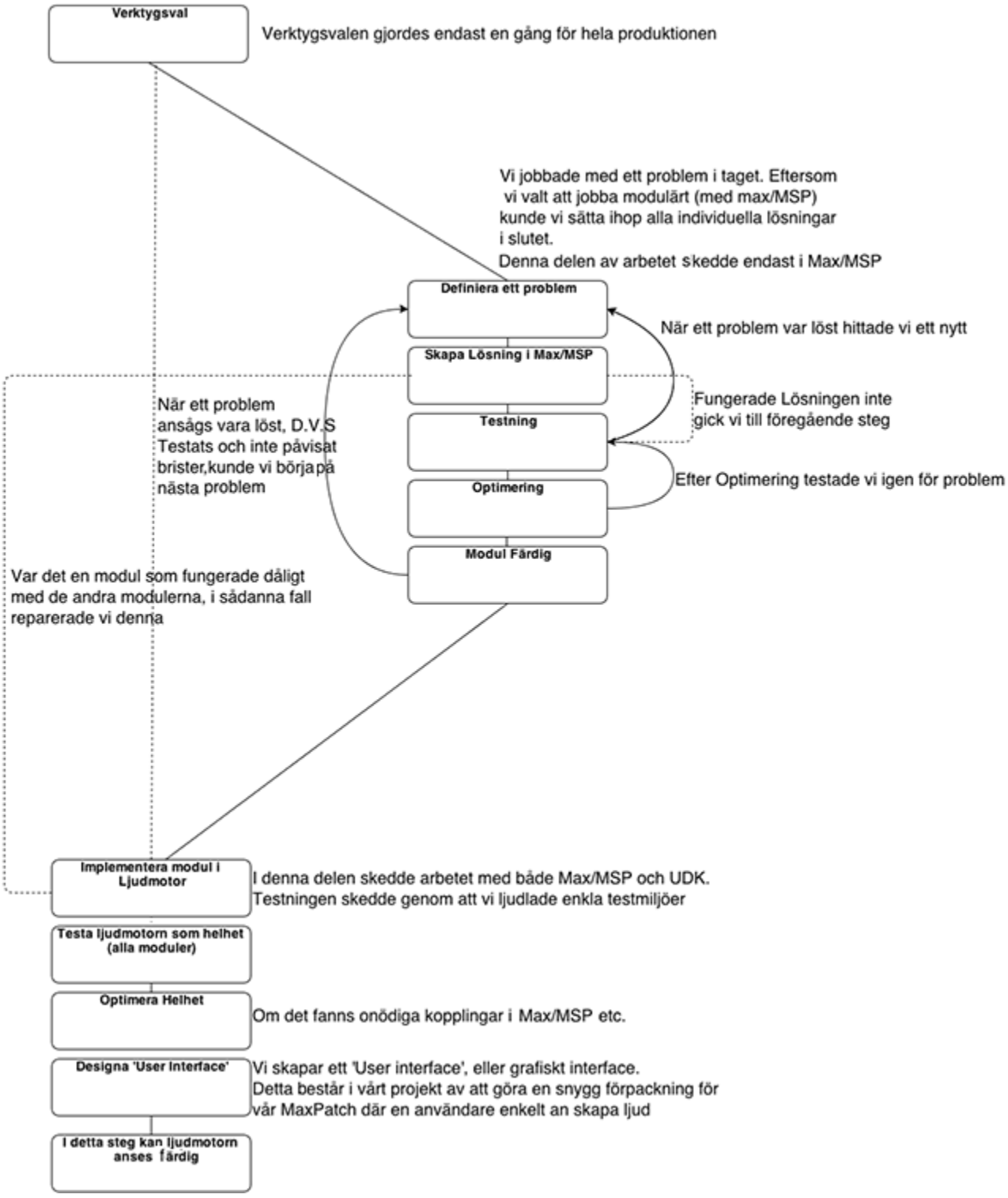


Fig. 5. Karta över arbetsprocess

3.1.5 Verktygsval

De verktyg vi använt oss av i vår produktion är *UDK*[13] och *Max/MSP*[14], i detta kapitlet kommer vi gå igenom hur vi använde dessa och hur de fungerar tillsammans i vår produktion.

Max/MSP är ett modulärt användargränssnitt för ljud och bild med mycket inbyggd signalbehandlings funktionalitet[15]. Beslutet att använda *Max/MSP* som utvecklingsmiljö byggde på ett antal faktorer, den största av vilka är att vi har föregående utbildning i programmet som vi fått genom kursen 'Modulär Ljuddesign' läst på BTH, och tidigare hobbyanvändning av programmet. Det finns ett antal liknande program, exempelvis Pd (*PureData*[16]), vars gränssnitt liknar *Max/MPS*'s och *Supercollider*[17] som är ett programmeringsspråk för ljud. Den stora skillnaden är den rikedom av dokumentation som finns för *Max/MSP*[18], exempelvis dess forum där man snabbt kan få svar på frågor samt det användargenererade innehåll som övningsvideor och andras forumtrådar. Vi övervägde till en början att använda programmet *Supercollider* men efter svårigheter att hitta information så som tutorials etc. för att få en inblick i hur utveckling i denna miljö går till bestämde vi oss för att inte använda det.

Vi ansåg att risken för att inlärningsprocessen skulle hindra vårt arbete var för stor.

Max/MSP's gränssnitt är objektorienterat[19] och programmen byggs av så kallade *Patcher*. En Patch kan vara ett slutgiltigt program men kan också instanseras (d.v.s. användas som ett objekt, inuti en annan patch, se: objektorienterat) i flera nivåer inuti en patch. Inom *Max/MSP* görs det skillnad på objekt som hanterar ljud (dessa kallas MSP-objekt) och objekt som hanterar annan data

---

[13] Unreal Development Kit http://www.unrealengine.com/udk/ [senast läst: 2013-05-10]
[14] http://cycling74.com/ [senast läst 2013-05-09]
[15] Signalbehandling innebär att manipulera signaler med hjälp av matematiska metoder.
[16] http://puredata.info/ [senast läst 2013-05-09]
[17] http://supercollider.sourceforge.net/ [senast läst 2013-05-09]
[18]Max/MSP Dokumentation: http://www.cycling74.com/docs/max5/vignettes/intro/docintro.html [senast läst 2013-05-09]
[18]Cycling74' Forum: http://www.cycling74.com/docs/max5/vignettes/intro/docintro.html[senast läst 2013-05-09]
[18]Exempel på utbudet av användarskapat innehåll kring Max/MSP http://www.youtube.com/playlist?list=PLD45EDA6F67827497 [senast läst 2013-05-09]
[19] Objektorienterad programmering innebär en programmeringsmetod där ett program kan innehålla ett antal objekt som interagerar med varandra.

(dessa kallas Max-objekt). Den största skillnaden mellan de här för oss som utvecklare är att MSP-objekten kan hantera data mycket snabbare än andra objekt, eftersom de arbetar med ljudkortets samplingsfrekvens. MSP-objekten är därför mer processorkrävande än Max-objekt, eftersom Max-objekt bara updateras efter en bestämd frekvens (minst 1ms, men i vårt fall c:a 50ms). Förutom patcher har man även möjlighet att programmera sina egna specialanpassade objekt. Det huvudsakliga programmeringsspråket för *Max/MSP* är *C*, men möjlighet finns att skriva objekt i *Java*. Eftersom *Java* hanterar ljuddata c:a 1,5–2,5 gånger så långsamt som *C* objekt (La Fata, 2000, s.43) bestämde vi oss för att av prestandaskäl inte skriva några egna MSP objekt i *Java*. Vi hade inte heller nog kunskap av *C* för att kunna skriva objekt i det.

Max/MSP har dock ett stort utbud av färdiga ljud-objekt och alla de signalbehandlingarna vi förväntade oss behöva göra, även de mer komplicerade, finns tillgängliga bland dessa. Java objekt användes för att filtrera den data som vi skickar från spelmotorn till *Max/MSP*. Exempelvis måste ljuddata, d.v.s. data om ljuds position, volym etc. och spelardata (information om spelarens position etc.) skiljas åt. De uträkningar som behöver göras väljer vi att göra till så stor del som möjligt i *UnrealScript*[20] snarare än *Java* och MAX objekt, eftersom det ger en mindre komplicerad MAX-patch.

På vilket sätt lämpar sig *Max/MSP* till att producera en ljudmotor avsedd till 3d-spel?

Tidigare projekt har gjorts och dokumenterats där *Max/MSP* använts som en ljudhanterare för spel.

Exempel på ett sådant projekt är denna videoserie[21] (*Max/MSP Audio Engine for Space Invaders*). I detta exempel har en student gjort en MAX-patch för att ljudlägga ett '*Space Invaders*' spel. Händelser så som när ett skott avfyras av spelaren startar ljud i *Max/MSP*. Ett annat exempel där en extern ljudutvecklingsmiljö används i samband med en spelmotor är Teleharmonium.

Teleharmonium är en installation, eller konsert, som till stor del bygger på *UDK*.

UDK (*Unreal Development Kit, Epic Games*) är en utvecklingsmiljö för dator och TV-Spel. Det som gör installationen relevant för vår produktion är att Teleharmonium genomförs med datatrafik mellan *UDK* och en extern ljudhanterare.

[20] http://udn.epicgames.com/Three/UnrealScriptReference.html [senast läst 2013-05-09]
[21] http://www.youtube.com/watch?v=-rcJZxIPg7w [senast läst 2013-05-09]

Detta sker genom ett tillägg till *UDK* som heter *UDKOSC*(*Unreal Developement Kit Open Sound Control*). *UDKOSC* använder sig av *UDP* (*User Datagram Protocol*) kommunikation mellan program. Efter att ha sett Teleharmonium samt ett par andra videobaserade exempel på *UDKOSC* bestämde vi oss för att detta var ett utmärkt sätt att skapa en länk mellan *Max/MSP* och en spelmiljö; Teleharmonium och de andra exemplena övertygade oss om att det var möjligt att använda *Max/MSP* som ljudmotor.

UDKOSC är utvecklat av *Robert Hamilton* vid *CCRMA*, avdelningen för musik, *Stanford University*, och är gratis[22]. Robert Hamilton var till mycket stor hjälp i det tidiga skedet av arbetet och delgav oss information om hur vi kunde använda *UDKOSC* samt svarade på de frågor vi hade om programmet. Exempelvis hjälpte han oss att få igång fungerande *UDP* traffik mellan *UDK* och *Max/MSP*.

3.1.6 Arkitektur

De programmen som nämnts kommer alltså att sammarbeta på detta sätt:

[22] https://ccrma.stanford.edu/~rob/software.php [senast läst 2013-05-09]

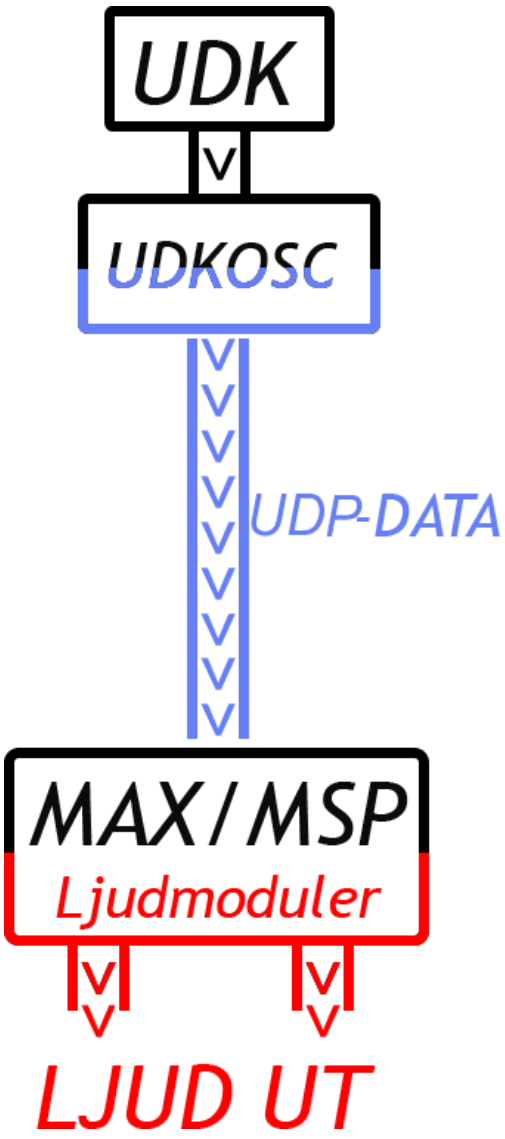


Fig. 6. karta över programvaror

UDK, hanterar alla data i spelet, utvalda data som är relevanta för att skapa ljudbilden skickas från *UDK* till *Max/MSP* via programtillägget *UDKOSC*. Dessa data är exempelvis spelarens position och rotation, samt information om ljudkällorna i scenen, deras ljudstyrka, hur mycket efterklang som skall appliceras, och om deras radie skall modifieras, exempelvis behöver ljudet av en fluga inte ha en stor radie, medan ljudet av en jumbojet behöver en mycket stor radie. Ytterligare *UDK*-objekt används till exempel för att ställa in rumsklangen i *Max/MSP*. Datan behandlas i *Max/MSP* och används för att skapa de effekterna vi nämnt i tidigare kapitel(se Kap. 2).

3.1.7 Arbetsprocess i Max/MSP

För att återskapa de binaurala effekter som beskrivits i den kontextualiserande delen (Kap. 2) av detta arbete använder vi *Max/MSP* objekt. ITD(se s4) görs genom att räkna ut respektive öras avstånd till varje ljudkälla. UDK-karaktärer har inga öron, så positioner för dessa måste antas, detta görs genom sinus och cosinus [23] av spelarens horisontella riktning, eller 'girning' (Engelskans yaw).

Ljuden skickas sedan genom *Max/MSP*-objekt som fördröjer höger respektive vänster kanal med bestämd tid. Vi behövde skapa extremt korta fördröjningstider, eftersom fördröjningstider så låga som 100 mikrosekunder gör stor skillnad på ljudupplevelsen.

Problemet med så här korta fördröjningar ligger i hur datorer processar ljud. Ljud processas och skickas ut till ljudkortet i flera steg. Först beräknas ljudet i segment, sedan skickas det ut till högtalare/förstärkare i segment. Segmenten kallas vanligtvis vektorer och storleken på de båda vektorerna definieras individuellt, och beskrivs i sampel. En normal vektorstorlek för ljudberäkning (signalvektor) kan ligga på 64 upp till 2048 sampel beroende på ljudkort och användarens preferenser. Lägre vektorstorlek leder till större beräkningskrav då processorn behöver behandla ett nytt segment oftare. Signalvektorns storlek begränsar den minimala tiden för en förändring i en MAX-patch och därför den minsta möjliga fördröjningstiden för en signal.

Signalvektorstorleken för att uppnå en fördröjning på 100 mikrosekunder kan härledas på följande sätt:

Antal sekunder per sampel = *1/44100*

Antal millisekunder per sampel = *1000/44100 = 0.022675*

[23] http://sv.wikipedia.org/wiki/Sinus [senast läst 2013-05-09]

Den största möjliga vektorstorlek för 100 mikrosekunders fördröjning är därför 4 sampel:

*0.022675*4 ≈ 0.1*

Denna vektorstorlek är väldigt låg och påverkar prestandan negativt.

3.1.8 Mätning för HRTF

Till motorn behövde vi göra HRTF-mätningar motsvarande de olika positionerna runt lyssnaren.

För mer information om HRTF-teknik se kapitel 2.4.3.1.

Vi tänkte oss att vi skulle ha en åhörare och sedan flytta en högtalare runt till olika positioner och med den spela upp vitt brus. Åhörare skulle även ha ett par *MK2*-mikrofoner från *Soundman* placerade i öronen som ett par hörlurar. För att kunna mäta på ett fast avstånd behövde vi en ställning som kunde flyttas runt på olika bestämda positioner, vi hade många olika tankar om hur den skulle kunna se ut och fungera (se bilaga 2). Det resulterade i att vi bestämde oss för att använda en del av ett mikrofonstativ som vi monterade ihop med ett cymbalstativ.

Resultatet blev ett stativ som kunde flyttas och vridas i nästan alla möjliga positioner och vinklar, vilket var väldigt användbart då vi behövde mäta i en sfärisk form runt åhöraren. Lösningen var dock något opraktisk då vi hela tiden fick hålla koll på att vi hade högtalaren på rätt avstånd (1m) från åhöraren och att vi var rakt över punkterna vi hade satt ut på golvet som vi skulle gå efter.

Vi valde att mäta på 1 meters avstånd på totalt 38 positioner runt åhöraren vid 5 olika höjder.

Mätningarna känns i efterhand tillräckliga men man hade kunnat göra det mer högupplöst och mätt på fler positioner både runt åhöraren och på fler höjder (läs mer om detta i bilaga 2)

Detta är något vi ser som en vidare utvecklingsmöjlighet av motorn.

Dessa impulssvar används sedan av *Max/MSP* för att skapa en så kallad FFT (Fast fourier Transform, se Bilaga FFT) filtrering.

Impulssvaret som matas till FFT objektet mixas med hjälp av spelarens relativa vinkel mot ljudkällan i två steg, relativ horisontell vinkel och sedan relativ höjdposition.

Genom att mata två ljudspår som är förskjutna med ***x*** (spelarens relativa vinkel) respektive 360-***x*** grader för respektive höger och vänster öra mot ljudkällan till FFT filtret får spelaren en ljudmix som efterliknar den som uppstår under liknande förhållanden i verkligheten.

3.2 Test

3.2.1 Testsyfte

Anledning till att vi genomförde ett test var delvis att få en uppfattning om hur *BaSE* förhåller sig till en alternativ ljudmotor, i detta fallet *UDK*'s '*Unreal Audio System*', genom att låta testpersonerna testa båda versionerna. Vi ville även testa *BaSE* på flera personer, eftersom ett längre test där UDK-banor måste startas och stängas om och om igen och ljud måste laddas in för varje test utsätter ljudmotorn för för större påfrestning än att bara testa den en gång med en bana. Detta leder till att fel som vi tidigare missat kan uppdagas. Vi ville också höra testarnas åsikter om motorn så att vi kan förbättra den efter den feedback vi får.

Testet gick ut på att deltagarna fick utföra två ljudrelaterade utmaningar, den ena i form av en väg med osynliga bilar som man skulle akta sig för, rumsligheten i detta test var som ett fritt fält (utan efterklang), eftersom vi ville fokusera framförallt på panoreringseffekten i de två olika versionerna (miljön hade heller inga väggar, så ett fritt fält tycktes naturligt).

Fig. 7. Vägen med de osynliga bilarna

Den andra banan var i form av en “ljudjakt”: 8 ljud var placerade i en labyrintisk bana där man skulle hitta alla ljud, ett åt gången.

Fig. 8. Scen ur ljudlabyrinten

Vi mätte tiderna för deltagarna att klara av ljudlabyrinten och ställde frågor om upplevelsen.

Vi försökte utforma frågorna för att de skulle ge oss en bild av skillnaderna mellan ljudmotorerna, ur lokaliseringsprespektiv, d.v.s. hur lätt det är att avgöra ett ljuds position. Vi ställde också frågor om den generella uppfattningen av ljudmotorerna.

3.2.2 Material

Vi lät alla testpersoner spela med samma dator och hörlurar, eftersom vi tror att vilka hörlurar som används kan påverka upplevelsen.

Vi gjorde undersökningen för att jämföra *BaSE* med en redan existerande, *UDK*’s inbyggda ljudmotor.

UDK's ljudmotor, även om den har vissa brister (exempelvis känns panoreringar plötsliga vid hörlurslyssning), används i populära spel, exempelvis *Tribes Ascend* (*Hi-Rez Studios*, 2012).

3.2.3 Testresultat

Vi utförde testet på fem personer, det var så många som vi hann med under dagen, eftersom det tog lång tid att utföra varje test. Tre av de fem av testpersonerna upplevde det lättare, d.v.s. de upplevde en större klarhet i var ljuden befan sig när de navigerade i den versionen som var ljudlagd med *BaSE*.

Den största skillnaden verkade ligga i att panoreringen var plötsligare i '*Unreal Audio System*'; testpersonerna upplevde det som om panoreringen gick för snabbt mellan höger och vänster.

En testperson skrev:

"*i version 1 (Unreal Audio System) så var panoreringarna väldigt hackiga. Man hörde ordentligt när den gick från höger-center-vänster.*" Några av testpersonerna upplevde att *BaSE* fungerade sämre och motiverade det med 'hackiga panoreringar'. De som svarat att detta var ett problem har dock kommenterat att det var det enda som gjorde skillnaden mellan vilken version de föredrog; "*Hackigt som sagt. Däremot lät ljuden i sig bättre, de "rörde" sig mer naturligt, förutom hacket då.*", "*Bara på grund av "hacket" jag beskrev tidigare.*"

I det andra testet valde fem av fem deltagare versionen med *BaSE* som den föredragna versionen.

Motiveringen var i några av fallen att man kände rumsligheten bättre, vilket troligtvis beror att efterklangen inte fungerade som vi ville i '*Unreal Audio System*', vilket gör detta test orättvist och ledde till att vi inte kunde tolka att alla valde denna som föredragna som ett rättvist resultat.

Däremot kommenterade många av deltagarna på saker som "*...i den andra fanns det väldigt mycket skillnader mellan att vara nära/långt ifrån ljudet, och vänster/höger och över/under*" och "(Det) *Går att urskilja vart ljudet är genom att kolla på det*"

3.2.4 Sammanställning

Vi upplever resultaten som positiva, trots en del kritik av vårt system. Anledningen till att vi kan tolka dessa resultat som positiva är att de som klagat på saker i *BaSE* främst klagat på saker som ‘gått snett’ i *Max/MSP*, det hackiga som beskrivs kommer utav att *Max/MSP* hade problem att processa data från *UDK*, detta troligtvis eftersom datorn stått på med flera versioner av både *UDK* och *Max/MSP* under en längre tid. Denna feedback ger oss även information om vilken optimering som behöver göras. Exempelvis ändrade vi vår FFT-filtrering enligt klagomålen på den ‘hackiga panoreringen’ som beskrevs av några testpersoner.

Kommentarerna på den andra delen av testet tyder på att den FFT-filtrering vi använde för att skapa panoreringen märktes av och gav bra inlevelse, och kommentaren “*Går att urskilja vart ljudet är genom att kolla på det*”, tyder på att den funktion som filtrerar och sänker ljudstyrkan på ljud när de befinner sig bakom en vägg också bidrar till inlevelsen.

Så även om testet inte hade så många deltagare fick vi ändå in bra feedback som hjälpte oss hitta buggar i *BaSE*, vilka nu är åtgärdade.

4. Resultat och diskussion

Syftet med denna del av arbetet är att presentera resultatet av arbetet i form av en granskning av den färdiga produkten och dess funktionalitet. Vi kommer att gå igenom det här och även hur arbete med *BaSE* fungerar i praktiken. Vi vill ägna en del till att jämföra de förväntningar vi hade i början av projektet med det slutgiltiga resultatet, inklusive vår frågeställning.

4.1 Resultat

Resultatet av detta arbete har blivit en fungerande ljudmotor: *BaSE*, som kan användas med *UDK* men som dock inte är begränsad till *UDK*; *BaSE* kan användas med alla spelmotorer som har möjlighet att använda UDP trafik.

BaSE har de funktioner som kan förväntas av en modern ljudmotor, så som efterklang som kan ändras beroende på spelminjön och även funktioner som vissa moderna ljudmotorer saknar, som

simulering av ljudskymning när ett objekt befinner sig mellan ljudkälla och lyssnare och efterklang per ljudkälla som kan anpassas beroende på avståndet. Det som gör att *BaSE* står ut mest i mängden ljudmotorer är användningen av FFT filtrering för att simulera en lyssnares position i ljudmiljön.

Arbetsflödet i *BaSE* är anpassat för att enkelt kunna användas av utvecklare som är vana vid *UDK*. Praktiskt sett fungerar användandet av *BaSE* likadant som arbetet i *Unreal Audio System*; unrealklassen som hanterar *BaSE* ljud har ungefär samma egenskaper som en vanlig unrealljudklass: skall ljudet spela, vilken ljudvolym har ljudet, vilken tonhhöjd har ljudet, skall det repetera etc. Det enda användaren behöver göra är att ställa in dessa parametrar som de skulle gjort i *UDK*, och sedan välja ett unikt ljudnamn i *BaSE* (i *Max/MSP*).

4.1.1 Möjlighet till vidareutveckling

Den slutgilitga lösning vi hittat på vår problemformulering löser problemet med binaural syntes, men vilka problem finns i vår lösning?

Det största problemet med vår lösning är att den är beroende av tredjepartsprogram. På ett antal plan är detta inget egentligt problem; de program vi använt finns tillgängliga för gratis distrubuering, d.v.s. om man som spelutvecklare vill använda *BaSE* som ljudmotor till ett spel finns all möjlighet att distribuera spelet utan extra kostnad jämfört med en annan kostnadsfri ljudmotor. Problemet ligger snarare i att göra *BaSE* attraktiv för spelutvecklare. Vi tror att ett system som kräver färre, eller inga externa mjukvaror skulle vara mer attraktivt för spelutvecklare, även om dessa externa mjukvaror inte kräver extra utgifter i form av licenser. Anledningen till detta är den upplevda tid som kommer att gå åt till att lära sig arbeta med ett nytt system snarare än det man är van vid, eller vad man skulle kunna sammanfatta som rädsla för att prova nya oprövade verktyg. Förhoppningsvis stämmer denna oro från vår sida inte överens med hur spelutvecklare resonerar i verkligheten. Det finns exempel på spelutvecklare som kombinerat flera ”färdiga” utvecklingsmiljöer för att uppnå en önskad kvalitet, exempelvis *Bioshock Infinite* (Irrational Games, 2013), där *Unreal Engine* (Epic Games, 2012) använts i kombination med ljudmjukvaran *Wwise* (Audiokinetic, 2012), något som definitivt skulle kunna jämföras med att använda vår ljudmotor i kombination med *UDK*.

4.2 Diskussion

Generellt sett är vi nöjda med resultatet av vårt arbete, vi har lyckats skapa en ljudmotor som uppfyller alla krav på ljudkvalitet och även överstiger i många aspekter de förväntningar vi hade i början av detta projekt.

Men har vi genom arbetet svarat på våra frågeställningar?

Vår frågeställning löd:

Hur utvecklar man en ljudmotor som fungerar efter binaural princip?

Ytterligare tycker vi att det är viktigt att ifrågasätta hur den forskning vi gjorde under *problemområde* relaterade till utvecklingen av *BaSE*; om forskningen inte kan relateras till vår utveckling och våra metoder, till vilken nytta var den, och borde vi gjort annorlunda?

När vi utvecklade *BaSE* stod den forskning vi gjorde under *problemområde* (Kap. 2) som främst till nytta som en referens till tekniker och fenomen vi skulle komma att återskapa i ljudmotorn.

Ett exempel på detta är då vi skulle återskapa ITD (se kap. 2.4.2.1) effekten i vår ljudmotor, utföra ljudmätningar, eller tillämpa FFT.

Vi känner dock att även om allt som all forskning gjord under *problemområde* var relevant till produktionen, skulle produktionsdelen kunna gjorts tydligare för oss själva; målet: att skapa en binaural ljudmotor kan *tyckas* vara enkelt definierat för en ljuddesigner.

Begreppet binauralt görs väldigt tydligt i vår text, men begreppet ljudmotor skulle behövt göras tydligare. Delvis för att en överenskommelse om vad som ingår i begreppet skall vara självklar mellan oss som skriver arbetet och gör produktionen. Definitionen behövs även för att påvisa relevans till vår forskning för några av processerna i vår produktion. Vad vi menar är att en ljudmotor är så mycket större än det binaurala även om det binaurala var i fokus och därmed det viktigaste så gör inte det binaurala en komplett ljudmotor. De begreppen som vi syftar på definieras tydligt i texten men kopplingen mellan dem och ljudmotor skulle behövt göras bättre. Exempelvis så beskrivs efterklangen och dess karaktär i förhållande till subjektiv uppfattning av ett ljuds position, men det kopplas inte till begreppet ljudmotor, trots att det ingår i att skapa en välljudande ljudmotor. Problemet upptäcktes dock aldrig förrän vi kom till vår produktionsdel, trots att vi gång på gång reflekterade kring vår frågeställning och dess relevans till den kommande produktionen.

Vi tror att det här problemet beror på att gränsen mellan vad som ingår i produktionen och vad som ingår i problemområdet inte är så självklar som vi inledningsvis trodde.

Vi tror att det här har berott på att vi valde att fokusera på de fenomen kring mänsklig hörsel vi skulle återskapa snarare än någonting rörande programvara och praktiskt genomförande.

Vi anser att vi genom produktionen av BaSE svarat på vår frågeställning och att baserat vår produktion på den forskning vi gjort i *problemområde*.

Något som vi i efterhand upplevde kunde ha varit bättre är vår tidsestimering och planering av arbetet. Det här var mestadels ett problem under *problemområde*, eftersom att produktionsdelen var en iterativ process, och därför inte behövde en sträng planering. Vi använde oss av deadlines på vissa delar som t.ex att impulsmätningarna skulle utföras vid ett visst tillfälle och att ett test av motorn skulle gå att utföra vid ett visst tillfälle men i resten av arbetet jobbade vi från dag till dag. Detta gjorde att det var svårt att få en överblick av arbetet som helhet.

5. Ordlista

Här finns återigen definitionerna som vi gett i fotnoterna.

Psykoakustik - Vetenskapen om människans subjektiva uppfattning av ljud.

Resonant frekvens - De flesta objekt har en inneboende resonant frekvens, om ett objekt blir utsatt för denna frekvens kommer det i medvibration.

Vitt brus - En ljudsignal som innehåller alla möjliga frekvenser och har en likvärdig energi för alla frekvensband.

Oriktad ljudkälla - Ljudkälla som ger lika ljudtryck i alla riktningar.

Phon - Skala för ljudintensitet baserad på mänsklig ljuduppfattning.

Klippning - Klippning uppstår när en signal är starkare än ett medium kan återge. Vid klippning uppstår övertoner eftersom sinustonens rundade form inte får plats. Detta leder till ofta oönskade övertoner, så kallad distorsion.

Linjär interpolering - Att anta ett värde för en punkt genom att utgå från andra kända värden. Linjär interpolering innebär att mellan varje känd punkt dra en rak linje.

Sinuston - En ren ton, tar sitt namn från sinus som beskriver avståndet i höjdled från centrum av en cirkel med radien 1 till en punkt på dess kant för en given vinkel. Detta ger sinustonen en rundad form. Frekvensen på sinustonen bestämmer hur ofta cirkeln varvas.

Resonant frekvens - De flesta objekt har en inneboende resonant frekvens, om ett objekt blir utsatt för denna frekvens kommer det i medvibration.

Psykoakustisk - Vetenskapen om människans subjektiva uppfattning av ljud.

Spatialiseras - Att rumsligt placera en ljudeffekt.

Signalbehandling - Signalbehandling innebär att manipulera signaler med hjälp av matematiska metoder.

Objektorienterad - Objektorienterad programmering innebär en programmeringsmetod där ett program kan innehålla ett antal objekt som interagerar med varandra.

Hz - Hertz, en enhet för att beskriva frekvens, Hz räknas per sekund.

Dopplereffekt – Dopplereffekt (inom ljud) är ett fenomen som uppstår då en ljudkälla närmar eller avlägsnar sig från en åhörare. Dopplereffekten påverkar ljudkällans tonhöjd; om ljudkällan närmar sig blir den högre men om källan avlägsnar sig blir den lägre.

6. Källförteckning

Algazy, V.R. Duda, R. O. Thompson, D. M. Avendano C. (2001) *THE CIPIC HRTF DATABASE* [PDF]
Tillgänglig via: http://interface.cipic.ucdavis.edu/data/doc/CIPIC_HRTF_Database.pdf

Blauert J. (1997) *Spatial Hearing Revised Edition.* MIT press

Brieger, S. and Göthner, F. (2011) *Spelares inställning till HRTF-teknik i FPS-spel*. Stockholm.

Davis, K A. (2005) *Contralateral Effects and Binaural Interactions in Dorsal Cochlear Nucleus.* JARO nummer 6. University of Rochester USA

Duraiswami, R. (n.d.) *Introduction to HRTFs*. [PDF] p.33. Tillgänglig via: http://www.umiacs.umd.edu/~ramani/cmsc828d_audio/HRTF_INTRO.pdf [Läst: 2013-02-19].

Everest, F A. Pohlman, K C. (2009) *Master Handbook of Acoustics.* 5 uppl. McGraw Hill

Hofman, P M. Van Rickwick, J G.A. Van Opstal A.J. (1998) *Nature neuroscience Volume 1 no 5*, [online] Tillgänglig på: http://www.mbfys.ru.nl/~johnvo/papers/nn98.pdf [Läst: 2013-02-05].

Lee, K. Lee, S. (2010) A Real-time Audio System for Adjusting the Sweet Spot to the Listener's Position från *IEEE Transactions on Consumer Electronics vol.56 nummer 2.*

La Fata, T. (2000) *Writing Max Externals in Java V 0.3* (PDF) Tillgänglig via: http://blog.lib.umn.edu/geers001/compmusprog2007/WritingMaxExternalsInJava.pdf [Läst 2013-05-10]

Moore, B. C. J. (2008) *An introduction to the psychology of hearing*. Bingley, Emerald.

Rapture 3D (2009) [Mjukvara]. USA: Blue Ripple Audio

Rumsey, F. McCormick, T. (2002) *Sound and recording: an introduction*. 4 uppl. Focal Press.

Tribes Ascend (2012) [Digitalt Spel]. Exekutiv Producent: Todd Harris. USA: Hi-Rez Studios

Unreal Tournament 3 (2007) [Digitalt Spel]. Producent: Jerry Huber. USA: Epic Games

Wallach, H. Newman, E B. Rosenweig M R. (1949) *The Precedence Effect in Sound Localization* från *The American Journal of Psychology Volym 62, Nummer 3*

Bilaga 1. FFT

(Fast Fourier Transform/ Snabb Fouriertransform)

Denna bilaga kommer inte att ge en komplett bild av snabb fouriertransform eftersom detta är ett komplicerat begrepp, bilagan syftar istället till att ge en mycket ytlig överblick av FFT samt visa hur FFT användes i vårt arbete.

FFT är ett begrepp som inom ljudvärlden mest används när man syftar på filtrering. All FFT är en typ av filtrering, men hur det används kan skilja sig mycket stort. FFT är en matematisk teknik som går ut på att en signal, i detta fallet en ljudsignal tas från tidsdomänen, som är det vanligaste sättet att se ljudsignaler, till frekvensdomänen, där man istället för tid och amplitud, har frekvens och amplitud som grundegenskaper.

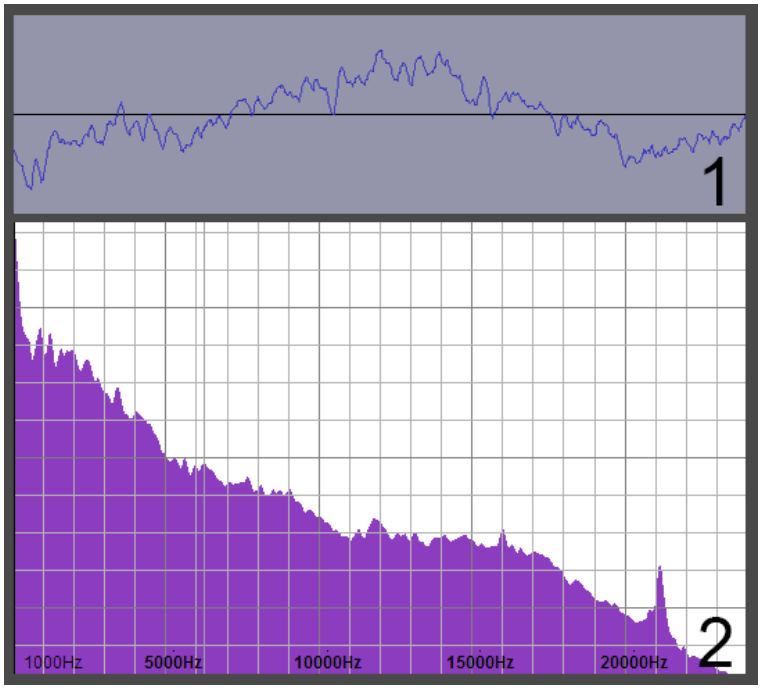


Fig. 1. Visualisering av en ljudvåg i tidsdomän (1) respektive frekvensdomän (2).

Genom att flytta ljudet från tidsdomänen till frekvensdomänen kan man göra operationer som inte är möjliga i tidsspektrat. I vårt fall filtrerade vi ljudfiler med hjälp av impulser genom att multiplicera de två signalernas amplitud för de olika frekvensbanden, denna operation kallas faltning.

Exemepel på FFT i *Max/MSP*[24]

Detta är är vår slutgiltiga FFT patch

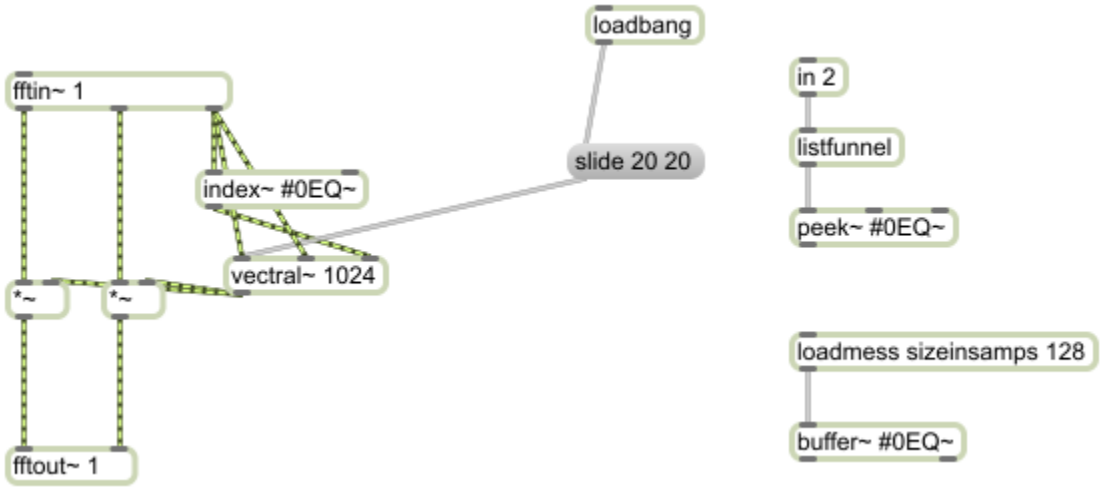
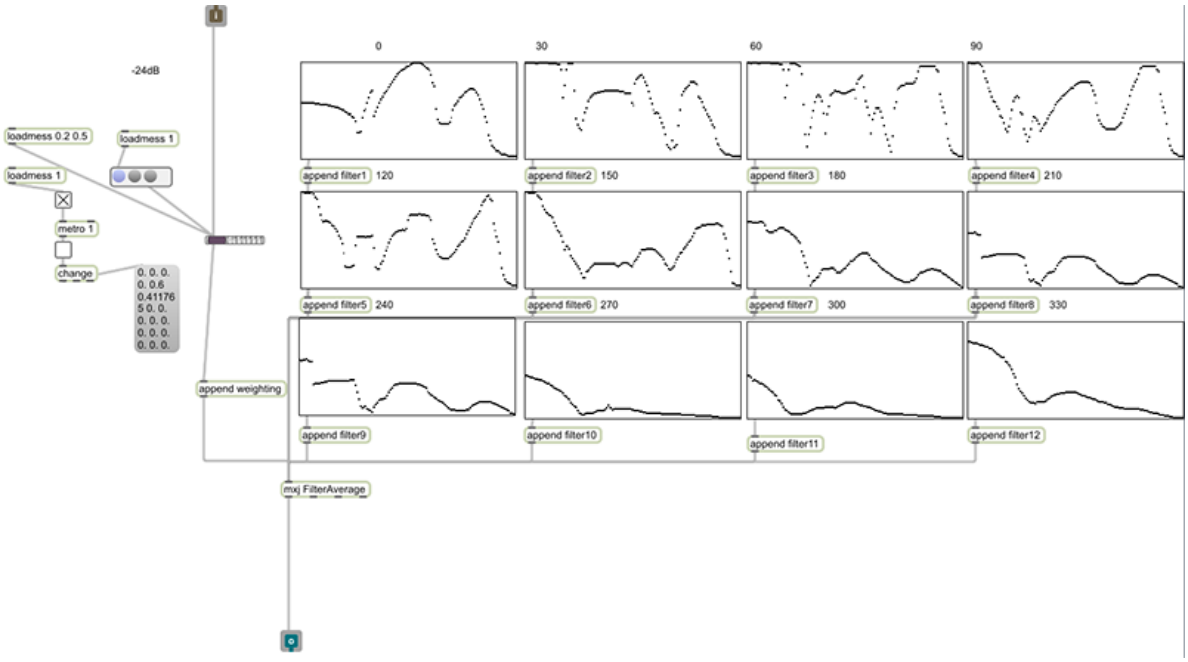


Fig. 1. FFT patch

Vectral objektet i patchen är en optimeringsåtgärd, den skapar en buffer av de föregående fft fönstrena, och skickar ut genomsnittet av dessa. Genom att använda vectral objektet kan man sänka updateringsfrekvensen på den data som skickas in i FFT objektet utan att skapa knaster eller andra missljud.



[24] http://www.youtube.com/watch?v=69A1kGNFYIc [Senast läst 2013-05-10]

FIG. 2: Kurvor som representerar de olika vinklarna

Bilden ovan är ett exempel på de filterkurvor (Graferna) som skickas in i FFT objektet.
Som en optimeringsåtgärd valde vi att inte använda impulssvaren vi spelat in, utan ‘översätta’ dessa till kurvor med en upplösning på 128 steg. Anledningen till detta är att en lösning med impulser måste hantera 74 (12*3+2 för varje öra) ljudsignaler för varje ljudfi som skall ‘binauraliseras’. Med detta systemet kan man åstadkomma samma sak men med filter som ger ett vägt genomsnitt istället. Detta genomsnitt updateras bara en gång var 50’e millisekund.

Bilaga 2. Impulsmätningar.

En stor del av motorn bygger på HRTF- impulser. De spelas in igenom att spela upp vitt brus ur en eller flera högtalare, dock bara en åt gången, som är positionerade runt en åhörare. Åhöraren kan vara i form av en människa med mikrofoner placerade i öronen eller ett så kallat *"dummyhead"*. De här impulserna används sedan i *Max/MSP* där för att filtrera de ljud som skall spelas upp.

Ett ljud som skall spelas upp med känslan att det kommer snett bakifrån till höger t.e.x. filtreras igenom den impulsen som är inspelad vid den positionen. Det här gör att ljudets spektrum påverkas på samma sätt som det hade gjort i verkligheten.

För att kunna göra en sådan inspelning behövde vi en ställning som vi kunde montera en eller flera högtalare på, som sedan kunde flyttas runt till olika positioner runt åhöraren.

Från början tänkte vi oss att ställningen skulle se ut såhär:

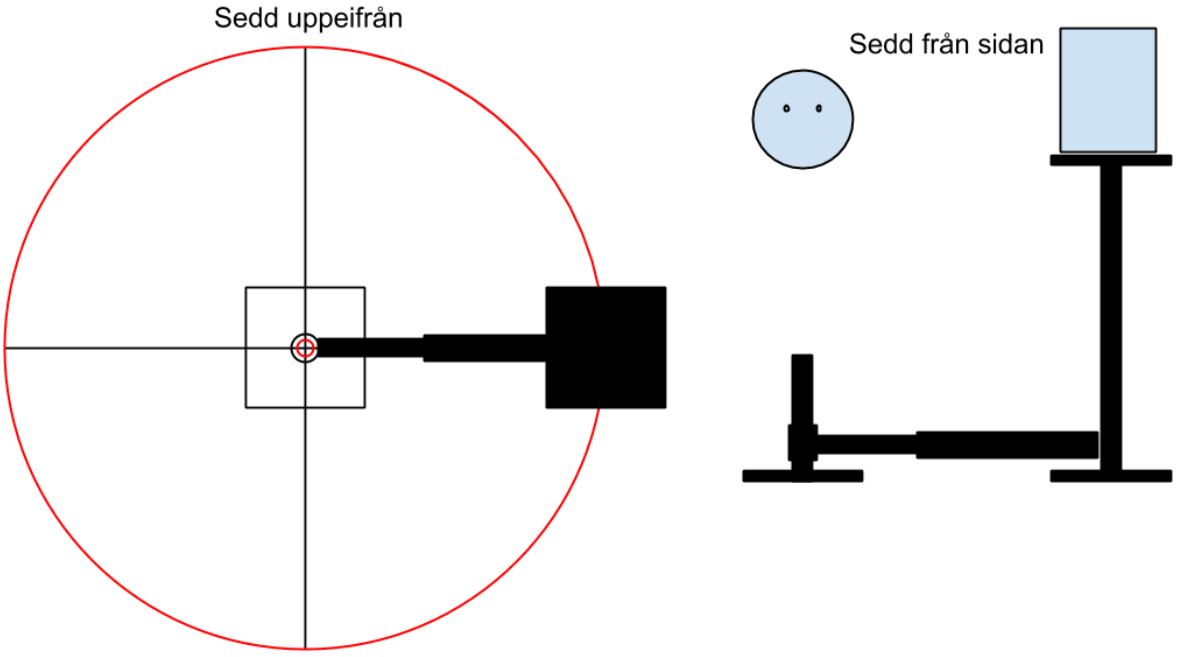


Fig. 1. Tidig idé till ställning (författares illustration)

Vi kom dock rätt snabbt fram till att den här ställningen inte skulle fungera då den inte går att ställa in på alla de positioner vi behöver spela in impulser på. Den går alltså enbart att flytta runt åhöraren i det horisontella planet och det går därför inte att spela in impulser snett uppifrån, uppifrån, underifrån och snett underifrån med den. De här möjligheterna krävs då impulser

behöver spela in i sfärisk form runt åhöraren så att ljud från alla möjliga positioner kan spelas upp och filtreras på rätt sätt i *Max/MSP*.

Vi hittade senare *THE CIPIC HRTF DATABASE* (V. R. Algazi, 2001) och använde detta dokument som grund för våra mätningar. I dokumentet finns t.e.x. information om antalet mätpunkter och det avstånd de mätte på (V. R. Algazi, 2001, s1). Den ställning de använde sig av var i form av en halvcirkel med flera högtalare monterade likt Fig 2.

Fig. 2. Inspelningsställning (Duraiswami s.11)

Då vi varken hade tillgång till en sådan ställning eller möjlighet att bygga en p.g.a. både praktiska och ekonomiska skäl fick vi komma på en annan lösning. Det hela resulterade i att vi använde oss av ett mikrofonstativ som vi monterade ihop med ett cymbalstativ. Denna rigg var väldigt användbar då den kunde vridas i alla möjliga positioner och som vi tidigare nämnt är detta viktigt då vi behövde spela in impulser från många olika vinklar i en sfärisk form.
Vi valde att spela in impulserna på 1m avstånd precis som det beskrivits i *THE CIPIC HRTF DATABASE*. Vi kunde inte spela in på lika många punkter som *CIPIC* beskriver utan valde att enbart spela in på 38. *CIPIC* beskriver en inspelning av 1250 punkter per öra.

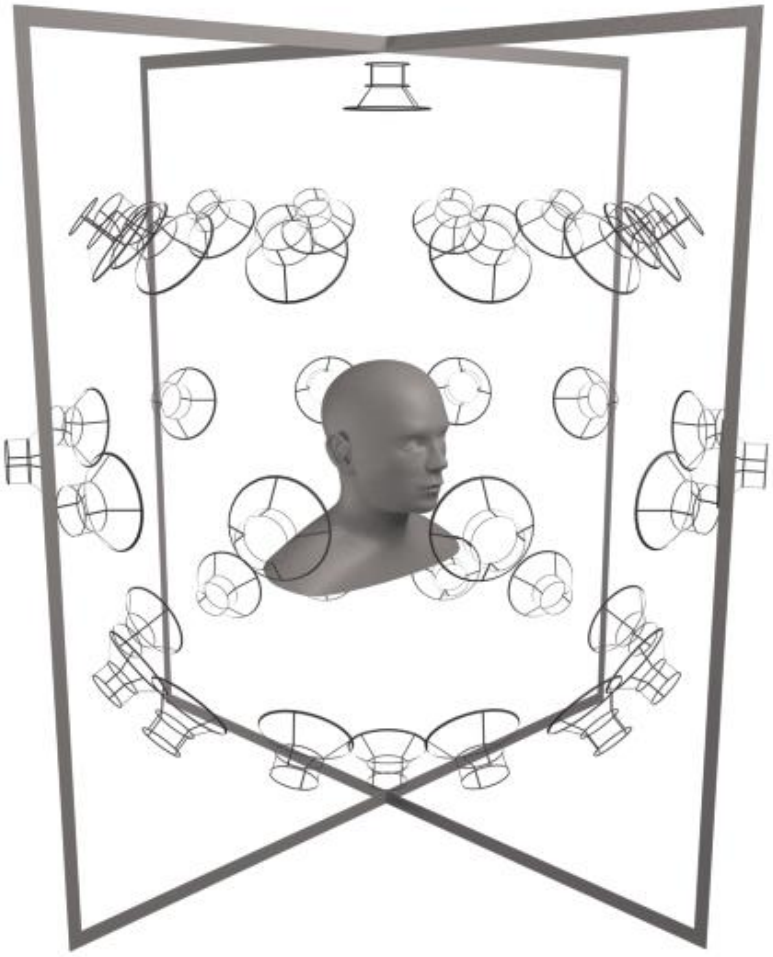


Fig. 3. Visualisering av inspelningspunkter (Författares illustration)

Anledningen till att vi valde att spela in på 38 punkter (se Fig 3) var delvis att vår ställning inte höll hög nog precision att göra många fler än så, dels för att vi vid fler punkter skulle belasta hårdvara oerhört i Max/MSP men också för att vi enbart ville testa principerna för binauralt ljud och därför ansåg att precision på 38 punkter räckte för att göra det. I efterhand upplevde vi att antalet punkter faktiskt var tillräckliga och effekten vi strävade efter fungerar som den skulle. Då vi inte hade något att jämföra med vet vi inte hur stor skillnaden är, en vidareutvecklingsmöjlighet är att mäta på fler positioner för att få större precision.
Fler mätningar skulle dock behöva annan mätutrustning exempelvis enklare rigg med flera högtalare och ett sätt att se till att åhörare sitter precis likadant varje mätning.
Nedan följer en bild från inspelningen

Fig. 4. Bild från impulsinspelningarna