

Masterproef Sound Quality

Studiegebied

Industriële wetenschappen en technologie

Opleiding

Master in de industriële wetenschappen:
elektromechanica

Afstudeerrichting

Elektromechanica

Academiejaar

2007-2008

Ward Jonckheere

Voorwoord

Deze masterproef is geschreven als een vereiste voor het verkrijgen van de graad van Master in de Industriële Wetenschappen: Elektromechanica. Ze is niet enkel een test van de technisch-wetenschappelijke bekwaamheid, maar ook van secundaire competenties zoals het doorzettingsvermogen. Ik ben dan ook de nodige dank verschuldigd aan dhr. Bruno Vanslambrouck, mijn promotor, die altijd de tijd nam en gaf om de voortgang te bespreken en in alle fasen van dit jaarwerk goede ideeën aanreikte.

Verder wil ik iedereen bedanken die op een of andere manier een aandeel in deze masterproef gehad heeft, alsook iedereen die geïnteresseerd mijn wedervaren in deze gevolgd heeft. Ik wil hier ook mijn ouders vernoemen, die begripvol met mijn sterk verminderde beschikbaarheid dit jaar omgingen.

Tot slot zou ik de lezer met expertise willen oproepen tot medewerking aan de online kennisdeling. De bereidwilligen die gedegen thematische, encyclopedische of educatieve websites opzetten en onderhouden of hun artikels vrij ter beschikking stellen, maken kennisvergarig drempelloos. In pre-internettijden zou deze eerder academisch gerichte masterproef nooit op een dergelijke mondiale referentielijst gestoeld geweest zijn. Die vele bronnen hebben het werk niet verlicht, maar de scriptie wel completer gemaakt.

Ward Jonckheere
Mei 2008

Inhoudsopgave

Voorwoord	ii
Inhoudsopgave	iii
Lijst van figuren	vi
Lijst van tabellen	viii
Lijst van symbolen	ix
1 INLEIDING.....	1
1.1 Situering van het onderwerp	1
1.2 Doel van de masterproef	3
2 AKOESTIEK EN PSYCHOAKOESTIEK.....	5
2.1 Inleiding	5
2.2 Grootheden van geluid	5
2.3 Bandenspectra in het frequentiedomein	7
2.4 Het menselijk auditief systeem	9
2.5 Maskering.....	11
2.5.1 <i>Spectrale maskering</i>	11
2.5.2 <i>Maskering in de tijd</i>	12
2.6 Toonhoogte.....	12
2.6.1 <i>Zuivere tonen</i>	12
2.6.2 <i>Complexe tonen</i>	13
2.7 Kritische banden.....	14
3 SOUND QUALITY PARAMETERS.....	16
3.1 Inleiding	16
3.2 Luidheid	17
3.2.1 <i>Isofonen</i>	17
3.2.2 <i>Wegingscurven</i>	18
3.2.3 <i>Bepaling van luidheid</i>	19
3.2.3.1 Luidheidsbepaling volgens Stevens	19
3.2.3.2 Luidheidsbepaling volgens Zwicker	21
3.3 Scherppte.....	22
3.4 Fluctuatiesterkte	23
3.5 Ruwheid	25
3.6 Tonaalheid.....	26
3.7 Pleasantness en annoyance.....	27

3.8	Productspecifieke parameters.....	29
4	BEOORDELING VAN SOUND QUALITY.....	31
4.1	Inleiding	31
4.2	Menselijke perceptie	32
4.3	Meetuitrusting	32
4.3.1	<i>Artificieel hoofd</i>	33
4.3.2	<i>Microfoon</i>	34
4.3.3	<i>Meetomgeving</i>	35
4.4	Testjury.....	35
4.5	Software	37
4.6	Enkele toepassingen	38
4.6.1	<i>Akoestisch comfort in een personenwagen</i>	38
4.6.2	<i>Koelkastlawaai in verschillende omgevingen</i>	40
4.6.3	<i>Geluidshinder van windmolens</i>	42
4.6.4	<i>Auditieve informatieoverdracht</i>	43
5	ONTWIKKELING VAN EEN COMPUTERPROGRAMMA.....	46
5.1	Inleiding	46
5.2	Captatie van het geluid	47
5.2.1	<i>Captatie van de geluidsdruk</i>	47
5.2.2	<i>Conversie van het analoge signaal</i>	47
5.2.2.1	Theoretisch	47
5.2.2.2	Praktisch	48
5.3	Verwerking van het geluidssignaal	49
5.3.1	<i>Kalibratie van het geluidssignaal</i>	49
5.3.1.1	Theoretisch	49
5.3.1.2	Praktisch	51
5.3.2	<i>Filtering van het geluidssignaal</i>	51
5.3.2.1	Berekening van het vermogenspectrum.....	52
5.3.2.2	Conversie naar decibel	53
5.3.2.3	Aanmaak en implementatie van de tertsbandfilters	53
5.4	Luidheid	54
5.4.1	<i>Codering</i>	54
5.4.2	<i>Verificatie</i>	57
5.5	Scherpte.....	57
5.5.1	<i>Codering</i>	57
5.5.2	<i>Verificatie</i>	59
5.6	Fluctuatiesterkte	61
5.6.1	<i>Codering</i>	61
5.6.1.1	Modulatiefrequentie	61
5.6.1.2	Maskeringsdiepte	61
5.6.1.3	Fluctuatiesterkte	63

5.6.2	<i>Verificatie</i>	64
5.7	Ruwheid	68
5.7.1	<i>Codering</i>	68
5.7.2	<i>Verificatie</i>	68
5.8	Tonaalheid	69
5.8.1	<i>Codering</i>	69
5.8.2	<i>Verificatie</i>	77
5.9	Pleasantness & annoyance	78
5.10	Hoofdprogramma	79
6	APPLICATIE VAN HET SQEP	83
6.1	Inleiding	83
6.2	Praktische applicaties	83
6.2.1	<i>Dampkappen</i>	83
6.2.2	<i>Haakse slijpmachines</i>	87
6.2.2.1	Luidheid	89
6.2.2.2	Scherpte	90
6.2.2.3	Tonaalheid	91
6.2.3	<i>Compressor</i>	94
6.2.4	<i>Ventilatoren</i>	96
6.3	Mogelijkheden in onderwijscontext	100
6.4	Mogelijkheden bij de lokale industrie	101
7	CONCLUSIE.....	102
7.1	Conclusie	102
7.2	Ideeën voor de toekomst	103
7.2.1	<i>Sound Quality algemeen</i>	103
7.2.2	<i>Sound Quality in de opleiding</i>	104
	BIBLIOGRAFIE	106
	BIJLAGE A: GENERATIE VAN TONEN EN RUIS IN MATLAB.....	114
	BIJLAGE B: HANDLEIDING BIJ HET SQEP	117
	BIJLAGE C: MATLAB CODES VAN HET SQEP	125
	BIJLAGE D: DIAPRESENTATIE.....	157

Lijst van figuren

Figuur 1.1: De te volgen sound quality cyclus om een onderbouwde productaanpassing te kunnen doen.	3
Figuur 2.1: Een zuivere toon is een perfecte sinus en heeft juist één frequentie.	5
Figuur 2.2 De tekeningen geven een indruk van getalswaarden in Pa en dB.	6
Figuur 2.3: Complexe geluiden zijn samengesteld uit vele frequenties.	7
Figuur 2.4: De octaaf- en tertsbandsfilter sluiten aan zodat het volledige spectrum wordt bedekt.	8
Figuur 2.5: Macroscopische tekening van het menselijk oor.	9
Figuur 2.6: schematische voorstelling van het menselijk auditief systeem.	9
Figuur 2.7: Groepje haarcellen dat bestaat uit drie rijen haarcellen.	10
Figuur 2.8: Het menselijk waarneembaar geluidsgebied.	10
Figuur 2.9: Het maskerings-effect bij smalbandlawaai.	11
Figuur 2.10: Schematische tekening van tijdsgebonden maskering. Bemerkt de verschillende tijdsorigines.	12
Figuur 2.11: Post-maskering bij verschillende maskeerniveaus.	12
Figuur 2.12: De volle lijn toont dat de verhouding pitch - frequentie enkel bij lage frequenties lineair is. De onderbroken lijn toont het logaritmisch verband tussen beiden bij hoge frequenties.	13
Figuur 2.13: Schematische voorstelling van een ontrold slakkenhuis met maataanduiding.	14
Figuur 3.1: Doel en methodes van de objectieve akoestiek, de psychoakoestiek en de subjectieve akoestiek.	16
Figuur 3.2: De isofonen voor zuivere tonen.	17
Figuur 3.3: Het A-wegingsniveau vergeleken met de geïnverteerde en op nul genormaliseerde 40dB-isofoon.	18
Figuur 3.4: Partiële luidheid volgens Stevens Mark VI.	20
Figuur 3.5: Een ingevulde grafiek ter berekening van de totale luidheid in een diffuus veld (fabrieksgeluid).	22
Figuur 3.6: De scherpte van smalband-, hoogdoorlaat- en laagdoorlaatgeluid.	23
Figuur 3.7: Een voorbeeld van een amplitudegemoduleerde (links) en een frequentiegemoduleerde sinus.	24
Figuur 3.8: De maskeringsdiepte is een belangrijke parameter bij de berekening van de fluctuatierisico.	24
Figuur 3.9: De relatie van de perceptieve ruwheid tot de modulatiefrequentie, -diepte en geluidsniveau.	25
Figuur 3.10: Tonaalheidsbeoordeling door vijf personen van een harmonisch tonencomplex met als grondfrequentie 700 Hz en SNR ratio 11 dB.	27
Figuur 3.11: Onderzoek naar de verschillen in de waardering van het geluid van een wagenportier bij Duitsers en Japanners aan de hand van productspecifieke parameters.	30
Figuur 4.1: Geluidsoptname van een harde schijf m.b.v. een artificieel hoofd.	33

Figuur 4.2: Een artificieel hoofd en torso geplaatst in een anechoïsche kamer.....	35
Figuur 4.3: Vragen gebruikt om de perceptie van golfers te kwantificeren.....	36
Figuur 4.4: Schematisch overzicht van de onderzoeksprocedures.....	39
Figuur 4.5: Grondplan van de flat met weergave van de meetposities.....	40
Figuur 4.6: Het verschil in de A-gewogen SPL van een koelkast in een flat en een anechoïsche kamer, is bij bepaalde frequenties erg groot.....	41
Figuur 4.7: Gemiddelde waarden voor de <i>lapping</i> , <i>swishing</i> en <i>whistling</i> van de verschillende wind turbine geluiden. De drie eigenschappen zijn in de opgesomde volgorde weergegeven.....	43
Figuur 4.8: Strategieën om met de verwarring van psychoakoestische parameters in auditieve displays om te gaan.....	44
Figuur 5.1: Matlab is zowat de standaard geworden wat het technisch programmeren betreft.....	46
Figuur 5.2: De nauwkeurigheid van het signaal na discretisering hangt af van de bemonsteringsfrequentie.....	48
Figuur 5.3: De effectieve of RMS waarde van een zuivere sinus kan gebruikt worden om het signaal te kalibreren.....	50
Figuur 5.4: De Hann functie is een veelgebruikte manier om te 'vensteren'.....	53
Figuur 5.5: De karakteristieken van een 4de en een 10de orde Butterworth filter.....	54
Figuur 5.6: Een automatische berekening van de luidheid van een dampkap in de drie standen.....	57
Figuur 5.7: Het verloop van de wegingsfunctie $g(z)$	58
Figuur 5.8: Grafische afleiding van de scherpte op vier punten.....	59
Figuur 5.9: Grafieken van de scherptes per bandwaarde van de vier bemeten smalbanden.....	60
Figuur 5.10: Een modulatieperiode wordt in vier gelijke stukken gesplitst.....	62
Figuur 5.11: De maskeerdiepte in de tien Bark band zou bijna vier maal groter zijn als bij negen Bark.....	64
Figuur 5.12: Weergave op dezelfde schaal van de maximale en minimale specifieke luidheid van het kalibratiesignaal.....	65
Figuur 5.13: Het kalibratiesignaal en zijn fluctuatiersterkte van 1 vacil.....	66
Figuur 5.14: Het data flow diagram van het programma om tonale componenten te berekenen.....	70
Figuur 5.15: Visualisatie van de tonaalheidsberekeningen voor een samengesteld signaal.....	78
Figuur 5.16: Schema van het hoofdprogramma.....	80
Figuur 5.17: Het eerste deel van de resultaten van het Sound Quality Evaluation Program.....	81
Figuur 6.1: De testopstelling bij Novy.....	84
Figuur 6.2: Het luidheidsspectrum van de dampkap met actieve koolfilter, op de vier verschillende afzuigsnelheden.....	85
Figuur 6.3: Vergelijking van het luidheidsspectrum van de drie dampkapconfiguraties in de power stand.....	86
Figuur 6.4: Geluidspname van de onbelast draaiende kleine slijpmachine.....	87
Figuur 6.5: Geluidscaptatie van de grote slijpmachine in gebruik.....	88
Figuur 6.6: De luidheidscurves van de slijpmachines.....	89

Figuur 6.7: De scherptecurves van de slijpmachines.	90
Figuur 6.8: De tonale componenten van de grote haakse slijpmachine in onbelaste werking.	91
Figuur 6.9: De tonale componenten van de grote haakse slijpmachine in belaste werking.	92
Figuur 6.10: Het frequentiespectrum van de grote slijpmachine in belaste toestand.	93
Figuur 6.11: De bemeten éencilinder compressor.	94
Figuur 6.12: Afbeelding in het tijdsdomein van het compressorgeluid ter bepaling van de modulatieperiode.	95
Figuur 6.13: Deel 1 van de resultaten van de berekeningen op het compressorgeluid.	96
Figuur 6.14: De testopstelling met de tien schoepen tellende ventilator.	97
Figuur 6.15: De A-gewogen geluidsniveaus van de ventilatoren voor de verschillende ventilatorsnelheden.	98
Figuur 6.16: Vergelijking van de ventilatoren, rekening houdende met tonaalheid en geluidsterkte.	99

Lijst van tabellen

Tabel 2.1: De genormaliseerde frequenties van de banden [Hz].	8
Tabel 2.2: De 24 kritische banden met hun frequenties.	15
Tabel 3.1: Engelse adjectieven voor geluidsevaluatie.	29
Tabel 4.1: Commerciële sound quality softwarepakketten en hun leveranciers.	38
Tabel 5.1: Verificatie van de sharpness code.	60
Tabel 6.1: De resultaten van de meting op de Head'line wandkap.	84
Tabel 6.2: De technische kenmerken van de twee slijpmachines.	88
Tabel 6.3: De stationaire luidheid van de haakse slijpers.	89
Tabel 6.4: De scherpte van de haakse slijpers.	90
Tabel 6.5: Resultaten van de ventilatortesten.	98
Tabel 6.6: Procentuele resultaten van de ventilatortesten.	99

Lijst van symbolen

Voor deze lijst is gekozen voor een atypische beperkte weergave in tabelvorm. Dit om de nodige verbanden zichtbaar te maken en het overzicht beter te behouden.

Grootheid NL	Grootheid EN	Grootheid afkorting	Eenheid	Eenheid afkorting
druk	pressure	p	pascal	Pa
geluidsdrukniveau	sound pressure level	L_p	(deci)bel	dB
frequentie	frequency	f	hertz	Hz
tijd	time	t	seconde	s
bandbreedte	bandwidth	z	Bark	Bark
toonhoogte	pitch	Z	mel	mel
luidheidsniveau	loudness level	L_N	foon	P
luidheid	loudness	N	soon	S
scherpte	sharpness	S	acum	acum
fluctuatiersterkte	fluctuation strength	F	vacil	vacil
ruwheid	roughness	R	asper	asper
tonaalheid	tonalness	T	tonality unit	tu

1 Inleiding

1.1 Situering van het onderwerp

"The most beautiful sound in the world is silence."

R.S. Hichens

Maar, om de contradictio in terminis verder te zetten: is stilte ook het meest aangename geluid? Bestaat 's werelds meest aangename geluid überhaupt wel? Op beide voorgaande vragen kan negatief geantwoord worden; zo simpel zit de zaak niet in elkaar. Een geluid wordt als aangenaam ervaren in functie van het object dat het geluid voortbrengt. Omdat dit een factor is die bepalend kan zijn voor de verkoopbaarheid van het product in kwestie, wordt er in de industrie intensief op gewerkt. De Sound Quality van een product wordt onderzocht en verbeterd teneinde zijn geluid van storend naar aangenaam te brengen, zonder de associatie ervan met het product te verliezen. Het geluid van een product geeft namelijk informatie over het functioneren ervan als een deel van de communicatie tussen het product en zijn gebruiker.

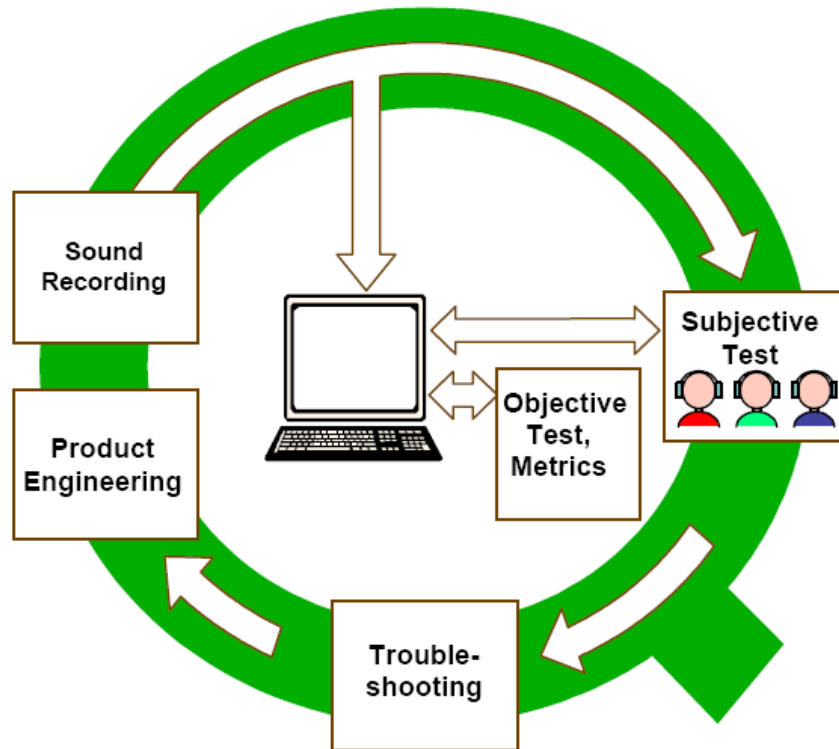
Bij het industrieel ontwerp van producten kunnen drie doelen onderscheiden worden. Het eerste is functionalisme, wat draait rond het bedenken van het product en maken dat het werkt. Het tweede, esthetiek, gaat over de vorm en verschijning van een product en de derde, humane wetenschap (*HFE, Human Factors Engineering*), houdt de communicatie met en de relatie tussen het product en zijn gebruiker in. De drie doelen worden chronologisch in rekening gebracht bij de evolutie van het ontwerp van een product omdat de markt de eisen eveneens chronologisch hoger legt. Typevoorbeeld hierbij is de wagen. Het wordt voor de automobielconstructeurs alsmear moeilijker zich als merk te differentiëren van de concurrentie. Tegenwoordig rijden namelijk alle wagens goed en zien ze er alle goed uit. Het verschil moet gemaakt worden in het derde ontwerpdoel. Het geluid van het product is daar een onderdeel van. Het hoeft dan ook niet te verwonderen dat de automobiellndustrie de eerste was die sound quality testen deed en dit op wagenportieren. Een blikken geluid bij het dichtslaan ervan roept negatieve associaties op bij de gebruiker. Dat is dodelijk in de showroom, waar de wagenkeuze van dergelijke ogenschijnlijke details afhangt.

Product Sound Quality evaluatie is dan ook het sterkst ingeburgerd in de transportmiddelensector (automobiel, motorfiets, trein, vliegtuig, helikopter...), in het ontwerpproces van huishoudtoestellen (wasmachines, mixers, stofzuigers, koelkasten...) en in kantoorapparatuur (airco, computer, printer...). Ook ontwerpers van onderdelen van bovenstaande producten moeten met het aspect van sound quality rekening houden. Dit neemt niet weg dat de aandacht voor sound quality toeneemt in andere sectoren maar ook in andere domeinen zoals architectuur, milieu, muziek en spraak, alwaar de term Speech Quality gebruikt wordt. Het sound quality onderzoeksdomein zelf is hoofdzakelijk opgebouwd uit de

disciplines trillingen, akoestiek, psychoakoestiek, fysiologie, psychologie, musicologie, communicatieleer en signaalverwerking.

De ergerlijkheid (*annoyance*) en aangenaamheid (*pleasantness*) van een geluid zijn twee objectief haast onmogelijk meetbare beoordelingscriteria. Net als bij de andere zintuigen verschillen ook op auditief vlak smaken van mens tot mens en van tijd tot tijd. Dit stelt een productontwerper voor enorme uitdagingen, met als risico een onverkoopbaar product. Aan de andere kant is er –bij bestaande producten– altijd een verwachtingspatroon wat het geluid betreft, denk bv. aan Harley-Davidson, dat zijn karakteristieke uitlaatgeluid als troef uitspeelt en er bij het ontwikkelen van een nieuw motormodel over waakt dat dit geenszins verandert. Prototypes van wagens waarbij de bestuurder geen enkel geluid waarneemt, worden niet gecommmercialiseerd omdat zaken als de staat van de weg en de rijnsnelheid ook auditief ingeschat worden. Het summum is hier, in tegenstelling tot de boodschap van het citaat bovenaan, niet de absolute stilte, maar een goede sound quality.

Bekende meetmethodes zoals de A-weging voor geluid zijn dan ook nutteloos bij sound quality bepaling. Ze zijn enkel op geluidsreductie gericht. Drie verschillende haardrogers kunnen hetzelfde A-gewogen geluidsniveau hebben, terwijl ze een sterk verschillend geluid produceren. Daarom moeten die geluiden op een andere manier beoordeeld en gewaardeerd worden. Daarbij kan een panel van luisteraars ingeschakeld worden (subjectieve evaluatie) of kan er beroep gedaan worden op computerberekeningen (objectieve evaluatie). Beiden hebben hun voor- en nadelen maar worden complementair aangewend. Enkel als de objectieve berekeningen resultaten gelijkaardig aan de (gemiddelde) menselijke waarderingen geven, kunnen ze deze vervangen. De berekeningen gebeuren op verschillende parameters (*metrics*). Algemeen zijn daarbij echter enkel de psychoakoestische parameters bruikbaar, omdat andere te productspecifiek zijn. Op Figuur 1.1, waar de sound quality cyclus is weergegeven, is te zien dat op basis van de subjectieve en objectieve bevindingen het probleem bij het product gezocht wordt en een aanpassing gebeurt. Dit kan door aanpassing van de trillingen aan de bron of aan de behuizing ervan. Het vinden van de link tussen een bepaald aspect van een geluid en de door te voeren constructieaanpassing is geen eenvoudige opdracht voor productontwerpde ingenieurs.



Figuur 1.1: De te volgen sound quality cyclus om een onderbouwde productaanpassing te kunnen doen.

1.2 Doel van de masterproef

Het belang van sound quality neemt zienderogen toe zodat binnen het lab Industriële Fysica en Toegepaste Mechanica (IFTM) van Howest dept. PIH interesse ontstond om dat vakdomein te verkennen. Met een pool van toekomstige ingenieurs in de elektromechanica, het industrieel ontwerp en de elektronica, zou de PIH eventueel een stimulerende rol kunnen spelen in de introductie van het vakdomein bij de plaatselijke industrie. Door een masterproef ter zake uit te schrijven, is de verkennende fase ingezet. Er is geen samenwerking met een bedrijf opgezet omwille van het embryonale stadium en de relatieve onzekerheid over het welslagen van het onderzoek naar het nieuwe vakdomein. De essentie van de masterproef is een fundering te bouwen die de verdere uitdieping van de vakdomeinkennis binnen de vakgroep faciliteert.

Omwille van het ontbreken van de kennis over sound quality, dient eerst een grondige literatuurstudie te gebeuren. Deze is veel uitgebreider dan doorsnee door de complexiteit en de interdisciplinariteit van het onderwerp. De relatieve abstractheid, de uitgebreidheid, de niet-uniformiteit en de Engelse vaktaal maken de studie er niet lichter op. Desondanks kan, na voldoende speurwerk, de nodige informatie over het vakgebied gevonden worden in boeken, cursussen, scripties en dissertaties, wetenschappelijke artikels en internetpagina's. Om later opzoekingswerk over bepaalde aspecten van sound quality vlotter te laten verlopen, is bij de vaktermen de Engelse en soms Duitse vertaling cursief toegevoegd. Naar de literatuur die

geraadpleegd is om de scriptie te schrijven, is niet verwezen in de tekst omdat dit de leesbaarheid ervan te sterk zou verminderen. De bibliografie achteraan is wel gerangschikt. De literatuurstudie omvat de omschrijving van het vakdomein, de gebruikte parameters, de conventionele berekeningswijzen van die parameters, de meetmethodes en -apparatuur en het aanhalen van gekende toepassingen. Ze is eveneens samengevat in een digitale presentatie met didactische geluidsfragmenten, geschikt als studiemateriaal voor studenten. Deze is te vinden in de digitale bijlage.

Om sound quality metingen te kunnen uitvoeren, moet de nodige apparatuur voorhanden zijn. Gevraagd wordt een marktonderzoek van bestaande commerciële sound quality pakketten uit te voeren en de mogelijkheden van de aanwezige apparatuur en software in het lab IFTM te verkennen.

Het uitvoerend gedeelte van de masterproef omvat het onderzoek naar de haalbaarheid en zo mogelijk de realisatie van de ontwikkeling van een rekenmethode –in het Matlab[®] rekenpakket– die toelaat om vanuit een geluidsopname met de aanwezige apparatuur de sound quality meeteenheden te berekenen. Dit om de dure sound quality software te kunnen omzeilen. Er staat ‘zo mogelijk realisatie’ omdat gespecialiseerde softwarepakketten verschillende berekeningswijzen hanteren en karig zijn met informatie over de niet-gestandaardiseerde formules. Om geluidsopnames digitaal te kunnen capteren en voor te bewerken, is kennis van digitale signaalverwerking noodzakelijk.

Om de industriële relevantie van het verwerven sound quality expertise na te gaan, wordt de bedrijfswereld gepolst naar interesse voor het vakdomein en naar praktische toepassingen. Deze kunnen dan eventueel fungeren als aanschouwelijke sound quality case studies ter curriculumuitbreiding van de studenten.

2 Akoestiek en psychoakoestiek

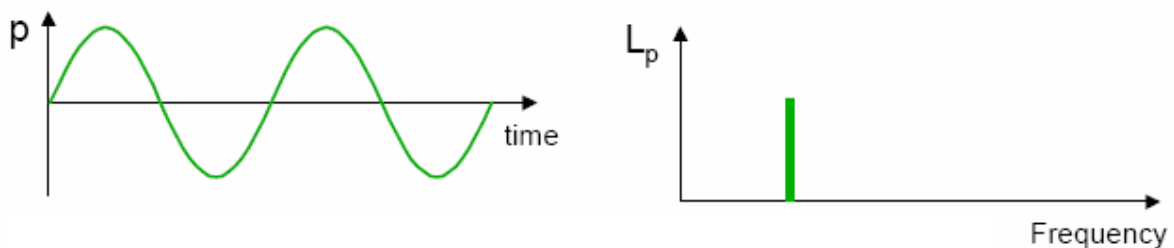
2.1 Inleiding

In dit hoofdstuk worden vooreerst summier de grondslagen en basisbegrippen uit het wetenschapsdomein van de akoestiek behandeld die voor de bepaling van sound quality relevant zijn. Dit om de niet met het vakdomein vertrouwde lezer in staat te stellen de scriptie te volgen. Voor een grondige kennismaking met de beginselen van de geluidsleer kan de lezer zich wenden tot de literatuur opgesomd in de bronnenvermelding.

Daarna behandelt dit hoofdstuk de belangrijkste en voor sound quality relevante begrippen van de psychoakoestiek. Het wordt de lezer duidelijk dat *what you hear*, niet altijd *what you get* is. Een psychoakoesticus tracht vast te leggen wat het auditief systeem van (combinaties van) inkomende geluidssignalen overhoudt. Enkele begrippen, die bepaalde auditieve fenomenen omschrijven, worden besproken.

2.2 Grootheden van geluid

Geluid plant zich voort in golfvorm, dus met een bepaalde amplitude die de geluidsterkte bepaalt en een bepaalde frequentie die de toonhoogte bepaalt. Dit geldt echter enkel voor zuivere tonen (zie Figuur 2.1).



Figuur 2.1: Een zuivere toon is een perfecte sinus en heeft juist één frequentie.

Regulier lawaai¹ bestaat uit samengestelde golven met vele frequenties en amplitudes. Fysisch zijn die golven drukverschillen. Die drukken, gesuperponeerd op de omgevingsdruk in het medium (atmosfeerdruk in lucht), worden opgevangen door het oor. Geluidsdruk is een scalaire grootheid en wordt gemeten in Pascal. Echter, het grote bereik en de logaritmische

¹ Het is nuttig het onderscheid duidelijk te houden tussen *lawaai*, *geluid*, *klank* en *toon*. Echter, vertaling naar het Engels van deze woorden levert resp. *sound*, *noise*, *sound* en *tone*. *Klank*, de samenstelling van zuivere tonen, heeft dus geen eigen benaming. *Noise* en *sound* worden echter ook niet consequent gescheiden gehouden (men bepaalt vb. de *sound quality* van *machine noise*) waardoor verwarring zou kunnen optreden. Om die reden zal in deze scriptie alle *noise* als *geluid* vertaald worden, want dat is het in essentie ook.

werking van het oor rechtvaardigen een verscaling naar decibel. De omzetting gebeurt met volgende formule:

$$L_p = 20 \cdot \log_{10} \left(\frac{p}{p_0} \right) \quad (2.1)$$

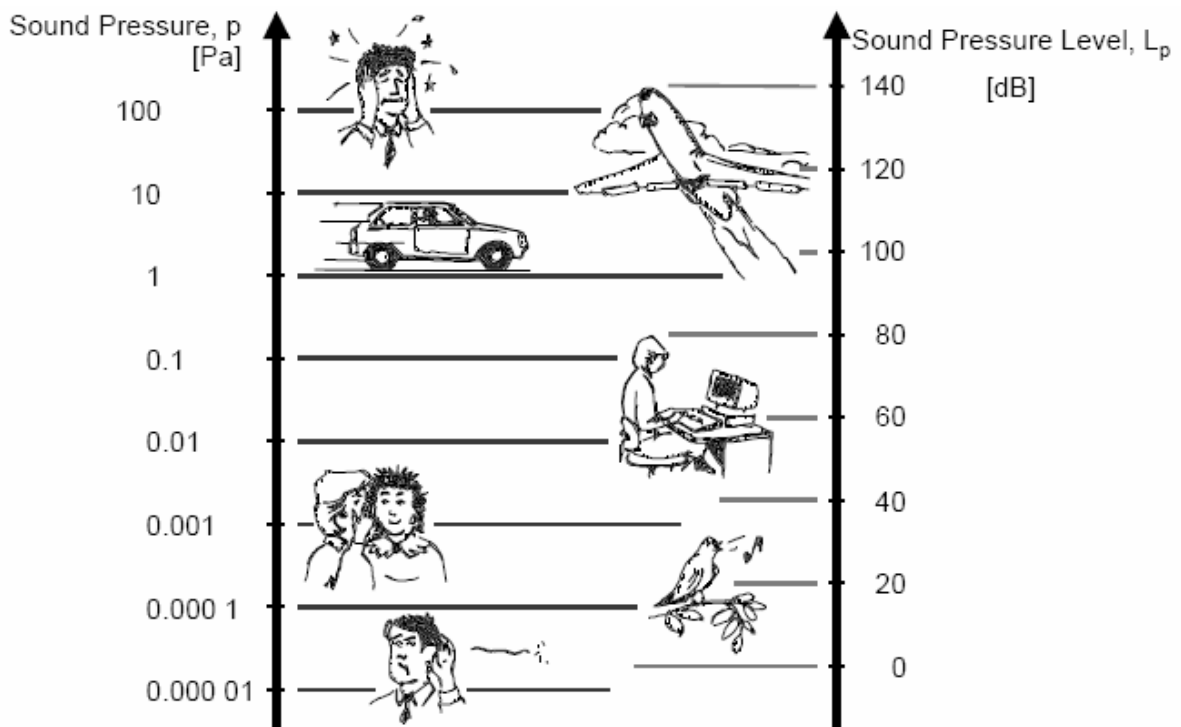
Waar

L_p is het geluidsdrumniveau (*sound pressure level, SPL*) [dB].

p is de geluidsdruk [Pa].

p_0 is de referentiedruk die de gehoorgrens (bij 1000Hz) van een normaal persoon weergeeft en bedraagt 20 μ Pa.

De decibelschaal is veruit de meest bekende voor de appreciatie van geluid. De twee getallenassen in Figuur 2.2 geven de corresponderende L_p -waarden bij enkele geluidsdrukken. De tekeningen geven een indruk van hun getalswaarden.

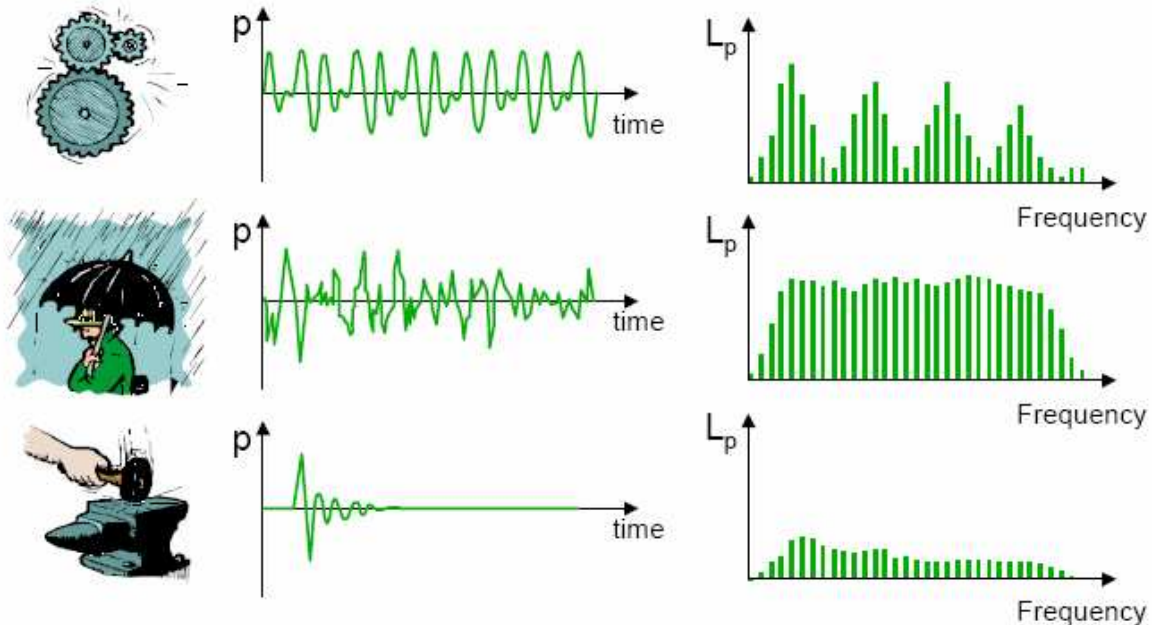


Figuur 2.2 De tekeningen geven een indruk van getalswaarden in Pa en dB.

Twee andere, in deze scriptie niet verder aangewende grootheden zijn geluidsvermogen (L_W) en -intensiteit (L_I). Eerstgenoemde wordt gebruikt voor de bepaling van (machine)lawaai, geluidsintensiteit hoofdzakelijk voor de lokalisatie van geluidsbronnen. Ook deze worden verschaald naar decibel. Om een bepaalde decibelwaarde dan nog als een geluidsdrumniveau te herkennen, wordt de suffix '(SPL)' toegevoegd, bv. $L_p = 86$ dB (SPL).

2.3 Bandenspectra in het frequentiedomein

Bij de verwerking van geluidsopnames is beschikking over hun frequentiespectra onontbeerlijk. Veruit de meeste geluiden zijn duidelijk geen zuivere tonen, zelfs geen klanken (samenstelling van tonen). Ze zijn samengesteld uit een groot aantal componenten en bevatten verschillende frequenties. De Figuur 2.3 laat zien hoe alledaagse geluiden het hele frequentiedomein bestrijken.



Figuur 2.3: Complexe geluiden zijn samengesteld uit vele frequenties.

Frequentiespectra van signalen kunnen bekomen worden door uitvoering van een fouriertransformatie. Deze frequentiespectra worden daarna nog eens opgesplitst in banden van gelijke breedte, weliswaar op logaritmische schaal. In de akoestiek worden de 1/1- en 1/3-octaaftandspectra gebruikt. Hun centrale frequenties, f_c , zijn gestandaardiseerd in de norm ISO 266-1997, zie Tabel 2.1. De onder- en bovenfrequentie van een 1/x-octaaftand worden bepaald met volgende formule:

$$f_{1,2} = f_c \cdot 2^{\pm \frac{1}{2}x} \quad (2.2)$$

Waar

f_1 is de benedenfrequentie [Hz]

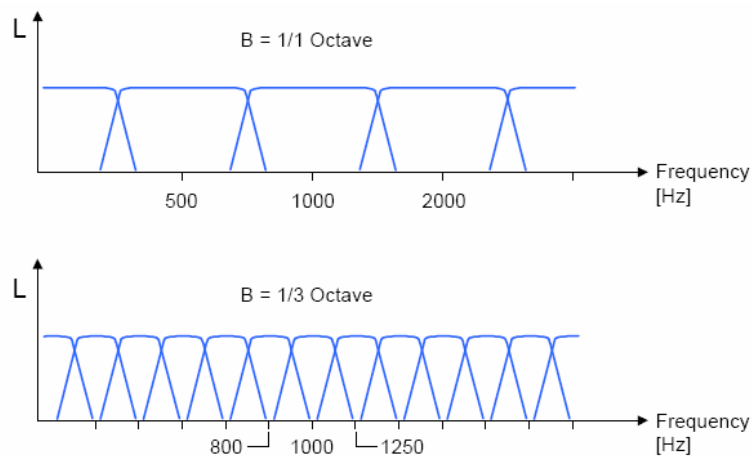
f_2 is de bovenfrequentie [Hz]

f_c is de middenfrequentie [Hz]

Tabel 2.1: De genormaliseerde frequenties van de banden [Hz].

Normalized frequency	1/1 oct	1/2 oct	1/3 oct		1/1 oct	1/2 oct	1/3 oct		1/1 oct	1/2 oct	1/3 oct
16	x	x	x	160			x	1600			x
18				180		x		1800			
20			x	200			x	2000	x	x	x
22.4		x		224				2240			
25			x	250	x	x	x	2250			x
28				280				2800		x	
31.5	x	x	x	315			x	3150			x
35.5				355		x		3550			
40			x	400			x	4000	x	x	x
45		x		450				4500			
50			x	500	x	x	x	5000			x
56				560				5600		x	
63	x	x	x	630			x	6300			x
71				710		x		7100			
80			x	800			x	8000	x	x	x
90		x		900				9000			
100			x	1000	x	x	x	10000			x
112				1120				11200		x	
125	x	x	x	1250			x	12500			x
140				1400		x		14000			
160			x	1600			x	16000	x	x	x

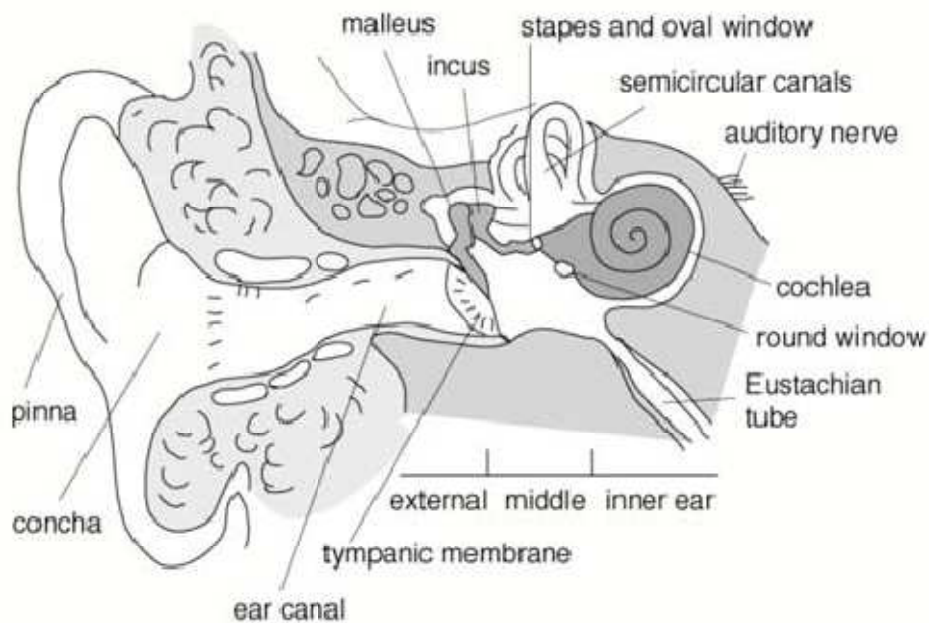
Op die manier wordt het volledige, menselijk hoorbare frequentiespectrum onderverdeeld (Figuur 2.4). De 1/3-octaf- of tertsbandspectra worden gebruikt omdat de gevoeligheid van het menselijk oor varieert naargelang de frequentieband. Bij geluidsberekeningen krijgen de frequentiebanden dan ook elk een wegingsfactor om de perceptie van het oor zoveel mogelijk te benaderen.



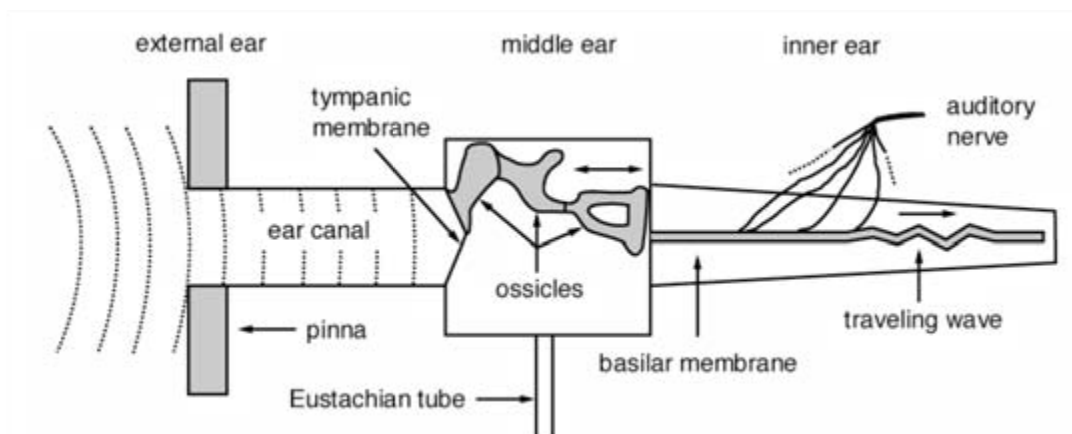
Figuur 2.4: De octaf- en tertsbandsfilter sluiten aan zodat het volledige spectrum wordt bedekt.

2.4 Het menselijk auditief systeem

Om de vreemde bevindingen van de psychoakoestiek te begrijpen, is elementaire kennis van de verwerking van de binnenkomende drukgolven door het menselijk gehoor noodzakelijk. Zoals te volgen op Figuur 2.5 brengen de inkomende drukverschillen het tympanisch membraan of trommelvlies aan het trillen. Deze beweging wordt via de ossicles of gehoorbeentjes in het middenoor (malleus of hamer, incus of aambeeld en stapes of stijgbeugel) doorgegeven aan het ovale venster. Een excitatie van laatstgenoemde zorgt voor een golfbeweging in de vloeistof binnen in de cochlea of het slakkenhuis.



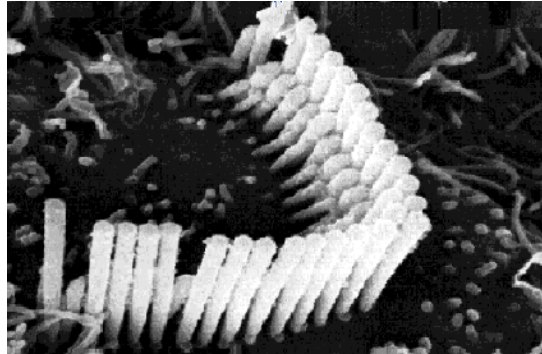
Figuur 2.5: Macroscopische tekening van het menselijk oor.



Figuur 2.6: schematische voorstelling van het menselijk auditief systeem.

Het ontrolde slakkenhuis in Figuur 2.6 toont hoe een golf doorheen het basilair membraan loopt. Op het basilair membraan ligt het orgaan van Corti, dat bedekt is met haarcellen

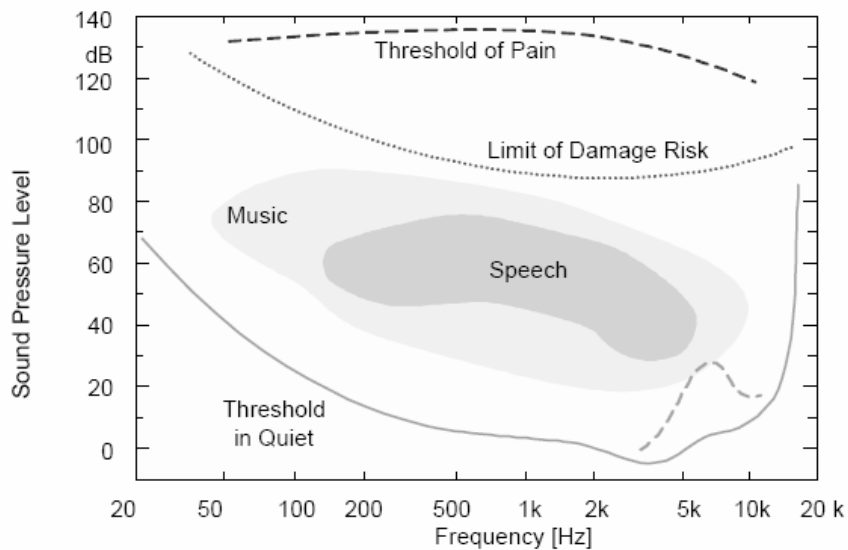
(ca. 15000 stuks, Figuur 2.7). Deze bewegen mee met de vloeistofgolven, en sturen een ionenstroom naar de neuronen. Via de zenuwvezels bereiken de elektrische impulsen uiteindelijk de hersenen. De trilhaartjes diep in de cochlea, of verst van het ovale venster, nemen de laagste frequenties waar. Op die manier voert het oor een signaalontbinding in frequenties uit, gaande van 20Hz tot 20kHz.



Figuur 2.7: Groepje haarcellen dat bestaat uit drie rijen haarcellen.

Het oor heeft echter geen constante gevoeligheid over zijn frequentiebereik. Dit wordt geïllustreerd in Figuur 2.8, die geen rechte lijnen bevat; de menselijke auditieve luidheidsperceptie stemt niet overeen met de objectieve L_p -waarden. Curven van gelijke luidheid dienden te worden opgesteld om geluidsterktemetingen correct op menselijke leest te kunnen schoeien.

Ook de gehoordrempel (*threshold in/of quiet*) is op de figuur aangegeven. Dat is de grens waarop we geluiden net (niet) waarnemen. Hij ligt hoog bij lage tonen, bereikt een minimum rond 4 kHz om dan terug sterk te stijgen vanaf 10 kHz. De gehoordrempel is belangrijk bij het bepalen van de subjectieve luidheid van geluiden; in welke mate vangt het auditief systeem bepaalde frequenties al dan niet op?



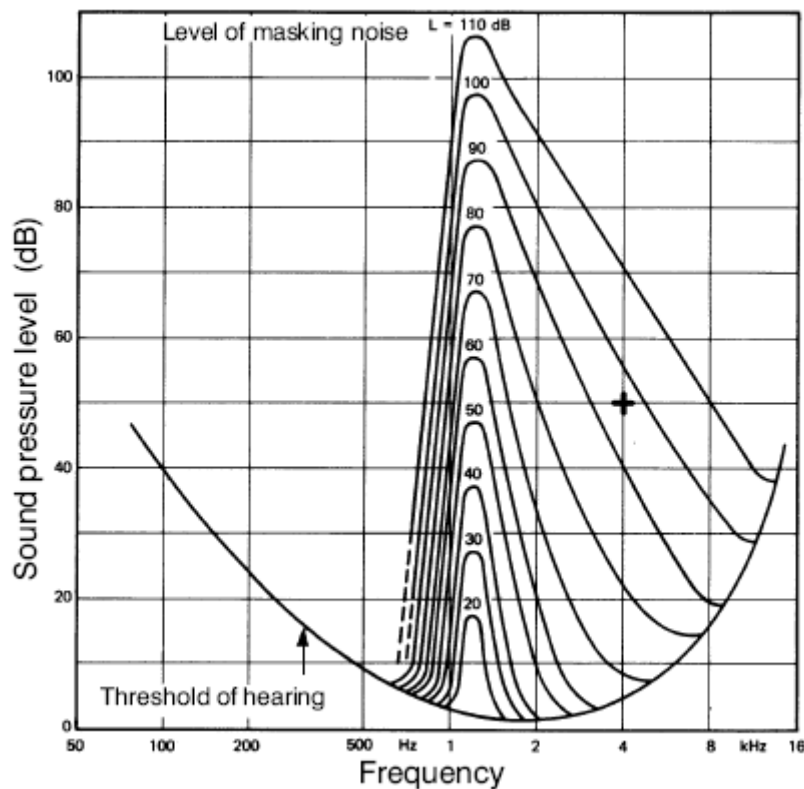
Figuur 2.8: Het menselijk waarneembaar geluidsgebied.

2.5 Maskering

Wie op straat wat met de buur staat te praten en een zware vrachtwagen passeert, zal de buur niet meer verstaan gedurende de passage. Het stemgeluid wordt gemaskeerd door de vrachtwagen. Maskering (*masking*, *Verdeckung*) vindt plaats wanneer een geluid overstemd wordt door een ander. Dit is een psychoakoestisch fenomeen. Maskering kan spectraal en tijdgebonden plaatsvinden.

2.5.1 Spectrale maskering

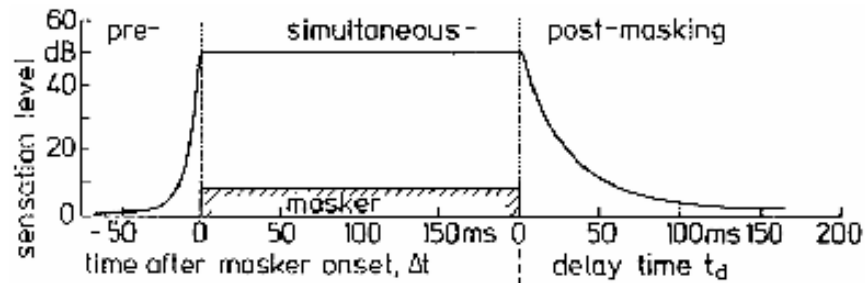
Figuur 2.9 toont een voorbeeld van spectrale maskering. Een 4 kHz toon, 50 dB (aangeduid met +), wordt afgespeeld samen met smalbandruis, gecentreerd rond 1200 Hz. Tot 90 dB ruis zal de toon te horen zijn, vanaf 100 dB niet meer. De figuur toont ook dat de maskering naar de hogere frequenties toe, veel groter is dan naar de lage. Ook geldt dat gelijk geëxciteerde smalbandruis van verschillende frequenties dezelfde maskeringskromme hebben, weliswaar op logaritmische schaal. Hoge frequenties hebben m.a.w. een groter gemaskeerd veld.



Figuur 2.9: Het maskeringseffect bij smalbandlawaai.

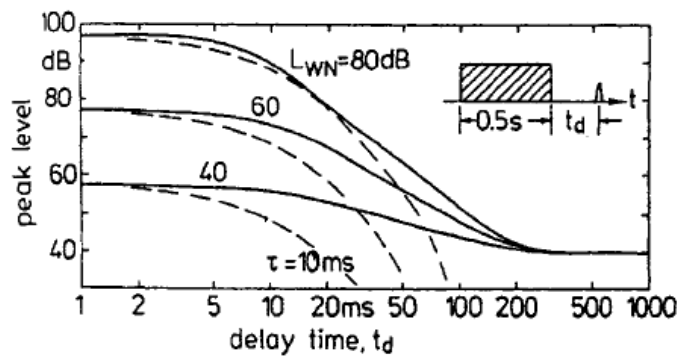
2.5.2 Maskering in de tijd

Maskering in de tijd kan opgesplitst worden in pre-, simultaneous- en post-masking (Figuur 2.10). De neurale reactietijd van het auditief systeem is recht evenredig met het geluidsniveau van het ontvangen signaal.



Figuur 2.10: Schematische tekening van tijdsgebonden maskering. Bemerkt de verschillende tijdsorigines.

Omdat het gehoor korte (< 200 ms) tonen als minder luid waarneemt kan pre-masking ontstaan: een kort geluid dat direct (< 50 ms) voorafgaat aan een luid tweede geluid, wordt nauwelijks tot niet waargenomen. Bij post-masking geldt hetzelfde, maar omgekeerd: tot 200 ms na een sterk geluid wordt een kort tweede geluidje gemaskeerd; het auditief systeem neemt het niet waar. Figuur 2.11 toont hoe het maskeerniveau 5 ms na het stoppen van de witte ruis (*white noise*) van 80 dB begint af te nemen. Na 200 ms is de maskering bij de drie testgeluiden gelijktijdig volledig weggefallen. Simultane maskering komt voor als korte geluidspulsen bovenop het eerste geluid onhoorbaar zijn.



Figuur 2.11: Post-masking bij verschillende maskeerniveaus.

2.6 Toonhoogte

2.6.1 Zuivere tonen

Testen hebben uitgewezen dat de perceptie van toonhoogte (*pitch*) niet altijd overeenstemt met de juiste frequentie. Waar men 220 Hz aanduidde als de halve toonhoogte van 440 Hz, was dit nog 1300 Hz als halve toonhoogte van 8 kHz. Bij hoge frequenties gaat de beleving van toonhoogteverhouding (*ratio pitch*) naar een grotere frequentieverhouding dan 2:1 (zie

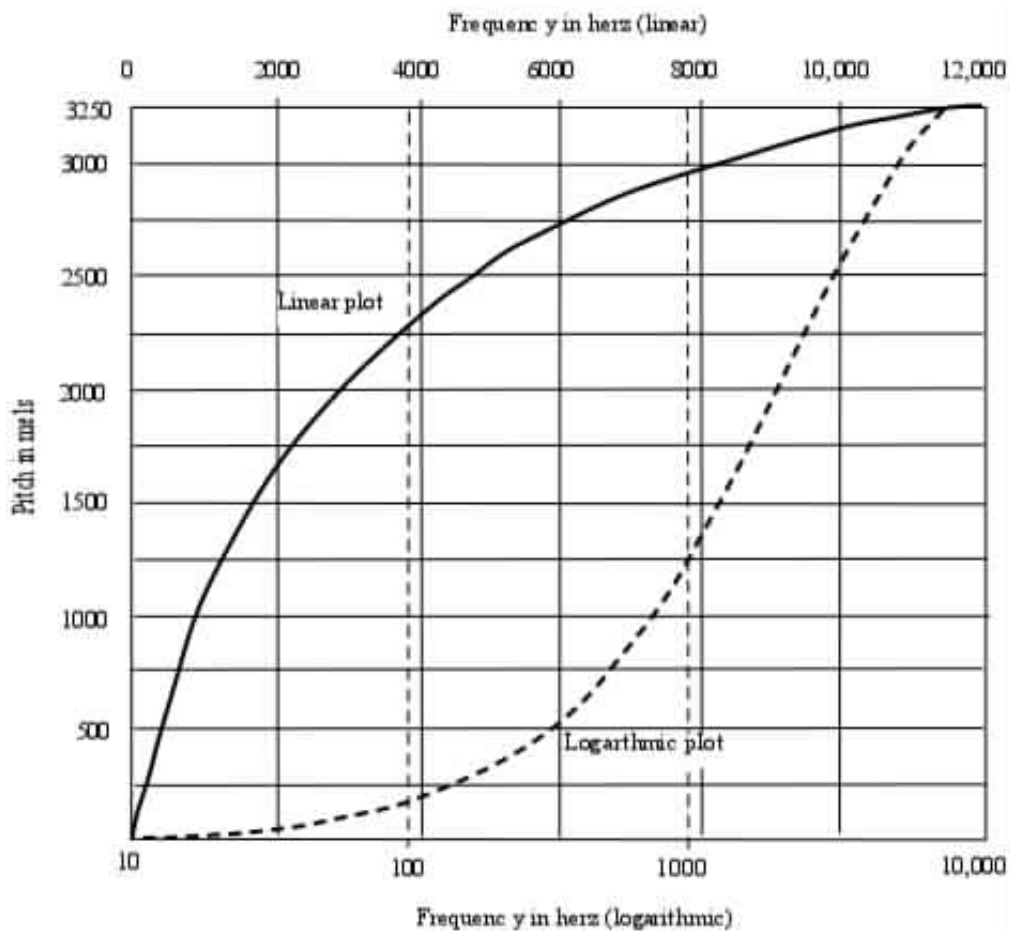
ook Figuur 2.12). Om de psychoakoestische parameter toonhoogte absoluut te kunnen vastleggen, was een referentie nodig. 125 mel (komt van melodie) werd gelijkgesteld aan 125 Hz; daar is de perceptieve frequentieverhouding wel nog gelijk aan 2:1. De onderstaande formule van Zwicker leert dat de reeds aangehaalde 8 kHz correspondeert met een toonhoogte van 2100 mel en 1300 Hz met 1050 Hz. Net de helft dus.

$$Z = 1000 \log_2(1 + f / 1000) \quad (2.3)$$

Waar

Z is de toonhoogte [mel]

f is de frequentie [Hz]



Figuur 2.12: De volle lijn toont dat de verhouding pitch - frequentie enkel bij lage frequenties lineair is. De onderbroken lijn toont het logaritmisch verband tussen beiden bij hoge frequenties.

2.6.2 Complexe tonen

Bij complexe tonen kunnen meerdere toonhoogtes onderscheiden worden. De perceptie ervan hangt in grote mate af van hun frequentie en in mindere mate van maskeereffecten. Die toonhoogtes, afgeleid van de spectrale inhoud van een geluid, worden spectrale toonhoogtes (*spectral pitches*) genoemd. Het komt ook voor dat men een bepaalde toonhoogte meent waar te nemen, terwijl deze niet aanwezig is. Dit komt omdat de hogere harmonischen ervan wel aanwezig zijn. Dit verschijnsel heet *virtual pitch* en is onderzocht door Terhardt.

True pitch is de gebruikte benaming als in de toonhoogteberekeningen het effect van de *pitch shift*, dit is het persoonsgebonden perceptieverschil tussen de ervaren en de eigenlijke toonhoogte, ingecalculleerd wordt. Bij *nominal pitch* wordt daarmee geen rekening gehouden.

2.7 Kritische banden

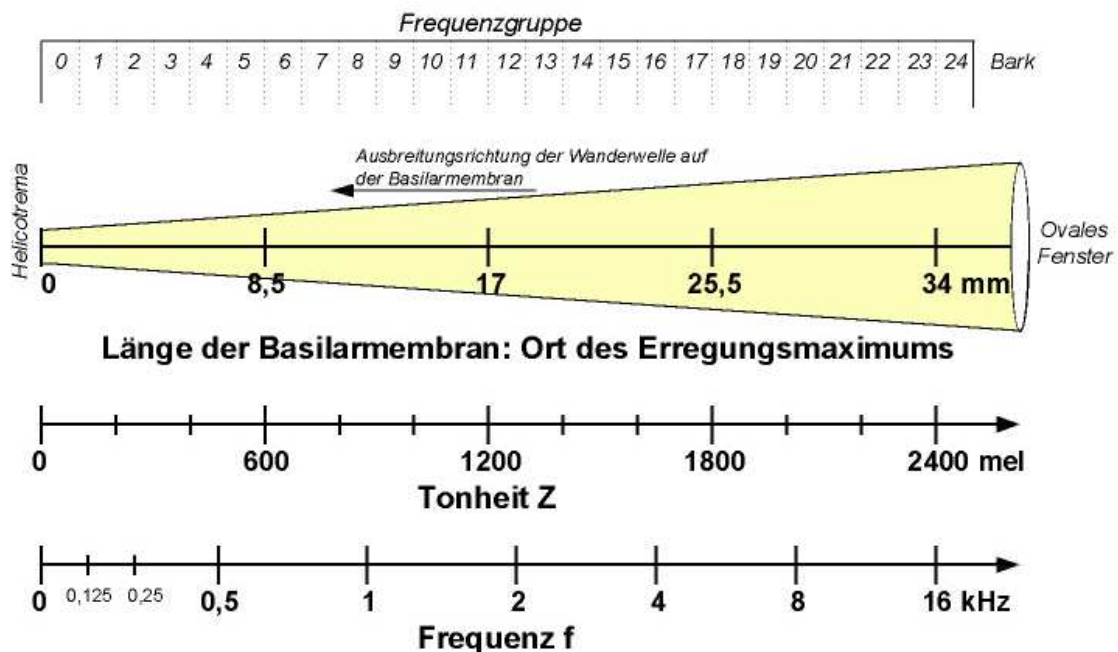
De maskerende ruisbandbreedte vergroten zal er op een bepaald punt niet meer toe leiden dat een zuivere toon op diens centrale frequentie nog meer gemaskeerd zal worden. Deze kritische punten zijn bepalend voor de kritische bandbreedtes (*critical bandwidth*). Enkel de energie dicht bij de toonfrequentie zal voor de maskering zorgen; binnen een kritische band zal het gehoor de energie sommeren. Deze kritische banden (*critical bands*, *Frequenzgruppen*) komen overeen met de niet-lineaire afstanden tussen de frequentiereceptoren op het basilair membraan. Zo een frequentieband is 1,3 mm lang en de cochlea 32 mm (zie Figuur 2.13), waaruit volgt dat er 24 kritische banden zijn in het menselijk auditief systeem. De ‘Bark-schaal’, uitgevonden door Zwicker in 1961 en genaamd naar Heinrich Barkhausen, volgt deze verdeling. Berekening gebeurt als volgt:

$$z = 13 \cdot \tan^{-1}(0,00076f) + 3,5 \cdot \tan^{-1}((f / 7500)^2) \quad (2.4)$$

Waar

z is de kritische band waarde [Bark].

f is de frequentie [Hz].



Figuur 2.13: Schematische voorstelling van een ontrold slakkenhuis met maataanduiding.

Zoals te zien in Tabel 2.2 beslaan de kritische bandbreedtes, met centrale frequentie onder de 500 Hz, rond de 100 Hz. Daarboven nemen ze ruwweg telkens met 20% toe. De correcte berekening gebeurt aan de hand van volgende formule:

$$CB = 25 + 75 \cdot (1 + 1,4 f_c^2)^{0,69} \quad (2.5)$$

Waar

CB is de Critical Bandwidth [Hz].

f_c is de centrale frequentie van de band [kHz].

Tabel 2.2: De 24 kritische banden met hun frequenties.

z	f_l, f_u	f_c	z	Δf_G	z	f_l, f_u	f_c	z	Δf_G
Bark	Hz	Hz	Bark	Hz	Bark	Hz	Hz	Bark	Hz
0	0				12	1720			
		50	0.5	100			1850	12.5	280
1	100				13	2000			
		150	1.5	100			2150	13.5	320
2	200				14	2320			
		250	2.5	100			2500	14.5	380
3	300				15	2700			
		350	3.5	100			2900	15.5	450
4	400				16	3150			
		450	4.5	110			3400	16.5	550
5	510				17	3700			
		570	5.5	120			4000	17.5	700
6	630				18	4400			
		700	6.5	140			4800	18.5	900
7	770				19	5300			
		840	7.5	150			5800	19.5	1100
8	920				20	6400			
		1000	8.5	160			7000	20.5	1300
9	1080				21	7700			
		1170	9.5	190			8500	21.5	1800
10	1270				22	9500			
		1370	10.5	210			10500	22.5	2500
11	1480				23	12000			
		1600	11.5	240			13500	23.5	3500
12	1720				24	15500			
		1850	12.5	280					

Dit verloop wordt nagebootst door de tertsbanden die, boven 500Hz, goed met de selectiviteit van het auditief systeem corresponderen.

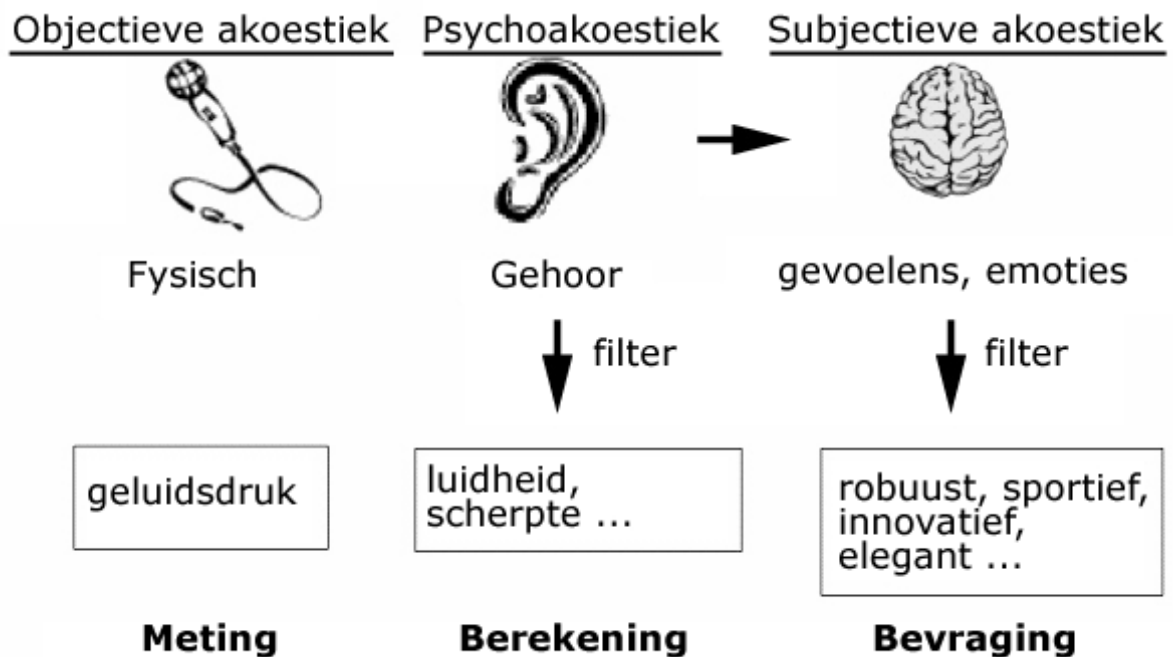
Naast de Bark-schaal bestaat ook nog de ERB-schaal (*Equivalent Rectangular Bandwidth*), bedacht door Moore & Glasberg om luidheid nauwkeuriger te kunnen berekenen (zie verder).

3 Sound Quality parameters

3.1 Inleiding

Op basis van de fundamentele bevindingen van de psychoakoestiek zijn parameters opgesteld, metrics in het Engels, die bepaalde kenmerken van het geluid dat het oor bereikt, kwantificeren. Ze zijn ontstaan vanuit de nood objectief geluidskennmerken vast te kunnen leggen. Om tot algemeen geldende definities en berekeningswijzen te komen, zijn dan ook grote testgroepen van normaal horende personen vereist. Alle grafieken dienaangaande zijn dan ook experimenteel bepaald.

De psychoakoestische parameters worden gebruikt in de muziekwereld, summier in de architectuur, in de akoestische milieuproblematiek, in gespreksruimtes en telecommunicatie, alwaar de term ‘speech quality’ gebruikt wordt, maar vooral bij productontwikkeling en –innovatie. Van de belangrijkste meeteenheden die in dit hoofdstuk nader toegelicht worden, is echter enkel ‘luidheid’ een gestandaardiseerde psychoakoestische grootheid. De andere, i.e. scherpte, fluctuatiersterkte, ruwheid, tonaalheid en sensory pleasantness zijn (nog) niet in normen opgenomen en zijn dus niet eenduidig vastgelegd. Om die reden, en om een optimale schikking van de scriptie te verkrijgen, zijn in dit hoofdstuk enkel de algemeenheden van laatstgenoemde grootheden opgenomen (kwalitatieve benadering). Formules/modellen ter bepaling van hun waarden zijn verderop terug te vinden, bij de uitleg over de opbouw van het Matlab programma.



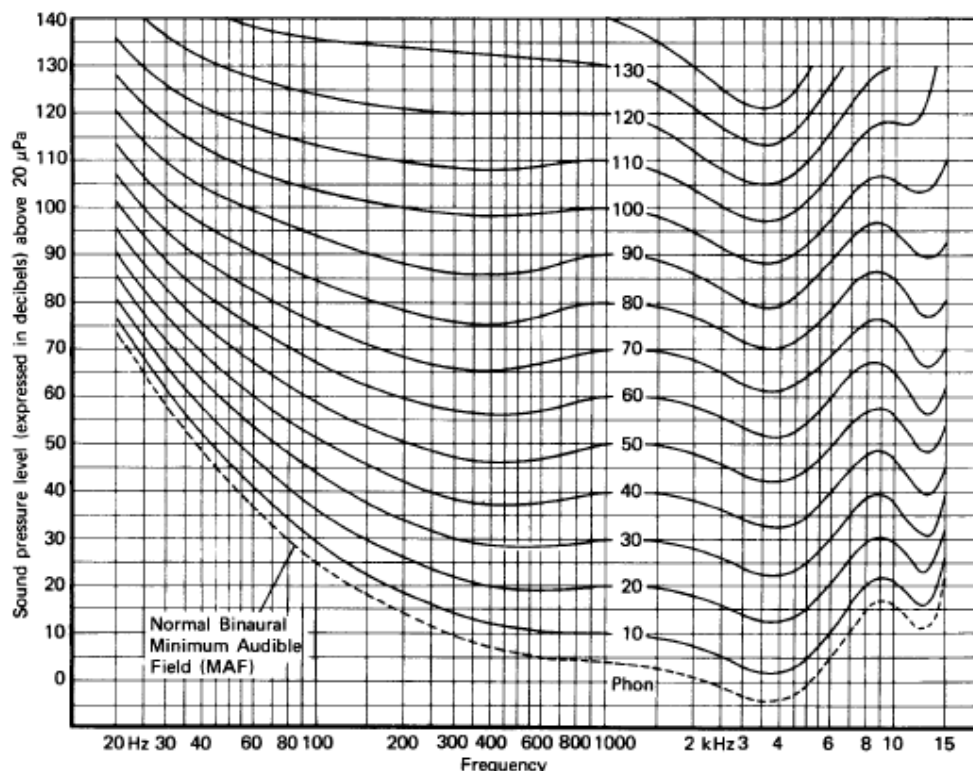
Figuur 3.1: Doel en methodes van de objectieve akoestiek, de psychoakoestiek en de subjectieve akoestiek.

De bepaling van product sound quality eindigt echter niet bij de bevindingen van de psychoakoestiek. Figuur 3.1 geeft daarover een overzicht. Vooreerst is er de objectieve akoestiek waarmee enkel de SPL kan gemeten worden. Het menselijk gehoor ervaart de geluidsgolven echter anders, waardoor het nodig is bijkomende psychoakoestische parameters te berekenen. Daarna maakt het brein van deze geluiden ook een eigen versie, onder andere afhankelijk van de gevoelens bij dat geluid of de emoties van het moment. Deze worden dan beschreven in adjectieven allerhande, omdat een exacte benaming van een geluid onmogelijk is. Op het aspect van de subjectieve akoestiek wordt ingegaan in de paragraaf ‘Productspectifieke parameters’.

3.2 Luidheid

3.2.1 Isofonen

Het objectieve geluidsdrukniveau (*SPL*) stemt niet overeen met de ervaring ervan door de mens. Twee geluiden met nochtans gelijke *SPL*, zullen niet als even luid ervaren worden. Hoe een geluidsterkte dan wel ervaren wordt, is weergegeven op de door Fletcher en Munson (1933) experimenteel afgeleide isofonen, weergegeven in de Figuur 3.2. De lijnen van gelijk luidheidsniveau (*loudness level*) of isofonen lopen niet parallel, en tonen aan dat het menselijk gehoor het gevoeliger is bij 4 kHz. De contouren van gelijke luidheid liggen vast in de ISO-norm 266:1987 en hebben als referentie dat een zuivere toon van 1kHz bij 40 dB (*SPL*) een luidheidsniveau van 40 foon (*phon*) heeft, de eenheid van luidheidsniveau.



Figuur 3.2: De isofonen voor zuivere tonen.

Een ander kenmerk van de menselijke luidheidswaarneming is dat een geluidssterktetoename van 10 dB (SPL) niet op alle niveaus gelijk wordt ervaren. Nochtans zou die toename een verdubbeling van de luidheid moeten inhouden. Om subjectieve luidheidsverschillen gemakkelijk kwantitatief te bepalen, omdat luidheid vooral relatief (tegenover een referentie) beoordeeld wordt en om interpretatie gemakkelijker te maken (linearisering) is de soon (*son*) afgeleid:

$$N = 2^{(L_N - 40)/10} \tag{3.1}$$

waar

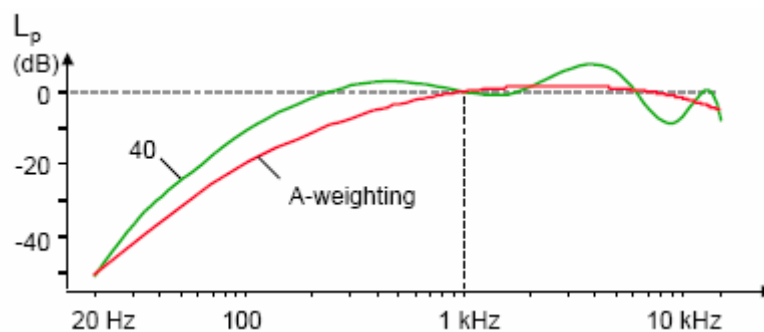
N is de luidheid [soon]

L_N is het luidheidsniveau [foon]

Eén soon stemt zodoende overeen met 40 foon. Een toon die dubbel zo luid wordt, zal nu een luidheid hebben van $2N$ en een luidheidsniveau van $L_N + 10$.

3.2.2 Wegingscurven

Om een betrouwbaar, aan de mens aangepast, beeld van geproduceerde luidheidsniveaus te krijgen, werden wegingscurven opgesteld. De A-, B-, C- en D-wegingscurven zijn vanuit verschillende oogpunten allemaal een benadering van de curven van gelijke luidheid. De A-weging wordt, bij gewone toepassingen, veruit het meest gebruikt. B-weging wordt nauwelijks nog gebruikt, C voor storend omgevingslawaai zoals rockconcerten en D voor vliegtuiglawaai. Er bestaat ook een Z-weging. Dit een weging die de specifieke, niet perfect lineaire frequentieresponsie van een microfoon compenseert zodat er netto een perfecte nulweging (*Zero*) van alle frequenties overblijft. Figuur 3.3 toont dat de algemeen toegepaste A-weging bij lage frequenties goed overeenstemt met de isofonen. Rond de 4 kHz echter wordt het luidheidsniveau onderschat.



Figuur 3.3: Het A-wegingsniveau vergeleken met de geïnverteerde en op nul genormaliseerde 40dB-isofoon.

Ook is de A-weging niet geschikt voor de luidheidsbepaling van breedbandgeluid; ze brengt het verband tussen bandbreedte en luidheid niet in rekening.

3.2.3 Bepaling van luidheid

Om de luidheid van breedband- en niet-zuivere geluiden toch te kunnen bepalen zijn er berekeningswijzen gedefinieerd die wel met deze en andere factoren, zoals maskering, rekening houden. Eerst wordt de specifieke luidheid (*specific loudness*) van elke kritische band bepaald. Dit gebeurt op basis van de wet van Stevens, die stelt dat er een relatie is tussen de perceptie van zijn intensiteit en de fysische grootte van een stimulus, waarbij de relatieve intensiteitsverandering proportioneel is met de luidheidsverandering. In plaats van het intensiteitsniveau wordt echter het excitatieniveau (*excitation level*) gebruikt.

De eersten die een luidheidsbepaling uitwerkten, zijn Stevens en Zwicker. Hun beider methodes zijn vastgelegd in de ISO-norm 532:1975 A (Stevens) en B (Zwicker). Eerstgenoemde gebruikt een octaafbandanalyse van het frequentiespectrum, laatstgenoemde een tertsbandsanalyse. Ondertussen zijn al andere, recentere methodes uitgewerkt, zoals deze van Aures en Moore & Glasberg, maar bovengenoemde blijven de enige officiële bepalingwijzen. Ze gelden overigens enkel voor stationaire geluiden (steady state sounds), wat duidelijk maakt waarom in deze context van statische luidheid gesproken wordt. De dynamische luidheid van een transiënt geluid mag zo niet bepaald worden. Onder andere Zwicker heeft toch geprobeerd tijdsgebonden geluidseffecten (temporal effects) in rekening te brengen, maar deze methode is (nog) niet als ISO-standaard geaccepteerd.

3.2.3.1 Luidheidsbepaling volgens Stevens

De methode genaamd Stevens Mark VI is enkel te gebruiken voor diffuse velden met relatief vloeiende breedbandspectra. Elke octaafband heeft een bepaalde partiële Sound Pressure Level. Deze wordt *Band Pressure Level* genoemd. Op het nomogram in Figuur 3.4 kan deze laatste omgezet worden in een partiële luidheidsindex. Met onderstaande formule wordt uit die indices een totale luidheid gedistilleerd:

$$S_t = S_m + F(\sum S - S_m) \quad (3.2)$$

Waar

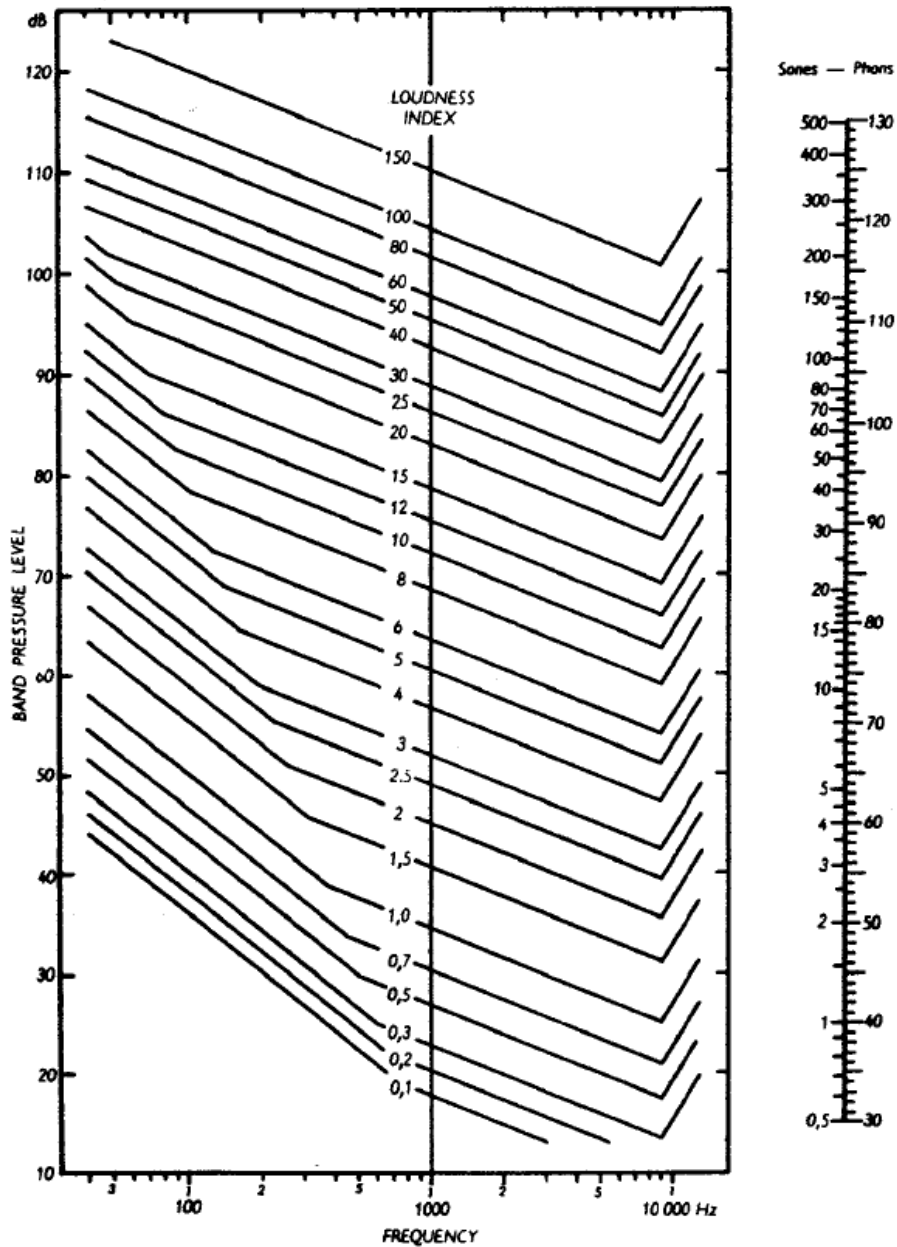
S_t is de totale luidheid [soon (OD)²].

S_m is de grootste van de luidheidsindices [soon(OD)].

$\sum S$ is de som van de partiële luidheid van elke band [soon(OD)].

F is een vaste factor die de maskeereffecten in rekening brengt. F bedraagt 0,3 voor octaafbandwaarden.

² De termen foon en soon zonder een kwalificerende afkorting kunnen enkel gebruikt worden bij de uitdrukking van directe subjectieve metingen. O staat voor *octave*, D voor *diffuse*.



Figuur 3.4: Partiële luidheid volgens Stevens Mark VI.

Er bestaat ook een (nog) niet-ISO-gestandaardiseerde Stevens Mark VII, waarin de partiële luidheidsberekening en de F -waarden wat meer verfijnd zijn.

3.2.3.2 Luidheidsbepaling volgens Zwicker

Zwickers' luidheidsbepaling wordt meer toegepast en staat erom bekend de subjectieve luidheidsbeoordeling het dichtst te benaderen. Ze is geldig van 25 tot 12 500 Hz en kan complexe breedbandgeluiden afkomstig van ofwel een frontaal geluid uit een vrij veld (L_{GF}) ofwel een diffuus geluidsveld (L_{GD}) beoordelen. Tien grafieken, vijf voor beide mogelijkheden zijn opgesteld. Figuur 3.5 toont een dergelijke grafiek, waarop de totale luidheid van een (diffuus) fabrieksgeluid bepaald is.

Onderstaand stappenplan kan gebruikt worden om de totale luidheid van een geluid waarop een tertsbandanalyse gedaan is, te bepalen.

1. Betreft het een frontaal geluid uit een vrij veld, dan wordt een GF-grafiek gekozen. Voor een diffuus veld is dat GD.
2. Uit het hoogste *Band Pressure Level*, bekeken over alle tertsbanden, wordt de grafiek met gelijkaardige dB-waarden geselecteerd.
3. De verticale 'ladders' stellen tertsbanden voor. Startende vanaf 280 Hz (tertsband met middenfrequentie 315 Hz) wordt een horizontaal lijntje getrokken per tertsband, waarvan de abscis afhankelijk is van het gemeten geluidsdrukniveau (SPL).
4. Omdat de kritische banden bij lage frequenties breder zijn dan de éénderde-oktaafbanden, moeten deze als volgt onder 280 Hz samengevoegd worden tot drie corresponderende kritische banden L_1 , L_2 en L_3 :
 - a. Combineer alle banden tot de bovengrens van 90 dB tot L_1 .
 - b. Combineer de drie banden van 90 tot 180 Hz (L_2).
 - c. Combineer de twee banden van 180 tot 280 Hz (L_3).

De formule ter omzetting tot de drie kritische banden wordt exemplarisch voorgesteld met de berekening van L_2 :

$$L_2 = 10 \log_{10} (\text{antilog}L_{100} / 10 + \text{antilog}L_{125} / 10 + \text{antilog}L_{160} / 10) \quad (3.3)$$

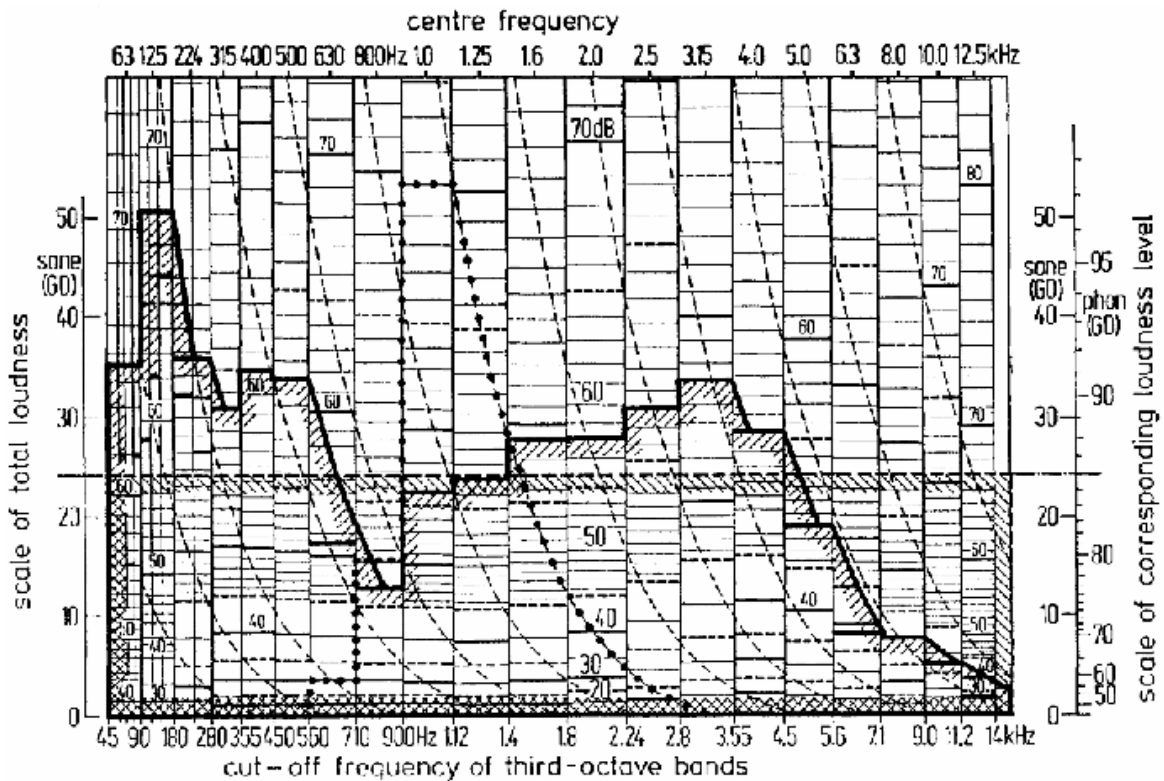
Waar

L_2 is het Band Pressure Level van de tweede kritische band [dB].

L_x (L_{100} , L_{125} , L_{160} ...) het Band Pressure Level van de respectieve tertsbanden met centrale frequentie 100 Hz etc. [dB].

Ook deze lijnen kunnen nu op de grafiek getekend worden.

5. Als de drukniveaus stijgen met de frequenties, moeten deze verbonden worden met verticale rakende lijnen. Is het rechts opvolgende niveau echter lager, dan moet, tot dat opvolgende niveau bereikt is, een lijn volgens de neerwaarts hellende onderbroken lijnen getekend worden. Deze brengen de maskering in rekening.
6. De ingesloten oppervlakte onder de curve wordt nu getransformeerd naar een rechthoek van gelijke oppervlakte, over de volledige breedte van de grafiek. De hoogte ervan geeft de totale luidheid in foon(GF/GD) of soon(GF/GD) aan. Een nauwkeuriger bepaling kan gebeuren met een planimeter.



Figuur 3.5: Een ingevulde grafiek ter berekening van de totale luidheid in een diffuus veld (fabrieksgeluid).

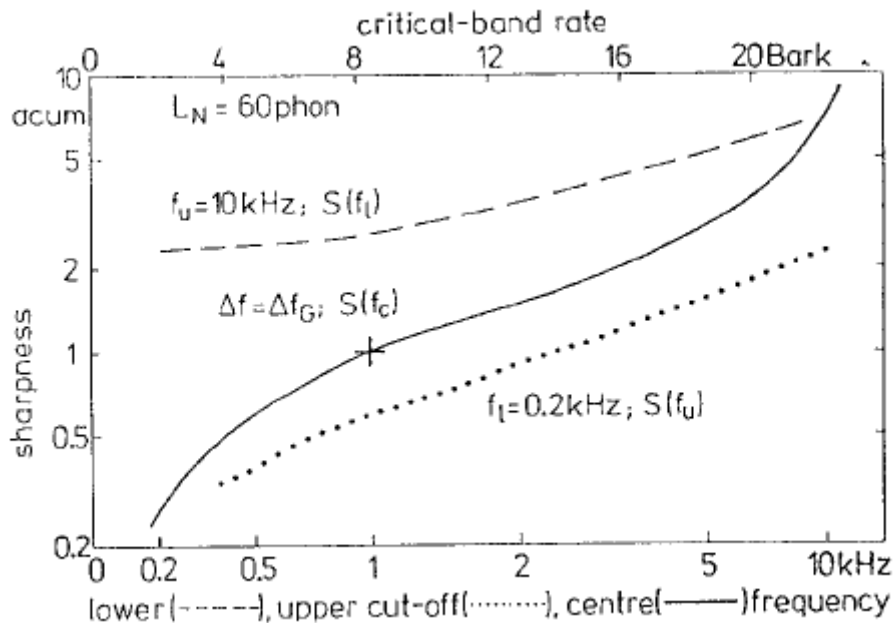
De totale luidheid van een geluid op dergelijke manier bepalen is tijdrovend en niet handig in het huidige computertijdperk. Daarom is o.l.v. Zwicker, een programmeercode, eerst in FORTRAN en daarna in BASIC, opgesteld om de luidheid automatisch te berekenen. De BASIC programmacode is opgenomen in de norm DIN 45 631. Meer hierover verderop in deze scriptie.

3.3 Scherpste

Alle geluiden hebben een bepaalde ‘scherpte’ of ‘scherpheid’ (*sharpness*, *Schärfe*) of, omgekeerd, dofheid. De kwantitatieve bepaling van scherpste gebeurt in ‘acum’ en als referentie voor één acum is smalbandruis met de kritische bandbreedte rond centrale frequentie 1 kHz op een niveau van 60 dB genomen. Dit punt is op Figuur 3.6 gemarkeerd met een kruis. Acum is afgeleid van het Latijnse ‘acer’, wat scherp betekent.

Binnen een kritische band neemt het auditief systeem geen scherpteverschil waar. Ook luidheidsniveau maakt geen verschil. De scherpste hangt vooral af van de spectrale inhoud en de centrale frequentie van smalbandgeluiden. De volle lijn in Figuur 3.6 stelt een geluid van één kritische band breed in functie van zijn centrale frequentie voor. Opvallend is de (sterke) scherptetoename bij hogere frequenties. Het belang van de bandbreedte, beperkt met een ondergrens van 200 Hz en een bovengrens van 10 kHz, wordt geïllustreerd door de

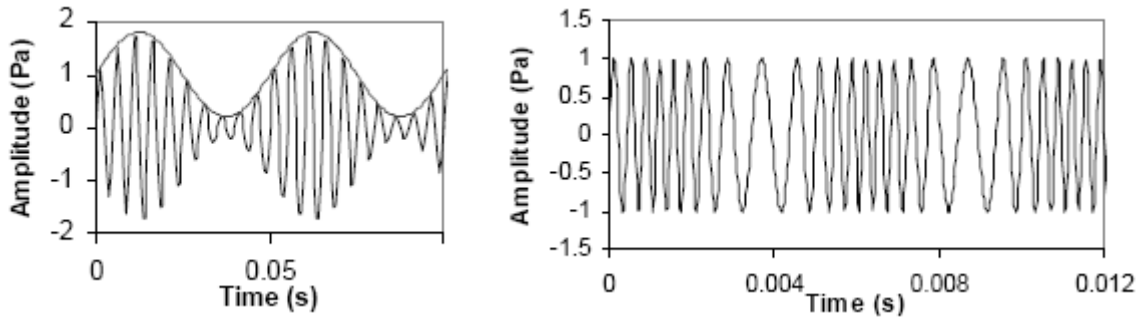
onderbroken en de stippellijn. De stippellijn is opgesteld vanuit een vaste grensfrequentie van 200 Hz. De waarden op de abscis geven nu de bovengrens aan. Een geluid met bandbreedte 0,2 tot 2 kHz heeft dus een scherpte van 1 acum. De onderbroken lijn geldt voor geluid met een bovengrensfrequentie van 10 kHz. Hier stelt de abscis dan de onderfrequentie voor. Opvallend is dat met het verlagen van de ondergrensfrequentie, ook de scherpte verlaagt. Of, met andere woorden, hoogfrequent smalbandgeluid heeft een hogere scherpte dan breedbandgeluid. Spectrumverbreding kan hier dan ook een oplossing ter reductie van de scherpte zijn. Het luidheidsniveau neemt dan wel toe. Bemerkt overigens dat de grafiek opgesteld is bij een constant niveau van 60 foon.



Figuur 3.6: De scherpte van smalband-, hoogdoorlaat- en laagdoorlaatgeluid.

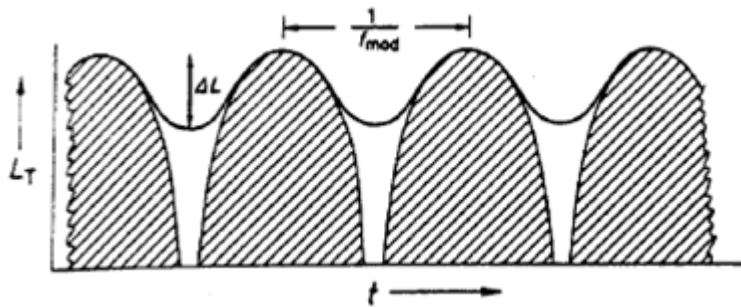
3.4 Fluctuatiesterkte

Waar de vorige psychoakoestische parameters ogenblikkelijke geluidsindrukken vastlegden, geldt voor de parameter fluctuatiesterkte (*fluctuation strength*, *Schwankungsstärke*) dat deze over een bepaalde tijd beoordeeld moet worden. Variaties van geluid in de tijd hebben algemeen een grote invloed op hun perceptie. Modulatie-effecten leggen de fluctuatiestructuur van een geluid vast. Figuur 3.7 toont links een amplitudegemoduleerde zuivere toon, de modulatie diepte bedraagt hier zo'n 0,75. Rechts is een frequentiegemoduleerde sinusgolf getekend. Een dergelijke toon zal een fluctuerende toonhoogte hebben.



Figuur 3.7: Een voorbeeld van een amplitudagemoduleerde (links) en een frequentiegemoduleerde sinus.

De indruk die bij dergelijke fluctuerende geluiden ontstaat is deze van een toe- en afnemende luidheid. Boven een fluctuatiefrequentie van 20 Hz, een enigszins arbitrair gekozen grens, wordt geen variërende luidheid meer ervaren. Dit is het gebied van de parameter ‘ruwheid’, die verderop besproken wordt. De fluctuatiesterkte neemt toe met de modulatie diepte of –graad en het luidheidsniveau. Bij een modulatiefrequentie van 4 Hz bereikt de fluctuatiesterkte een maximum. Overigens wordt laatstgenoemde sterk gedempt door het post-maskeringseffect, gevisualiseerd op Figuur 3.8. Door de maskering zijn de korte geluidsloze periodes onhoorbaar.



Figuur 3.8: De maskeringsdiepte is een belangrijke parameter bij de berekening van de fluctuatiesterkte.

De algemene berekeningswijze van de fluctuatiesterkte is in onderstaande evenredigheid voorgesteld. Deze wordt dan aangepast tot een bepaalde vergelijking, afhankelijk van het soort geluid dat bemeten is. Bemerkt het voorkomen van de 4 Hz component.

$$F \sim \frac{\Delta L}{\frac{f_{mod}}{4} + \frac{4}{f_{mod}}} \tag{3.4}$$

Waar

F is de fluctuatiesterkte [vacil].

ΔL is de maskeringsdiepte [dB].

f_{mod} is de modulatiefrequentie [Hz].

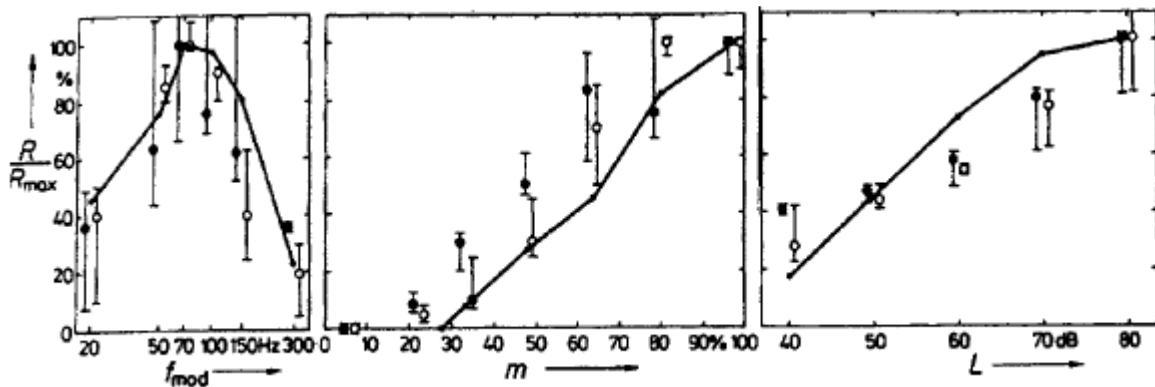
De eenheid voor fluctuatiesterkte is ‘vacil’ en 1 vacil is de fluctuatiesterkte van een 1 kHz-toon van 60 dB die een sinusoidale amplitudemodulatie met een frequentie van 4Hz ondergaat bij een modulatiegraad van 1. Vacillare is het Latijnse werkwoord voor ‘schommelen’.

3.5 Ruwheid

Ontstaan vanuit de noodzaak aan beoordeling van motorgeluid is de psychoakoestische parameter ‘ruwheid’ (*roughness, Rauhigkeit*) ontstaan. Ruw geluid kan soms gepast zijn, bv. bij een motorfiets, maar is meestal ongewenst. Zoals hierboven reeds aangehaald is de parameter ruwheid in essentie gelijk aan de fluctuatiesterkte. Ze worden echter totaal verschillend ervaren; voor het menselijk auditief systeem heeft een ruw geluid een constante luidheid. Dit is echter niet zo, de amplitude wijzigt wel degelijk maar de snelheid ervan ligt zo hoog dat de wijzigingen gemaskeerd worden.

In de eerste van de drie experimenteel bepaalde grafieken in Figuur 3.9 is het verloop van de ruwheid bij verschillende frequenties weergegeven. Bij lage modulatiefrequenties, vanaf 20 Hz, is de ruwheid klein, ondanks de grote maskeerdieptes bij deze frequenties (zie hierboven Figuur 3.8). Rond de 70 Hz bereikt de ruwheidscurve een maximum, om dan terug af te nemen en zelfs helemaal te verdwijnen. De maskeerdiepte wordt immers almaar kleiner. Bij dergelijke hoge frequentiemodulaties, ruwweg van 150 tot 300 Hz, worden drie afzonderlijke tonen waargenomen.

De tweede grafiek in Figuur 3.9 toont de bij benadering lineaire relatie tussen ruwheid en modulatie diepte, terwijl de derde grafiek aantoont dat de perceptie van ruwheid toeneemt met het geluidsniveau. Ruwheid neemt overigens bijna kwadratisch toe met de modulatiegraad.



Figuur 3.9: De relatie van de perceptieve ruwheid tot de modulatiefrequentie, -diepte en geluidsniveau.

Bovenstaande vaststellingen leiden tot de volgende evenredigheid:

$$R \sim \Delta L \cdot f_{mod} \tag{3.5}$$

Waar:

R is de ruwheid [asper].

ΔL is de maskeringsdiepte [dB].

f_{mod} is de modulatiefrequentie [Hz].

Kwantitatieve bepalingen van ruwheid gebeuren met de eenheid ‘asper’. 1 asper (Lat. ‘ruw’) is de ruwheid die optreedt bij de sinusoidale amplitudemodulatie van een 60 dB 1 kHz-toon. De modulatiefrequentie bedraagt 70 Hz en de modulatiegraad 1.

3.6 Tonaalheid

Sommige geluiden bezitten een of meerdere frequentiecomponenten die veel luider klinken dan de andere. Denk bv. aan het rolgeluid van een autoband op de weg. Het bevat een tonale component met snelheidsafhankelijke frequentie. Het horen van een dergelijke ‘fluittoon’ kan erg irritant zijn en is vastgelegd in de psychoakoestische grootheid ‘tonaalheid’ (*tonalness*, *Tonhaltigkeit*). Het zijn overigens de tonale componenten die het onderscheid maken tussen klank en ruis. Ruis heeft helemaal geen tonaalheid! Tonaalheid heeft een duale verhouding met de menselijke perceptie. Een verhoging van de tonale componenten is, zoals het aangehaalde praktijkgeval bewijst, geen garantie op een aangenamer geluid; evengoed kan ze de menselijke ergernis in de hand werken.

Soms wordt ‘tonaliteit’ (*tonality*) als benaming voor de parameter gebruikt, doch deze valt af te raden omdat tonaliteit al langer bestaat als begrip in de muziektheorie. De eenheid van tonaalheid is ‘tonality unit’, afgekort ‘tu’ en heeft als eenheidsreferentie een 1 kHz toon van 60 dB.

Geluiden worden als tonaal ervaren als ze een pure toon of een frequentiecomponent, smaller dan een kritische band, bevatten. De DIN 45 681-norm stelt dat van een tonaal geluid gesproken kan worden als de kritische band (1/3-octaaftband) van deze tonale component een geluidsniveau bezit dat 6 dB hoger ligt dan het rekenkundig gemiddeld niveau van de twee aangrenzende banden. Dit is de ‘*prominence ratio*’, wat vertaald kan worden als de ‘uitsteekverhouding’. Een andere maat is de ‘*tone-to-noise ratio*’, die het aantal zuivere tonen in een signaal weergeeft.

De tonaalheid van een geluid hangt af van vele factoren. De frequentie en de bandbreedte van de tonale component en het luidheidsniveau (met en zonder de tonale componenten) zijn bepalend. Modellen voor tonaalheid, onder andere opgesteld door Aures in zijn dissertatie, houden met deze factoren rekening maar zorgen voor een sterke filtering van de tonale componenten. Dit geeft een eenduidiger maar minder compleet beeld van de tonaalheid. Figuur 3.10 toont hoe ook het aantal harmonischen van de tonale component de tonaalheid beïnvloedt. Deze factor is bijvoorbeeld niet in de berekening opgenomen.



Figuur 3.10: Tonaalheidsbeoordeling door vijf personen van een harmonisch tonencomplex met als grondfrequentie 700 Hz en SNR ratio 11 dB.

Het is duidelijk dat de rekenkundige formulering van tonaalheid nog niet op punt staat. Het voorstel van Aures is geldig voor tonale componenten vanaf 100 Hz en kan enkel bij stationaire geluiden aangewend worden.

$$T = \sqrt{\sum_{i=1}^n [W_1(\Delta z_i) W_2(\Delta z_i) W_3(\Delta z_i)]^2} \quad (3.6)$$

Waar:

T is de tonaalheid [tu].

De term $W_1(\Delta z_i)$ de afhankelijkheid van het verschil in kritische bandbreedte van de i -de tonale component representeert.

De term $W_2(\Delta z_i)$ de afhankelijkheid van de frequentie van de i -de tonale component representeert.

De term $W_3(\Delta z_i)$ het effect van het geluidsniveaoverschil van de i -de tonale component representeert.

De drie termen in formule (3.6) zijn op zich ook de uitkomst van formules die hier niet nader toegelicht worden.

3.7 Pleasantness en annoyance

‘Pleasantness’, wat vertaald kan worden als aangenaamheid, en ‘annoyance’, i.e. ergernis, zijn als parameters net als twee communicerende vaten. De aangenaamheid van een geluid neemt toe met de afname van de ergerlijkheid. Beiden kunnen wel tegelijkertijd voorkomen! Beide meeteenheden zijn allesomvattende sound quality parameters. Deze zijn handig als van een product allerhande eigenschappen vermeld moeten worden. Dan volstaat de auditieve pleasantness als gegeven naast bv. zijn afmetingen. Al lange tijd zijn er pogingen ondernomen om een dergelijk onderscheid te maken. Zo zijn termen als ‘sensory euphony and disphony’

(*sensorischer Wohlklang und Übelklang*), wat zintuiglijke welluidendheid en onwelluidendheid betekent, al sinds 1910 in zwang. Vandaag zijn deze meeteenheden wetenschappelijk beter onderbouwd. Zwicker en Fastl benaderen relatieve sensory pleasantness holistisch met een formule waarin zowel luidheid, scherpte, ruwheid als tonaalheid vervat zitten.

$$\frac{P}{P_0} = e^{-0,7 \frac{R}{R_0}} \cdot e^{-1,08 \frac{S}{S_0}} \cdot \left(1,24 - e^{-2,43 \frac{T}{T_0}}\right) \cdot e^{-\left(0,023 \frac{N}{N_0}\right)^2} \quad (3.7)$$

Waar

P is de sensory pleasantness.

R is de ruwheid [asper].

S is de scherpte [acum].

T is de tonaalheid [tu].

N is de luidheid [soon].

Vanaf 14 soon wordt de sensory pleasantness nog weinig beïnvloed door de luidheid. Om een absolute pleasantness waarde te verkrijgen wordt dan ook een constante luidheid verondersteld. De andere referentiewaarden worden dan genormaliseerd naar 14 soon.

- N_0 wordt dus 1 soon, wat correspondeert met een sinusoidale toon van 1 kHz met $L = 40$ dB.
- $R_0 = 1,3$ asper bij $N = 14$ soon, $f = 2$ kHz, $f_{mod} = 70$ Hz en $m = 1$.
- $S_0 = 9,6$ acum bij $N = 14$ soon en $f = 8$ kHz.
- $T_0 = 1$ tu bij $N = 14$ soon en $f = 500$ Hz.

Met P_0 is 1 bij een 1 kHz toon van 60 dB wordt dat:

$$P = e^{-0,55R} \cdot e^{-0,113S} \cdot \left(1,24 - e^{-2,2T}\right) \cdot e^{-(0,023N)^2} \quad (3.8)$$

Daartegenover is ook een formule opgesteld die de annoyance bepaalt, meer bepaald de ‘unbiased annoyance’ (*unbeeinflusste Lästigkeit*), wat staat voor onbevooroordeelde ergernis. Met onbevooroordeeld worden laboratoriumomstandigheden bedoeld. Er kan geen fout ontstaan door een subjectieve houding van testluisteraars!

Dergelijke all-in-one formules worden dikwijls aangepast naar een model dat enkel voor een bepaalde klasse van producten geldig is. Dat model krijgt dan een naam als *Sound Quality Index for Hairdryers* en zou dan een goede indicatie moeten geven over de sound quality van een haardroger.

3.8 Productspecifieke parameters

Zoals aangehaald in de inleiding bij dit hoofdstuk zullen de psychoakoestische parameters meestal niet volstaan om een geluid volledig te omschrijven. Ze zullen eveneens meestal niet volstaan om aan de sound quality vereisten van een bepaald product te voldoen. De subjectieve akoestiek levert allerhande persoonlijke benamingen voor een geluid op. Geprobeerd wordt deze gevoelens zoveel mogelijk te standardiseren en te kanaliseren om alsnog een zo onpersoonlijk beeld van het geluid te verkrijgen. Woordenlijsten met mogelijke adjectieven bij geluid en meer specifiek bij productgeluid, zijn aangelegd om dat geluid zo concreet mogelijk te kunnen benoemen. Zo heeft Lyon de moeite genomen uit 23 000 Engelse woorden 1 300 geluidsgerelateerde te halen. Hij kon ze in drie categorieën splitsen naargelang hetgeen ze beschreven: de maker van het geluid, onze responsie op het geluid of het geluid zelf. Door nog eens de (bijna-)synoniemen en de weinig gebruikte termen weg te filteren, bekwam hij een lijst, zie

Tabel 3.1, van 62 beschrijvende woorden. Wegens de dikwijls voor zich sprekende woorden en eventuele onvertaalbaarheid ervan, is gekozen om deze lijst in het Engels over te nemen.

Tabel 3.1: Engelse adjectieven voor geluidsevaluatie.

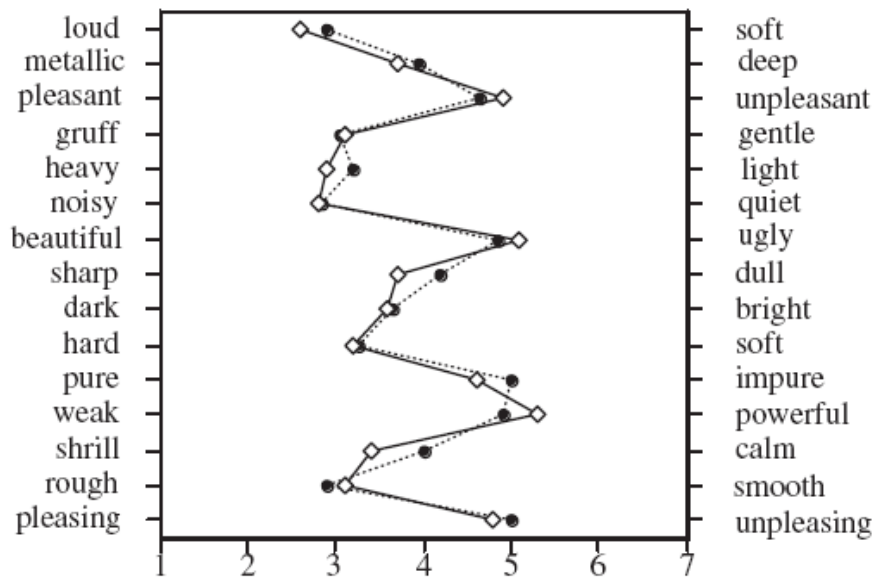
bassy	dull	jangling	rhythmic	ticking
booming	echoey	knocking	ringing	tinny
bright	even	mellow	roaring	tonal
buzzy	grating	melodic	rumbling	uneven
clanking	grinding	modulated	sharp	vibratory
clear	groaning	pinging	shrill	whirring
damped	gurgling	pounding	smooth	whistling
deep	gushing	pulsing	soft	muffled
discordant	harsh	purring	soothing	open
distant	hissing	rapping	squeaking	piercing
distorted	hollow	raspy	strong	
dripping	humming	rattling	swooshing	
droning	impacting	resonant	thin	

Tabel 3.1 is natuurlijk niet meer dan een hulpmiddel bij een geavanceerde sound quality bepaling van een product. Om fijne subtiliteiten, enkel voorkomend bij het te onderzoeken product, te kunnen benoemen, kan het nodig zijn neologismen aan te wenden.

Een voorbeeld daarbij is de dissertatie van Dedene; *Integration of sound quality in the design process of exhaust systems* [ref. 40]. Om de storende elementen uit het uitlaatgeluid van een willekeurige personenwagen te kunnen verwijderen door aanpassingen aan de constructie van de uitlaat, moesten deze nauwkeurig benoemd worden. Een team van experts beluisterde allerhande opnames van uitlaatgeluiden en gaf een naam aan de waargenomen storende

elementen. Dedene definieerde deze 24 elementen in een catalogus en reduceerde ze dan naar 9, onder andere door gelijkaardige elementen zoals *singing noise*, *whine* (gejank) en *turbo-charger whistle* samen te nemen onder *whistle* (gefluit). De negen storende factoren bij uitlaatgeluid zijn de volgende: gedreun/gebrom, acceleratiegeluid, gefluit, hol geluid, gejuuw, metallisch geluid, gekletter, geknal/gepruttel en stromingsgeluid.

Van de uitverkoren beschrijvende woorden relevant voor het product of de productklasse, worden dan parameters gemaakt. Dit gebeurt door ze een waarderingsschaal toe te kennen, meestal gewoon van nul tot tien. Scores worden toegekend door een test jury (zie , waarover meer in het volgende hoofdstuk, of, als een voldoende grote correlatie tussen bepaalde eigenschappen van een geluid en de appreciatie van een bepaalde parameter bewezen is, automatisch door een computeralgoritme. Een parameter wordt dan niet meer gekwantificeerd door bevraging, maar door berekening.



Figuur 3.11: Onderzoek naar de verschillen in de waardering van het geluid van een wagenportier bij Duitsers en Japanners aan de hand van productspecifieke parameters.

Door sterk te investeren in de ontwikkeling en verfijnen van geluidsparameters, enkel bruikbaar voor het eigen product, kan een bedrijf een knowhow en kennis opbouwen die het op akoestisch gebied een grote voorsprong geeft ten opzichte van concurrenten.

4 Beoordeling van Sound Quality

4.1 Inleiding

Vroeger, toen van een sound quality vakdomein nog geen sprake was, werden sommige producten, alhoewel niet intensief, toch al enigszins op hun sound quality getest. Een team normaal horenden oordeelde of het geproduceerde geluid aangenaam was, of sterk stoorde. Dit geeft wel een indicatie omtrent eventuele klankverbetering, maar is geen sluitende werkwijze. Nadelen zijn de tijd- en geldrovendheid van de procedure, maar vooral de te grote subjectiviteit van de beoordeling. De testgroep is te klein om van een objectief resultaat te kunnen spreken. Vandaag worden sound quality evaluaties op een wetenschappelijk meer verantwoorde manier aangepakt. Problemen die daarbij rijzen zijn zowel fysisch als psychologisch van aard. Sound Quality parameters kunnen niet gemeten worden, enkel berekend op basis van empirische formules die alle maar één variabele, de geluidsdruk, kunnen hebben. Daarenboven beïnvloeden allerlei factoren de menselijke perceptie.

Sound quality beoordelingen bestaan nu meestal uit twee delen. Eén deel bestaat erin opnames van het productgeluid voor te leggen aan een ‘testjury’. Op die manier kunnen ook gemakkelijk geluiden vergeleken worden; door verschillende voorstellen voor de klank van een product na elkaar te spelen, kan de luisteraar eenvoudig zijn voorkeur bepalen. Het tweede en meest geavanceerde deel is dat waarbij het opgenomen signaal door een computer verwerkt wordt. Na analyse zal deze dan een objectief sound quality oordeel vellen, zonder subjectieve, menselijke tussenkomst. De combinatie van een jury test en een objectieve benadering met software blijft meestal noodzakelijk. De sound quality parameters zijn namelijk gebaseerd op empirische vergelijkingen en kunnen verkeerde uitslagen geven. Enkel indien voor een specifiek product een correlatie tussen de subjective en de objectieve bevindingen gevonden wordt, kan volledig overgegaan worden op de computerberekende sound quality evaluatie.

Om geluidsoptnames voor sound quality doeleinden te maken, is de gepaste instrumentatie benodigd. Dit onderwerp komt aan bod opdat de lezer probleemloos de sound quality beoordelingsmethoden zou begrijpen.

Het hoofdstuk wordt afgesloten met enkele mooie voorbeelden van volledige sound quality onderzoeken. Deze concrete toepassingen stellen de lezer in staat de theoretische hoofdstukken 2, 3, en 4 enigszins uit de abstractie te halen.

4.2 Menselijke perceptie

Het grote probleem (uitdaging?) bij sound quality evaluatie is dat ze gebaseerd is op de menselijke perceptie en niet op puur fysische grootheden. Uw buurjongen zal het geluid dat zijn crossmotor produceert helemaal anders beoordelen dan u. En als het daarbovenop op nachtelijke uren is dat hij zijn rijkunsten bijschaaft, dan schiet uw beoordelingscijfer waarschijnlijk nog verder de diepte in. Nochtans betreft het hier telkens een identiek geluid!

In voorgaande hoofdstukken werd reeds aangetoond dat hetgeen we horen niet hetzelfde is als het geproduceerde geluid. Hoe we over dat geluid denken is echter nog een ander paar mouwen. Vooreerst hangt dat nauw samen met de prikkels die de andere zintuigen (zicht, reuk-, smaak- en tastzin) geven. Verder, en zelfs nog gecompliceerder, hebben ook niet-perceptuele aspecten een invloed op de sound quality beoordeling. Dit zijn de cognitieve invloeden en worden door Bodden in drie groepen verdeeld:

- Bron-gerelateerde factoren:
De gebruiker stelt zich een beeld voor van de bron (i.e. het product).
bv. Denk aan een stofzuiger.
- Situatie-gerelateerde factoren:
Een product wordt voor een specifieke activiteit gebruikt, de gebruiker kan interageren met de bron.
bv. Beeld u in dat u aan het stofzuigen bent.
- Persoon-gerelateerde factoren:
Mensen hebben hun persoonlijke verwachting, motivatie, smaak, voorkeur of aversie.
bv. Hoe vindt u dat een stofzuiger moet klinken?

Daaruit volgend kan één fysisch geluid compleet verschillende sound qualities hebben. Dit vergemakkelijkt de creatie van objectieve sound quality meeteenheden niet en maakt het voor een fabrikant dus moeilijker om een zo algemeen mogelijk publiek voor zijn product aan te trekken. Om toch enig inzicht in deze materie te verkrijgen, nemen ontwerpers, vooral deze van de auto's, psychologen onder de arm. In team met de ingenieurs proberen de psychologen zoveel mogelijk rekening te houden met de denkprocessen van de consument (*the human factor*).

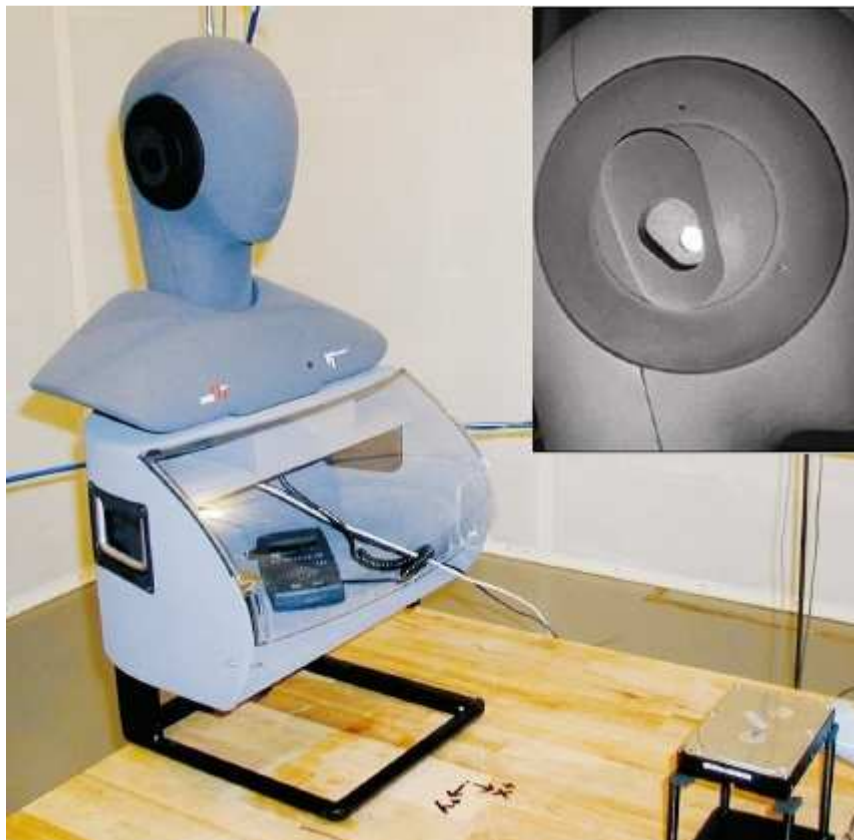
4.3 Meetuitrusting

Voor de evaluatie van sound quality is een meting met een gewone geluidsmeter ontoereikend. Alhoewel enkel een geluidsdrukmeting vereist is, moeten de nodige aanpassingen in de meetopstelling gebeuren om een opname te doen die zo goed mogelijk de responsie van het menselijk gehoor representeert.

4.3.1 Artificieel hoofd

Een artificieel of kunstmatig hoofd (*artificial head, dummy head*) is opgebouwd uit een kunsthoofd, geplaatst op een halve torso (*HATS; Head And Torso Simulator*). Hun basis bematingen zijn vastgelegd in de IEC-norm 959. In het hoofd (zie Figuur 4.1) zitten twee microfoons aan het einde van een gehoorgang ingebouwd. Een alleenstaande microfoon wordt niet gebruikt omdat het menselijk uitwendig oor een richtingsfilter (*directional filter*) is, wat betekent dat signalen uit bepaalde richtingen beter ontvangen worden dan andere. Dit komt door diffractie, reflecties en resonanties veroorzaakt door de bouw van het uitwendig oor. Een conventionele microfoon zal op alle signalen gelijk reageren.

Een tweede reden voor het gebruik van een artificieel hoofd zijn de twee microfoons die de twee oren voorstellen (*binaural hearing*). De mens heeft twee gehoorpaden en kan daardoor bronnen lokaliseren en een 3D-beeld van de omgeving maken.



Figuur 4.1: Geluidsofname van een harde schijf m.b.v. een artificieel hoofd.

Bij opnames met een artificieel hoofd moet wel een gelijkstelling (*equalisation*) doorgevoerd worden. De frequentie transferfunctie van een hoofd varieert met de richting van de geluidsinval, terwijl dat bij een standaard microfoon niet zo is. Om compatibiliteitsredenen worden de microfoonsignalen in het hoofd daarom aangepast. De buitenoor transferfunctie wordt opgesplitst in een richtingsafhankelijk gedeelte, beïnvloed door de externe menselijke geometrie (borst, hoofd, oorschelp), en een richtingsonafhankelijk gedeelte, dat uit de resonanties van de oorschelpholte en oorgang ontspringt.

Soms, als het plaatsen van een artificieel hoofd onmogelijk is, of de aanschaf ervan niet opportuun is, wordt ook gebruikt gemaakt van oortjes die in plaats van kleine luidsprekertjes, kleine microfoontjes zijn. Zodoende kan ook een waarheidsgetrouwe opname gemaakt worden.

Afhankelijk van het geluidsveld kan nu gekozen worden voor de instellingen ‘vrij veld’ (*free field*), diffuus veld (*diffuse field*) of richtingsonafhankelijk (*independant of direction*). De ‘vrij veld’ gelijkstelling zal dan de richtingsonafhankelijke transferfunctie verwijderen, maar de richtingsonafhankelijke gelijkstelling benadert de transferfunctie van een standaard microfoon het meest en staat het dichtst bij de werkelijkheid, namelijk dat een precieze bronlokalisatie bij sound quality bepaling meestal moeilijk is.

4.3.2 Microfoon

Voor de captatie van geluid worden condensatormicrofoons gebruikt. Ze zijn stabiel en betrouwbaar in hun werking en hebben een hoge precisie. Een condensatormicrofoon werkt volgens het principe van de afnemende elektrische capaciteit bij toenemende afstand tussen de condensatorplaten. Een van de twee platen is een erg licht diafragma, dat beweegt in responsie op akoestische (lucht)drukvariaties. De variërende plaatafstand zorgt voor een variërende capaciteit, die op zijn beurt voor veranderingen in het potentiaalverschil over de condensator zorgt. Een voltmeter(tje) in het circuit detecteert dat verschil. Als nu de vaste verhouding van de spanning ten opzichte van de druk in de microfoon gekend is, kunnen de oorspronkelijke geluidsdrukken gereconstrueerd worden.

4.3.3 Meetomgeving



Figuur 4.2: Een artificieel hoofd en torso geplaatst in een anechoïsche kamer.

Belangrijk bij metingen is de omgeving of ruimte waarin gemeten wordt. Idealiter vinden de opnames plaats in een anechoïsche kamer (zie Figuur 4.2); dat is een kamer waar alle geluidsgolven geabsorbeerd worden en geen echo is, als in een open veld. Een semi-anechoïsche kamer is gelijkaardig, maar heeft een volledig weerkaatsende bevloering. Deze geeft, zeker bij wegtransportvoertuigen, een realistischer geluidsimmissie. Soms worden de metingen echter gewoonweg op de normale locatie van het te onderzoeken product gedaan, bv. een koelkast in de keuken. Dit omdat een geluid soms dermate vervormt in een anechoïsche kamer dat er geen relevante informatie m.b.t. de sound quality meer uitgehaald kan worden. Nadeel is dan dat objectief vergelijken tussen productgeluiden moeilijk wordt.

4.4 Testjury

Gemaakte opnames kunnen op hun sound quality beoordeeld worden door een selecte groep luisteraars. Dit gebeurt met een hoofdtelefoon en een invulformulier. Voordelen van werken in laboratoriumomstandigheden zijn:

- De test is reproduceerbaar.
- Alle subjecten hebben identieke testomstandigheden.
- Geluiden kunnen direct vergeleken worden.
- Mogelijke subjectiviteit veroorzaakt door de andere zintuigen wordt uitgeschakeld.

- De snelle opeenvolging van geluidsfragmenten resulteert in tijdsefficiënt werk.
- Allerhande kunstmatige geluiden kunnen aan de reeks toegevoegd worden. Ook geluiden, aangepast aan de antwoorden van de luisteraar. Dit kan de bouw van overbodige prototypes sterk beperken.

Werken onder laboratoriumcondities houdt echter ook bepaalde nadelen in. De kans bestaat dat een subject anders reageert op een geluid als hij het product bij zich heeft. Alle zintuigen worden terug ingeschakeld en deze beïnvloeden het denkproces sterk. Dit is het reeds beschreven cognitieve aspect. Om in de sfeer te blijven; buurjongen zou ook niet gelukkig zijn moest zijn crossbrommer als een grasmachine klinken. Bij gekende producten, die dus al langere tijd op de markt zijn, is er namelijk een verwachtingspatroon (*expectancy*) wat het geluid betreft.

Daarvoor zijn twee mogelijke oplossingen. Ofwel wordt de test ‘in het veld’ gedaan, ter plaatse dus, ofwel worden de cognitieve factoren ondervangen door de testjury de nodige informatie mee te geven. Deze deelt mee over welk product het gaat (bron-gerelateerd) en legt de situatie uit aan het subject waarin hij/zij zich bevindt (situatie-gerelateerd) zodat hij/zij zich daarop kan baseren bij de geluidsbeoordelingen. De persoon-gerelateerde factoren die dan opspelen zullen sterker zijn naarmate het subject beter vertrouwd is met het product en de situatie. Een vragenlijst darover kan duidelijkheid scheppen.

Er zijn twee belangrijke manieren om de testjury te bevragen. Ten eerste met relatieve vragen, genre ‘Welk geluid, A of B, is het meest aangenaam?’, *paired comparison* geheten. De andere, minder tijdsintensieve manier van enquêtering is met absolute vragen. Hierbij moet de luisteraar op een bepaalde schaal, bv. van hard tot zacht, oordelen hoe hij het afgespeelde geluid ervaart. Figuur 4.3 toont een dergelijke bevraging handelend over de evaluatie van het impactgeluid bij de slag op een golfbal. Die testmethode wordt *semantic differential test* genoemd of ook nog *direct-magnitude estimation test*.

<i>How did the shot feel?</i>										
Unpleasant	1	2	3	4	5	6	7	8	9	Pleasant
<i>How did the impact sound?</i>										
Very dull	1	2	3	4	5	6	7	8	9	Very crisp, sharp
<i>How loud was the impact sound?</i>										
Quite, dead sound	1	2	3	4	5	6	7	8	9	Loud, explosive sound
<i>How did the impact feel in your hands?</i>										
Very soft	1	2	3	4	5	6	7	8	9	Very hard
<i>How quickly did you perceive the ball to have come off the clubface?</i>										
Dead, ball came off slowly	1	2	3	4	5	6	7	8	9	Lively, ball came off quickly

Figuur 4.3: Vragen gebruikt om de perceptie van golfers te kwantificeren.

De verzamelde gegevens van de testjury worden na afloop van de enquête statistisch verwerkt. Gebruikte statistische methoden zijn de paarsgewijze T-test, betrouwbaarheidsintervallen, correlatie en lineaire regressie. Deze moeten aantonen of er consistente verbanden bestaan tussen geluiden en de waardering ervan door bepaalde personen.

Om het vele werk dat daar bij komt kijken te verlichten hebben grote fabrikanten van akoestisch materiaal softwarepakketten ontwikkeld, waarmee vlot de ‘jury tests’ kunnen worden ingevuld en waarbij de nodige berekeningen automatisch gebeuren. Voorbeelden zijn Brüel & Kjær met hun ‘Psychoacoustic Test Bench’ en Head Acoustics met ‘SQuare’.

4.5 Software

Om sound quality beoordelingen en analyses nog sneller en uniformer te doen verlopen, moet de schakel mens als jury van tussen de ketting gehaald worden en vervangen door een computer. Het signaal van de microfoon wordt gedigitaliseerd; het continu signaal wordt een discreet signaal, ondergaat een fouriertransformatie en wordt opgesplitst in frequentiebanden. Met die gegevens worden de algoritmes, op basis van formules, doorlopen om de sound quality parameters te berekenen. Deze formules zijn zo opgesteld dat ze een objectief resultaat geven voor de verschillende parameters. Op die manier kunnen verschillende geluiden op een identieke manier onderzocht worden, zonder twijfel over ongelijke perceptie. Nadeel bij softwareberekening van sound quality is het totaal afwezig zijn van inlevingsvermogen. Een crossbrommer of een grasmachine hebben voor de computer evenveel emotionele waarde: geen. Dat is een lacune die de gebruiker van de software moet invullen met zijn ervaring.

Een nadeel van andere aard is de moeilijke opstelling van de formules en algoritmes om tot een resultaat te komen dat de menselijke perceptie zo dicht mogelijk benadert. Er zijn al verdienstelijke pogingen gedaan om de perfectie te bereiken, maar zoiets ligt niet voor de hand. Bedrijven die sound quality softwarepakketten ontwikkelen doen dit ofwel vanuit hun knowhow over geluid en trillingen (*NVH*), ofwel hebben zij een geschiedenis in de akoestische wereld.

Tabel 4.1 Tabel 4.1 somt enkele bekende softwarepakketten op. Deze laten allen toe om de gebruikelijke psychoakoestische sound quality parameters te berekenen evenals aanvullende, productspecifieke parameters, zoals bv. sportiviteit bij een auto-uitlaat. Elke fabrikant probeert de formules nog wat bij te schaven zodat ze dicht bij de ‘perfecte objectieve meeteenheid’ komen. Standaard zijn die formules overigens enkel geldig voor mono geluiden (*monaural hearing*). Ook kunnen geluiden virtueel bijgewerkt worden zodat hun eigenschappen verbeteren. Alles is sterk gevisualiseerd, met grafieken allerhande. De pakketten zijn echter duur in aankoop wegens hun specificiteit.

Tabel 4.1: Commerciële sound quality softwarepakketten en hun leveranciers.

Bedrijf	Product	Ref.
01dB-Metravib	dBsonic	[61], [62]
Akustik Technologie Göttingen	Si++ Workbench	[63]
Brüel & Kjør	Pulse Sound Quality Software – Type 7698	[64]
Head Acoustics	ArtemiS	[65]
LMS	CADA-X Sound Quality Engineering	[66]
Müller-BBM	PAK Psychoacoustics Module	[67]
ViAcoustics	NAE Sound Quality Analyzer	[68]

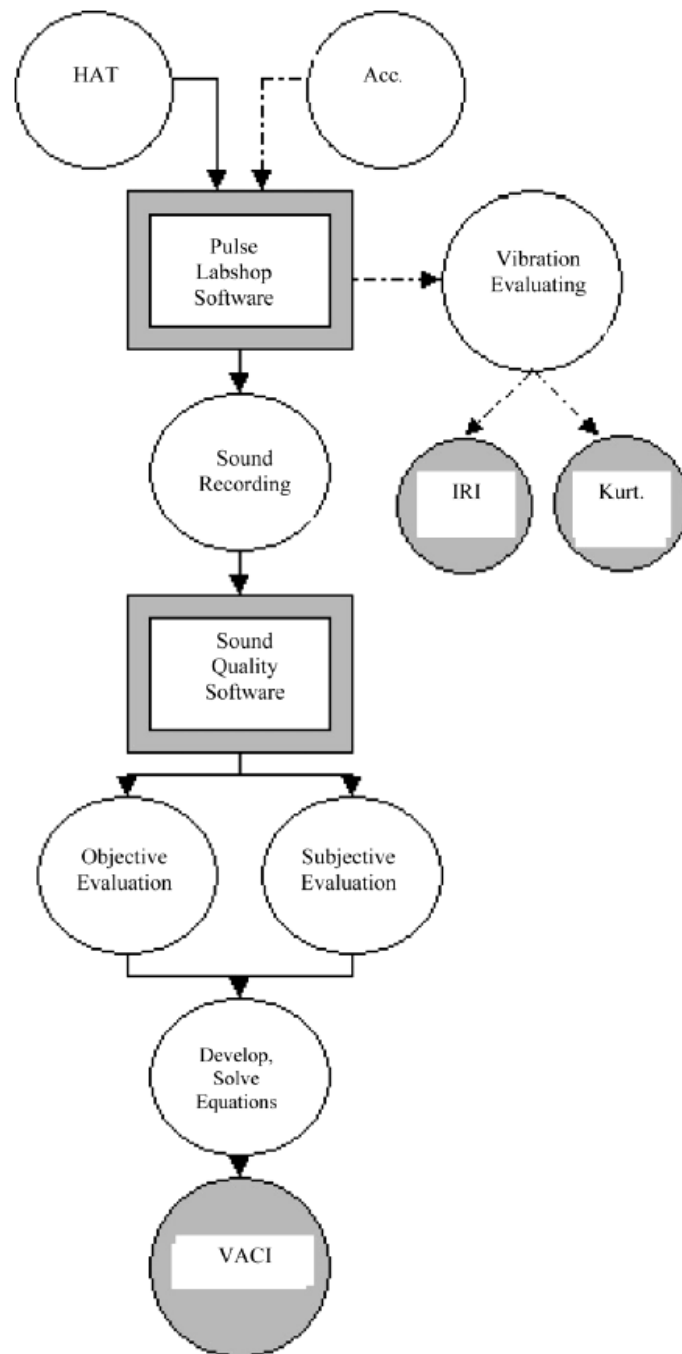
4.6 Enkele toepassingen

In de inleiding werd reeds vermeld dat onderzoek naar sound quality vooral in de transport- en huishoudtoestellensector gebeurt. Dit wordt in deze paragraaf gestaafd met twee typevoorbeelden. Het derde mooie sound quality project betreft een poging de hinder van windmolens te verminderen en het vierde is een buitenbeentje. Hier wordt namelijk gepoogd om met behulp van de psychoakoestische parameters informatie auditief over te brengen. Alle vier leggen ze, elk vanuit hun invalshoek, de nadruk op een ander aspect van de sound quality meting.

4.6.1 Akoestisch comfort in een personenwagen

Referentie [71]: Nor MJM et al., *Index for vehicle acoustical comfort inside a passenger car*.

De auteurs van bovenvernoemd artikel, onderzoekers uit Maleisië en Iran, wilden een index ontwikkelen die zowel subjectieve als objectieve sound quality parameters bevat. Deze index, in opzet vergelijkbaar met de allesomvattende annoyance parameter, moet er voor zorgen dat de akoestiek in een personenwagen ideaal is in zoveel mogelijk verschillende omstandigheden. Met verschillende omstandigheden bedoelen ze de wegdekken, die sterk kunnen variëren. Vijf verschillende types wegen werden bereiden op verschillende snelheden, terwijl opnames gemaakt werden met een HATS (Head And Torso Simulator, zie paragraaf 3). Een accelerometer, bevestigd aan de vloerplaat diende om de hobbeligheid of ruwheid van de weg te meten. Er bestaat namelijk een maat, de *International Roughness Index (IRI)*, die de ruwheid van het wegdek in functie van een voertuig aangeeft. De auteurs poogden een verband te vinden tussen deze index en de *Vehicle Acoustic Comfort Index (VACI)* die ze zelf ontwikkelden. Voor de evaluatie van de trillingen gebruikten ze ook nog de kurtosis. Dat is een maat voor de ‘piekvormigheid’ of impulsiviteit van een signaal. Deze meeteenheid wordt nog wel meer gebruikt bij sound quality analyse. Figuur 4.4 geeft een overzicht.



Figuur 4.4: Schematisch overzicht van de onderzoeksprocedures.

Zowel voor de captatie van het geluid en de trillingen als de verwerking ervan werd met Brüel & Kjær apparatuur gewerkt. De jury, die de subjectieve evaluatie deed, kon aan geluiden scores toekennen op een schaal van 5 terwijl de objectieve evaluatie uit de bepaling van Zwicker luidheid, scherpte, fluctuatiesterkte en ruwheid bestond. Kunst is dan het bekomen van de VACI door correcte statistische analyse (het voorlaatste cirkeltje van Figuur 4.4). De significantie van de psychoakoestische parameters voor de verschillende wegdekken werd berekend, regressieanalyses werden uitgevoerd en correlatiefactoren werden gezocht. Dit resulteerde in vijf verschillende indices, afhankelijk van het wegdek. Zo is de formule voor de VACI op een autosnelweg de volgende:

$$VACI = -0,3L - 4,1S + 14 \quad (4.1)$$

Waar

L is de stationaire luidheid [soon].

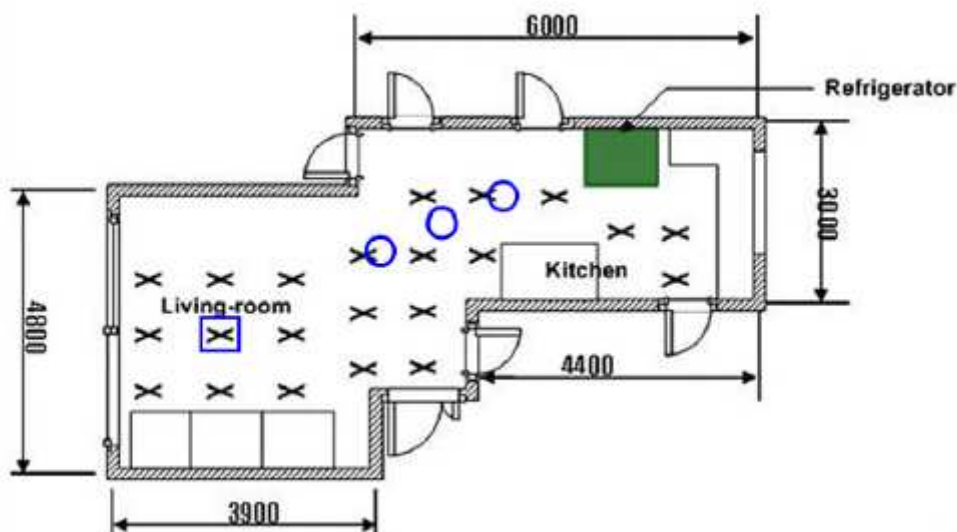
S is de scherpte [acum].

Verder toonden de onderzoekers niet veel wereldschokkends meer aan. Ze zagen dat de kurtosis toeneemt met de ruwheid van het wegdek en de snelheid van de wagen en dat de International Roughness Index omgekeerd evenredig is met de VACI. Dit ligt in de lijn der verwachtingen maar is hiermee toch op een wetenschappelijk onderbouwde manier bewezen.

4.6.2 Koelkastlawaai in verschillende omgevingen

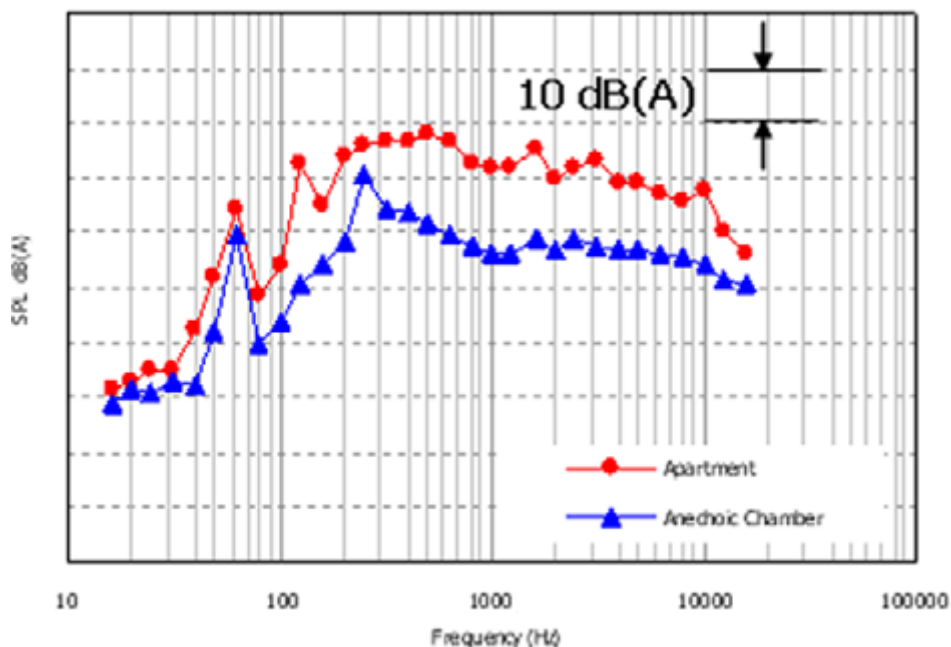
Referentie [72]: Jeon J.Y. et al., *Sound radiation and sound quality characteristics of refrigerator noise in real living environments*.

Ondanks de blijvende inspanningen van fabrikanten om het lawaai te reduceren, blijft een koelkast –en specifiek zijn compressor en ventilatoren– een ergerlijke geluidsbron. Omdat op het vlak van vermindering van de geluidssterkte maar weinig progressie meer te maken valt, kan het lonen om ook eens een sound quality onderzoek te doen op koelkasten. Het bijzondere aan deze studie is dat de metingen ook in een echte leefomgeving verricht zijn. Dit omdat een anechoïsche kamer een andere geluidskarakteristiek geeft dan een keuken. De geluidsuitstraling, frequentiekarakteristieken en geluidsdrukniveaus zijn er anders, wat een andere subjectieve consumentenrespons geeft. Praktisch werden vijf verschillende koelkasten in een flat in Seoel (de thuisstad van de auteurs) geplaatst, te zien op Figuur 4.5. De figuur toont ook de 25 plaatsen in de flat waar de geluidsdruk gemeten is.



Figuur 4.5: Grondplan van de flat met weergave van de meetposities.

Een eerste feit dat uit de metingen valt af te leiden, is dat in een anechoïsche kamer het koelkastlawaai beduidend stiller is dan in de reële leefomgeving. Op de samengevoegde grafieken in Figuur 4.6 valt duidelijk op dat de midden- en bovenfrequenties meer dan tien decibel luider klinken in de flat. De oorzaak ligt bij de geluidsreflectie van de muren van het gebouw. De meting voltrok zich overigens telkens op één meter afstand van de koelkast.



Figuur 4.6: Het verschil in de A-gewogen SPL van een koelkast in een flat en een anechoïsche kamer, is bij bepaalde frequenties erg groot.

Honderd testpersonen vormden het subjectieve deel van een sound quality beoordeling. Zij namen plaats in de flat en beoordeelden allerhande koelkastgeluiden, gereproduceerd door een luidspreker. Evaluatie gebeurde op de *noisiness* (lawaaierigheid) en *amenity* (aantrekkelijkheid) van het geluid en met een *semantic differential test*. Voor deze laatste werden 24 adjectieven en hun antoniemen geselecteerd, die onderverdeeld kunnen worden in drie groepen; *booming*, *metallic* en *discomforting sound*. De vijf koelkastgeluiden kregen een waardering in elk van de drie groepen, waarna een correlatie gezocht werd tussen deze en de klassieke sound quality parameters. Na onderzoek met lineaire regressie werd tenslotte een Sound Quality Index opgesteld, vergelijkbaar met de VACI in de vorige toepassing, maar nu geldig voor koelkastlawaai. Deze maakt gebruik van de berekenbare psychoakoestische parameters en zou in staat moeten zijn om de sound quality van een koelkastgeluid te voorspellen.

4.6.3 Geluidshinder van windmolens

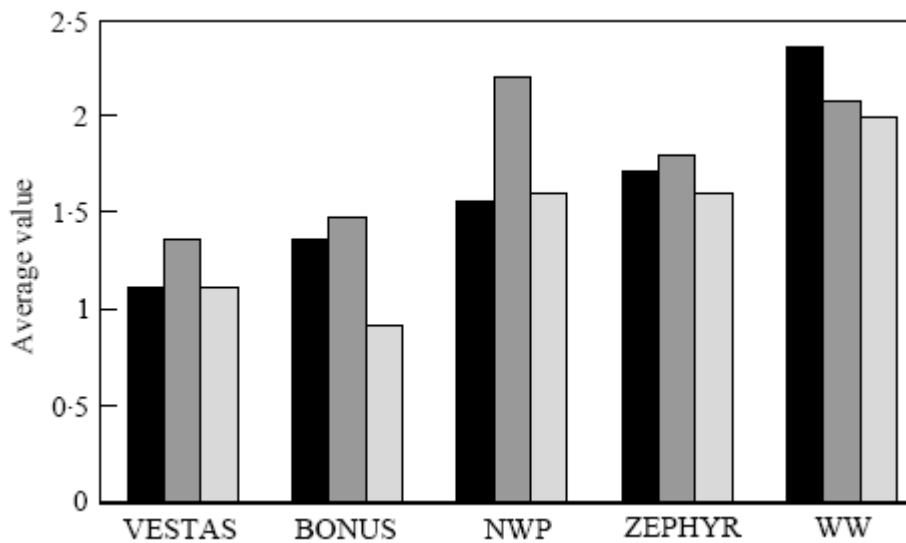
Referentie [70]: Persson Wayne K. & Öhrström E., *Psycho-acoustic characters of relevance for annoyance of wind turbine noise*.

Omdat een eerdere studie nauwelijks een verband kon aantonen tussen de geluidssterkte en de storendheid van een windmolen voor de omwonenden, wilden bovenstaande Zweden onderzoeken of het verband wel te leggen was met de psychoakoestische parameters van Zwicker en Fastl. Daartoe namen ze, op 100 m afstand, het geluid van vijf verschillende types windmolens op. Ze gebruikten daarbij een windscherm om ook bij hoge windsnelheden (tot 11 m/s) en dus bij hogere windmolenactiviteit, metingen te kunnen doen. Ze zochten stukjes van 20 s uit de opnames waar geen achtergrondlawaai aanwezig was en speelden die elk in een lus (*loop*) van meerdere minuten, alle met een gelijke L_{Aeq} van 40 dB(A). Om de in totaal 25 jury luisteraars, opgedeeld in groepjes van vijf, in de juiste sfeer te brengen, werd de kamer waarin de opnames afgespeeld werden, gedecoreerd als een openluchtomgeving met tuinstoelen en een parasol. Tijdens het binnenkomen van de kamer was eveneens vogelgefluit te horen. Ter gewenning aan het geluid van de windmolens konden ze gedurende tien minuten wat lezen.

De subjecten kregen vier vragenlijsten voorgeschoteld. De eerste polste naar hun houding tegenover windmolens, de tweede vroeg hen in welke mate ze het windmolengeluid vervelend vonden en gewaar waren. Naar het laatste werd bv. als volgt gevraagd: *How long did you pay attention to the wind turbine noise?* Er kon geantwoord worden op een schaal van vijf niveaus gaande van *not/nearly not/ aware of it at all* tot *the whole time*. De derde vragenlijst wilde weten in welke mate bepaalde psychoakoestisch gevoelige geluiden (*descriptors*) zoals een kunstmatig ruw, laag, schurend... geluid storend waren. Als laatste moesten de jury luisteraars de relatieve ergerlijkheid van de vijf windmolengeluiden waarden. Parallel hieraan werden de vijf windmolengeluiden ook aan een softwarematige (Brüel & Kjør) sound quality analyse onderworpen.

Significantietesten en correlatieanalyses van de verkregen gegevens gaven de mogelijkheid tot wetenschappelijk onderbouwde conclusies. Zo werd duidelijk dat enkel de kunstmatig gegenereerde geluiden die met de adjectieven *lapping*, *swishing* en *whistling* omschreven worden, significant tot de ergerlijkheid of het bewustzijn van het windmolengeluid bijdroegen. Deze kunnen het best naar het Nederlands vertaald worden als respectievelijk kabbelen, suizen en fluiten. Gezocht werd dan in welke mate deze eigenschappen in het geluid van de verschillende windmolentypes voorkwamen. Het resultaat is af te lezen op de Figuur 4.7. De auteurs achterhaalden daarbovenop nog andere correlaties, maar die waren minder invloedrijk op de perceptie van het windmolengeluid. Een andere vaststelling was dat de ergerlijkheid van de vijf windmolengeluiden verschilde, hoewel zij met dezelfde geluidssterkte afgespeeld werden. Een poging dit te verklaren met de gekende

psychoakoestische parameters liep echter op niets uit; de auteurs vonden geen aantoonbaar verband.



Figuur 4.7: Gemiddelde waarden voor de *lapping*, *swishing* en *whistling* van de verschillende wind turbine geluiden. De drie eigenschappen zijn in de opgesomde volgorde weergegeven.

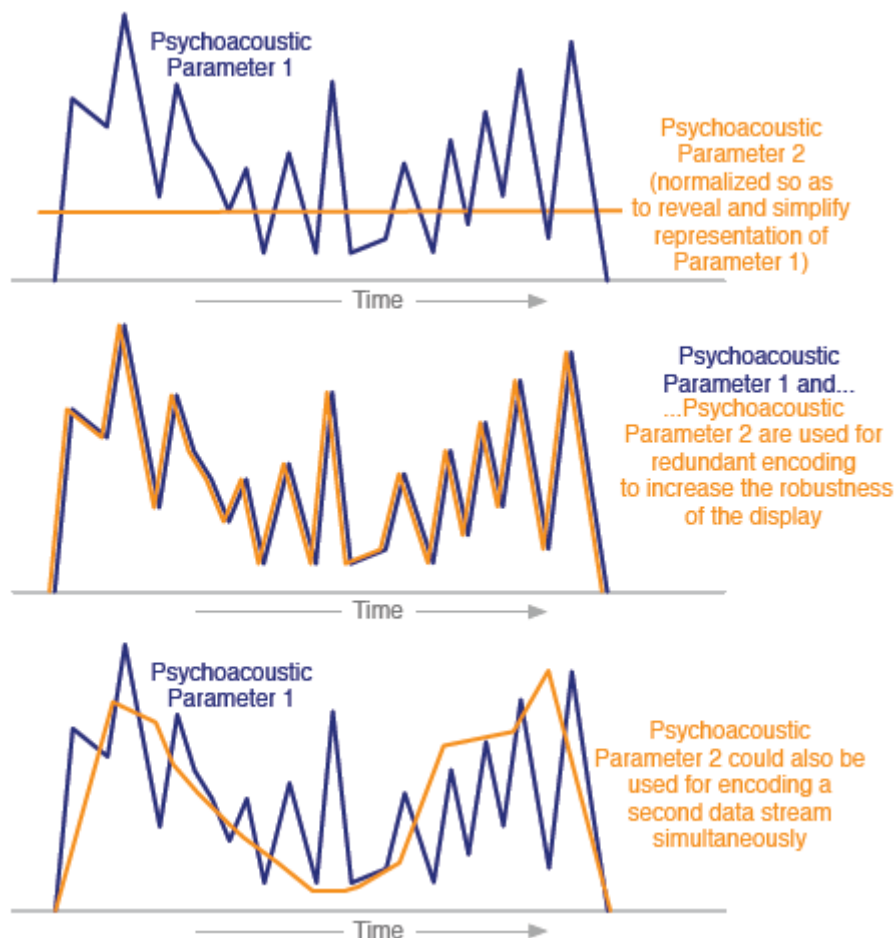
4.6.4 Auditieve informatieoverdracht

Referentie [69]: Ferguson S., Cabrera D. et al., *Using Psychoacoustical models for information sonification*.

De vierde en laatste toepassing van een sound quality analyse is de vreemde eend in de bijt. De auteurs gebruiken de psychoakoestische parameters namelijk niet om de Product Sound Quality te bepalen, maar om efficiënter auditieve informatie over te brengen. De auditieve overdracht van computergegevens heet *auditory display*, wat vertaald kan worden als auditief afleesinstrument en is een alternatief voor visualisatietechnieken. Een voorbeeld is de hartslagmachine die de chirurg tijdens de operatie met een regelmatige beep op de hoogte houdt van de toestand van de patiënt, zonder dat deze de ogen van de patiënt moet afwenden. Een auditieve display ondersteunt de gebruiker in het begrijpen van informatie. Dit kan gebeuren met of zonder spraak. Dat laatste heet sonificatie en is een vakdomein dat nog in de kinderschoenen staat. In deze studie werd onderzocht of de psychoakoestiek een rol kan spelen in de ontwikkeling van sonificatietechnieken. Door de sound quality cyclus toe te passen op mogelijke geluiden van een auditieve display, kunnen deze verfijnd worden, kunnen eventuele onbedoelde perceptieve effecten ervan getraceerd worden en kunnen ze gemakkelijker te interpreteren gemaakt worden. (cf. resp. pleasantness, annoyance en expectancy).

Voor de kwantitatieve benadering van de perceptie van de informatiedragende geluiden, worden de psychoakoestische meeteenheden luidheid, scherpte, ruwheid en fluctuatiersterkte gebruikt zoals opgesteld door Zwicker en Fastl. Daarnaast wordt ook de tonaalheid (*pitch perception*) bepaald. Deze parameters worden uitgebreid toegelicht in het artikel van de Australiërs. In deze bespreking van het artikel wordt dit deel weggelaten daar dit al gebeurd is in hoofdstuk 3.

Voor luisteraars is het niet zo moeilijk een fundamentele frequentie te ontwaren uit een geluid. Dat wordt een andere zaak als het over de psychoakoestische parameters gaat, die daarenboven alle in elk geluid in meerdere of mindere mate voorkomen. Daar kan op getraind worden, maar de parameters die niet beoordeeld moeten worden, kunnen ook anders weergegeven worden om de gevraagde parameter te versterken. De auteurs onderscheiden daartoe drie methodes, afgebeeld op Figuur 4.8. De eerste is de normalisatie van de andere parameters om de verwarring te reduceren. Als tweede worden de andere parameters naar een parallelle grafiek gemanipuleerd zodat ze de te bevragen parameter versterken, terwijl de derde de andere psychoakoestische sensaties gebruikt om verschillende data stromen weer te geven.



Figuur 4.8: Strategieën om met de verwarring van psychoakoestische parameters in auditieve displays om te gaan.

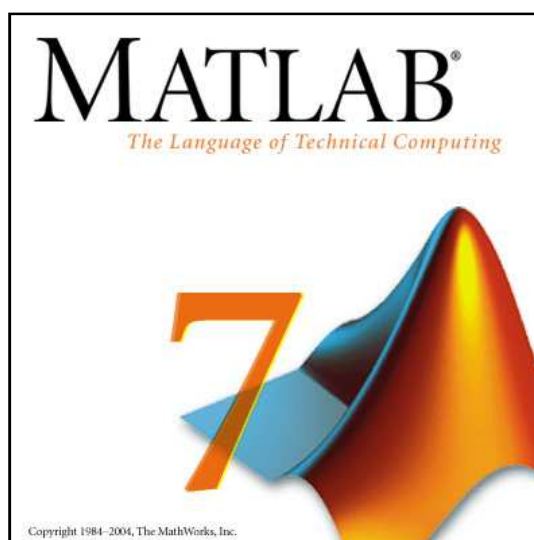
De opzet van het onderzoek was theoretisch; er is geen bestaand geluid geëvalueerd. Er werd een pool van geluiden aangelegd, waarbij alle parameters zowel maximaal als minimaal zijn. De geluiden varieerden voldoende om telkens een merkbare verandering in één parameter te veroorzaken. Daarbij was het nodig over kennis van de parametermodellen te beschikken, omdat een andere parameter soms ongewild mee verandert. Van al de eventueel aangepaste geluiden in de matrix worden dan hun bijhorende waarden van de vijf psychoakoestische parameters bepaald. Sonificatie kan dan gebeuren op basis van psychoakoestiek. De jury testen dienen wel met hoofdtelefoons te gebeuren omdat de frequentieresponsie van luidsprekers en de akoestische responsie van de kamer, net als de positie van de luisteraar in de kamer, alle factoren zijn die de perceptie kunnen beïnvloeden. De auteurs geven ook aan dat de hoofdtelefoon dient gekalibreerd te worden en dat de signalen van hoge kwaliteit moeten zijn.

Afsluitend maken de auteurs de bedenking dat hun werkwijze wel erg abstract is, maar ze vermoeden dat hun vernieuwende aanpak toekomst heeft. De beperkingen van hun aanpak liggen bij het vele werk voor de luisteraar die alle psychoakoestische meeteenheden moet evalueren en bij de modellen van die meeteenheden, die niet op punt staan.

5 Ontwikkeling van een computerprogramma

5.1 Inleiding

Meerdere fabrikanten hebben een sound quality programma op de markt. Deze analyseren een geluidssignaal en berekenen de psychoakoestische parameterwaarden, met daarbij eventueel nog zelfbedachte productspecifieke parameters. Het geheel is grafisch mooi ingekaderd en bevordert een vlotte sound quality bepaling. Elke fabrikant claimt het beste pakket te bezitten en kan dat ongestraft doen, want op luidheid³ na, is geen enkele parameterformule gestandaardiseerd. Bij elke release van een nieuwe versie van hun pakket zijn de gebruikte formules lichtelijk aangepast, om zo een perfectere benadering van de menselijke waarneming te verkrijgen. Deze formules of programmeercodes worden vanzelfsprekend niet vrijgegeven, alhoewel ze in essentie allemaal uitgaan van de bevindingen daaromtrent van autoriteiten in het vakgebied, zoals Zwicker, Fastl, Aures en Terhardt. De aankoopprijs van dergelijke hoogtechnologische software moedigt echter aan om een eigen poging tot ontwikkeling van een automatische berekening van de psychoakoestische parameters te ondernemen. De meest geschikte omgeving voor dergelijke implementaties van wiskundige formules is deze van het sterk ingeburgerde Matlab[®] (Figuur 5.1). De technisch georiënteerde programmeertaal van Matlab faciliteert het programmeren en Matlab zal databewerkingen sneller doorvoeren dan een ‘gewone’ computerprogrammeeromgeving.



Figuur 5.1: Matlab is zowat de standaard geworden wat het technisch programmeren betreft.

³ De berekening van luidheid volgens Zwicker is vastgelegd in de ISO532B norm, maar is digitaal onbruikbaar. Daarom is de DIN45631 norm uitgevaardigd, waarin een BASIC programmeercode voor de berekening van luidheid volgens Zwicker zorgt. Echter, andere onderzoekers zoals Moore & Glasberg en Aures ontwikkelden ondertussen andere (nauwkeuriger) berekeningswijzen voor luidheid. Dikwijls kan in de commerciële SQ-pakketten dan ook gekozen worden op welke manier de gebruiker de luidheid wenst te berekenen.

Om een geluid te kunnen beoordelen moet het vanzelfsprekend eerst opgenomen worden. Hoe dat aangepakt wordt en hoe dat verloopt wordt in de eerste paragraaf uiteengezet. De tweede behandelt de verwerking van het geluidssignaal om het ‘gebruiksklaar’ te maken voor de berekening van de psychoakoestische meeteenheden. Hierbij komt een groot stuk digitale signaalverwerking (*DSP, Digital Signal Processing*) kijken. Daarna komen de in hoofdstuk drie behandelde psychoakoestische parameters aan bod. Er wordt uitgelegd welke formules gebruikt worden en hoe deze als functies in het Matlab-programma verwerkt zijn. Als laatste wordt de structuur van het hoofdprogramma, **sqep.m**, uit de doeken gedaan. Het is dat script dat moet opgeroepen worden om de volledige sound quality analyse uit te voeren. In bijlage kunnen alle codes gevonden worden.

5.2 Captatie van het geluid

5.2.1 Captatie van de geluidsdruk

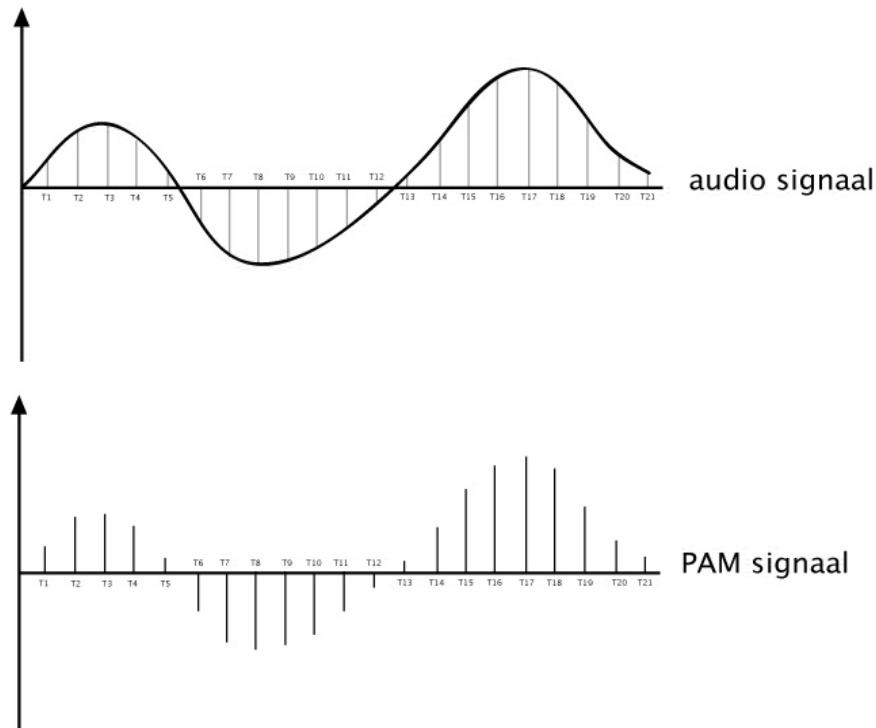
Idealiter gebeurt de geluidsdrukcaptatie met een artificieel hoofd, hoogkwalitatieve microfoons en in een aangepaste omgeving. Om reden van de dure aanschafprijs, die voor de school buiten proportie zou zijn, is dergelijke uitrusting niet voorhanden. Wel is een *LMS Pimento[®] Multichannel Analyzer* met een *PCB[®] electret ICP microfoon* ter beschikking, maar de beperkte mobiliteit en de hoge bezettingsgraad van de analyzer noopten tot onderzoek naar alternatieven. De eenvoudigste meetopstelling is deze waarbij een microfoon rechtstreeks op de geluidskaart van de computer aangesloten wordt. De computer leest dan de binnenkomende analoge elektrische signalen in.

5.2.2 Conversie van het analoge signaal

5.2.2.1 Theoretisch

Zoals beschreven in hoofdstuk 4 zet een microfoon de variërende geluidsdrukgolven om in evenredige elektrische, continue spanningssignalen. Elke microfoon heeft dan ook een bepaalde gevoeligheid, bv. 40 mV/Pa. Deze analoge spanningen worden op de geluidskaart eerst door een ADC (Analog to digital Converter) gediscretiseerd. Dit gebeurt met een bepaalde bemonsteringsfrequentie (*sample frequency, sampling rate, f_s*). Dat is de frequentie waarop, zoals zichtbaar op de Figuur 5.2, de continue waarden gevolgd worden. Volgens het Nyquist-Shannon-criterium moet deze dubbel zo groot zijn als de hoogste gemeten of relevante signaalfrequentie, om ondersampling tegen te gaan of, anders gezegd, het continue signaal terug te kunnen opbouwen. Hoe hoger de bemonsteringsfrequentie, hoe beter de originele golfvorm gereconstrueerd wordt. Ook de bestandsgrootte zal echter toenemen. Zo bedraagt de bemonsteringsfrequentie van digitale audio (muziek) standaard 44100 Hz. De helft ervan, 22050 Hz, is –met een marge– gekozen omdat de menselijke auditieve frequentiewaarneming toch niet verder reikt dan 20 kHz. Een 5 seconden durend signaal,

gesampled met een frequentie van 44100 Hz, zal dus 176 400 getalswaarden opleveren. Dit omzettingsproces heet de PAM: Pulse Amplitude Modulation.



Figuur 5.2: De nauwkeurigheid van het signaal na discretisering hangt af van de bemonsteringsfrequentie.

Een computer werkt echter binair; analoge data kan niet decimaal verstuurd worden. De noodzakelijke conversie van de analoge samples naar het binair stelsel heet kwantificeren. Hoe nauwkeuriger (getallen na de komma) het sample, hoe groter het bestand. Stel een monogeluid van cd-kwaliteit waarvan een sample uit 16 bits (= 2 byte) bestaat en de sample frequency 44100 Hz bedraagt. Uit die gegevens kan de bitsnelheid (*bitrate*) berekend worden; ze bedraagt 705,6 kbits/s. Deze conversie heet PCM, wat staat voor Pulse Code Modulation.

De gekende WAV geluidsbestanden zijn simpelweg de signalen in PCM-formaat, omhuld met een jasje waarop enkele bijkomende gegevens zoals titel en uitvoerder van het fragment staan. Het probleem van WAV-bestanden is hun grootte. Compressieformaten zoals MP3 nemen vele malen minder plaats op de harde schijf door onhoorbare geluiden weg te laten.

5.2.2.2 Praktisch

De doorsnee gebruiker heeft geen weet van de hierboven beschreven conversiemethode. Dat is niet nodig ook, want er zijn voldoende computerprogramma's waarmee geluidsopnames kunnen gedaan worden. Ook Matlab heeft dergelijke functionaliteit, met als bijkomend voordeel dat direct alle samplewaarden ter beschikking staan in een vector. Opnames in Matlab gebeuren met het volgende commando `wavrecord(N, Fs, CH)`.

Input

N is het aantal samples dat genomen moeten worden, bv. $4 * FS$ (=4 seconden).

FS is de sampling frequency.

CH is het aantal kanalen (1=mono, 2=stereo).

Matlab stelt ook nog de aanverwante functies `wavplay`, `wavread` en `wavwrite` ter beschikking. Het opgenomen geluid bevindt zich in de Matlab Workspace in matrixvorm. De getalswaarden van het databestand zijn de analoge, discrete amplitudes. Met het `plot` commando kan het opgenomen signaal in het tijdsdomein bekeken worden.

5.3 Verwerking van het geluidssignaal

5.3.1 Kalibratie van het geluidssignaal

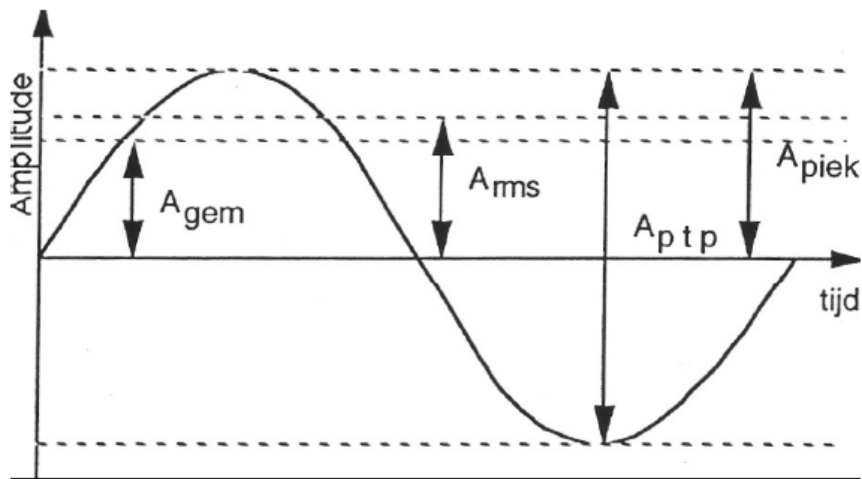
5.3.1.1 Theoretisch

Het probleem bij de omzetting naar een WAV-bestand is dat de amplitudewaarden wel relatief, d.i. ten opzichte van elkaar, correct zijn, maar niet absoluut. Idealiter zou de absolute geluidsdruk verkregen worden door de eenvoudige deling van de amplitudewaarden, die de binnengelezen microfoonspanning representeren, door de microfoongevoeligheid. Dan zouden de amplitudes van het signaal perfect de oorspronkelijke geluidsdrukken voorstellen. Echter, de amplitude van een geluidssignaal bepaalt het volume ervan en dat absolute volume heeft geen enkel nut bij het beluisteren van geluid (muziek), want de luisteraar draait zelf zijn volumeknop naar een niveau naar keuze. Daarom zijn alle amplitudes van een WAV-bestand verschaald tot waarden tussen -1 en 1. Dit maakt deze waarden echter onbruikbaar om luidheids- en andere metingen op te doen, daar zij niet de opnamesituatie weerspiegelen.

Er zijn meerdere oplossingen voor het probleem. Zo zouden de spanningswaarden rechtstreeks, zonder verscaling kunnen binnengelezen worden en dan gedeeld door de microfoongevoeligheid. Deze manier van werken blijkt echter heel moeilijk omwille van de geïntegreerde werking van een geluidskaart. Speciale computerprogramma's, die als een oscilloscoop de spanningen traceren en naar buiten brengen bestaan wel, maar geven enkel een analoog signaal, terwijl de digitale waarden noodzakelijk zijn. Deze zijn misschien wel naar buiten te krijgen, maar vereisen kennis van computerarchitectuur en lage programmeertalen en dat ligt te ver buiten het bestek van deze masterproef. De *LMS Pimento* analyzer en software bieden overigens wel de mogelijkheid om, na opgave van de microfoongevoeligheid, opgenomen geluidssignalen als geluidsdrukken in Pascal weer te geven.

Een tweede mogelijkheid bestaat uit het binnenlezen van de spanningssignalen via de seriële poort. Matlab ondersteunt communicatie via de seriële poort, wat de captatie heel wat gemakkelijker maakt. Spijtig genoeg is ook deze optie weinig realistisch, omdat het RS 232 protocol met vaste spanningen werkt. Een externe voeding is noodzakelijk, samen met een microcontroller om buiten de computer de analoog-digitaal conversie uit te voeren. Deze manier van werken zou de mobiliteit van de opstelling sterk beperken en ligt ook niet voor de hand. Daarbovenop is de gevoeligheidsverhouding van de microfoon vereist, en deze is dikwijls maar bij benadering correct.

Uiteindelijk lijkt de beste manier van werken deze waarbij tegelijkertijd een opname van het geluidssignaal en van L_p - of het geluidsdrukniveau, meetbaar met een decibelmeter, gebeurt. Als van beide hun waarde op een identiek tijdstip gekend is, kan, na terugrekening naar lineaire geluidsdruk, het verband tussen de signaalwaarden en de geluidsdruk berekend worden. Praktisch is het echter onmogelijk om beider waarden op exact hetzelfde ogenblik te verkrijgen. Daarom wordt de effectieve waarde (*RMS Value; Root Mean Square*) van het signaal gebruikt. Zie Figuur 5.3.



Figuur 5.3: De effectieve of RMS waarde van een zuivere sinus kan gebruikt worden om het signaal te kalibreren.

De formule voor de berekening van de effectieve waarde van een verzameling van n getallen gaat als volgt:

$$x_{eff} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \quad (5.1)$$

Waar

x_{eff} is de effectieve waarde.

n het aantal getallen.

x_i het i -de getal in de reeks.

Tegenover de effectieve waarde van het signaal staat het equivalent geluidsniveau (L_{eq}) van de decibelmeter. Deze stelt net hetzelfde voor, maar voor continue signalen is een integratie nodig. De formule verheldert:

$$L_{eq,T} = 10 \log_{10} \frac{1}{T} \int_0^T \left[\frac{p(t)}{p_0} \right]^2 dt \quad (5.2)$$

Waar

$L_{eq,T}$ is het equivalente geluidsdrukniveau over een periode T [dB].

T is de periode waarover de meting gebeurt [s].

$p(t)$ is de ogenblikkelijke geluidsdruk op het tijdstip t [Pa].

p_0 is de referentiedruk van 20 μ Pa.

5.3.1.2 Praktisch

In de praktijk betekent voorgaande manier van werken dat met twee opnameapparaten gewerkt moet worden. Dit is nadelig voor de nagestreefde mobiliteit en compactheid van de testopstelling, en bevordert eveneens de nauwkeurigheid niet. Daarom zijn andere pistes onderzocht. Deze van de in het lab Industriële Fysica en Toegepaste Mechanica beschikbare Ono Sokki LA 220 decibelmeter bleek de eenvoudigste. Het AC outputkanaal op de decibelmeter wordt d.m.v. een TRS-connector (Tip Ring Sleeve) verbonden met de geluidskaart van de computer. Captatie van het geluidssignaal gebeurt nu via de microfoon van de decibelmeter. De decibelmeter zelf wordt ingesteld op een ongewogen, equivalente geluidsdrukmeting van bv.10 seconden (keuze niet vrij), en in Matlab wordt tegelijk een `wavrecord` uitgevoerd, eveneens van 10 seconden. Na het verstrijken van de ingestelde tijd toont de decibelmeter dan het equivalente geluidsniveau, berekend over de ingestelde periode.

In de Matlab programmeercode is een functie aangemaakt, **calibratesignal.m**, waarin de tijdsvector die alle amplitudewaarden van het opgenomen signaal bevat, vermenigvuldigd wordt met een kalibratiecoëfficiënt c , zodat de amplitudewaarden terug de originele geluidsdrukken voorstellen en dus de eenheid Pascal kunnen krijgen. De volledige code kan in bijlage gevonden worden, maar de functie werkt als volgt:

```
[y]=calibratesignal(signal,Lp)
```

Input

`signal` is de niet-gekalibreerde tijdsvector van het signaal.

`Lp` is het equivalente geluidsdrukniveau, afgelezen op de decibelmeter.

Output

`y` is de gekalibreerde lineaire tijdsvector.

5.3.2 Filtering van het geluidssignaal

Om de luidheid van een geluid te kunnen bepalen, moet de energie-inhoud van alle kritische banden gekend zijn. Daartoe moeten technieken uit de digitale signaalverwerking aangewend worden. De nodige Matlab programmeercode is, omwille van de complexiteit van de materie, in grote mate overgenomen van Hastings' Matlab applicatie om luidheid te berekenen. Omdat

het toch nodig is te weten wat allemaal gebeurt in de signaalverwerkende functies, worden ze beknopt uitgelegd. Doel van deze voorbereidende ‘behandelingen’ van het signaal is het te filteren in tertsbanden. Dat is nodig om de luidheid te kunnen berekenen. Filteren van een signaal betekent het doorlaten van bepaalde frequenties, in casu een tertsband (*bandpass*) en het onderdrukken van de andere frequenties (*bandstop*).

5.3.2.1 Berekening van het vermogensspectrum

Als eerste wordt de functie **Powspec.m** opgeroepen. Deze berekent het vermogensspectrum van het signaal. Dat is een weergave van het vermogen bij elke (berekende) frequentie van dat signaal. De echte berekening gebeurt met het Matlabcommando `psd` (*Power Spectral Density*) dat werkt volgens de methode van Welch. De methode van Welch houdt in dat het frequentiespectrum van het signaal eerst in stukken gesplitst wordt. Het aantal secties waarin het spectrum opgedeeld wordt, ligt in **Powspec.m** vast in de variabele `NFFT`. Ze bedraagt oorspronkelijk, Nyquist indachtig, de halve bemonsteringsfrequentie, maar wordt dan naar boven afgerond tot de eerstkomende macht van twee. Uiteindelijk zal het vermogensspectrum de helft van dat laatste aantal waarden bevatten. De gebruikte vensterfunctie⁴ bij het filteren is de Hann functie (zie Figuur 5.4).

Daarna wordt van elke sectie het aangepaste periodogram, dat een weergave is van de spectrale (energie)dichtheid, berekend. Dit gebeurt door een discrete fouriertransformatie (DFT) uit te voeren en dan de bekomen waarden te kwadrateren. Dit zijn nu de excitaties, uitgedrukt in Watt. De syntaxis van de functie gaat als volgt:

```
[Yxx, f]=PowSpec(y, Fs, df)
```

Input

`y` is de gekalibreerde lineaire tijdsvector van het signaal.

`Fs` is de sampling frequency.

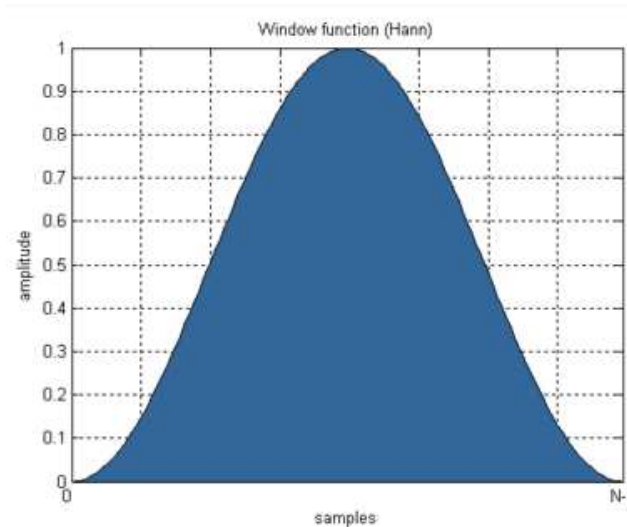
`df` is de tijd tussen twee samples of de frequentieresolutie, maar hier gewoon met waarde 2 ingeladen om de halve sampling frequency te kunnen berekenen. Later, in het hoofdprogramma, krijgt `df` wel zijn correcte waarde.

Output

`Yxx` is het vermogensspectrum van het signaal `y`.

`f` is een vector met alle frequentiewaarden van het vermogensspectrum.

⁴ Omdat bij het filteren een onmiddellijke scheiding tussen een niet en een wel doorgelaten frequentie voor oscillaties zorgt ter hoogte van deze scheiding -het Gibbs fenomeen- zijn methodes gezocht om deze scheiding geleidelijker te doen verlopen. Dit is nodig omdat anders, door de grote rimpel ter hoogte van de overgangen, amplitudes bij de ene frequenties overschat en bij de andere onderschat wordt. Voorkomen van het Gibbs fenomeen wordt voorkomen door anders dan rechthoekig te vensteren (*windowing*). Bekende windows zijn het Hamming window, het Hann window, het Kaiser window en het Chebyshev window. Deze hebben elk hun voor- en nadelen en worden gekozen naargelang de vereisten. Wiskundig gezien zijn de windows een functie waar alle amplitudes van het signaal doorgevoerd worden. De functiewaarde is afhankelijk van de frequentie.



Figuur 5.4: De Hann functie is een veelgebruikte manier om te 'vensteren'.

5.3.2.2 Conversie naar decibel

In **Convert2dB.m** worden de lineaire vermogenswaarden, zoals de naam het aangeeft, omgezet naar decibels. In paragraaf 3.2 over luidheid is reeds vermeld dat deze excitatieniveaus gebruikt worden om de luidheid te berekenen. De syntaxis van de functie is de volgende:

```
[YdB]=Convert2dB(Yxx, cal)
```

Input

Y_{xx} is het vermogensspectrum van het signaal y .

cal is een kalibratiecoëfficiënt. Die is hier niet van toepassing en wordt op 1 gezet.

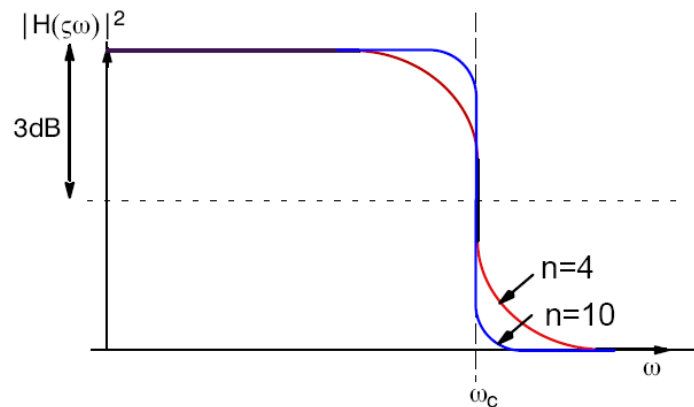
Output

Y_{dB} is het vermogensspectrum van het signaal y in decibels.

5.3.2.3 Aanmaak en implementatie van de tertsbandfilters

Het luidheidsmodel vereist een verdeling van het frequentiespectrum in tertsbanden. De aanmaak van de verschillende filters daarvoor gebeurt in de functie **GenerateFilters.m**. Ze stelt 29 tertsbandfilters op, met centrale frequenties van 25 tot 16000 Hz. De banden met een centrale frequentie onder 220 Hz worden geresampled naar een bemonsteringsfrequentie die lager ligt dan deze van het signaal. Alle tertsbanden worden aangemaakt met de functie **Oct3dsgn.m**, ontwikkeld door Couvreur. Die functie gebruikt een oneindige impuls responsiefilter (*IIR, Infinite Impulse Respons Filter*), meer bepaald een Butterworth filter. Een Butterworth filter wordt gekenmerkt door de frequentieresponsie die in de doorlaatband maximaal effen is. Voor de aanmaak van de tertsbandfilters wordt een derde orde filter gegenereerd. De invloed van de orde specifiek voor een Butterworth filter is zichtbaar op Figuur 5.5. De keuze voor een Butterworth filter heeft als reden dat deze gebaseerd is op een bilineaire transformatie (*bilinear transform*). Dit is gevraagd in de standaard

ANSI S1.11-2004, die specificaties voor analoge en digitale filters van (fractionele) octaafbanden vastlegt.



Figuur 5.5: De karakteristieken van een 4de en een 10de orde Butterworth filter.

De functie **GenerateFilters.m** heeft de volgende syntaxis:

`[H]=GenerateFilters(f,Fs)`

Input

f is een vector met alle frequentiewaarden van het vermogensspectrum.

F_s is de sampling frequency.

Output

H is een matrix met de frequentieresponsies van de 29 tertsbanden.

Buiten de functie, in het hoofdprogramma, wordt H gecorrigeerd naar beneden toe. Dit is omdat de totale energiewaarde van de tertsbanden anders groter zou uitkomen dan het origineel. Ook de eigenlijke berekening van de excitatieniveaus per tertsband gebeurt in het hoofdprogramma. De 29ste band, rond 16 kHz, wordt daarbij niet gebruikt. De dus 28 niveaus worden opgeslagen in de vector L_t .

5.4 Luidheid

5.4.1 Codering

De specifieke en totale luidheid worden bepaald in de functie **loudness.m**, die in essentie een omzetting is van de BASIC code, vermeld in de Duitse norm DIN45631. Haar syntaxis gaat als volgt:

`[N,Ns]=loudness(Lt,SF)`

Input

L_t is de vector met de excitatieniveaus van de 28 tertsbanden.

SF staat voor het geluidsveld, met de waarde D of F, die aangeven of het geluid in een diffuus resp. vrij veld (frontaal geluid) ontstaan is.

Output

N_S is de specifieke luidheid van de 24 kritische banden (onderverdeeld in trapjes van 0,1).

N is de totale luidheid.

Na definiëring van allerlei variabelen (centrale frequenties, experimenteel afgeleide correctiefactoren, wegingscoëfficiënten, conversieconstanten) worden eerst de elf laagste tertsbanden samengevoegd tot drie banden, net zoals bij de grafische methode. Bij de laagste frequenties zijn de 1/3-octaaftanden namelijk veel smaller dan de kritische banden. Samen met de overige tertsbanden worden zo $(28-11)+3=20$ banden bekomen. Na aanpassing aan het geluidsveld en enkele correctiefactoren wordt nog een eenentwintigste band toegevoegd. De specifieke luidheid van deze band zal niet gemeten worden, maar de waarde van de maskeercurve van de twintigste band bedragen. Deze 21 banden beslaan nu samen dezelfde bandbreedte als de 24 kritische banden van het menselijk auditief systeem, maar hebben vanzelfsprekend andere tussengrenzen. Dit wordt opgelost door de vector ZUP , die de tertsbanden herverdeelt volgens de kritische banden.

De specifieke luidheidsbepaling zelf gebeurt op de theoretische basis van de reeds vermelde evenredigheid tussen de specifieke luidheid en het excitatieniveau:

$$\frac{\Delta N'}{N'} = k \frac{\Delta E}{E} \quad (5.3)$$

Waar

$(\Delta) N'$ is de (verandering van de) specifieke luidheid [soon/Bark].

$(\Delta) E$ is de (verandering van het) excitatieniveau [dB].

k is een proportionaliteitsconstante.

Op basis van bovenstaande evenredigheid, en gebruik makend van de gehoordrempel voor de excitatie en een referentie voor de specifieke luidheid, hebben Zwicker en Fastl een formule opgesteld voor de bepaling van de specifieke luidheid via de spectrale verdeling van het excitatieniveau. Bemerkt de afhankelijkheid van de kritische bandwaarde:

$$N'(z) = N'_0 \left(\frac{1}{s} \cdot \frac{E_{HS}(z)}{E_0} \right)^{e_1} \cdot \left[\left(1 - s + s \cdot \frac{E(z)}{E_{HS}(z)} \right)^{e_1} - 1 \right] \quad (5.4)$$

Waar

$N'(z)$ is de specifieke luidheid in functie van de kritische bandwaarde z [soon/Bark].

N'_0 is de referentie specifieke luidheid, gekozen voor een SPL van 40 dB bij 1 kHz.

Deze bedraagt ongeveer 0,068 soon/Bark.

s is de verhouding tussen de intensiteit van een juist hoorbare testtoon en de intensiteit van een breedbandgeluid die in de zelfde kritische band voorkomt en wordt berekend als volgt: $s \approx 10^{(0,22-0,005z)} - 1$.

$E_{HS}(z)$ is de gehoordrempel.

E_0 stemt overeen met de referentiewaarde van geluidsintensiteit, d.i. $I_0=10^{-16} W/cm^2$.

e_1 is een experimenteel bepaalde exponent en bedraagt 0,23. Deze wordt echter aangepast naar 0,25 bij de berekening omdat daar met tertsbanden gewerkt wordt. $E(z)$ is het excitatieniveau over de oppervlakte van z .

In de Matlab code wordt deze formule, weliswaar in twee delen, als volgt herschreven:

$$N' = 0,0635 \cdot 10^{0,025 L_{HS}} \cdot \left[\left(1 - s + s \cdot 10^{0,1(L_E - L_{HS})} \right)^{0,25} - 1 \right] \quad (5.5)$$

Waar

N' is de specifieke luidheid [soon/Bark].

L_{HS} is de gehoordrempel logaritmisch. Deze wordt berekend met de volgende formule:

$L_{HS} = 3,64e^{-0,8 \ln f} - 6,5e^{-0,6(f-3,3)^2} + 0,001(f)^4$ met f de frequentie in kHz. In de code worden echter vaste, afgeronde waarden gebruikt. De eenheid van L_{HS} is dus [dB].

L_E is het gecorrigeerde excitatieniveau, bekomen met de functie **Convert2dB.m** [dB].

Nadien worden de specifieke luidheden, rekening houdend met spectrale maskering, de totale luidheid berekend.

$$N = \int_{z=0}^{24} N'(z) dz \quad (5.6)$$

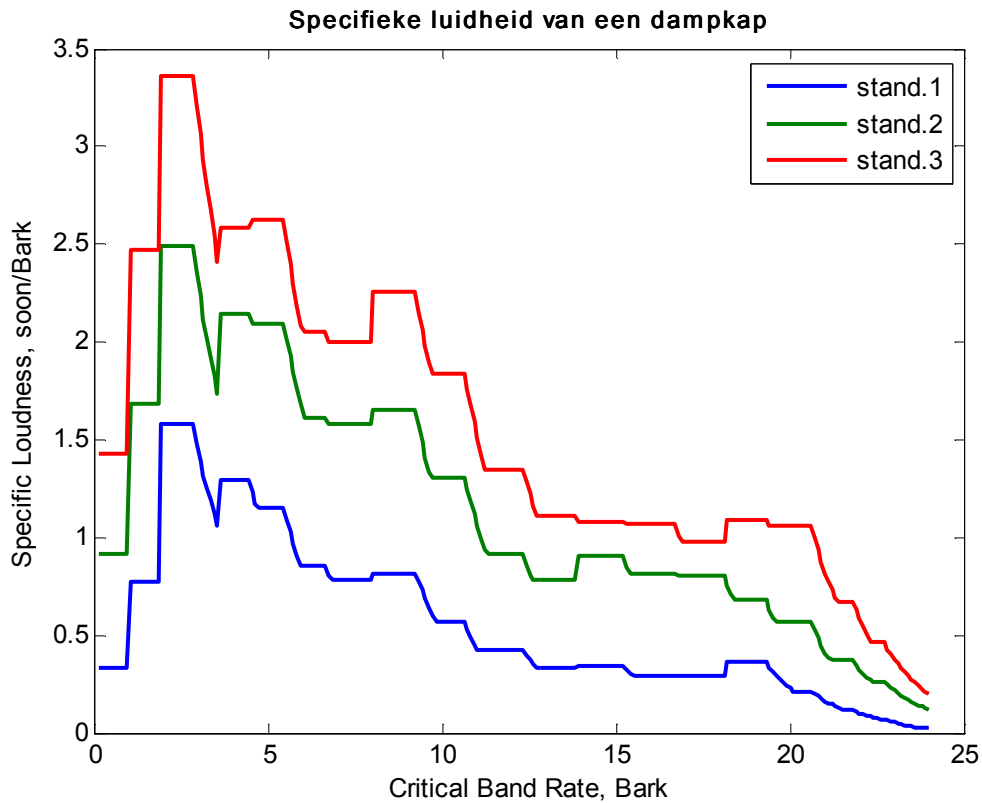
Waar

N is de totale luidheid [soon].

z is de kritische bandwaarde [Bark].

$N'(z)$ is de specifieke luidheid, afhankelijk van de kritische bandwaarde z [soon/Bark].

De totale luidheid wordt in de programmacode in 240 trapjes opgebouwd. Dit is zichtbaar in de dimensie van N_s , die 1x240 bedraagt. Dit is om de maskeringscurves waarheidsgetrouw te kunnen weergeven. Figuur 5.6 illustreert de treffende gelijkennis met een grafisch bepaalde luidheid. De spectrale maskering naar boven toe is zichtbaar in de exponentieel dalende curves naar rechts toe.



Figuur 5.6: Een automatische berekening van de luidheid van een dampkap in de drie standen.

5.4.2 Verificatie

De deugdelijkheid van de Matlab code in **loudness.m** staat buiten kijf. Naast het feit dat ze een omzetting is van een officiële norm, is deze code al uitvoerig getest, onder andere door vergelijking door Hastings met het Brüel & Kjær Pulse systeem. Een relatieve fout van mogelijk twee procent moet wel ingecalculeerd worden, maar een dergelijke mate van nauwkeurigheid is toch niet vereist.

5.5 Scherppte

5.5.1 Codering

De berekening van de scherppte gebeurt door de specifieke luidheden te integreren over alle kritische bandwaarden en dan te delen door de totale luidheid:

$$S = C \cdot \frac{\int_{z=0}^{24} N'(z)g(z)dz}{\int_{z=0}^{24} N'(z)dz} \tag{5.7}$$

Waar

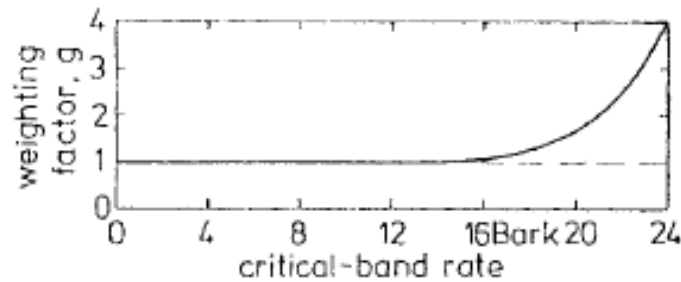
S is de totale scherppte [acum].

C is een proportionaliteitsconstante die de berekening naar het referentiegeluid mogelijk maakt en bedraagt ongeveer 0,11.

z is de kritische bandwaarde [Bark].

$N'(z)$ is de specifieke luidheid [S/Bark] volgens Zwicker.

$g(z)$ is een wegingsfunctie die de sterke scherptestijging bij hogere frequenties in rekening brengt door een exponentiële stijging. Figuur 5.7 toont zijn verloop.



Figuur 5.7: Het verloop van de wegingsfunctie $g(z)$.

In het Matlab programma is deze berekeningswijze, bedacht door von Bismarck en uitgewerkt door Zwicker & Fastl, toegepast. Anderen, zoals Aures, hanteren licht aangepaste formules. Ook wat de wegingsfunctie $g(z)$ betreft, zijn verschillende benaderingen in omloop. In de functie **sharpness.m** werd de volgende gebruikt:

$$g(z) \begin{cases} z < 15 \rightarrow g(z) = 1 \\ z \geq 15 \rightarrow g(z) = 0,2e^{0,308(z-15)} + 1 \end{cases} \quad (5.8)$$

Waar

$g(z)$ is een wegingsfunctie gebruikt bij de berekening van de scherppte.

z is de kritische bandwaarde [Bark].

Naar buiten toe ziet **sharpness.m** er uit als volgt:

`[S, Ss] = sharpness (Ns, N)`

Input

N_s is een vector met de specifieke luidheid van de kritische banden in 240 secties.

N is de totale luidheid.

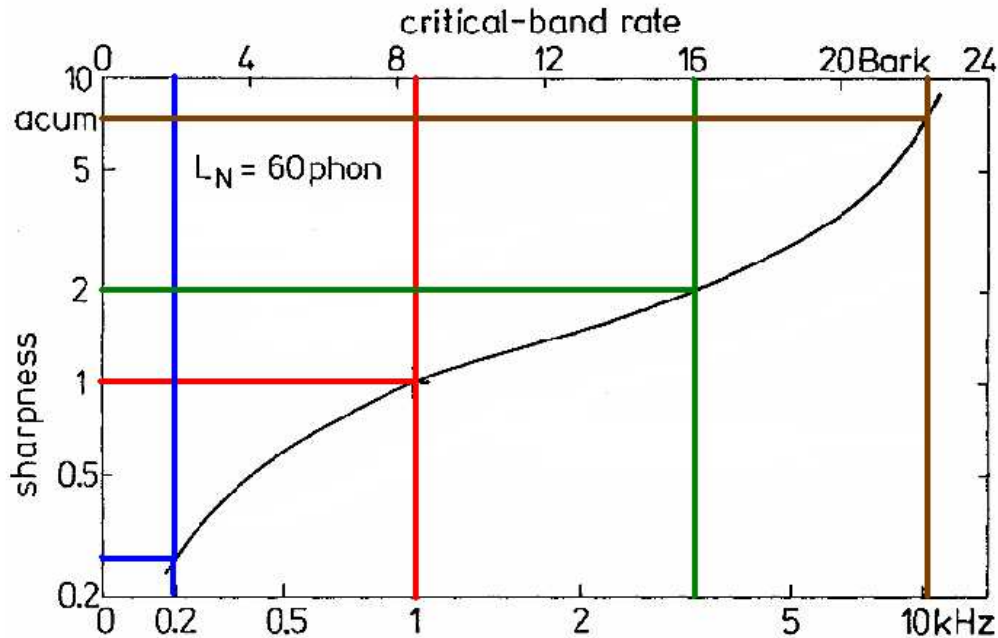
Output

S_s is een vector met de specifieke scherppte van de 240 secties.

S is de totale scherppte.

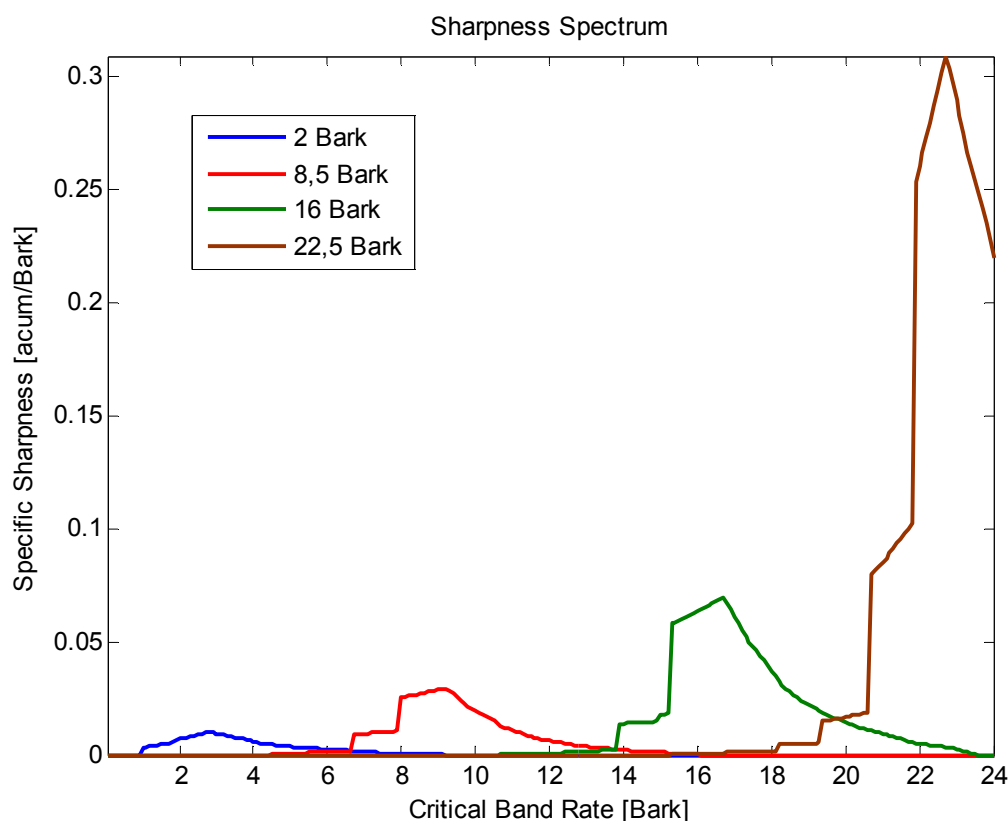
5.5.2 Verificatie

De verificatie of het algoritme van **sharpness.m** in al zijn eenvoud wel correcte waarden berekent, gebeurt aan de hand van de berekening van vier punten op de empirisch opgestelde grafiek op Figuur 5.8.



Figuur 5.8: Grafische afleiding van de scherpte op vier punten.

De signalen zijn smalbandruis van één kritische bandwaarde breed. Smalbandruis wordt gegenereerd door *white noise* te filteren op de gewenste bandbreedte (zie bijlage). Een band op de twee uiterste grenzen is getest, evenals de band waarbij de scherpte volgens de definitie juist één acum zou moeten bedragen en een band tussenin. Figuur 5.9 toont het scherptespectrum van de vier banden. Er moet ook over gewaakt worden dat het luidheidsniveau voor alle metingen zestig foon of vier soon bedraagt.



Figuur 5.9: Grafieken van de scherptes per bandwaarde van de vier bemeten smalbanden.

Tabel 5.1 vat de uitkomsten samen. De volgende conclusies kunnen getrokken worden:

- De 2 Bark band heeft een fout van een vijftal procent. Deze fout mag verwaarloosd worden omdat met dermate kleine getallen gewerkt wordt dat de nauwkeurigheid bij het aflezen van de grafiek en de berekening van de scherpte toch te klein is.
- Bij de kritische band rond 1000 Hz is de Matlab code perfect.
- De 16 Bark band vertoont een ernstige fout in de scherpteberekening van 25%. Dit is te wijten aan de functie $g(z)$, die hier klaarblijkelijk de scherpte overdrijft.
- Bij de bovenste berekende band van 22,5 Bark schiet de berekende scherpte dan weer 9% te kort. Ook dit komt door de wegingsfunctie $g(z)$. Vanaf 15 Bark stijgt de wegingsfunctie in waarde. Blijkbaar is die eerst te groot, om dan bij de bovenste frequenties te klein te zijn. Dit doet besluiten dat voor het programma geen andere wegingsfunctie gekozen moet worden, daar deze zo goed als mogelijk de ideale lijn, weergegeven in Figuur 5.7, volgt.

Tabel 5.1: Verificatie van de sharpness code.

Kritische bandwaarde [Bark]	2	8,5	16	22,5
Centrale frequentie [Hz]	200	1000	3150	10500
Bandbreedte [Hz]	100	160	500	2500
Kleur	blauw	rood	groen	bruin
Afgelezen scherpte [acum]	$\pm 0,3$	1	2	8
Berekende scherpte [acum]	0,31	1	2,5	7,3
Fout	ong. +5%	0%	+25%	-9%

5.6 Fluctuatiesterkte

5.6.1 Codering

Vanuit de algemene proportionaliteit van de fluctuatiesterkte met de maskeringsdiepte en de modulatiefrequentie, weergegeven in formule (3.4), hebben Zwicker & Fastl een model opgesteld. Hun empirische studies wezen uit dat voor de berekening van de fluctuatiesterkte van een geluid de volgende formulering best aangewend wordt:

$$F = \frac{0,008}{\left(\frac{f_{mod}}{4}\right) + \left(\frac{4}{f_{mod}}\right)} \cdot \int_{z=0}^{24} \Delta L(z) dz \quad (5.9)$$

Waar

F is de fluctuatiesterkte [vacil].

f_{mod} is de modulatiefrequentie [Hz].

z is de kritische bandwaarde [Bark].

$\Delta L(z)$ is de maskeringsdiepte in de kritische band z [dB].

5.6.1.1 Modulatiefrequentie

Een eerste probleem bij de implementatie van bovenstaande formule ligt bij de modulatiefrequentie, die manueel zal moeten geschat worden. Dit kan best gebeuren vanaf een weergave van het geluidssignaal in een periodisch tijdsdomein. Daartoe is de functie **Tmod.m** ontwikkeld. De gebruiker geeft daar, na inzoomen en aflezen uit de grafische signaalweergave, de modulatieperiode van het signaal in. Dit werkt vlotter dan de modulatiefrequentie -zijn inverse- daar de periode rechtstreeks kan afgelezen worden. Syntaxis:

$[Tm] = Tmod(y, Fs)$

Input

y is de (gekalibreerde) lineaire tijdsvector van het signaal.

Fs is de sampling frequency.

Output

Tm is de modulatieperiode van het signaal.

5.6.1.2 Maskeringsdiepte

Een tweede kritiek punt in deze formule is de maskeringsdiepte (ΔL). Deze valt niet eenvoudig te bepalen. Voor computerprogramma's kan de volgende benadering gebruikt worden:

$$\Delta L(z) \approx \log \left(\frac{N'(z)_{max}}{N'(z)_{min}} \right) \quad (5.10)$$

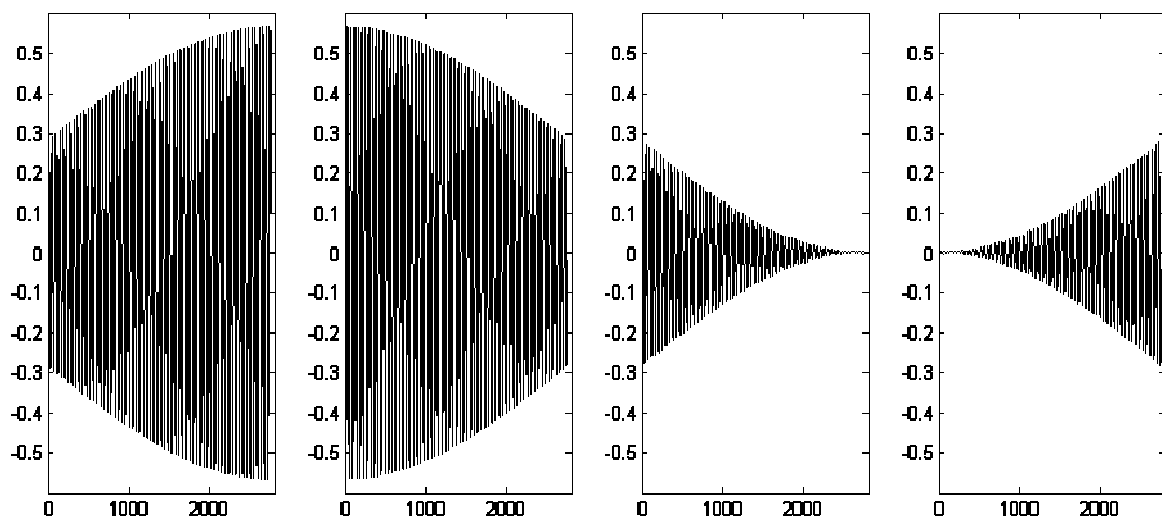
Waar

$\Delta L(z)$ is de maskeringsdiepte in de kritische band z [dB].

$N'(z)_{max}$ is de maximale specifieke luidheid in de kritische band z [soon/Bark].

$N'(z)_{min}$ is de minimale specifieke luidheid in de kritische band z [soon/Bark].

Voor de bovenstaande specifieke luidheden kunnen niet deze uit de voorgaande luidheidsberekening genomen worden, omdat ze uit het vermogenspectrum van het hele signaal afgeleid zijn. De maskeringsdiepte is daarentegen een fenomeen dat enkel in het tijdsdomein kan waargenomen worden. Daarvoor is een nieuwe Matlab functie aangemaakt, genaamd **specloud.m**. Die functie zal de specifieke luidheid berekenen van vier stukken van gelijke lengte uit één modulatieperiode van het signaal (zie Figuur 5.10). Door nu de maximale en minimale luidheid per kritische band te zoeken over de vier stukken heen, kan een benadering gemaakt worden voor de maskeringsdiepte. De keuze voor vier stukken is voor een stuk arbitrair (compromis tussen nauwkeurigheid en rekentijd), maar uitvoerig testen toonde aan dat bij een kleinere onderverdeling de minimale specifieke luidheid in sommige stukken over alle banden nul wordt. Dit is een nauwkeurigheidfout maar ze maakt wel de maskeringsdiepte voor alle banden gelijk aan nul. Het is evident dat dit in de praktijk niet zo is.



Figuur 5.10: Een modulatieperiode wordt in vier gelijke stukken gesplitst.

De functie wordt als volgt opgeroepen:

```
[Nsmax, Nsmin]=specloud(x, Fs, SF, Tm)
```

Input

x is de gekalibreerde lineaire tijdsvector van het signaal.

F_s is de sampling frequency.

SF staat voor het geluidsveld, met de waarde D of F, die aangeven of het geluid in een diffuus resp. vrij veld (frontaal geluid) ontstaan is.

T_m is de modulatieperiode van het signaal.

Output

N_{max} is een vector met 21 maximale specifieke luidheden over de acht stukken heen.

N_{min} is een vector met 21 minimale specifieke luidheden over de acht stukken heen.

Om te beginnen vraagt **specloud.m** op welk tijdstip in het signaal de onderverdeling van de modulatieperiode gestart moet worden. Dit kan handig zijn als bij het aflezen daarvan in **Tmod.m** op een bepaalde plaats een mooie curve gevonden is. Standaard wordt op één seconde gestart met het splitsen van de modulatieperiode in het signaal in vier stukken met lengte $T_m/4$. Daarna worden alle voorbereidende stappen voor de luidheidsberekening doorlopen, om uiteindelijk de luidheid zelf te berekenen. Daarvoor wordt echter niet de gewone luidheidsfunctie opgeroepen, maar een aangepaste, waarin de spectrale maskering niet berekend wordt. Dit omdat die maskering de waarde van de specifieke luidheid –vooral naar hogere banden toe– vervormt, terwijl spectrale componenten zoals vermeld geen invloed op de maskeringsdiepte hebben. De aangepaste functie heet **loudnessnomask.m**. Alle amplitudes van het signaal worden met tien vermenigvuldigd alvorens de luidheid ervan te berekenen. Dit omdat de fragmenten door hun korte duur en soms kleine amplitudes anders een luidheid van nul soon meekregen. Deze ingreep is toegelaten omdat er met de verhouding van de luidheden verder gewerkt wordt, waardoor de factor tien geneutraliseerd wordt.

Omdat de berekening van de specifieke luidheid ongeveer per tertsband gebeurt (21 stuks) en niet per kritische band, moet ook voor de bepaling van de minimale en maximale waarden per band die bandbreedtes gevolgd worden. De 240 waarden van N_s mogen niet zomaar per tien in 24 delen gesplitst worden. Dit kan gebeuren door opnieuw via de vector *ZUP* te werken. Voor elk van die banden wordt dan de gemiddelde luidheid berekend. Deze is gelijk aan de andere waarden in die band want de maskering is weggenomen. De gemiddelde waarde is hier gewoon handig om zo tot 21 in plaats van 240 waarden te komen. Dit gebeurt voor alle acht de fragmenten van het signaal, waardoor een matrix N_{smean} met dimensie $[4 \times 21]$ bekomen wordt. Per kolom wordt daarvan de maximale en minimale waarde gezocht en meegegeven in de vectoren N_{max} en N_{min} .

5.6.1.3 Fluctuatiesterkte

In de functie **fluctuationstrength.m** wordt eerst de maskeringsdiepte ΔL berekend. Per band wordt de maximale en minimale luidheid ingegeven om dL en daaruit F_{sp} , de specifieke fluctuatiesterkte, te kunnen berekenen. De eindformule is aangepast, omdat met de modulatieperiode i.p.v. –frequentie gewerkt wordt. Samenvattend is de syntaxis van **fluctuationstrength.m**:

```
[F, Fsp]=fluctuationstrength(Nmax, Nmin, Tm)
```

Input

N_{smax} is een vector met 21 maximale specifieke luidheden over de acht stukken heen.

N_{smin} is de een vector met 21 minimale specifieke luidheden over de acht stukken heen.

T_m is de modulatieperiode van het signaal.

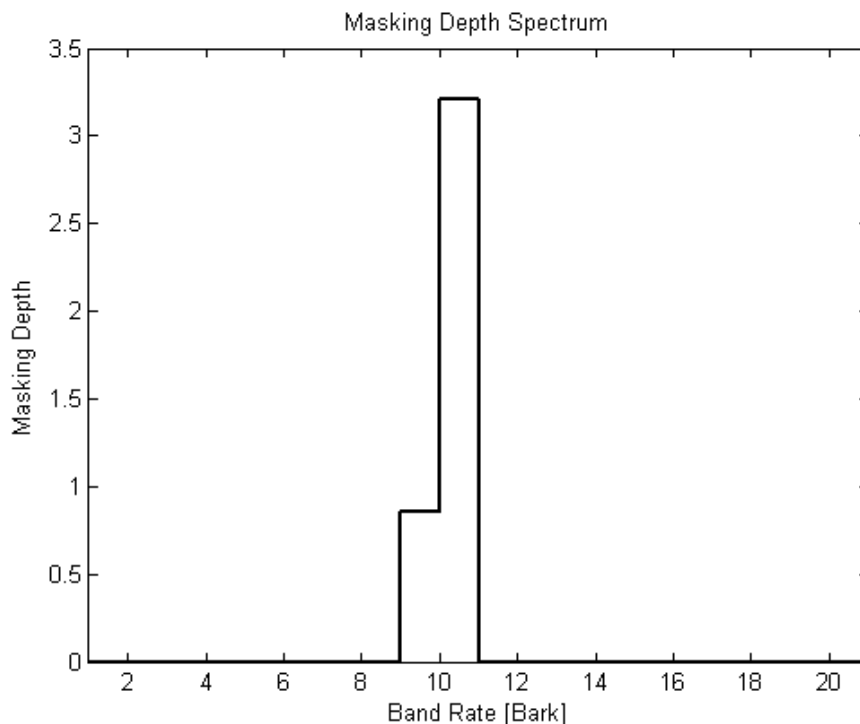
Output

F_{sp} is de specifieke fluctuatiersterkte van de 21 kritische banden.

F is de totale fluctuatiersterkte.

5.6.2 Verificatie

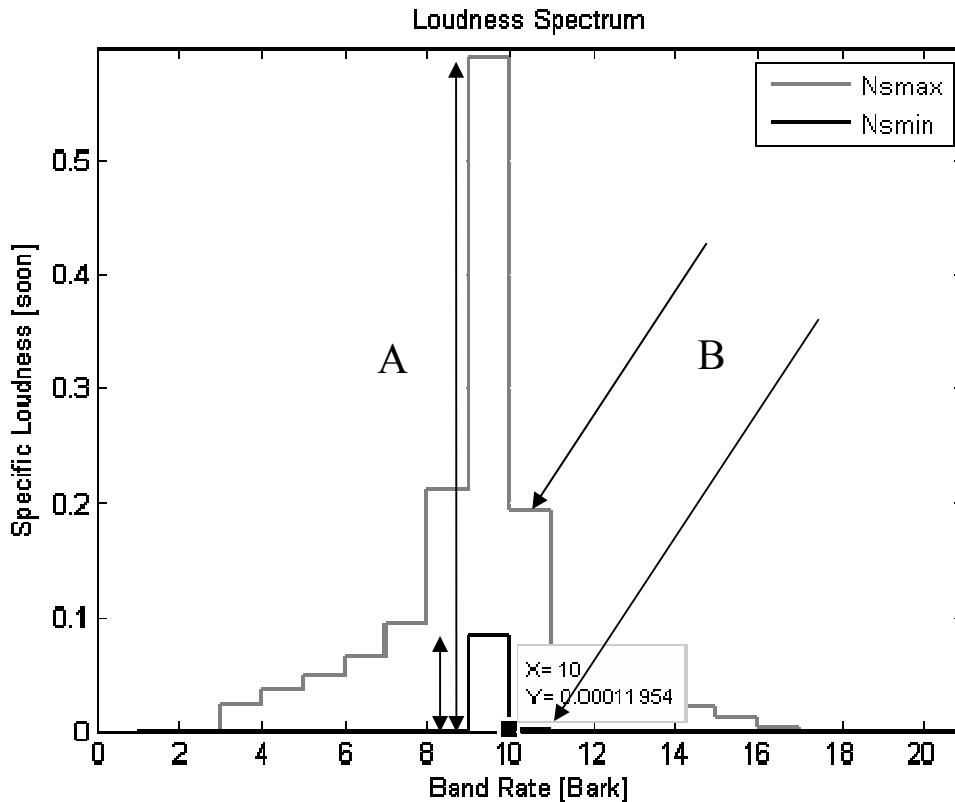
De juistheid van **fluctuationstrength.m** kan getest worden door het signaal dat juist één vacil zou moeten zijn, in het programma te laden. Dit is een 1 kHz-signaal van 60 dB⁵ dat een sinusoidale amplitudemodulatie met een frequentie van 4Hz ondergaat met een modulatiegraad van 1. Onderzoek naar de reden van een totaal verkeerde waarde bracht een fout aan het licht in de verhouding van de maximale en minimale luidheid per band. De Figuur 5.11 toont hoe de maskeringsdiepte ΔL vier maal groter is in de tien Bark band dan in de negen Bark band. Het 1 kHz signaal valt nochtans volledig binnen de negen Bark bandwaarde.



Figuur 5.11: De maskeerdiepte in de tien Bark band zou bijna vier maal groter zijn als bij negen Bark.

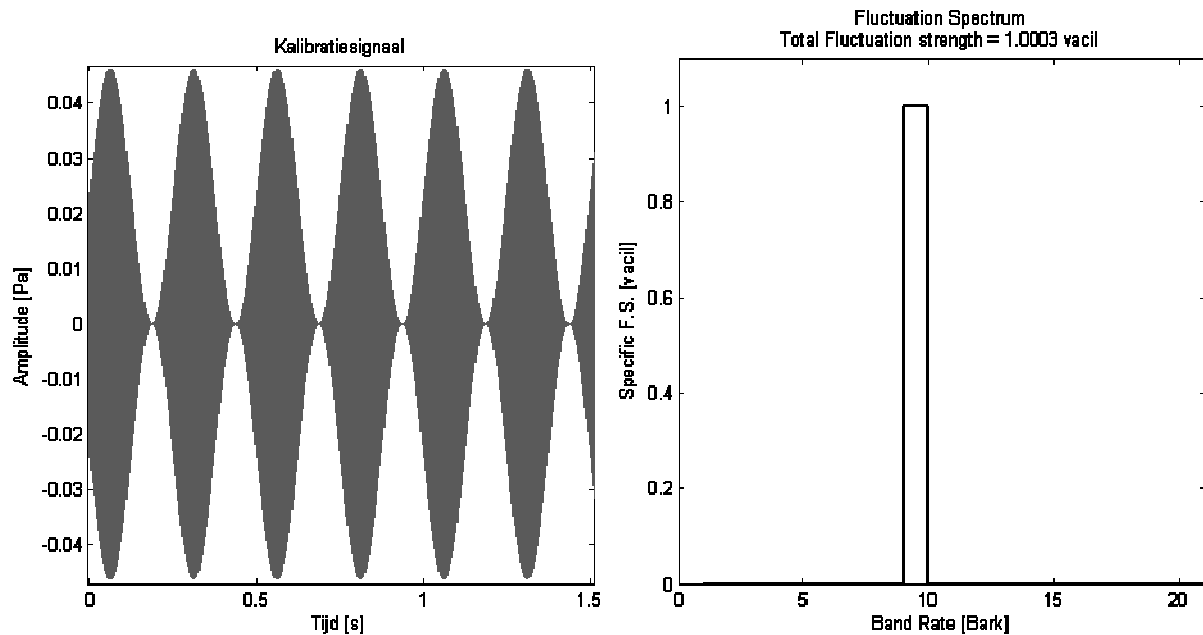
⁵ Er moet op gelet worden dat de kalibratie van een gemoduleerd signaal correct gebeurt. De opgegeven L_{eq} is deze van het niet-gemoduleerd signaal, de zuivere toon of ruis dus. Na generatie ervan moet dus al de kalibratie gebeuren, waarna de modulatie opgelegd kan worden. In het programma zelf moet dan geen kalibratie meer doorgevoerd worden. Vanzelfsprekend wordt geen weging toegepast.

De reden voor deze fout zichtbaar op Figuur 5.12. De band waarin het 1 kHz signaal zich bevindt, steekt ver boven de andere uit (A), maar het is de verhouding tussen maximum en minimum die telt. Bij tien Bark (B) bedraagt de minimale luidheid een luttele 0,0001954 soon, maar de verhouding met N_{smax} wordt daardoor wel erg groot. Om dit te verhelpen worden alle waarden kleiner dan 10% van de grootste waarde van N_{smin} weggelaten in de berekening van de verhouding, of anders gezegd, de verhouding in die banden wordt op 1 gesteld. Dit een enigszins arbitraire grens; in de literatuur is daarover niets te vinden!



Figuur 5.12: Weergave op dezelfde schaal van de maximale en minimale specifieke luidheid van het kalibratiesignaal.

Herneming van de berekening van het kalibratiesignaal geeft een waarde van amper 0,030747 vacil. In een poging deze fout te compenseren werd de constante factor, volgens Zwicker & Fastl 0,008, aangepast naar 0,26. Op Figuur 5.13 is te zien dat nu wel één vacil bekomen wordt.



Figuur 5.13: Het kalibratiesignaal en zijn fluctuatiesterkte van 1 vacil.

Vertrekkende van het gegeven dat het kalibratiesignaal –geforceerd– één vacil bedraagt, kan het nu telkens met één parameter gewijzigd worden. Zo wordt zichtbaar wat het verloop van de fluctuatiesterkte is. Hieronder worden de veranderingen met hun invloed op de berekende fluctuatiesterkte opgesomd:

- De verhoging van het effectieve geluidsniveau naar 100 dB levert 4,5 vacil op. Volgens Zwicker & Fastl zorgt een verhoging van 40 dB voor een stijging van de fluctuatiesterkte met een factor 2,5 (1,7 tot 3). De bekomen waarde is dus te hoog.
- Een wijziging van de modulatiefrequentie moet voor een vermindering van de fluctuatiesterkte zorgen. Dit gebeurt niet, integendeel; de fluctuatiesterkte stijgt in waarde.
- Een vermindering van de modulatiegraad geeft een daling van de fluctuatiesterkte ($F=0,34$ vacil met $m=25\%$), maar deze is niet groot genoeg (i.c. $F\approx 0,15$ vacil). De modulatie diepte komt niet rechtstreeks voor in de formule, maar wordt er onrechtstreeks toch in betrokken doordat de maskeringsdiepte ervan afhankelijk is. Zwicker & Fastl stellen de relatie tussen de fluctuatiesterkte en modulatie diepte als een S-curve vanuit de oorsprong voor.
- De frequentie van de toon heeft ook een sterke invloed. De fluctuatiesterkte haalt zijn maximum bij 1 kHz en zakt dan zowel bij hogere als lagere frequenties. Echter, in de berekening klimt ze mee met de hoogte van de draagfrequentie ($f_c=2500\text{Hz}$; $F=1,37$ vacil). Dit komt door de vermenigvuldiging van ΔL met de kritische bandwaarde z . Deze is nochtans vereist om de integratie zoals opgelegd in formule (5.9) correct te voltrekken.

Daarnaast moet **fluctuationstrength.m** ook in staat zijn de fluctuatiesterkte van frequentiegemoduleerde signalen te bepalen. Als testsignaal wordt hetzelfde als dat in het boek ‘Psychoacoustics’ gebruikt, namelijk een frequentiegemoduleerde sinus met centrale

frequentie 1500 Hz, modulatiefrequentie 4 Hz, frequentiedeviatie 700 Hz en geluidsniveau 70 dB. Dit levert 8,96 vacil, terwijl dat eigenlijk 2 vacil zou moeten zijn. Wordt de frequentiedeviatie verkleind naar 32 Hz, dan bedraagt de fluctuatiesterkte 0,52 vacil. Aflezing op de lineaire grafiek leert dat dit maar 0,25 vacil zou mogen zijn (de helft). Verhoging van de centrale frequentie naar 2,5 kHz met opnieuw een frequentiedeviatie van 700 Hz geeft 7,96 vacil. De empirische grafiek leert dat dit zo'n 70% van het 1,5 kHz signaal zou moeten zijn (hier 89%).

Als laatste wordt ook nog verwacht van de functie dat ze breed- en smalbandgeluid weet te analyseren. In testomstandigheden wordt ruis gebruikt, die dan een amplitudemodulatie meekrijgt. In **fluctuationstrength.m** wordt breedbandruis van 60 dB met een frequentiemodulatie van 4 Hz en een modulatiediepte van 40dB (i.e. een modulatiegraad van 98%) echter schromelijk overschat met 18.15 vacil waar die breedbandruis 1,8 vacil zou moeten bedragen.

De totale balans voor de fluctuatieberekening helt over naar de negatieve kant. Voor ruis werkt ze niet. Voor amplitudegemoduleerde en frequentiegemoduleerde sinussen wel, maar hoewel de berekende waarden wel de juiste richting uitgaan, is de absolute fluctuatiesterkte altijd verkeerd. Dit doet vermoeden dat de toevoeging van een exponent aan de formule betere resultaten zou geven. De meest significante oorzaak van de fouten ligt waarschijnlijk niet bij formule (5.9) die in haar grofheid wel correct is, maar bij de formule (5.10). De wiskundige benadering voor de waarde van de maskeringsdiepte is ontoereikend. Omdat het een psychoakoestisch gegeven is, valt de maskeringsdiepte niet uit een signaal af te leiden. Om er toch een waarde aan toe te kunnen kennen, is voorgesteld ΔL te benaderen als het logaritme van de verhouding van de maximale luidheid op de minimale luidheid en dit binnen een kritische band. Eventueel kan nog een constante factor toegevoegd worden, die dan samengenomen wordt met de constante in de hoofdformule. De essentie is evenwel die verhouding van luidheden en daar wringt de schoen. Zelf opgelegde restricties en gedane veranderingen aan de formules zijn *trial and error* gewijs toegevoegd om een beter resultaat te verkrijgen maar zijn niet wetenschappelijk onderbouwd. Zo is de opsplitsing van een modulatieperiode een methode die het signaal perfect kan analyseren, maar daarmee is de maskeringsdiepte zelf niet berekend. De implementatie in het algoritme van een formule die de pre- en postmasking kwalitatief bepaalt, kan misschien een oplossing zijn. Zelfs daarmee zal het ideaal nog niet bereikt zijn, want het spectraal effect van modulatie hangt af van de moduleringswijze (amplitude- of frequentiemodulatie), de moduleringsfrequentie en het soort geluid (eentonig, smalband of breedband).

5.7 Ruwheid

5.7.1 Codering

Zoals reeds aangegeven is de parameter ruwheid in grote mate gelijk aan de fluctuatiesterkte, met dat verschil dat ruwheid voor modulatiefrequenties van zo'n 20 tot 300 Hz geldt. Ze kunnen niet met dezelfde formule berekend worden. Zwicker & Fastl stelden voor ruwheid volgend empirisch model op:

$$R = 0,3 f_{\text{mod}} \cdot \int_{z=0}^{24} \Delta L(z) dz \quad (5.11)$$

Waar

R is de totale ruwheid [asper].

f_{mod} is de modulatiefrequentie [kHz].

z is de kritische bandwaarde [Bark].

$\Delta L(z)$ is de maskeringsdiepte in de kritische band z [dB].

De functie **roughness.m** is qua opbouw gelijk aan **fluctuationstrength.m**, met dat verschil dat de constante term voor de integraal verschillend is. Dit is haar syntaxis:

```
[R, Rs]=roughness(Nsmax, Nsmin, Tm)
```

Input

Ns_{max} is een vector met 21 maximale specifieke luidheden over de acht stukken heen.

Ns_{min} is een vector met 21 minimale specifieke luidheden over de acht stukken heen.

T_m is de modulatieperiode van het signaal.

Output

R_s is de specifieke ruwheid van de 21 kritische banden.

R is de totale ruwheid.

5.7.2 Verificatie

Ook **roughness.m** is getest met haar kalibratiesignaal. Dat is een sinusoïdaal amplitudegemoduleerde 1 kHz-sinus van 60 dB met een modulatiefrequentie van 70 Hz en een modulatiegraad van 1. De ruwheid die daarbij bekomen wordt, bedraagt echter nul asper. De oorzaak ligt bij de berekening van de luidheid; ook N_s bevat enkel nulwaarden waardoor alle verdere bewerkingen ook op nul uitdraaien. Het nul zijn van alle specifieke luidheden komt door de erg korte fragmentjes waarvan de luidheid bepaald moet worden. $70 \text{ Hz} = 0,0142857 \text{ s}$. Wordt die modulatieperiode nog eens in vier gedeeld dan bedraagt de duur van de fragmentjes nog 3,6 ms. Dit is te kort om er de luidheid op te kunnen berekenen. Een oplossing werd gezocht in de creatie van een nieuwe vector die hetzelfde fragment tien maal na elkaar bevat. Door de vier zelf samengestelde signalen dan door **loudnessnomask.m** te

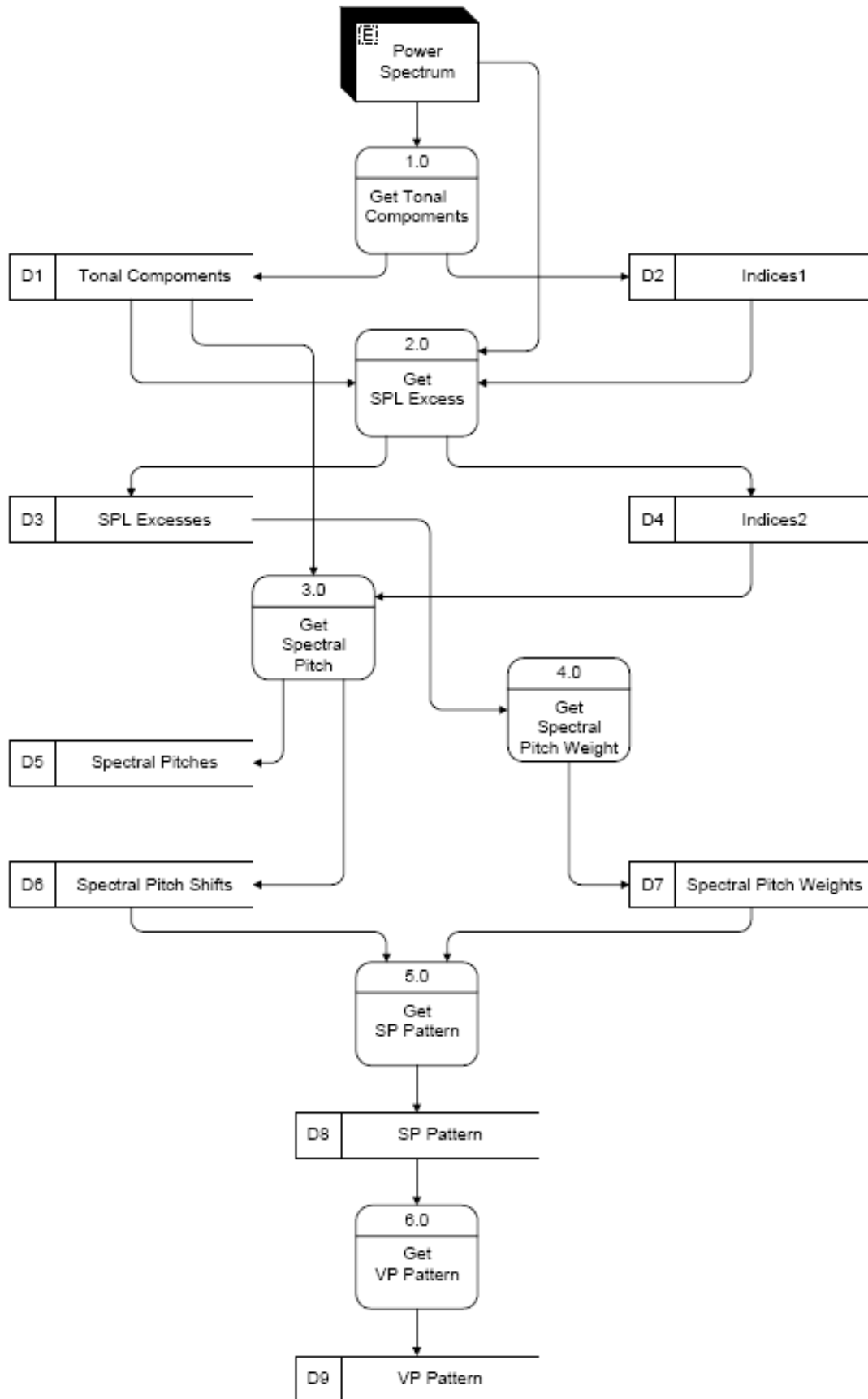
laten lopen, krijgt de luidheid toch een beduidende waarde. Nader onderzoek leerde dat dit echter totaal onbetrouwbare resultaten zijn, waarmee het geen zin heeft verder rekenwerk uit te voeren. **roughness.m** is dan ook disfunctioneel, tenminste als een algoritme bekomen moet worden, opgebouwd volgens de regels van de kunst.

5.8 Tonaalheid

5.8.1 Codering

Voor de bepaling van de tonaalheid van een geluid is gebruik gemaakt van een bestaand Matlab programma. Het is van de hand van Zhang en Shrestha en gebaseerd op het algoritme van Terhardt. Ze ontwikkelden de code voor hun masterproef in de informatica in samenwerking met Brüel & Kjær (ref. [106]). Er kon ook geopteerd worden om de formules van Aures zelf te implementeren maar dat ligt niet voor de hand omdat zaken als het surplus in geluidsniveau van een tonale component moeten ingegeven worden. Het overgenomen programma maakt geen kwantitatieve berekening van de tonaalheid (in tu), maar geeft wel aan waar de spectrale tonale componenten liggen, met hun grootte. Uiteindelijk is dat hetgeen een ontwerper wil weten over de tonaalheid van zijn product. Daarnaast toont het programma ook de *virtual pitch*. Zoals uitgelegd in paragraaf 3.6 is dat een toon die de mens meent waar te nemen, maar die eigenlijk niet aanwezig is. Die virtuele toon ontstaat door het wel aanwezig zijn van de harmonischen van die toon. Hieronder wordt uitgelegd hoe het programma in het eigen sound quality programma geïntegreerd is en hoe het ineen zit. Op het data flow diagram in Figuur 5.14 is het interne verloop van de functies en de variabelen te volgen.

Het dient opgemerkt dat het gebruikte algoritme van Terhardt niet identiek is aan de DIN 45681-norm. Volgens die norm is een geluid tonaal als het geluidsniveau van een component 6 dB hoger ligt dan het rekenkundig gemiddeld niveau van de twee aangrenzende kritische banden. In bovenvernoemde thesis werd, na empirisch onderzoek op complexe geluiden, 7 dB als norm gesteld en wordt niet met tertsbanden gerekend. Dit is in de norm wel zo om met gewone geluidsmeters –die een tertsbandanalyse kunnen uitvoeren– vast te kunnen stellen of een geluid tonaal is. Deze benadering is echter te grof voor een Sound Quality analyse. Hoeveel de *sample spacing* dan wel moet bedragen is onduidelijk. In de functies die hier gebruikt worden, is dat afhankelijk van de *sample rate* van het vermogenspectrum. Dat vermogenspectrum wordt, zoals beschreven in paragraaf 5.3, gegenereerd in de functie **Powspec.m**, waarvan de sample rate afhangt van de bemonsteringsfrequentie. De sample spacing zal in deze toepassing dan ook altijd enkele Hertz bedragen. De variabiliteit van de sample spacing is geen obstakel zolang signalen met gelijke sample frequency opgenomen worden. Het is overigens altijd mogelijk om signalen te *downsamplen* zodat de sample spacing groter wordt. Dit heeft als gevolg dat componenten die zich niet op één enkele frequentie maar meer als een piek voordoen, ook als tonaal herkend zullen worden.



Figuur 5.14: Het data flow diagram van het programma om tonale componenten te berekenen.

1. De tonale componenten

Als eerste worden de tonale componenten in het signaal gezocht en dit vanuit het vermogensspectrum.

`[y, z] = Get_tonal_comp(ps)`

Input

`ps` is het vermogensspectrum van het signaal.

Output

`y` bevat de SPL-waarden en frequenties van de tonale componenten.

`z` bevat de indices van de maximale samples.

In de functie **Get_tonal_comp.m** wordt gekeken of de SPL-waarde van een sample hoger is dan zijn omliggende. Is die waarde zeven decibel hoger dan de zes omliggende samples, dan wordt deze als een tonale component aanzien. Om de correcte frequentie terug te vinden van de tonale component wordt de volgende interpolatieformule gebruikt:

$$f_c = f_i + 0,46(L_{i+1} - L_{i-1}) \quad (5.12)$$

Waar

f_c is de frequentie van de tonale component [Hz].

f_i is de frequentie van het i -de sample van het spectrum [Hz].

$L_{i\pm 1}$ zijn de aanliggende SPL-waarden [dB].

Na de bepaling van de spectrale tonale componenten, moet ook hun eventuele maskering onderzocht worden. Een gemaskeerde toon moet vanzelfsprekend niet in rekening gebracht worden. Dit wordt behandeld in de onderstaande puntjes 2 en 3. Het eerste onderzoekt welke componenten onhoorbaar of minder hoorbaar zouden kunnen zijn door spectrale maskering. Daarmee wordt de *nominal spectral pitch* gevonden. De tweede functie bekijkt of tonen eventueel verschuiven (*pitch shift*) door wederzijdse partiële maskering. Dat is de berekening van de *true spectral pitch*.

2. De SPL uitsprong

De mate waarin een tonale component uitsteekt wordt beschreven door de geluidsniveau uitsprong (*SPL excess*) in de functie **Get_spl_excess.m**:

`[y, idx] = Get_spl_excess(ps, x, z)`

Input

`ps` is het vermogensspectrum van het signaal.

`x` bevat de SPL-waarden en hun frequenties van de tonale componenten.

`z` bevat de rijnummers van `x` waar een SPL uitsprong voorkomt.

Output

`y` is de matrix van SPL uitsprongen met hun frequentie.

`idx` bevat de rijnummers

De hoofdformule ter calculatie van de SPL uitsprong gaat als volgt:

$$LX_{\mu} = L_{\mu} - 10 \log_{10} \left[\left(\sum_{\substack{v=1 \\ v \neq \mu}}^N 10^{\frac{L_{Ev}(f_{\mu})}{20}} \right)^2 + I_{N\mu} + 10^{\frac{L_{TH}(f_{\mu})}{10}} \right] \quad (5.13)$$

Waar

LX_{μ} is de SPL uitsprong van de tonale component μ [dB].

L_{μ} is de SPL van de tonale component μ [dB].

$L_{Ev}(f_{\mu})$ is het excitatieniveau geproduceerd op de frequentie f_{μ} door de v -de tonale component [dB]. v neemt toe van 1 tot N (het aantal tonale componenten), waarbij μ wordt overgeslagen.

$I_{N\mu}$ is de geluidsintensiteit van de kritische band rond de tonale component μ [W/m^2].

$L_{TH}(f_{\mu})$ is de gehoordrempel op de frequentie f_{μ} [dB].

De bovenstaande termen worden apart berekend in subfuncties bij **Get_spl_excess.m**. Voor L_{Ev} is een aparte functie aangemaakt, **Lev.m**, omdat deze later nogmaals gebruikt zal worden. De berekening gaat als volgt:

$$L_{Ev}(f_{\mu}) = L_v - s(z_v - z_{\mu}) \quad (5.14)$$

Waar

L_v is de SPL van de tonale component v [dB].

z_v en z_{μ} zijn de kritische bandwaarde van de tonale componenten v respectievelijk μ [Bark]. De omzetting van frequentie naar kritische bandwaarde gebeurt volgens de formule (2.4) in de subfunctie $y=z(x)$.

s is de hellingsgraad (*steepness of the slopes*) van het excitatieniveau in functie van de kritische bandwaarde [dB/Bark]. De bepaling van s gebeurt in de subfunctie $y=s(f_u, f_v, L_v)$ onder **Lev.m**. Haar algoritme is gebaseerd op de volgende formule:

$$s \begin{cases} f_{\mu} \leq f_v \rightarrow s = 27 \\ f_{\mu} > f_v \rightarrow s = \left[-24 - \left(\frac{0,23}{f_v} \right) + (0,2L_v) \right] \end{cases} \quad (5.15)$$

Waar

s is de hellingsgraad van het excitatieniveau in functie van de kritische bandwaarde [dB/Bark].

f_{μ} en f_v zijn de frequenties van de tonale componenten μ respectievelijk v [kHz].

L_v is de SPL van de v -de tonale component [dB].

De geluidsintensiteit $I_{N\mu}$ wordt in de subfunctie $y=IN(f, x, idx)$ verkregen door de intensiteiten toe te voegen van de samples in het interval dat met de breedte van een kritische band rond de tonale component μ ligt. Het interval strekt dus van $z_{\mu} - 0,5$ Bark tot $z_{\mu} + 0,5$ Bark. Voor de waarde van het intensiteitsniveau wordt het

ongeveer gelijke geluidsdrumniveau genomen. De omliggende samples, die mee de tonale component vormen, worden overgeslagen.

Als laatste wordt ook de gehoordrempel gemodelleerd, waar deze in de loudness functie in een voorgedefinieerde vector vast ligt. De formule die in de subfunctie $y=LTH(x)$ toegepast wordt is de volgende:

$$L_{TH}(f_{\mu}) = \left[3,64(f_{\mu})^{-0,8} - 6,5 \cdot e^{-0,6(f_{\mu}-3,3)^2} + 10^{-3}(f_{\mu})^4 \right] \quad (5.16)$$

Waar

$L_{TH}(f_{\mu})$ is de gehoordrempel op de frequentie f_{μ} [dB].

f_{μ} is de frequentie van de tonale component μ [kHz].

3. Verschuiving van de toonhoogte

Door de interactie van simultane spectrale componenten in het auditieve systeem zijn de spectrale toonhoogtes niet exact gelijk aan de toonhoogtes van geïsoleerde tonen met dezelfde frequentie. Daar wordt rekening mee gehouden in de functie

Get_spectrum_pitch.m:

`[y, z] = Get_spectrum_pitch(x, idx2)`

Input

x bevat de tonale componenten met hun bijhorende frequentie.

$idx2$ bevat de rijnummers van x waar een SPL uitsprong voorkomt.

Output

y is het pitch spectrum.

z bevat de verschuiving van de toonhoogtes met de bijhorende frequenties.

Hier gaat de hoofdformule als volgt:

$$H_{\mu} = f_{\mu}(1 + v_{\mu}) \quad (5.17)$$

Waar

H_{μ} is de spectrale toonhoogte van de tonale component μ [pu]⁶.

f_{μ} is de frequentie van de tonale component μ [kHz].

v_{μ} is de geïnduceerde toonhoogteverschuiving van maximaal een paar procent.

Ze is enkel vereist voor tonale componenten met een SPL uitsprong groter dan nul en wordt in een subfunctie $y=Vu(x, u)$ bepaald met de volgende formule:

$$v_{\mu} = 2 \cdot 10^{-4} (L_{\mu} - 60) \cdot (f_{\mu} - 2) + 1,5 \cdot 10^{-2} \cdot e^{\left(\frac{-LX_{\mu}}{20}\right)} \cdot \left((3 - \ln f_{\mu}) + 3 \cdot 10^{-2} \cdot e^{\left(\frac{-LX_{\mu}}{20}\right)} \cdot (0,36 + \ln f_{\mu}) \right) \quad (5.18)$$

⁶ pu staat voor Pitch Unit, of eenheid van toonhoogte. Het is een eenheid bedacht door Terhardt en wordt dan ook gebruikt in deze berekeningswijze van zijn hand.

Waar

v_μ is de geïnduceerde toonhoogteverschuiving.

L_μ is de SPL van de tonale component [dB].

f_μ is de frequentie van de tonale component μ [kHz].

LX'_μ en LX''_μ zijn de specifieke voorstellingen van de SPL uitsprong met betrekking tot de tonale component μ [dB]. Ze stellen respectievelijk de lagere en de hogere uitsprongen voor. Hun berekening in de subfunctie [LXu1, LXu2]=specific_LXu(x, u) is nodig om de interferentie met de component μ te kennen:

$$LX'_\mu = L_\mu - 20 \log_{10} \sum_{v=1}^{\mu-1} 10^{L_{Ev}\left(\frac{f_\mu}{20}\right)} \quad (5.19)$$

$$LX''_\mu = L_\mu - 20 \log_{10} \sum_{v=\mu-1}^N 10^{L_{Ev}\left(\frac{f_\mu}{20}\right)} \quad (5.20)$$

Waar

LX'_μ en LX''_μ zijn de specifieke voorstellingen van de SPL uitsprong met betrekking tot de tonale component μ [dB].

L_μ is de SPL van de tonale component [dB].

$L_{Ev}(f_\mu)$ is het excitatieniveau geproduceerd op de frequentie f_μ door de v -de tonale component [dB]. v neemt toe van 1 tot N , het aantal tonale componenten, waarbij μ wordt overgeslagen. Het algoritme voor de berekening ervan zit in **Lev.m**.

4. Weging van de componenten

De mate waarin elke tonale component bijdraagt tot de totale tonale perceptie hangt af van de SPL uitsprong en zijn frequentie. De functie **Get_sp_weight.m** geeft een wegingsfactor aan de componenten.

[y]=Get_sp_weight(x)

Input

x is de matrix van SPL uitsprongen met hun bijhorende frequentie.

Output

y is de vector met de wegingsfactoren.

De wegingsformule is de volgende:

$$WS_{\mu} \begin{cases} LX_{\mu} < 0 \rightarrow WS_{\mu} = 0 \\ LX_{\mu} \geq 0 \rightarrow WS_{\mu} = \frac{1 - e^{\left(\frac{-LX_{\mu}}{15}\right)}}{\sqrt{1 + 0,07 \left(\frac{f_{\mu}}{0,7} - \frac{0,7}{f_{\mu}}\right)^2}} \end{cases} \quad (5.21)$$

Waar

WS_{μ} is de wegingsfactor van de tonale component μ [dB].

LX_{μ} is de SPL uitsprong van de tonale component μ [dB].

f_{μ} is de frequentie van de tonale component μ [kHz].

5. Opbouw van het spectrale pitch patroon

In de functie **Get_SP_pattern.m** worden de wegingsfactor en het pitch spectrum samen in een matrix gezet.

```
sp = Get_SP_Pattern(A, B)
```

Input

A bevat de verschuiving van de individuele toonhoogtes met hun bijhorende frequenties.

B is de vector met de wegingsfactoren.

Output

sp is een matrix met drie kolommen: de frequenties waarop tonale componenten liggen, de pitch shifts en hun wegingsfactor.

6. De virtuele tonen

Een virtuele toon is een subharmonische van een van de relevante spectrale tonen. Hun m -de subharmonische is te vinden door hun frequentie te delen door m . Zo wordt de *nominal virtual pitch* bekomen. Het is echter interessanter om met de *true virtual pitch* –de toonhoogte gepercipieerd door het gehoor– te werken. Voor de berekening van de true virtual pitch wordt rekening gehouden met de toonhoogteverschuivingen en de uitrekking van de subharmonische intervallen. De syntaxis van de functie **Get_VP_Pattern.m** is de volgende:

```
y = Get_VP_Pattern(sp)
```

Input

sp is een matrix met drie kolommen: de frequenties waarop tonale componenten liggen, de pitch shifts en hun wegingsfactor.

Output

y is een matrix met drie kolommen: de frequenties, de virtuele tonen en hun wegingsfactor.

In de formule voor true virtual pitch berekent de eerste factor de nominal pitch, terwijl de –lange– tweede de correctie naar true virtual pitch maakt. v_i zorgt ervoor dat de pitch shift in rekening gebracht wordt, terwijl dat voor het subharmonische interval

gebeurt door de term na de signumfunctie. De formule wordt berekend in de subfunctie $y=H(i, m, sp_sorted)$:

$$H_{im} = \frac{1000f_i}{m} \left(1 + v_i - \text{sgn}(m-1) \cdot 10^{-3} \left[18 + 2,5m - (50 - 7m) \frac{f_i}{m} + 0,1 \left(\frac{m}{f_i} \right)^2 \right] \right) \quad (5.22)$$

Waar

H_{im} is de virtuele toonhoogte van de m -de subharmonische van de i -de relevante component [pu].

f_i is de frequentie van de tonale component i [kHz].

v_i is de pitch shift van de component i [dB].

m is het nummer van de subharmonische die berekend wordt.

In principe zou de bovenstaande formule nu voor alle mogelijke subharmonischen doorlopen moeten worden. Door een gerichte aanpak kan dit aantal echter gereduceerd worden tot de meest significante virtuele toonhoogtes. Deze aanpak start met het sorteren in aflopende orde van de wegingsfactor zodat een tabel ontstaat met de frequenties, pitch shifts en het gewicht van de spectrale tonen. Verder worden, op empirische gronden, enkel de tonen die meer dan 70% van de amplitude van de grootste toon bedragen, in rekening gebracht. Ook wordt m begrensd tot twaalf. Er wordt, in de subfunctie $y=C(i, j, m, sp_sorted)$ ook een coïncidentiecoëfficiënt (*coincidence coefficient*) gedefinieerd. Daarmee wordt bepaald in hoeverre twee harmonischen een zelfde subharmonische zouden kunnen hebben:

$$C_{ij} \begin{cases} \gamma \leq \delta \rightarrow C_{ij} = \sqrt{\frac{WS_i WS_j}{mn}} \left(1 - \frac{\gamma}{\delta} \right) \\ \gamma > \delta \vee n > 20 \rightarrow C_{ij} = 0 \end{cases} \quad (5.23)$$

Waar

C_{ij} is de coïncidentiecoëfficiënt tussen de componenten i en j .

WS_i en WS_j zijn de gewichten van de spectrale tonale componenten i en j .

m en n zijn het nummer van de subharmonischen van de componenten i en j .

De waarde van m wordt ingesteld door het algoritme, maar n moet in de subfunctie $y=n(i, j, m, sp_sorted)$ elke keer opnieuw bepaald worden. Dit gebeurt met de onderstaande formule. Daarna wordt n afgerond naar een natuurlijk getal.

$$n = \frac{mf_j}{f_i} + 0,5 \quad (5.24)$$

δ is de breedte van het coïncidentie-interval en bedraagt 0,08.

γ is de coëfficiënt die de graad van inharmonicititeit van het geteste componentenpaar weergeeft. Hij wordt berekend in de subfunctie $y=\text{gamma}(i, j, m, sp_sorted)$ met de volgende formule:

$$\gamma = \left(\frac{nf_i}{f_j} \right) \quad (5.25)$$

Het gewicht dat aan een virtuele toon toegekend wordt, wordt als volgt berekend:

$$W_{im} = \beta \sum_{\substack{j=1 \\ j \neq i}}^R C_{ij} \quad (5.26)$$

Waar

W_{im} is het gewicht van de m -de subharmonische van de i -de relevante component.

C_{ij} is de coïncidentiecoëfficiënt tussen de componenten i en j . j neemt toe van 1 tot R , het aantal relevante tonale componenten, waarbij i wordt overgeslagen.

β is een factor, genaamd het fundamentele frequentiegewicht. Hij is opgesteld in de subfunctie `y=beta(i, m, sp_sorted)` en heeft een laagdoorlaatkarakteristiek:

$$\beta = \frac{1}{1 + \left(\frac{H_{im}}{800} \right)^4} \quad (5.27)$$

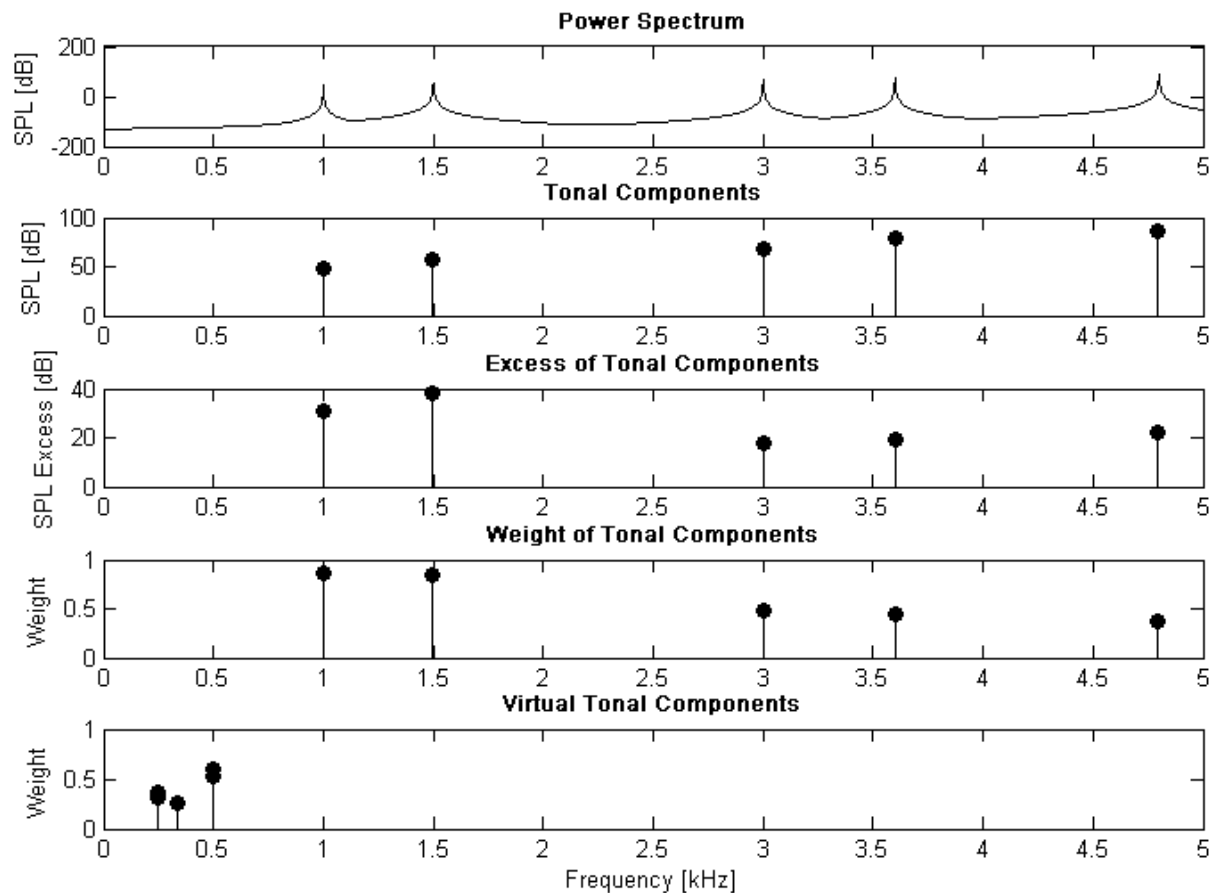
Waar

β is het eenheidsloze fundamentele frequentiegewicht.

H_{im} is de virtuele toonhoogte van de m -de subharmonische van de i -de relevante component [pu].

5.8.2 Verificatie

Net als bij de verificatie van de functie **loudness.m** is het voor de tonaalheid niet meer nodig de correcte werking van de functie uitgebreid te verifiëren. Te meer daar het algoritme niet door de masterstudenten zelf ontworpen is, maar door de autoriteit Terhardt. Zij hebben het algoritme enkel in Matlab geïmplementeerd. Brüel & Kjaer was daarbij hun stagebedrijf, wat het aanzien van de verkregen code nog doet toenemen. Op Figuur 5.15 zijn de grafieken te zien die uit de berekeningen van de tonaalheid functie voortkomen. Het signaal dat hiervoor gegenereerd werd, is de samenstelling van een 1000, 1500, 3000, 3600 en 4800 Hz signaal. Deze willekeurig gekozen signalen kregen een L_p -waarde van respectievelijk 50, 60, 70, 80 en 90 dB. Deze tonale componenten en hun stijgende geluidsdruk zijn te zien op de eerste twee grafieken; het vermogensspectrum en de tonale componenten. Op de derde grafiek is de uitsprong gevisualiseerd. Dat is mate waarin de tonale component boven zijn omgeving uitsteekt. De vierde grafiek toont dat de ervaring van tonaalheid door het gehoor verschilt naargelang de frequentie van de toon. Zo is de 1 kHz component groter geworden dan de 1,5 kHz component. De vijfde en laatste grafiek toont de virtuele tonen, deze die men dus denkt te horen. Dat zijn tonen met frequentie 250, 333 en 500 Hz. Alle drie zijn het subharmonischen van de 1000, 1500 en 3000 Hz componenten.



Figuur 5.15: Visualisatie van de tonaalheidsberekeningen voor een samengesteld signaal.

5.9 Pleasantness & annoyance

De sensory pleasantness zoals besproken in paragraaf 3.7 wordt niet berekend in het programma, alhoewel de formule allesbehalve ingewikkeld is. De eerste reden is dat de pleasantness geen absolute, maar een relatieve waarde is. Om ze te kunnen berekenen, moeten met andere woorden altijd minstens twee objecten bemeaten worden. Dit valt lastig te implementeren in het programma, dat gebaseerd is op de absolute bepaling van de psychoakoestische parameters van één geluid.

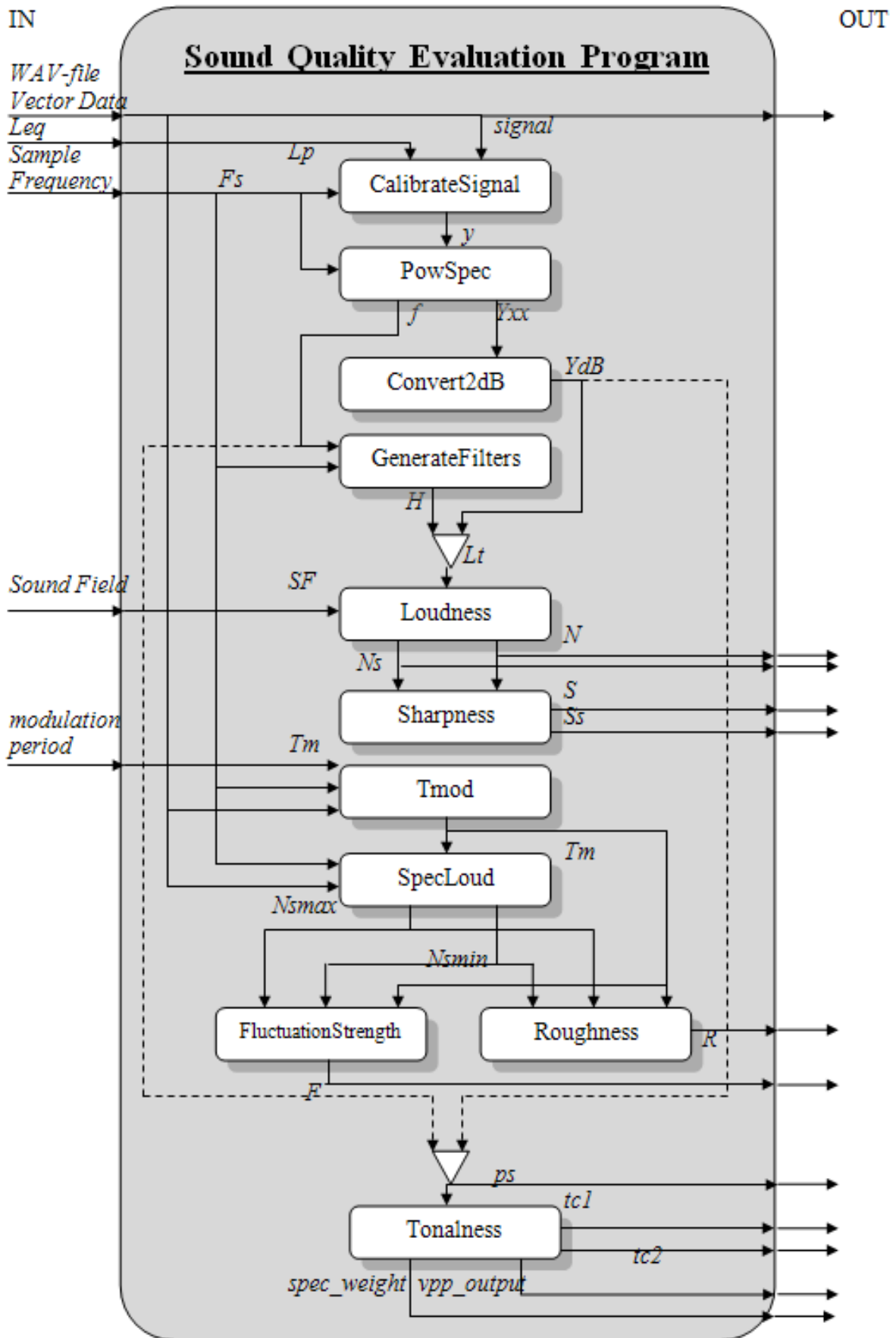
De tweede reden is dat de sensory pleasantness een weinig gebruikte parameter is. Zoals beschreven in paragraaf 4.6, waar enkele concrete sound quality analyses gedaan worden, is men snel geneigd een eigen Sound Quality Index –want dat is de sensory pleasantness– te ontwikkelen. De specificiteit van de ideale sound quality er product maken een algemene parameter als sensory pleasantness haast irrelevant. Overigens is het een kleine moeite om de pleasantness apart toch te berekenen. Met eerst het sound quality programma met minstens twee geluiden door te voeren en de bekomen waarden voor de psychoakoestische parameters dan in de pleasantness formule in te vullen is dat geregeld.

5.10 Hoofdprogramma

De hierboven besproken functies (*functions*) zijn geïntegreerd in een hoofdprogramma. Dat is geen functie, maar een script. Het verschil is dat er geen in- en output moet zijn bij een script. Het hoofdprogramma is **SQEP**, Sound Quality Evaluation Program gedoopt en zit vervat in het bestand **sqep.m**. Om het te starten dient gewoon `sqep` in het Matlab Command Window ingegeven te worden. Daarvoor moet de map waarin het programma zich bevindt wel in de Matlab Search Path lijst staan. De opbouw van het SQEP programma is afgebeeld op Figuur 5.16. De witte kaders zijn daarbij de hoofdfuncties die rechtstreeks in het script aangeroepen worden. De keuze voor een dergelijke opbouw is gemaakt om het programma zo overzichtelijk en flexibel mogelijk te houden. Dit stelt personen die later wijzigingen zouden willen aanbrengen of een specifieke functie uit het programma elders implementeren in staat dit vlot en zonder problemen te doen. Een lineaire opbouw valt sowieso beter te begrijpen dan een spinnenweb, waar aanpassingen aan een bepaald algoritme onverwachte negatieve consequenties voor het hele programma zouden kunnen inhouden. Ook is voldoende commentaar bij de verschillende stappen toegevoegd, zowel in het script als in de functies. Het commentaar is, net als het volledige programma, Engelstalig. Dit omdat het sound quality vakgebied een internationaal gegeven is. Eventuele niet-Nederlandstalige geïnteresseerden in het SQEP hebben dan geen taalbarrière te overwinnen. Een beknopte uitleg over de functies wordt bekomen door `help naam van de functie` (bv. `help loudness`) in het Command Window in te geven.

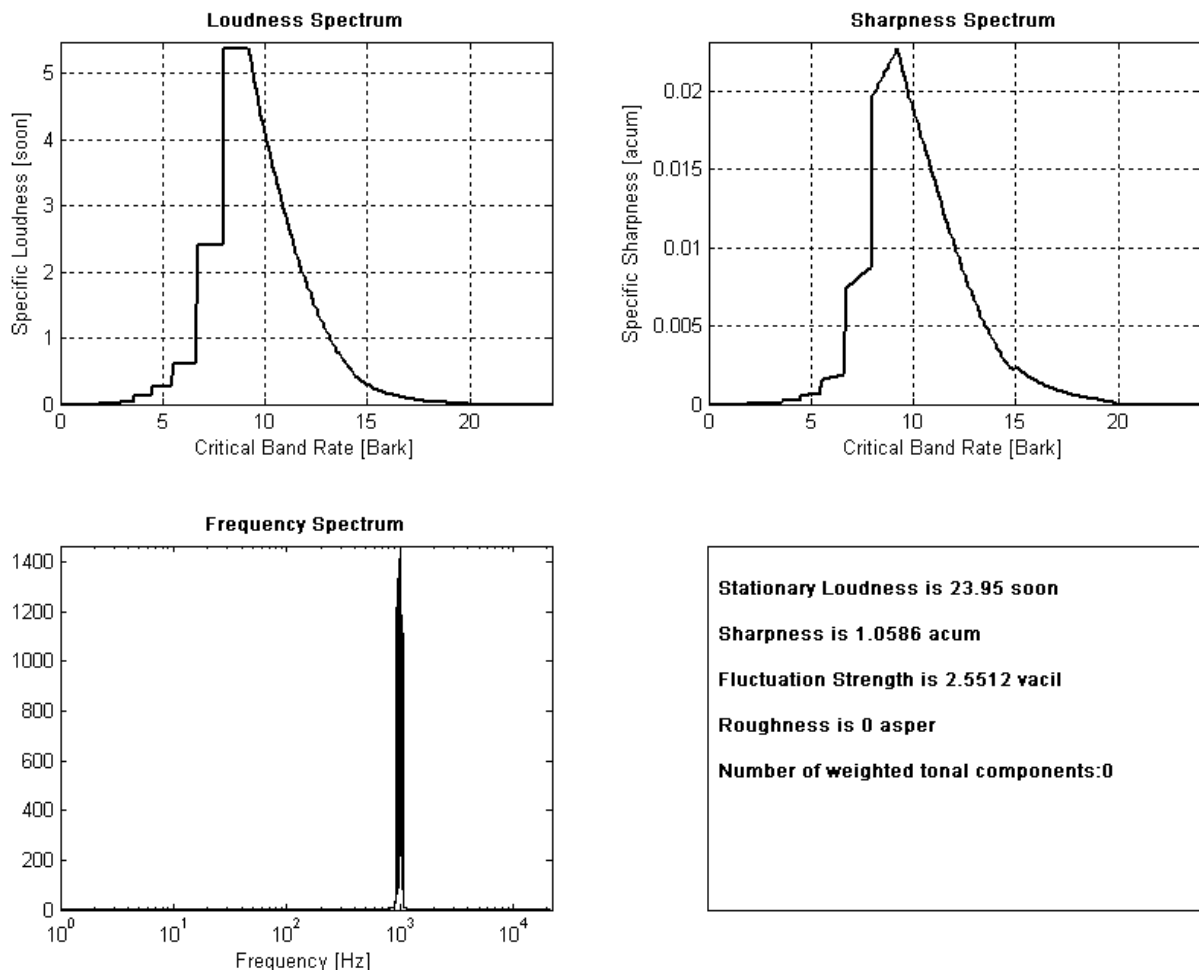
Naar de gebruikers toe is het programma zo eenvoudig en logisch mogelijk gehouden. Met een basiskennis over Matlab en akoestiek kan het SQEP doorlopen worden. Over de werking van de geïntegreerde functies hoeft de gebruiker niets te weten, wat voor een sterk verminderde kans op menselijke fouten zorgt. Een handleiding bij het SQEP is als bijlage bij deze scriptie toegevoegd. Daarin valt onder andere te lezen welke gegevens allemaal dienen ingevoerd te worden om de berekeningen te kunnen uitvoeren. Dit zijn achtereenvolgens:

- het antwoord op de vraag of het signaal een WAV-bestand dan wel een datavector is.
- de locatie van het WAV-bestand of de naam van de vector en zijn bemonsteringsfrequentie.
- het antwoord op de vraag of het signaal dient gekalibreerd te worden, en indien, wat het gemeten effectieve geluidsdrukniveau van het signaal is.
- het geluidsveld: vrij of diffuus.
- de modulatieperiode van het signaal.
- het tijdstip waarop de luidheidsmeting van één modulatieperiode dient te starten.



Figuur 5.16: Schema van het hoofdprogramma.

Na de berekeningen, waarvan de duur afhangt van de lengte en de bemonsteringsfrequentie van het signaal, worden twee vensters geopend, *figures*, waarin de resultaten grafisch en tekstueel meegegeven worden. Een voorbeeld van het eerste venster is te zien op Figuur 5.17. Het is het resultaat van een berekening op amplitudegemoduleerde (modulatiefrequentie = 4 Hz) smalbandruis rond 1 kHz. Van de luidheid, de scherpte en de frequenties van het signaal wordt het spectrum geplot. Deze laatste kan handig zijn om verschijnselen in de eerste twee figuren mee te verklaren. Rechts onderaan zijn in een kadertje de totale resultaten berekend. Daarin valt op dat de ruwheid nul asper bedraagt. Dat komt omdat, afhankelijk van de modulatiefrequentie, of de fluctuatiesterkte of de ruwheid berekend wordt. Op het schema in Figuur 5.16 is dit duidelijk gemaakt door beide functies lateraal tegenover elkaar te plaatsen. Het aantal gewogen tonale componenten bedraagt nul, wat logisch is want ruis is atonaal. Anderzijds zou bij meting volgens DIN 45681 de smalbandruis wel als tonaal bestempeld worden omdat in de norm de *prominence ratio* per kritische band berekend wordt.



Figuur 5.17: Het eerste deel van de resultaten van het Sound Quality Evaluation Program

Het tweede venster dat zich opent, is reeds weergegeven in Figuur 5.15. het is volledig aan de tonaalheidsberekeningen gewijd en geeft door de verticale ordening een mooi beeld van de evolutie van de berekeningen.

In het Command Window wordt ondertussen gevraagd onder welke naam de resultaten dienen opgeslagen te worden. Na het ingeven van die naam zal een *struct* variabele met die naam aangemaakt worden, die alle relevante variabelen bevat (16 stuks). De andere worden gewist. De structuurvariabele wordt aangemaakt om de overzichtelijkheid te bewaren en om een specifieke naam voor de relevante variabelen te verkrijgen. Dit is handig als de gebruiker twee verschillende metingen wil vergelijken, bv. door ze op één grafiek te plaatsen. Als die variabelen nog dezelfde naam zouden hebben, zou een conflict ontstaan.

Als laatste komt het *Save Workspace As* venster te voorschijn, dat alle Workspace variabelen opslaat in een bestand met extensie *.MAT. Later zijn deze variabelen dan nog beschikbaar door het *.MAT bestand her in te laden in Matlab.

6 Applicatie van het SQEP

6.1 Inleiding

Het Sound Quality Evaluation Program, waarvan de opbouw in vorig hoofdstuk uiteengezet is, is voorlopig enkel onder testomstandigheden beoordeeld. In dit hoofdstuk wordt dit hiaat weggewerkt met enkele illustratieve berekeningen op geluidsmetingen, waarbij de sound quality eigenschappen van bestaande producten onderzocht wordt. Het is daarbij spijtig dat de in het SQEP berekende waarden voor de fluctuatiersterkte en de ruwheid onbetrouwbaar zijn, wat impliceert dat deze niet in rekening gebracht kunnen worden. Het objectief bemeten beeld van de sound quality van een product is dan ook niet compleet. Dat neemt niet weg dat bepaalde deelaspecten ervan onderzocht kunnen worden. In dit hoofdstuk worden de resultaten die daar uit voortvloeiden besproken, evenwel beperkt tot enkele interessante bevindingen. De geluiden en de resultaten van de berekeningen er op zijn te vinden in de digitale bijlage.

Verder in dit hoofdstuk worden de mogelijkheden van het SQEP in de didactische omgeving van het lab Industriële Fysica en Toegepaste Mechanica bekeken, alsook deze in industriële omgevingen. De industriële relevantie van sound quality evaluatie is namelijk een punt dat niet uit het oog mag verloren worden.

6.2 Praktische applicaties

6.2.1 Dampkappen

Voor de dampkappenproducent Novy uit Kuurne zijn enkele metingen gedaan van het lawaai dat hun topmodel, de *Head'line wandkap*, voortbrengt. Deze kap is afgebeeld op de Figuur 6.1 en wordt schuin boven het fornuis gepositioneerd. De afzuiging gebeurt lang de randen en dat met drie mogelijke snelheden plus een power functie, bedoeld voor korte periodes. De wandkap kan in drie configuraties geleverd worden. Vooreerst is er de standaard installatie, dat is met een ingebouwde motor die de afgezogen damp naar buiten blaast. Een andere mogelijkheid bestaat er in de motor buiten te plaatsen en de damp van daar uit naar de buitenlucht te zuigen. De laatste configuratie is bedoeld voor keukens waar geen afzuigkanaal naar de buitenkant van het gebouw gelegd kan worden. De afgezogen damp wordt gewoon terug de keuken ingeblazen, maar niet zonder eerst door de actieve koolfilter die in de dampkap ingebouwd zit, te gaan. De actieve kool vangt de geuren in de damp op, zodat de keuken toch geurloos kan blijven.



Figuur 6.1: De testopstelling bij Novy.

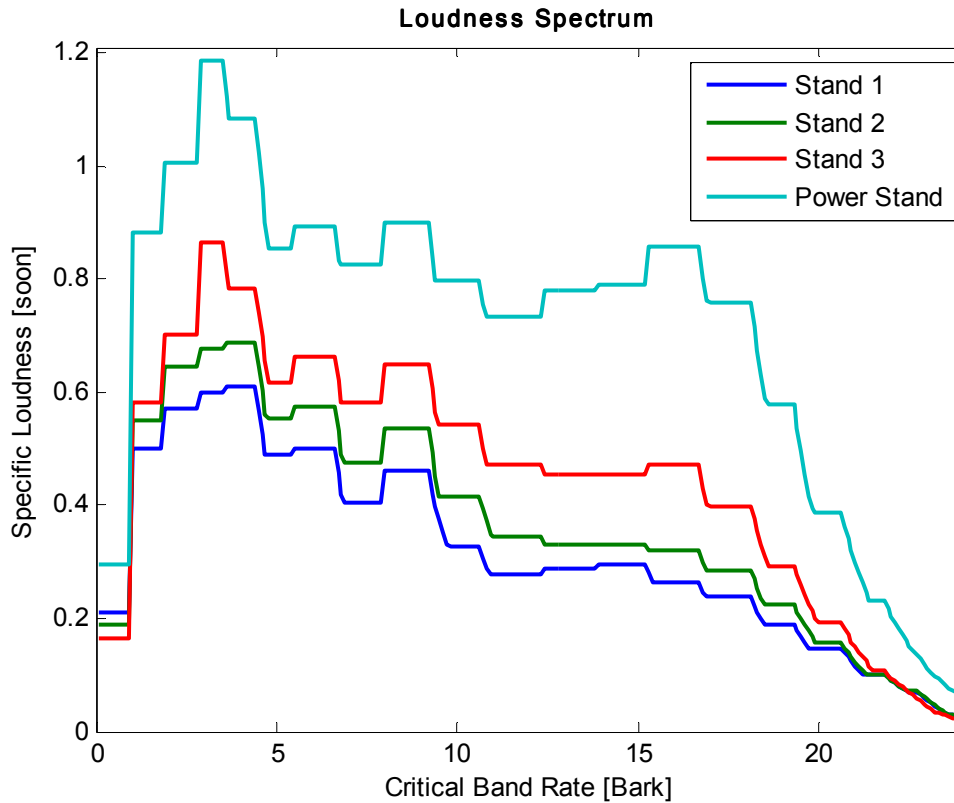
Op lawaaiverschil had Novy de verschillende configuraties nog niet onderzocht, evenals de verschillende afzuigsnelheden. Een meting op een testexemplaar bracht soelaas. Figuur 6.1 toont de op oorhoogte opstelde microfoon en de Matlab draaiende laptop. In Tabel 6.1 zijn de bekomen resultaten voor de stationaire luidheid van de verschillende wandkapconfiguraties en –afzuigsnelheden opgelijst.

Tabel 6.1: De resultaten van de meting op de Head'line wandkap.

Luidheid (soon) SPL (dB)	Stand 1		Stand 2		Stand 3		Power Stand	
Standaard	6,2	57,5	6,9	57,8	9,0	60,8	17,4	68,4
Motor op afstand	5,6	57,2	6,9	59,3	6,9	60,3	8,4	62,5
Met koolfilter	7,7	58,7	8,8	59,4	10,8	61,0	17,0	66,5

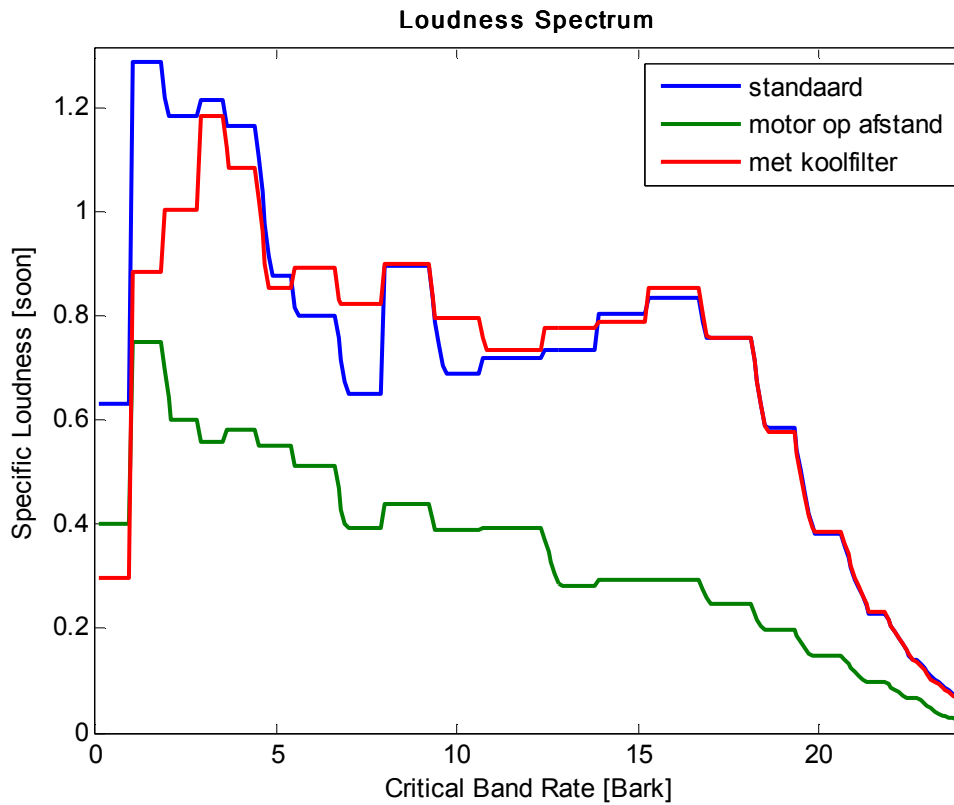
Tijdens de uitvoering van de metingen was er echter een achtergrondlawaai van 6,6 soon (56,8 dB). Dit kwam vooral door draaiende servers op de achtergrond. Deze verstoorden de meting dusdanig, dat de meetresultaten die in de tabel van een lichtgrijze achtergrond voorzien zijn, waardeloos zijn. De andere zijn nog bruikbaar als relatieve waarden. Zo kunnen op Figuur 6.2 enkele interessante zaken afgelezen worden van de luidheidsspectra van de vier standen van de dampkap met koolfilter. Vooreerst valt op dat de power stand in alle banden een heel stuk luider is dan de drie andere standen, die over het volledige spectrum dicht bij elkaar liggen. Verder valt op dat de vorm van stand 1 en 2 ietwat afwijkt van deze van stand 3

en de power stand. Meerbepaald in de drie Bark band vertoont de curve van de power stand een uitschieter die in eerstgenoemde niet aanwezig is. Onderzoek in het fft-beeld naar de waarde van de frequenties binnen die band kunnen uitsluitel geven en zijn eventueel een punt ter verbetering, maar vallen buiten het bestek van deze masterproef.



Figuur 6.2: Het luidheidsspectrum van de dampkap met actieve koolfilter, op de vier verschillende afzuigsnelheden.

Een andere samenstelling die opgesteld kan worden is deze van de power stand voor de drie verschillende wandkapconfiguraties. Figuur 6.3 toont dat de configuratie met motor op afstand over het hele spectrum een stuk stiller is dan de andere twee. Deze optie is dus zeker te overwegen voor iemand die de stilte in de keuken apprecieert. De andere twee zijn in grote lijnen gelijk. Enkel is de standaard configuratie luidruchteriger in de laagste banden, terwijl de uitvoering met koolfilter dit tussen vijf en tien Bark is. Deze verschillen kunnen onderzocht worden, zodat de oorzaak ervan duidelijk wordt. Dat maakt het mogelijk gerichte aanpassingen aan het product te doen teneinde het stiller te krijgen.



Figuur 6.3: Vergelijking van het luidheidsspectrum van de drie dampkapconfiguraties in de power stand.

Meer berekeningen naar sound quality analyse toe kunnen niet gedaan worden door het gebrek aan betrouwbare meetresultaten.

6.2.2 Haakse slijpmachines

Een machine waarvan niemand zal ontkennen dat ze nog heel wat progressie kan maken op auditief vlak, is de slijpmachine. Naast het feit dat haar volume de gehoorschadegrens overschrijdt, is ook haar snerpende lawaai een erg onaangenaam gegeven. In een poging dit eens in kaart te brengen, is het SQEP toegepast op twee draagbare haakse slijpmachines, een kleine en een grote. Er werden opnames gemaakt zowel in onbelaste (zie Figuur 6.4) als belaste toestand.



Figuur 6.4: Geluidspname van de onbelast draaiende kleine slijpmachine

Figuur 6.5 toont de grote haakse slijpmachine in belaste toestand; er wordt een snede gemaakt in een vijf mm dikke buis. De microfoon is telkens op veertig centimeter afstand geplaatst. Dit is in dit geval dezelfde afstand als deze van de slijpmachine tot het oor. Die afstand varieert natuurlijk naargelang de houding van de gebruiker, maar kan als normaal aangenomen worden. Het achtergrondgeluidsniveau was verwaarloosbaar.



Figuur 6.5: Geluidscaptatie van de grote slijpmachine in gebruik.

De slijpmachines gebruikt voor de opnames zijn alletwee van het merk Metabo[®] en zijn nog in goede staat. In Tabel 6.2 zijn hun belangrijkste technische gegevens, zoals opgegeven in de productcatalogus, opgesomd. Daarbij dient ten eerste opgemerkt dat de kleine slijper op het maximale toerental ingesteld stond en ten tweede dat het opgegeven toerental van 10.500 min^{-1} bij nominale belasting duidelijk verkeerd is en dus waardeloos.

Tabel 6.2: De technische kenmerken van de twee slijpmachines.

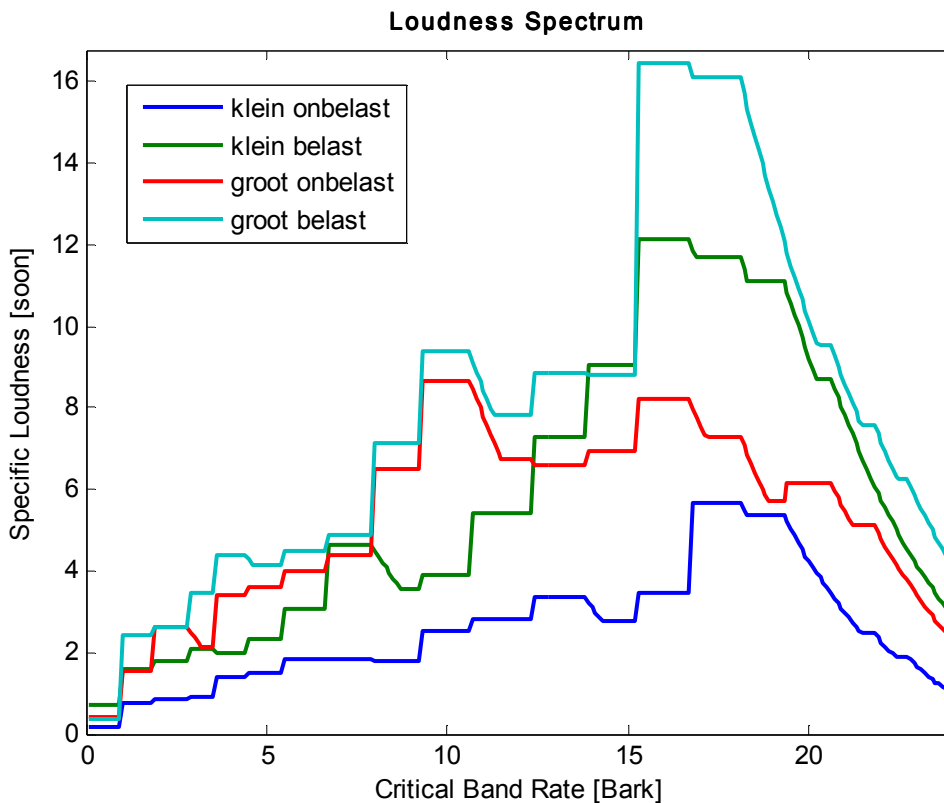
Formaat van de haakse slijpmachine:	Groot	Klein
Slijpschijf-/slijpsteen-Ø	230 mm	125 mm
Onbelast toerental	6.600 /min	7.000 - 10.500 /min
Nominaal opgenomen vermogen	2.300 W	1.400 W
Afgegeven vermogen	1.580 W	800 W
Toerental bij nominale belasting	4.600 /min	10.500 /min
Gewicht	4,9 kg	1,9 kg

6.2.2.1 Luidheid

De totale luidheden van de slijpmachines zijn opgesomd in Tabel 6.3. Opvallend zijn de erg hoge waarden; de 189,6 soon die de grote slijpmachine in belasting voortbrengt staat gelijk met een effectief geluidsdrumniveau van 105,2 dB. Dit is ver boven de gehoorschadegrens en dient eigenlijk eerst aangepakt te worden alvorens de ontwikkeling naar een aangenaam geluid kan gebeuren. Daarbij kan het luidheidsspectrum helpen. Op de Figuur 6.6 zijn de spectra vergeleken. Opvallend is de sterke stijging van de luidheid vanaf 15 Bark bij belasting en dit zowel voor de grote als de kleine slijpmachine. Deze wordt veroorzaakt door het lawaai dat voorkomt uit de wrijving tussen de slijpschijf en het metaal. Deze stijging is bijna integraal verantwoordelijk voor het luidheidsverschil tussen de belaste en de onbelaste toestand.

Tabel 6.3: De stationaire luidheid van de haakse slijpers.

Luidheid SPL	Onbelast		Belast	
Grote slijper	128,2 soon	98,5 dB	189,6 soon	105,2 dB
Kleine slijper	61,7 soon	86,8 dB	141,4 soon	100,8 dB



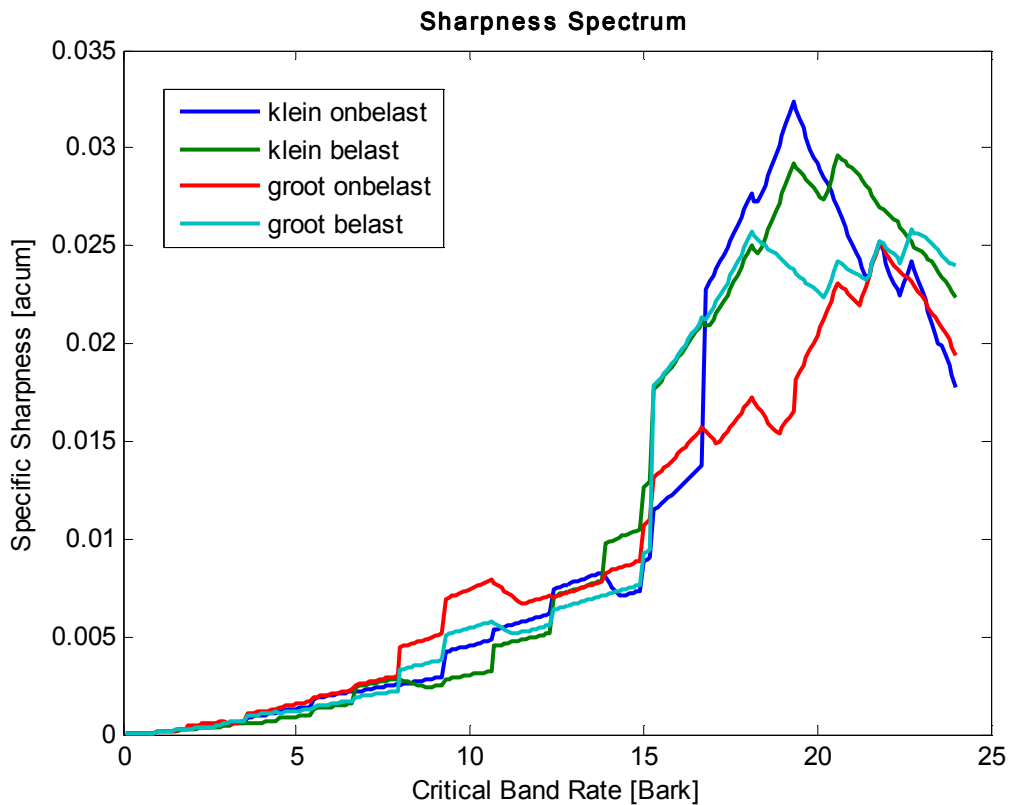
Figuur 6.6: De luidheidscurves van de slijpmachines.

6.2.2.2 Scherppte

Dat de waarden voor de scherppte van de slijpmachines hoog liggen, is geen verrassing. Tabel 6.4 toont echter wel verrassende cijfers; de scherppte van de kleine haakse slijpmachine ligt namelijk hoger dan deze van de grote. Dit bewijst dat het luidste geluid niet automatisch het scherpste is. De verschillen zijn echter niet zo groot en dan is het handig als een programma in staat is absolute waarden aan de scherppte van het geluid vast te kleven. Van de grafieken in Figuur 6.7 valt af te lezen dat alle curves gelijk opgaan tot 15 Bark, waar de specifieke scherppte van de belaste slijpmachines sterk stijgen. Ook de specifieke scherppte in de onbelaste toestanden stijgt naderhand, waarbij de kleine slijper onbelast zelfs hoger piekt dan belast. Toch blijft de totale scherppte van de slijper zonder belasting lager dan deze met belasting, of anders gezegd, de oppervlakte onder de grafiek is kleiner.

Tabel 6.4: De scherppte van de haakse slijpers.

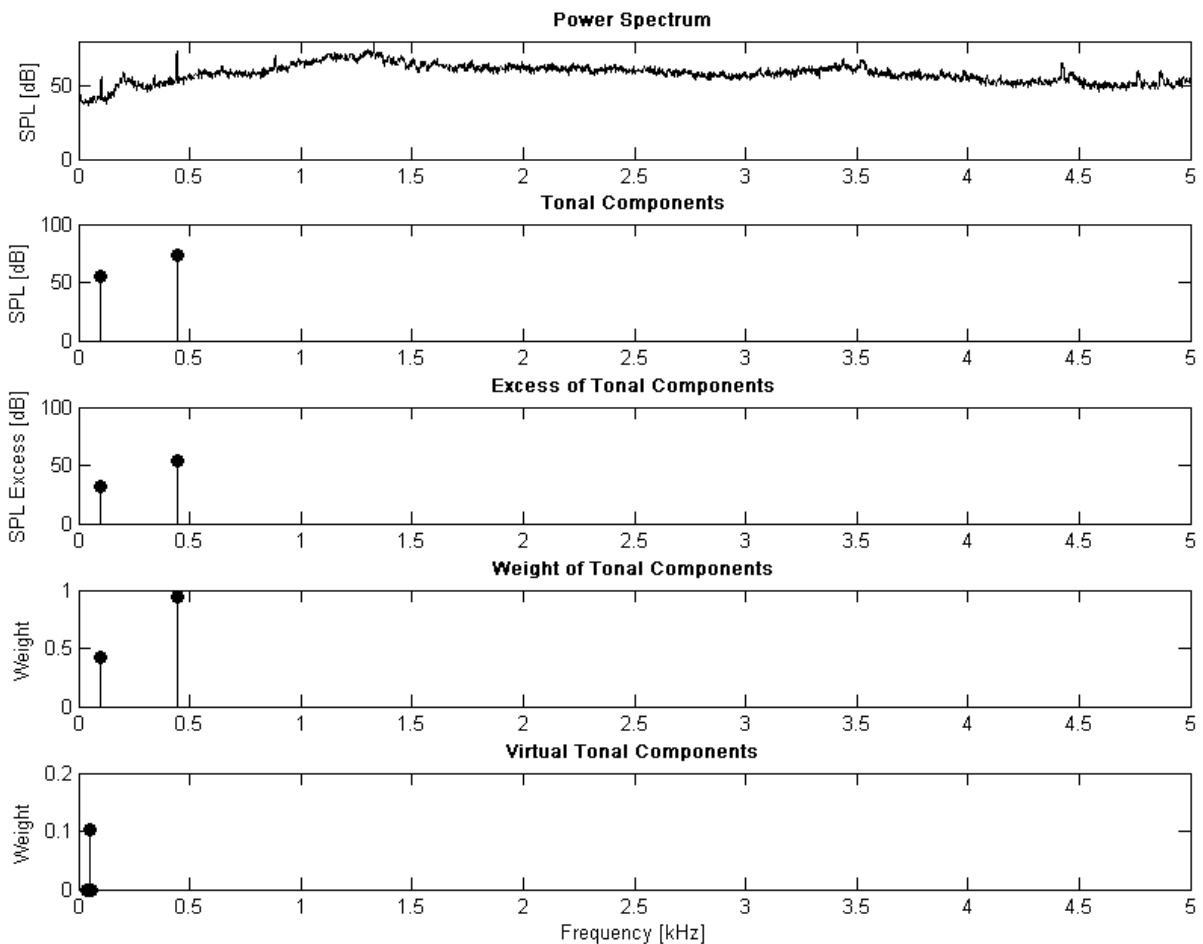
Scherppte in [acum]	Onbelast	Belast
Grote slijper	2,29	2,55
Kleine slijper	2,56	2,68



Figuur 6.7: De scherpptecurves van de slijpmachines.

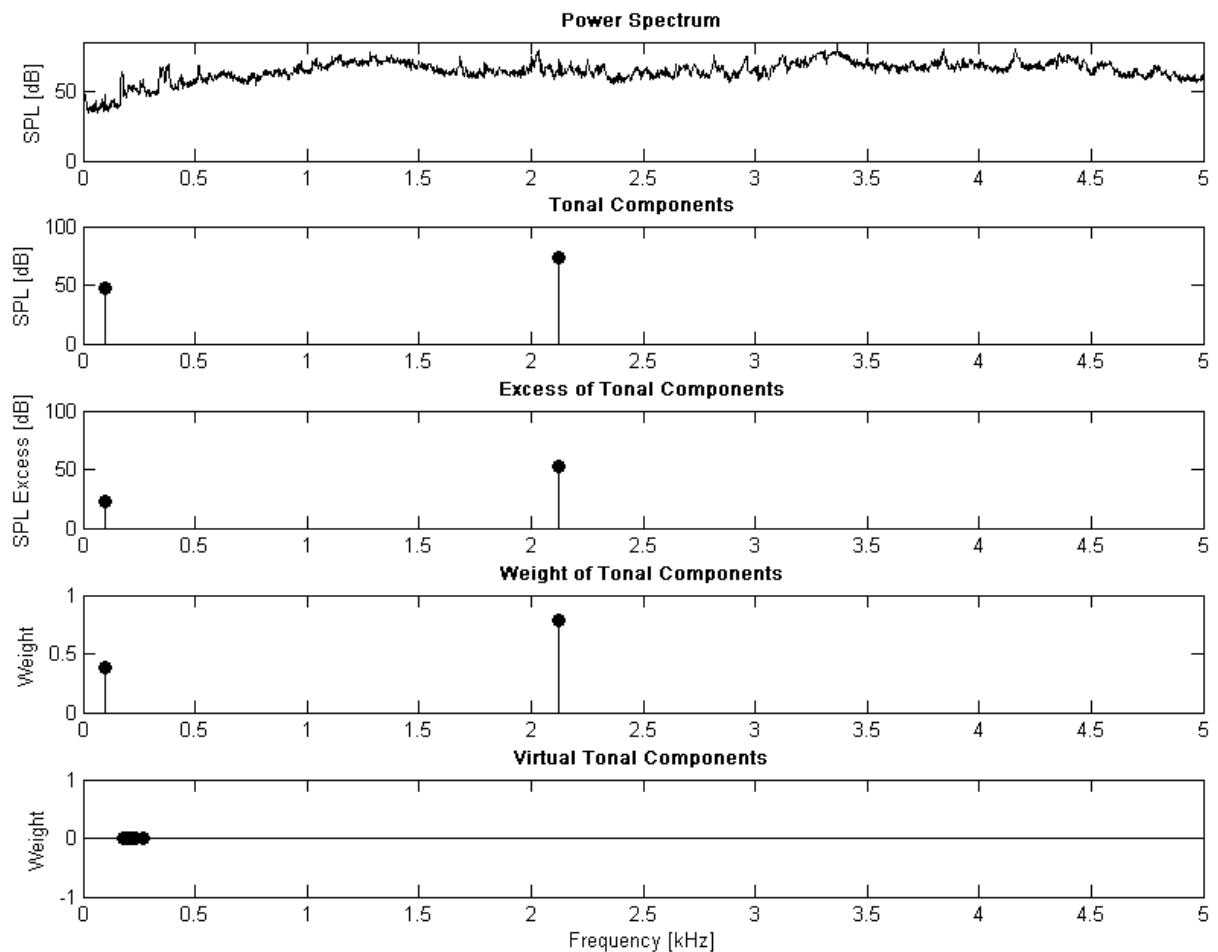
6.2.2.3 Tonaalheid

Wat de tonaalheid betreft, kunnen ook enkele significante zaken belicht worden. Dit zal enkel gebeuren voor de grote haakse slijpmachine daar deze betere resultaten leverde dan de kleine. Op Figuur 6.8 is te zien hoe het SQEP twee tonale componenten in het geluid vond. Eén op 100 Hz en één op 442 Hz. Deze zijn de rotatiefrequenties van respectievelijk de slijpschijf en de universeelmotor. 100 Hz is namelijk gelijk aan 6000 min^{-1} , wat het toerental van de slijpschijf zal zijn. Volgens de productcatalogus zou het onbelaste toerental 6600 min^{-1} moeten zijn, maar dit wordt dus niet gehaald. In het kader van de sound quality is het interessant te zien dat, hoewel de tonale componenten in SPL niet zoveel verschillen, zij dit wel doen in de menselijke perceptie; gewogen is de 442 Hz tonale component dubbel zo groot als de 100 Hz component. Aanpassingen ter reductie van de tonaalheid van de machine gebeuren dan ook best eerst bij de motor. Er is ook een virtuele toon aanwezig van 50 Hz, de helft van de 100 Hz component.

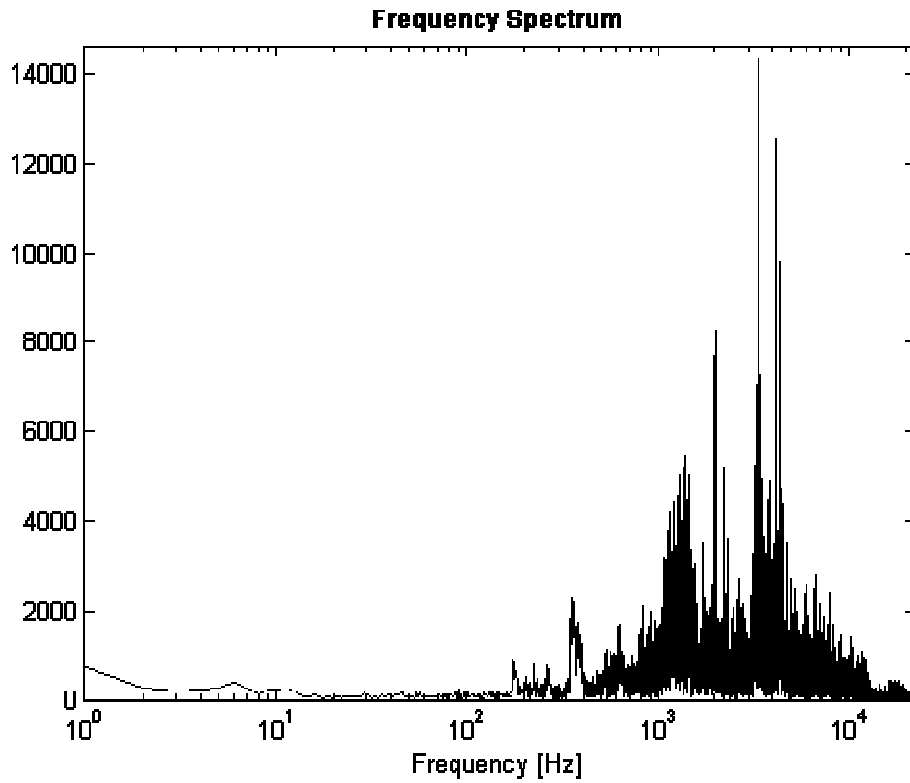


Figuur 6.8: De tonale componenten van de grote haakse slijpmachine in onbelaste werking.

De resultaten van de belaste grote slijper, gevisualiseerd in Figuur 6.9, tonen dat een en ander veranderd is. De 100 Hz component is nog steeds aanwezig, deze van 442 Hz niet. Na nader bekijken van het vermogenspectrum en het fft-beeld op de Figuur 6.10 is die component wel degelijk nog aanwezig, maar iets lager (rond 333 Hz) en met een spreiding. Dit komt door het licht variërend toerental, afhankelijk van de belasting, tijdens de meting. Interessant is de nieuwe tonale component gelegen op 2126 Hz, die een harmonische moet zijn. Deze blijkt namelijk niet uit het fft-beeld op de Figuur 6.10. Twee pieken, gelegen op 3369 en 4161 Hz springen in het oog, maar geen van deze blijkt een tonaal effect te creëren. Dat het SQEP dit traceert kan interessant zijn voor ingenieurs die de tonaalheid van de machine willen verminderen.



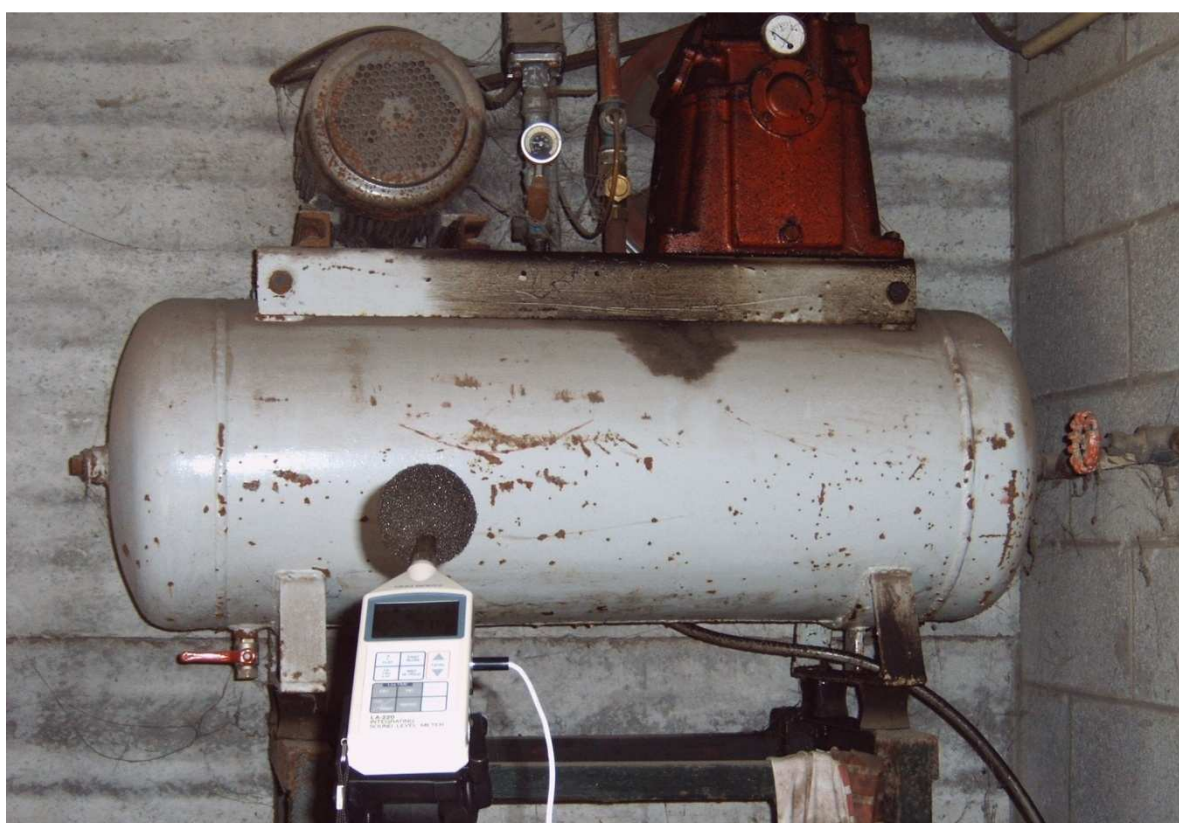
Figuur 6.9: De tonale componenten van de grote haakse slijpmachine in belaste werking.



Figuur 6.10: Het frequentiespectrum van de grote slijpmachine in belaste toestand.

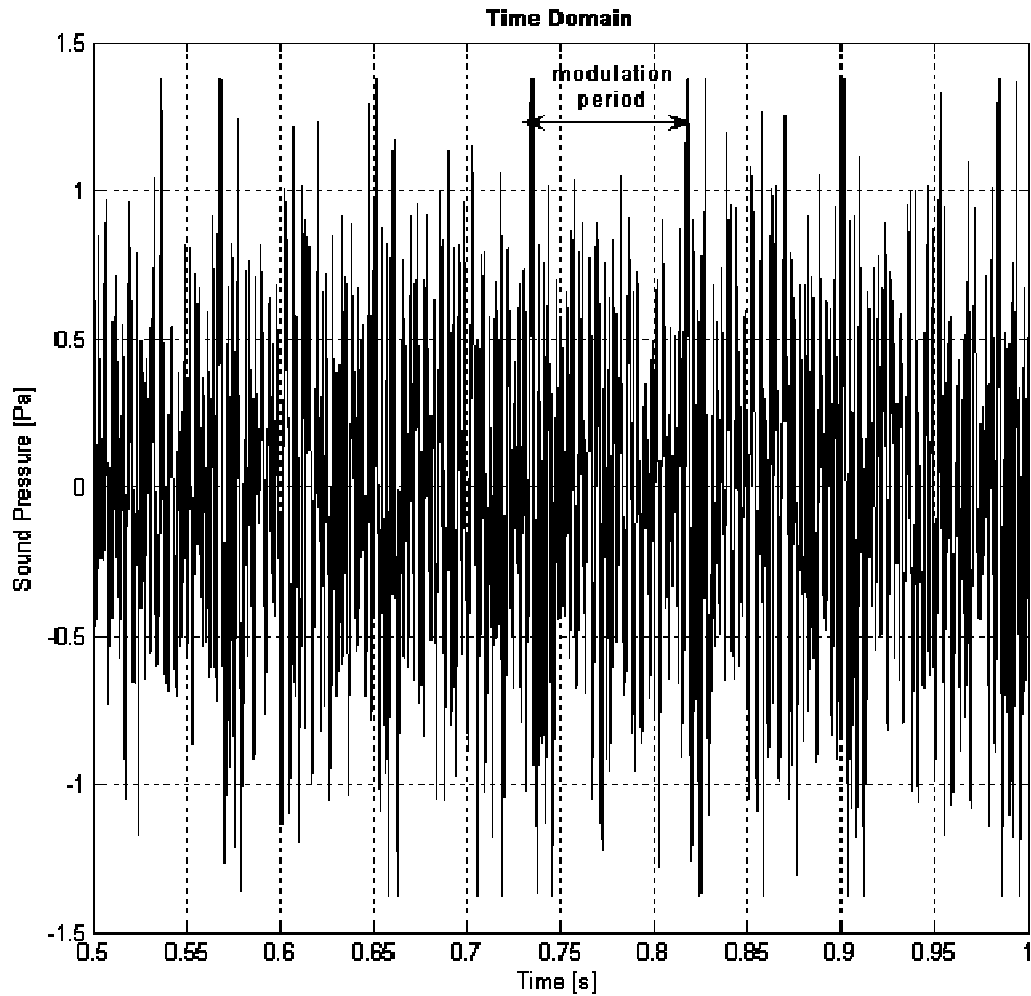
6.2.3 Compressor

Als laatste wordt het geluid van een al wat verouderde persluchtcompressor onderzocht. Hij is opgebouwd uit een elektrische motor, een eencilinder compressor en een drukvat (zie Figuur 6.11). Omdat de compressor een stampend geluid maakt, is het interessant zijn fluctuatiersterkte eens te bepalen. In hoofdstuk vijf wordt wel beschreven dat de uitkomst van de fluctuatieberekeningen onbetrouwbaar is, maar het doel van de meting ligt eerder bij de schatting van de modulatiefrequentie. De schatting moet manueel gebeuren, wat vanzelfsprekend goed lukt bij kunstmatige amplitude- of frequentiegemoduleerde signalen, maar niet voor de hand ligt bij productgeluid. Er zit altijd ruis op en het heeft bijlange geen mooie sinusvormige modulaties.



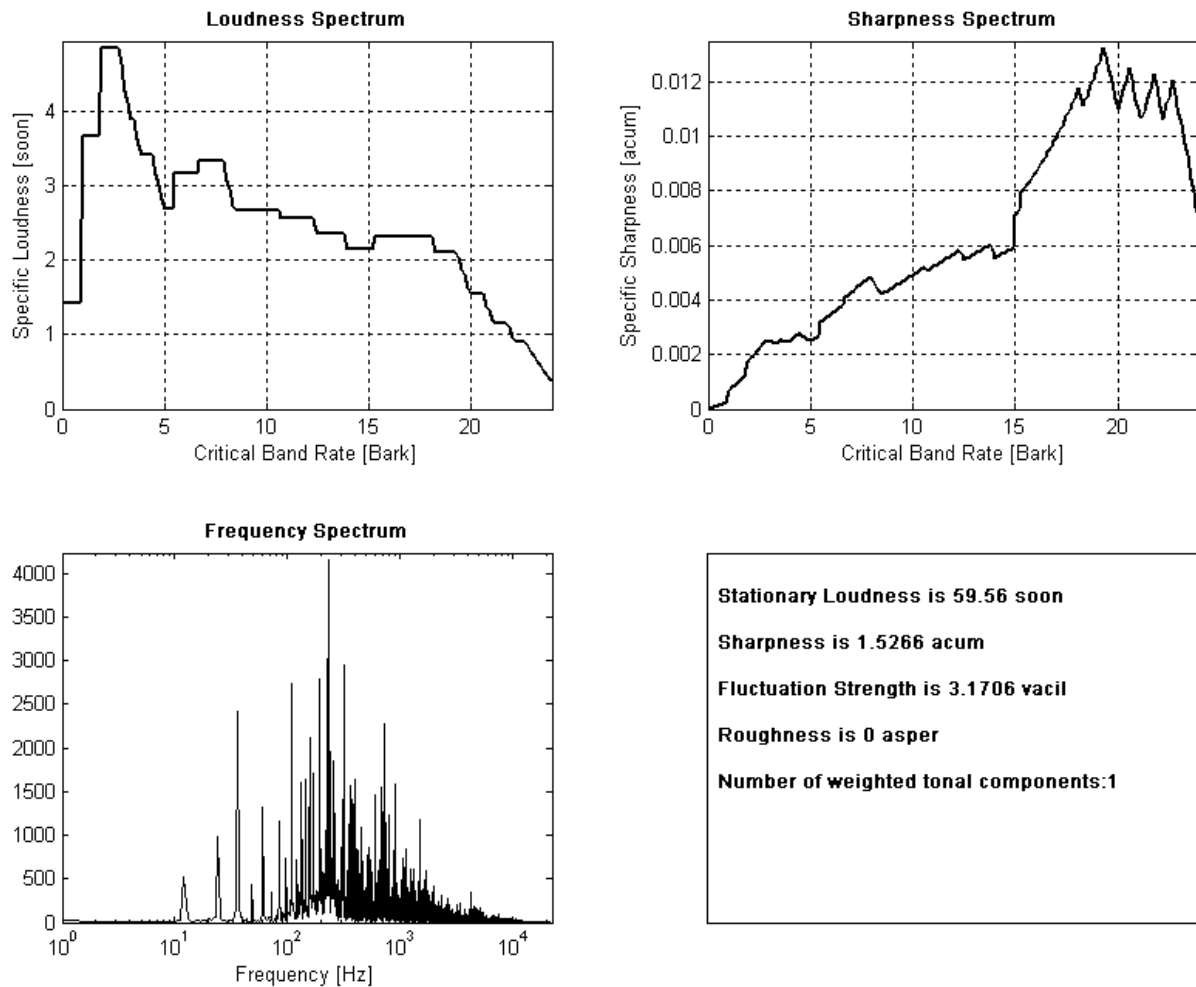
Figuur 6.11: De bemeeten éencilinder compressor.

Op de Figuur 6.12 is een stuk van het opgenomen compressorgeluid in het tijdsdomein te zien. De plot wordt automatisch gegenereerd in het SQEP, waarna gevraagd wordt welke de modulatieperiode is. Op de plot zijn de regelmatig terugkerende amplitudepieken uiterst geschikt om de modulatieperiode vast te leggen. De amplitudemodulatie hoeft niet sinusvormig te zijn! Met de dubbele pijl is de duur van een modulatieperiode afgebakend. Ze bedraagt hier 0,083 s, wat gelijk is aan een modulatiefrequentie van 12 Hz. Het is duidelijk dat dit de rotatiefrequentie van de compressor is.



Figuur 6.12: Afbeelding in het tijdsdomein van het compressorgeluid ter bepaling van de modulatieperiode.

Na ingave in het SQEP van deze waarde worden de psychoakoestische parameters berekend. Het eerste deel van de resultaten is weergegeven in Figuur 6.13. Zoals verwacht ligt het zwaartepunt van het luidheidsspectrum bij de lage kritische banden. Ook de fluctuatiersterkte is berekend en bedraagt 3,17 vacil, maar is dus een onbetrouwbare waarde.



Figuur 6.13: Deel 1 van de resultaten van de berekeningen op het compressor geluid.

6.2.4 Ventilatoren

Ventilatoren in alle vormen en groottes zijn een veel voorkomende bron van storend geluid. Ze komen voor in enorm veel apparaten en zijn al dikwijls herontworpen om een lawaaireductie te verkrijgen. Dat is ook het doel van de ontwerpers bij de firma Omnistor uit Menen, onderdeel van de Zweedse multinational Thule. Ze ontwerpen er onder andere ventilatoren, in te bouwen in de dakluiken van campers, *Omni-Vent* genaamd. Om het geluid te reduceren kunnen zij het schoepenaantal, de schoepdikte en de draaisnelheid wijzigen. Daarbij mag het doel van een ventilator, een zo groot mogelijk luchtdebiet verplaatsen, niet uit het oog verloren worden. Daar komt nog bij dat het stroomverbruik zo laag mogelijk gehouden moet worden, want deze komt van de auto-accu.

In november 2007 zijn in de geluidsarme kelder van het schoolgebouw luchtdebiet-stroomverbruik- en geluidsniveautesten uitgevoerd op twee bestaande en een nieuw ventilatormodel voor de *Omni-Vent*: een vijschoepige ventilator, een tienschoepige ventilator en een tienschoepige ventilator met een verstevigingsring rondom. Het geluidsdruk niveau werd gemeten –in dB(A)– met een decibelmeter en tegelijkertijd geïntegreerd met de LMS

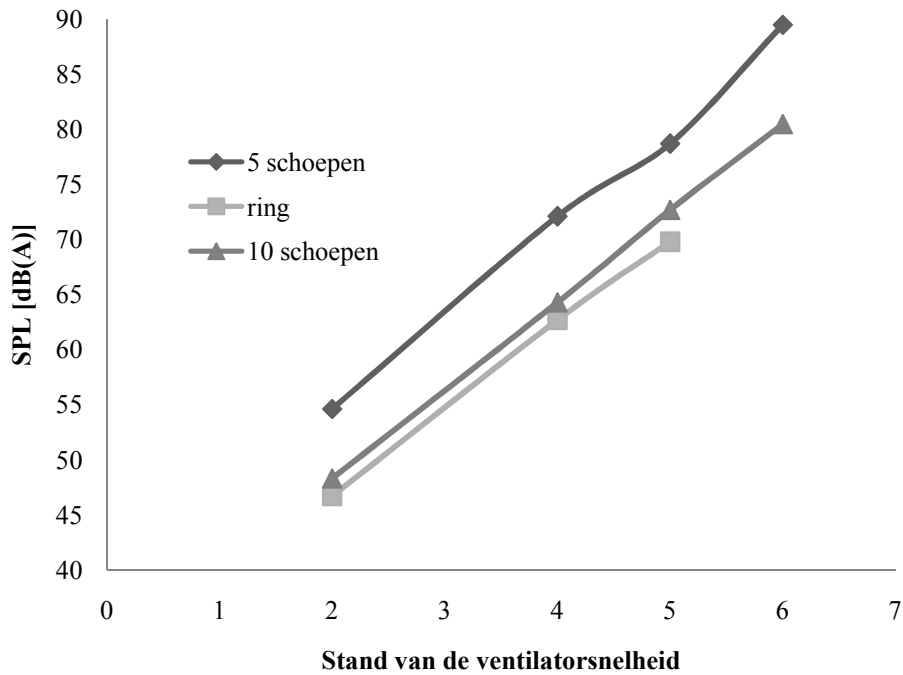
Pimento Analyzer, waarover sprake in paragraaf 5.2. De microfoons waren geplaatst op 60 cm afstand van de ventilatormotor, die op zes draaisnelheden kon worden ingesteld. Daarvan werden enkel de standen 2,4,5 en 6 bemeten, zowel links- als rechtsdraaiend, of zuigend en blazend. Hieronder wordt enkel de lucht in de camper blazende draairichting besproken. Figuur 6.14 geeft een impressie van de testopstelling.



Figuur 6.14: De testopstelling met de tien schoepen tellende ventilator.

Het doel van de metingen was de ventilatoren te vergelijken teneinde te weten te komen welke van de drie ventilatoren het best geschikt zou zijn om in productie te nemen, gelet op de verschillende kritische parameters. Figuur 6.15 vergelijkt daartoe de A-gewogen geluidsniveaus⁷ van de drie ventilatoren. Deze nemen lineair toe met de draaisnelheid. De vijf schoepen ventilator is duidelijk luidruchtiger dan de andere twee, die nauwelijks van elkaar verschillen. Van de ringvormige ventilator zijn geen gegevens in stand zes beschikbaar omdat de motorvoeding onvoldoende stroom kon leveren voor deze draaisnelheid.

⁷ Er wordt niet met de berekende stationaire luidheid gewerkt omdat geen waarden voor het ongewogen effectieve geluidsdrukniveau van de opgenomen fragmenten bekend zijn. Ten tijde van de opnames was het nog niet geweten dat deze daarvoor noodzakelijk zouden zijn. Ook de andere psychoakoestische parameters kunnen daardoor niet berekend worden, uitgenomen de tonaalheid. Daarvan is de absolute waarde (in dB) van de tonale componenten wel onbruikbaar, maar niet het aantal componenten.



Figuur 6.15: De A-gewogen geluidsniveaus van de ventilatoren voor de verschillende ventilatorsnelheden.

Bij het horen van de ventilatorgeluiden waren sommige ervan ergerlijk storend en zeker niet geschikt om te weerklinken terwijl personen in de camper verblijven. Op de grafieken in Figuur 6.15 valt deze nochtans belangrijke factor niet af te lezen. De ergerlijkheid van het geluid is natuurlijk recht evenredig met het volume ervan, maar hangt ook van andere zaken af. Zo is bij de ring- en tienschoepige ventilator de SPL (nochtans gewogen) ongeveer gelijk, terwijl eerstgenoemde bij de hoge toerentallen aanmerkelijk minder ergerlijk lawaai voortbrengt. Om dit toch te kunnen valoriseren, is het interessant de tonale componenten van het geluid ook in rekening te brengen. Tabel 6.5 geeft hun aantal weer, samen met het aantal gewogen componenten.

Tabel 6.5: Resultaten van de ventilortesten.

ventilatorvorm		stand van de ventilatorsnelheid			
		2	4	5	6
ring	SPL [dB(A)]	46,7	62,7	69,8	
	n tonale comp.	70	8	4	
	n gewogen tonale comp.	5	5	3	
vijf schoepen	SPL [dB(A)]	54,6	72,1	78,7	89,5
	n tonale comp.	46	20	27	32
	n gewogen tonale comp.	9	9	8	9
tien schoepen	SPL [dB(A)]	48,3	64,3	72,7	80,5
	n tonale comp.	23	7	9	7
	n gewogen tonale comp.	3	6	7	6

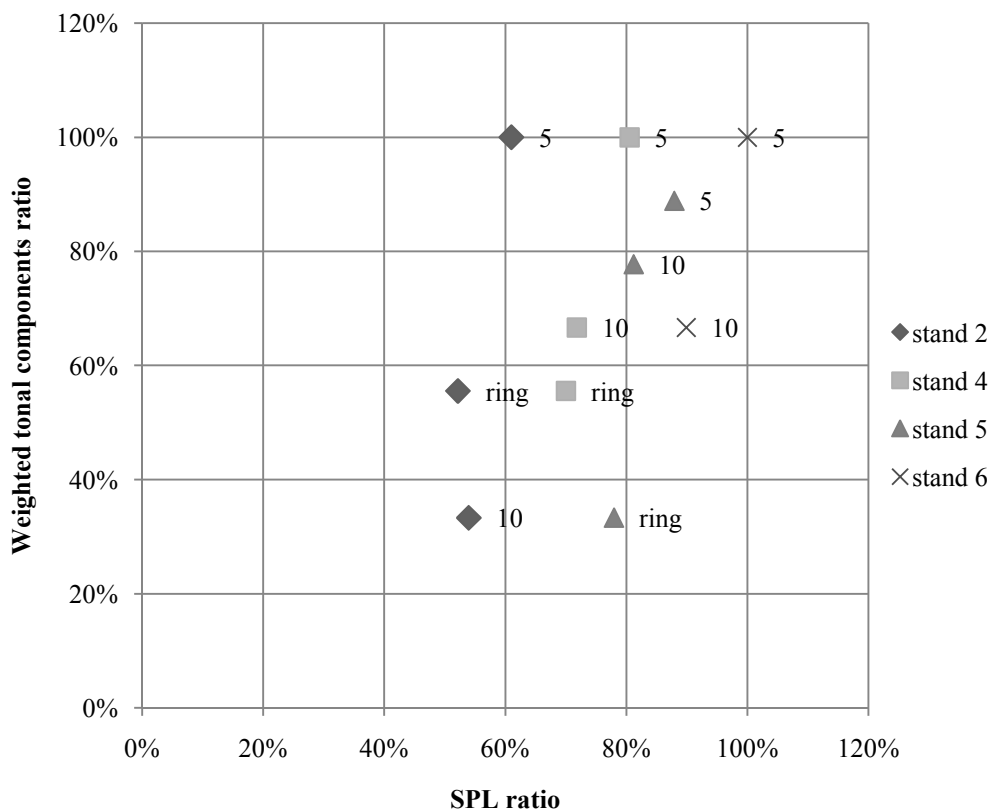
De ventilatorgeluiden bevatten aanzienlijk veel tonale componenten, maar na de weging blijven er maar enkele meer over. Het blijven er echter voldoende om het geluid tot een onaangenaam lawaai te degraderen. Hoe meer tonale componenten een geluid bevat, hoe

lelijker het zal klinken, op voorwaarde dat het geen harmonischen zijn (cf. dissonanten). Dit is hier zeker niet het geval want de meeste tonen worden veroorzaakt door de lucht die ‘fluit’. Het aantal gewogen tonale componenten wordt dus best ook in rekening gebracht bij de zoektocht naar de auditief aangenaamste ventilator. Omdat vooral de SPL-waarden te ver uiteenliggen om een praktische grafische voorstelling te kunnen maken, zijn in Tabel 6.6 alle waarden relatief weergegeven tegenover de hoogste, die in grijs gemarkeerd is.

Tabel 6.6: Procentuele resultaten van de ventilatortesten.

ventilatorvorm		stand van de ventilatorsnelheid			
		2	4	5	6
ring	SPL [dB(A)]	52%	70%	78%	
	n gewogen tonale comp.	56%	56%	33%	
vijf schoepen	SPL [dB(A)]	61%	81%	88%	100%
	n gewogen tonale comp.	100%	100%	89%	100%
tien schoepen	SPL [dB(A)]	54%	72%	81%	90%
	n gewogen tonale comp.	33%	67%	78%	67%

Op Figuur 6.16 zijn nu zowel tonaalheid als luidheid worden in rekening gebracht. Per draaisnelheid kan nu de minst storende ventilator gevonden worden. Dat is deze die het dichtst bij de oorsprong van de grafiek ligt. Zo is voor stand 2 de tienschoepige ventilator de minst ergerlijke, terwijl dat voor de standen 4 en 5 de ringvormige is. Dat beantwoordt aan de perceptie, net als het feit dat de vijfschoepige voor alle standen de slechtste optie is.



Figuur 6.16: Vergelijking van de ventilatoren, rekening houdende met tonaalheid en geluidsterkte.

Over alle draaisnelheden heen gezien, is de ringvormige ventilator de beste keuze. Spijtig genoeg geldt dit enkel op geluidsvlak. Met het hoogste stroomverbruik (auto-accu!) en een debiet dat drie maal zo laag ligt als bij de vijfschoepige ventilator is de keuze toch niet zo snel gemaakt...

6.3 Mogelijkheden in onderwijscontext

Het educatieve aspect van het Sound Quality Evaluation Program is tijdens de ontwikkeling ervan voortdurend in het achterhoofd gehouden, wat het geschikt maakt voor didactische doeleinden. Zoals in paragraaf 5.10 vermeld, is de opbouw zo duidelijk mogelijk gehouden en is de code van voldoende commentaar voorzien. Voor de elektromechanicus in opleiding kan het de sound quality theorie verduidelijken en hem helpen in de gerichte verbetering, voornamelijk door mechanische aanpassingen, van een productgeluid. Om geen praktische problemen met het SQEP te kennen is een handleiding, te vinden in bijlage bij deze scriptie, geschreven. Naast de elektromechanicus kan het programma ook van waarde zijn voor de studenten Industrieel Ontwerp. Zij moeten rekening houden met de *Human Factor*, aangehaald in paragraaf 4.2, waarin ook de aangenaamheid van een geluid vervat zit. Het SQEP kan daarbij een hulpmiddel zijn in de meting ervan.

In een labo-omgeving zou een opstelling gemaakt kunnen worden, waarbij de sound quality van twee gelijkaardige producten vergeleken kan worden. In een meer gedetailleerde benadering zou dan de rechtstreekse oorzaak van bepaalde psychoakoestische parameterwaarden kunnen gezocht worden. Bij grote apparaten is dan zelfs een geluidsintensiteitsmeting nodig om de bron van bepaalde storende frequenties te vinden. Deze opstelling is niet gerealiseerd omdat de sound quality berekening in het SQEP onvolledig is. Dat neemt niet weg dat, mits een basiskennis akoestiek, productgeluiden allerhande opgenomen, in het SQEP ingegeven en geanalyseerd kunnen worden, zoals gebeurd in paragraaf 6.2.

6.4 Mogelijkheden bij de lokale industrie

Dit voorjaar lanceert Novy een nieuwe imagocampagne. Voor de nieuwe campagne werd resoluut gekozen om de stilte, de meest karakteristieke eigenschap van een Novy-kap voorop te zetten. Op een eigentijdse en interpellierende manier wordt duidelijk gemaakt dat het plaatsen van een NOVY dampkap toelaat om weer te praten in te keuken. Een belangrijk element in de keuken van vandaag. "Heel stille dampkappen" zal niet enkel zichtbaar worden in een nieuwe mediacampagne in magazines en kranten, ook de catalogus en website worden in een volledig nieuw kleedje gestopt.

Bovenstaande tekst is te vinden op de website van de dampkappenproducent Novy. De firma, gevestigd in Kuurne, speelt de relatieve geruisloosheid van hun dampkappen uit als troef ten opzichte van concurrenten. Opvallend daarbij is de keuze voor een niet-technisch kenmerk van de dampkap, waarbij dat kenmerk niet het design is. Om die profilering geloofwaardig te maken, moeten de gebruikers natuurlijk het verschil in volume merken. Daarom is continu onderzoek naar geluiddempende aanpassingen vereist. Na contactering bleek de firma dan ook enthousiast om meer te weten te komen over het vakdomein van de sound quality. Dit is interessant om weten omdat zo duidelijk is dat onderzoek naar sound quality wel degelijk relevant is voor de industrie in de regio. Een kwantitatief onderzoek naar die industriële relevantie is weliswaar niet gevoerd, maar het is onwaarschijnlijk dat andere bedrijven die consumentenapparaten ontwikkelen het aspect van de product sound quality niet in rekening wensen te brengen. In de regio worden onder andere wasmachines, machines voor tuinonderhoud en plotters ontworpen en gefabriceerd, allemaal producten waarop sound quality analyse nuttig kan zijn. Ook samenwerking met firma's actief in de constructie van transportvoertuigen (treinen, bussen) kan een optie zijn. Als laatste zijn er dan nog de talloze onderdelenleveranciers, zoals het hierboven vermelde Thule, die een goede sound quality kunnen uitspelen naar hun afnemers toe.

Tot op een zeker niveau kan het SQEP de industrie helpen in de zoektocht naar de wegwerking van storende componenten van een geluid en de creatie van een aangenaam geluid. Dit is aangetoond met enkele voorbeelden in paragraaf 6.2. Het programma werkt echter niet volledig en is te eenvoudig om geavanceerde product sound quality evaluaties te doen. Zoals aangehaald is de kostprijs van gespecialiseerde software echter niet min, waardoor het programma als opstapje kan dienen ter inwijding in de materie.

Met die evaluatie is de kous natuurlijk niet af; het proces van *product re-engineering* moet dan nog aanvatten.

7 Conclusie

7.1 Conclusie

Deze masterproef over Sound Quality heeft als hoofddoelstelling het voor het lab Industriële Fysica en Toegepaste Mechanica nog onverkende vakdomein nader te bestuderen. De eerder theoretische inslag van de masterproef heeft als gevolg dat moeilijk concrete verwezenlijkingen aan de opgelegde doelstellingen getoetst kunnen worden.

Vooreerst is een intensieve literatuurstudie uitgevoerd, die een beeld schetst van de hedendaagse ontwikkelingen rond sound quality, geplaatst in een brede wetenschappelijke context. Daarin bleken niet alleen de akoestiek maar ook de psychoakoestiek een belangrijke rol te spelen. De relevante psychoakoestische parameters zijn dan ook grondig onderzocht. Ook hoe te werk gegaan wordt bij de bepaling van product sound quality en wat er voor de softwarematige berekening ervan op de markt beschikbaar is, is nagegaan.

In het kader van het onderzoek naar de haalbaarheid en realisatie van een eigen computerprogramma voor de evaluatie van sound quality is een Matlab[®] applicatie ontwikkeld die de psychoakoestische parameters automatisch berekent. De captatie van de geluidssignalen gebeurt met een gebruiksvriendelijke mobiele opstelling. De Matlab codes bestaan zowel uit eigen ontworpen delen als uit algoritmes die door anderen ontwikkeld zijn. Spijtig genoeg blijken na verificatie de resultaten voor twee van de psychoakoestische parameters –fluctuatiersterkte en ruwheid– onbetrouwbaar te zijn. De oorzaak ervan is achterhaald en ligt bij de ontoereikende, overigens niet-gestandaardiseerde formules voor de berekening ervan. Met het zelfgemaakte Sound Quality Evaluation Program (SQEP) zijn daarna enkele demonstratieve metingen gedaan op bestaande producten en machines. Aansluitend zijn daarbij de nodige contacten met de bedrijfswereld gelegd. Zo kan de vakgroep haar specialisatie in het onderwerp in de vorm van een project of een nieuwe masterproef zeker verder uitdiepen.

7.2 Ideeën voor de toekomst

7.2.1 Sound Quality algemeen

Stellen dat het sound quality vakgebied nog in de kinderschoenen staat, is een understatement. Stellen dat het vakdomein volgroeid is, is echter een overstatement. Het is nog in volle ontwikkeling, met nog heel veel ruimte voor evolutie. Zo kunnen de bestaande psychoakoestische meeteenheden duidelijk nog verbeterd worden en hopelijk ooit eens gestandaardiseerd, in navolging van luidheid. Dit ligt alsmaar minder voor de hand door de steeds verder uiteenlopende formules die de verschillende softwarefabrikanten hanteren. Ook nieuwe parameters zullen zeker nog ontwikkeld worden. Zo is bv. in referentie [114] achterhaald dat er een niet te verwaarlozen *end-effect* is bij het beoordelen van geluiden. Dat betekent dat, enigszins logischerwijs, de mens zijn oordeel over een geluid vooral van het einde ervan doet afhangen. Het is maar een voorbeeld van een van de psychoakoestische ontdekkingen die nog veel stof tot onderzoek zullen opleveren. Het dichter betrekken van aanliggende studiegebieden zoals psychologie, neurologie, natuurkunde en musicologie is daarbij vereist. Wat een algemene sound quality quoteringscode betreft, gaan stemmen op om een kwaliteitslabel op te richten of om na de $L_W(A)$ -waarde –die op veel producten verplicht vermeld is– een eenvoudig begrijpbare sound quality waarderingcode toe te voegen zoals een A B C D-schaal of plustekens naargelang de sound quality van het apparaat. Dit lijkt echter een erg moeilijk realiseerbaar ideaal. Ook de productspecifieke sound quality parameters zullen als een onderdeel van een algemene product- of taakspecifieke oplossing naar geluid toe nog op een veel hoger niveau ontwikkeld en verfijnd worden. Er worden zelfs al pogingen gedaan om, vanuit een veranderd sound quality beeld van een apparaat, malfunctie ervan te diagnosticeren.

Een gigantische progressie zal nog gemaakt worden op het vlak van sound quality voorspelling. Waar sound quality nu maar bepaald kan worden eens het product –eventueel als prototype– volledig af is, zal later veel tijd en geld uitgespaard kunnen worden door dit predictief te doen. Zoals altijd nemen ook hier de automobielingenieurs het voortouw met de bouw van virtuele wagens waarvan ze pogen de sound quality te voorspellen. Dit is echter aartsmoeilijk, door de vele situaties waarin de wagen kan verkeren (versnelling, motortoerental, wegdek, snelheid...), de niet-stationaire geluiden die hij produceert en de geavanceerde digitale signaalverwerkingstechnieken die vereist zijn. Voor veel producten zou het echter al een omwenteling zijn als de sound quality ervan überhaupt eens geëvalueerd zou worden.

7.2.2 Sound Quality in de opleiding

In het curriculum van de opleiding tot elektromechanisch of ontwerpingenieur zou sound quality een plaatsje kunnen krijgen in de cursus Mechanische Trillingen en Geluid en als onderzoeksdomein voor studenten kunnen dienen. Daartoe zijn mogelijke verderzettingen in het vakdomein, steunend op deze masterproef, opgelijst.

Naar de verdere uitwerking van de objectieve sound quality bepaling met een computerprogramma toe, zijn de volgende uitbreidingen mogelijk:

- Het meest stekend is het deficit aan het SQEP, nl. de onbetrouwbaarheid van de berekening van de parameters fluctuatiesterkte en ruwheid. De formule voor de maskeringsdiepte is bekend als de oorzaak ervan. Ze kan misschien weggewerkt worden door andere parametermodellen te coderen. Deze zullen echter niet gratis verkrijgbaar zijn. Eventueel kan de dissertatie van W. Aures een oplossing bieden. Ook aan de andere psychoakoestische parameters kan nog gesleuteld worden. Ze zouden kunnen vergeleken worden met een bestaand sound quality softwarepakket. Daarbij dienen wel de opmerkingen gemaakt te worden dat zo een pakket duizenden euro's kost en dat, luidheid uitgezonderd, de parameters niet gestandaardiseerd zijn!
- Uitbreiding van het SQEP kan het waardevoller maken. Zo lijkt het niet onmogelijk om naast stationaire ook transiënte luidheid te berekenen. De implementatie van fysische meetresultaten zoals octaaf- en tertsbandanalyse en A- en C-weging is mogelijk met de Matlabfuncties geschreven door C. Couvreur, maar vallen dus buiten de sound quality bepaling. Commerciële sound quality programma's laten de gebruiker overigens zelf zaken als de *window function* instellen. Dit vraagt wel een specialisatie in digitale signaalverwerking. Hulp kan misschien gevonden worden in het boek *Signals, Sound and Sensation* van W. Hartmann.
- Er zou een geluidsmodificerend programma kunnen ontwikkeld worden, dat bepaalde componenten uit of eigenschappen van een geluid weet weg te filteren zodat de ergerlijkheid ervan vermindert of de aangenaamheid verhoogt. Dit zou een luisterjury veel beter in staat stellen een waardeoordeel te geven aan een bepaald productgeluid. Aansluitend kan een geluidendatabank aangelegd worden, zodat snel vergelijken mogelijk wordt.

Daarnaast kan de volgende stap in de sound quality cyclus gezet worden; de *product re-engineering*. Zoals de naam laat uitschijnen wordt hierbij op één product of een productsoort gefocust. Het dient daarbij opgemerkt dat door de specificiteit van een product de kennisvergaring ter zake altijd maar een relatief belang kan hebben. Na een wetenschappelijk verantwoorde sound quality evaluatie kunnen gerichte aanpassingen aan het product gebeuren teneinde een betere sound quality te bekomen. Daarbij kan verschillende richtingen uitgegaan worden:

- Vooraan in de chronologie staat een volledig sound quality onderzoek op een product, eventueel vergeleken met een gelijkaardig product. Het vinden van correlaties tussen

subjectieve (met luisterjury's) en objectieve metingen van het productgeluid maken het mogelijk nieuwe productspecifieke parameters te ontwikkelen.

- Een ander paar mouwen is het vinden van de oorzaak van ongewenste waarden van sound quality parameters. Dit vergt een grondige ontleding van het apparaat in kwestie en de nodige kennis van geluidsbronnen en –overdracht in een machine. Een hulp daarbij kan het boek *Designing for Product Sound Quality* van R.H. Lyon zijn. Gerichtte constructie- of materiaalaanpassingen of wijzigingen aan de instellingen kunnen dan voor een goede sound quality zorgen zonder het product aan te veel andere positieve eigenschappen te laten inboeten. Uiteindelijk blijft het doel het product van een minder ergerlijk, aangener of aan de verwachtingen voldoende geluid te voorzien en zodoende een meerverkoop te realiseren.

Bibliografie

De bibliografie is genummerd per hoofdstuk of paragraaf waar mogelijk, daar exacte referering moeilijk is. Vele behandelde onderwerpen komen namelijk meermaals terug, zodat één bron niet echt gegeven kan worden zonder verscheidene andere te vermelden, wat tot onoverzichtelijkheid binnen de scriptie zou leiden. De website <http://www.wikipedia.org> is niet opgenomen in de bibliografie omdat deze weinig toegevoegde waarde aan de lijst biedt. Desondanks was Wikipedia een grote bron van informatie, die dikwijls de eerste noden lenigt bij problemen met vakdomeinspecifiek lexicon.

Hoofdstuk 1: Inleiding

- [1] eMercedesBenz Inc, *The New Mercedes C-Class: Superlative Ride Comfort Based On Digital Prototypes*. 01/04/08.
http://www.emercedesbenz.com/Jan07/09_The_New_Mercedes_C_Class_Superlative_Ride_Comfort_Based_On_Digital_Prototypes_1.html
- [2] Gade S., ‘What is Sound Quality?’. *Brüel & Kjær Magazine*, 1 (2007), 20-23.
- [3] Pierson R., Bosmoski A., ‘Harley-Davidson’s 100th anniversary – the sound of a legend’. *Sound and Vibration* (maart 2003).
http://findarticles.com/p/articles/mi_qa4075/is_200303/ai_n9223788/pg_1
- [4] LMS International, ‘Harley-Davidson: The Sound of a Legend’. *LMS News* 17-3 (2002) 8-10.

Hoofdstuk 2: Akoestiek en psychoakoestiek

- [5] Bohn D., Rane Corporation, *Pro Audio Reference*. 01/04/08.
<http://www.rane.com/digi-dic.html>
- [6] Brüel & Kjær, “*Basic Concepts of Sound*” *Brüel & Kjær Lecture Note BA 7666-11*. 1998
- [7] Brüel & Kjær, “*Basic Frequency Analysis of Sound*” *Brüel & Kjær Lecture Note BA 7669-11*. 1998
- [8] Greisbach R (Institut für Phonetik), *Skalierung der Empfindungen*. 31/03/08.
<http://phonetik.sprachsignale.de/grundlagen/aud18.html>
- [9] Hill D., *A Conceptionary for Speech & Hearing in the Context of Machines and Experimentation*. 01/04/08. <http://pages.cpsc.ucalgary.ca/~hill/papers/conc/index.htm>
- [10] Hogeschool Gent, *Milieuverontreiniging en –technologie “Geluid”*, Gent: Hogeschool Gent.
- [11] Intec, *Cursus Geluidshinder*. Gent: Universiteit Gent
- [12] Jensen J., *Virtual Pitch Algorithm of Terhardt*. 13/12/07.
<http://home.austin.rr.com/jmjensen/VirtualPitch.html>.

- [13] Karjalainen M., *Communication Acoustics*. Finland: HUT Acoustics Laboratory, 2007
- [14] Lichtblau C., ‘Tonhöhenwahrnehmung’, In: Universität Ulm, *Seminar Sehen und Hören*. Duitsland: Universität Ulm, 2004.
- [15] LMS International, ‘Acoustics and Sound Quality’. *The LMS Theory and Background Book*. Leuven: LMS, 2000.
- [16] Nave R., *HyperPhysics Concepts: Sound and Hearing*. 01/04/08.
<http://hyperphysics.phy-astr.gsu.edu/hbase/sound/soucon.html>
- [17] Nederlandse Vereniging voor Audiologie, *Nederlands Leerboek Audiologie*. 31/03/08.
<http://www.audiologieboek.nl/niveau2/indexniv2.htm>
- [18] Smith III J. & Abel J., *Equivalent Rectangular Bandwidth*. 09/04/08.
http://ccrma.stanford.edu/~jos/bbt/Equivalent_Rectangular_Bandwidth.html
- [19] Terhardt E., *Topics of Research*. 13/12/07.
<http://www.mmk.ei.tum.de/persons/ter.html>.
- [20] Traunmüller H., *Conversion between Hz, semitones, bark, ERB, etc.* 31/03/08.
<http://www.ling.su.se/staff/hartmut/umrechnung.htm>
- [21] Vanslambrouck B., *Trillingen en Geluid*. Kortrijk: Hogeschool West-Vlaanderen, 2007
- [22] Zwicker E. & Fastl H., *Psychoacoustics, Facts and Models*, third edition. Berlin: Springer, 2007

Hoofdstuk 3: Sound Quality parameters

Luidheid

- [23] Sengpiel E., *Tontechnik-Rechner*. 01/04/08.
<http://www.sengpielaudio.com/calculatorSonephon.htm>
- [24] Wolfe J., University New South Wales, *Equal loudness contours and audiometry – Test your own hearing*. 01/04/08. <http://www.phys.unsw.edu.au/jw/hearing.html>
- [25] Järveläinen H. et al., ‘Noise annoyance study in the cab of mobile work machines’, Finland: Helsinki University of Technology.
- [26] International Organization for Standardization, *Acoustics – Vol.1*, second edition. Genève: ISO, 1995
- [27] Nilsson M.E., ‘A-weighted sound pressure level as an indicator of short-term loudness or annoyance of road-traffic sound’. *Journal of Sound and Vibration*, 302 (2007), 197-207.

Ruwheid

- [28] Pressnitzer D., McAdams S., ‘Influence of Phase Effects on Roughness Modelling’. *ICMC: International Computer Music Conference, Thessaloniki, Griekenland, 1997*.
<http://mediatheque.ircam.fr/articles/textes/Pressnitzer97a/>

- [29] Terhardt E. *Auditory Roughness*. 31/03/08.
<http://www.mmk.e-technik.tu-muenchen.de/persons/ter/top/roughness.html>
- [30] Hoeldrich R., 'An Optimized Model for the Objective Assessment of Roughness Sensations in Vehicle Noise', *Sound Engineering Conference*, Graz, 1999.
- [31] Hoeldrich R., M. Pflueger, 'A Parameterized Model of Psychoacoustical Roughness for Objective Vehicle Noise Quality Evaluation', Graz: Joanneum Research Forschungs, 1998.
- [32] Vassilakis P., *SRA: An online research tool for spectral and roughness analysis of sound signals*. DePaul University Chicago, USA: MERLOT (2007).

Tonaalheid

- [33] Buss S., Weber R., 'Perception of tonalness of tyre/road noise and objective correlates'. *The 33rd International Congress and Exposition on Noise Control Engineering*, 2004.
- [34] Hastings A. et al., 'Measurement of the attributes of complex tonal components commonly found in product sound'. *Noise Control Engineering Journal*, 51 (4), 195-209.
- [35] Patsouras C., Fastl H. et al., 'Psychoacoustic evaluation of tonal components in view of sound quality design for high-speed train interior noise'. *Acoustic Science & Technology*, 23 (2002) 113-116.
- [36] Püschel D., 'Beurteilung und interaktive Filterung von Tonalität und Modulationen'. *Lärmbekämpfung Bd.2*, 4 (2007), 149-154.
- [37] Vormann M. et al., 'A new approach for the evaluation of tonal noise (tonality)', *World Scientific*, 2002, 1-5
- [38] Vormann M. et al., *Subjective Rating of Tonal Components in Noise with an Adaptive Procedure*.

Pleasantness & Annoyance

- [39] Västfjäll D. et al., 'Affective evaluations of and reactions to exterior and interior vehicle auditory quality'. *Journal of Sound and Vibration*, 255 (2002), 501-518.

Productspezifische parameters

- [40] Dedene L., *Integration of sound quality in the design process of exhaust systems*, Ph.D. thesis, Brussel: Vrije Universiteit Brussel, 2001.
- [41] Hashimoto T., 'Sound quality approach on vehicle interior and exterior noise –Quantification of frequency related attributes and impulsiveness-'. *Journal of the Acoustical Society of Japan* 21, 6 (2000) 337-340.

- [42] Kuwano S., Fastl H. et al., 'Quality of door sounds of passenger cars'. *Acoustic Science & Technology* 27-5 (2006), 309-312.
- [43] Lee S.-K., 'Objective evaluation of interior sound quality in passenger cars during acceleration'. *Journal of Sound and Vibration* (2007).
- [44] Lyon R.H., 'Product Sound Quality – from Perception to Design', *Sound and Vibration*, 2003, 18-22.

Hoofdstuk 4: Beoordeling van Sound Quality

Menselijke perceptie

- [45] Bodden M., *Perceptual Sound Quality Evaluation*.
- [46] De Baene W., A. Vandierendonck et.al., *Roughness perception of amplitude modulated tones is context dependent*. Gent, Universiteit Gent, 2003
- [47] Nordtest, 'Acoustics: Human sound perception – measurement positions for measurement of loudness', Finland: Nordtest.
- [48] Piker G., *Evaluation of product sound design within the context of emotion design and emotional branding*, Master's thesis, Izmir: Izmir Institute of Technology, 2005

Meetuitrusting

- [49] Core Sound LLC, *Core Sound Binaural microphone set*. 31/03/08.
<http://www.core-sound.com/mics/1.php>
- [50] Microphone Madness Inc, *Cardioid Stereo Microphones*. 31/03/08.
<http://www.microphonemadness.com/products/mmhigsencars.htm>
- [51] Roberts J.R. et al., 'Evaluation of impact sound on the 'feel' of a golf shot'. *Journal of Sound and Vibration*, 287 (2005), 651-666.
- [52] Bodden M., 'Instrumentation for Sound Quality Evaluation', *Acta Acustica* 83, (1997), 775-783.
- [53] Brüel & Kjær, *Product Data Sound Quality Head and Torso Simulator – Types 4100 and 4100D*, 2001.
- [54] Jiang L., Macioce P., *Sound Quality for Hard Drive Applications*.
- [55] Hassall J.R., K. Zaveri, *Acoustic Noise Measurements*, fifth edition. Denemarken: Brüel & Kjær, 1988.

Testjury

- [56] Brüel & Kjær, *Product Data I-deas Jury Evaluation Software*. 2006.
- [57] Edwards B., *Psychoacoustic testing of modulated blade spacing for main rotors*. NASA, 2002.

- [58] Hardy A.E.J., 'Measurement and assessment of noise within passenger trains'. *Journal of Sound and Vibration* 231 (2000), 819-829.
- [59] Head Acoustics, *Jury Testing and Listening Rooms*. 13/12/07.
http://www.headacoustics.com/sound_vibes/products/jury_testing.html
- [60] Otto N., S. Amman et al., 'Guidelines for Jury Evaluations of Automotive Sounds'. In: *Sound & Vibration*, (2001), pp.1-14.

Software

- [61] 01dB-Metravib, *dBsonic: Sound Analysis made easy!*
- [62] 01dB-Metravib, *Product Data: dBsonic. Sound Quality. Checking and Improving*, 2006.
- [63] Akustik Technologie Göttingen, *Si++ Workbench*. 2007.
- [64] Brüel & Kjær, *Pulse Sound Quality Software – Type 7698*. 2005.
- [65] Head Acoustics, *Produkte – Artemis Analyse-Software*. 13/12/07. <http://www.head-acoustics.de/2s-und-v/artemis/P-arte-d.htm>.
- [66] LMS International, *LMS Sound Quality Engineering. An Objective Solution to a Subjective Problem*.
- [67] Müller-BBM, *PAK*. 2005.
- [68] ViAcoustics, *Nelson Acoustics Sound Quality Analyzer*. 13/12/07.
<http://www.viacoustics.com/naesqtools.html>.

Toepassingen

- Behandeld

- [69] Ferguson S., Cabrera D. et al., 'Using Psychoacoustical models for information sonification'. *Proceedings of the 12th International Conference on Auditory Display*, 2006, 113-120.
- [70] Persson Waye K. & Öhrström E., 'Psycho-acoustic characters of relevance for annoyance of wind turbine noise'. *Journal of Sound and Vibration* 250 (2002) 65-73.
- [71] Nor MJM et al., Index for vehicle acoustical comfort inside a passenger car, *Applied Acoustics* (2007), 1-11.
- [72] Jeon J.Y., J. You et.al., 'Sound radiation and sound quality characteristics of refrigerator noise in real living environments'. *Applied Acoustics*, 68 (2007), 1118-1134.

- Niet behandeld

- [73] de Diego M. et al., *Subjective evaluation of actively controlled interior car noise*.
Universidad Politénica de Valencia.
- [74] Genuit K., 'Sound engineering of vehicle noise'. *HEAD acoustics*, Duitsland.

- [75] Parizet E. et al., 'Analysis of car door closing sound quality', *Applied Acoustics* (2007).
- [76] Sirkka A., *Component Sound Quality. A Power Window Study*, Master's thesis, Zweden: Lulea University of Technology, 2007.
- [77] Susini P. et al., 'Characterizing the sound quality of air-conditioned noise'. *Applied Acoustics* 65 (2004) 763-790.

Hoofdstuk 5: Ontwikkeling van een computerprogramma

Matlab handleidingen

- [78] Bendermacher N., de Graaff Stoffers R., *Matlab, een korte inleiding*. 2005.
- [79] Fujimoto K. *Sound by Matlab*. 29/03/08.
<http://www.h6.dion.ne.jp/~fff/technique/auditory/matlab.html>
- [80] Hastings A. *ISO 532 B / DIN 45631 Loudness*. 31/03/08.
<http://www.auditory.org/mhonarc/2000/msg00498.html>
- [81] Kerber S. *Matlab Übersetzung des BASIC Programms aus DIM 45631*. 31/03/08.
<http://www.mmk.ei.tum.de/~kes/DIN45631.m>
- [82] Martens F. et al., *Interactieve MATLAB Cursus. Technische Universiteit Eindhoven, fac. Werktuigbouwkunde en fac. Biomedische Technologie (2001)*.
- [83] Mathworks Inc, The, *Signal Processing Toolbox User's Guide*. Natick, MA (2008).
- [84] Speech and Hearing Group, Department of Computer Science University of Sheffield, UK. *Matlab Auditory and Speech Demos*. 31/03/08.
<http://www.dcs.shef.ac.uk/~martin/MAD/docs/mad.htm>
- [85] Timoney J., Lysagh T. Et al., 'Implementing loudness models in Matlab' *DAFx '04 Proceedings*, Napels, Italië (2004), 177-180.

Captatie van het geluid

- [86] Benford T., *Welcome to... PC sound, music, and midi. From mystery to mastery*. New York: MIS:Press, 1993.
- [87] Brüel & Kjær, *Measuring Sound. BR 0047-13*. Denemarken: Brüel & Kjær, 1984.
- [88] Chen A., Shehad N. et al., *W.A.V.S. Compression*. 01/04/08.
<http://is.rice.edu/~welsh/elec431/index.html>
- [89] Cisco Systems, *Waveform Coding Techniques*. Doc ID:8123 (2006)
- [90] ePanorama.net, *PC soundcard page*. 01/04/08.
http://www.epanorama.net/links/pc_sound.html
- [91] Harris S. & Sanchez C., *Personal Computer Audio Quality Measurements*. Cirrus Logic: Crystal Audio Division.
- [92] LMS International, *LMS Pimento: The Multichannel Analyzer with Portable Power*. Leuven: LMS, 2003.

Verwerking van het geluidssignaal

- [93] Andringa T., *Fouriertransformatie?* 01/04/08.
http://www.ai.rug.nl/~tjeerd/Onderwijs/Spraak/main/practicum/week1_2pract.html
- [94] Culling J, *Online, cross-indexed dictionary of DSP terms.* 01/04/08.
http://www.cf.ac.uk/psych/home2/CullingJ/frames_dict.html
- [95] DLI Engineering, *Vibration Manual.* 01/04/08.
<http://www.dliengineering.com/vibman/Alan2-toc.htm>
- [96] Introduction to Naval Weapons Engineering: *Frequency Modulation.* 29/03/08.
<http://www.fas.org/man/dod-101/navy/docs/es310/FM.htm>.
- [97] Kaplan I., *A Notebook Compiled While Reading 'Understanding Digital Signal Processing' by Lyon R.* 01/04/08.
http://www.bearcave.com/misl/misl_tech/signal/index.html
- [98] Rocchesso D., *Introduction to Sound Processing.* Università di Verona. 2003.
- [99] LMS International, 'Signal Processing'. *The LMS Theory and Background Book.* Leuven: LMS, 2000.
- [100] LMS International, 'Time data processing'. *The LMS Theory and Background Book.* Leuven: LMS, 2000.

Formules voor de psychoakoestische parameters

- [101] Cabrera D., "'Psysound": A computer program for psychoacoustical analysis', Proc. Of the Australian Acoustical Society Conference, Melbourne (1999), 47-54.
- [102] Byongho C., Nonmetric multidimensional scaling of complex sounds. Dimensions of preference ratings and perceived similarity of vehicle noise, Ph.D. thesis, Oldenburg, University of Oldenburg, 2001.
- [103] Head Acoustics, Loudness and Sharpness Calculation with Artemis. Head Application Note, 2005.
- [104] Wolters M., Sapp M. et al., Adaptive Algorithm For Detecting And Reducing Sibilants In Recorded Speech. 29/03/08. <http://www.omniaaudio.com/tech/de-esser.htm>
- [105] Wonho Y., Enhanced modified bark spectral distortion (EMBSD): An objective speech quality measure based on audible distortion and cognition model, Ph.D. thesis. Washington: Temple University, 1999
- [106] Zhang Z., Shrestha M., Sound Quality User-defined Cursor Reading Control – Tonality Metric. Master's Thesis, Denemarken: Danmarks Tekniske Universitet, 2003.

Hoofdstuk 6: Applicatie van het SQEP

- [107] Metabo Belgium, *Productcatalogus Handgeleid Elektrisch Gereedschap*. 14/05/08.
<http://www.metabo.be/Productcatalogus-handgeleid-elektrisch-gereedschap.24144+M5bf0e807042.0.html>.
- [108] Novy, Productcatalogus. Kuurne: NOVY, 2008.
- [109] Omnistor, *Omni-Vent*. 16/05/08. <http://www.omnistor.be/nl/products/omnivent.php>

Hoofdstuk 7: Conclusie

- [110] Lim T. C., ‘Correlations between deficiencies in power window systems influencing sound quality and some psychoacoustic metrics’. *Applied Acoustics*, 62 (2001), 1025-1047.
- [111] LMS International, *Improving the sound quality of the closing sound of a car door*.
- [112] LMS International, ‘Towards Sound Quality Engineering of the Virtual Car. From a Computer Game to an Engineering Tool’. *LMS News*, 15-1 (2000), 14-17.
- [113] Risberg T. *The Silent PC – Noise Labels & Acoustical Noise Declarations*. 01/04/08.
<http://www.silent.se/labels.php>
- [114] Västfjäll D. ‘The “end effect” in retrospective sound quality evaluation’. *Acoustic Science & Technology* 25, 2 (2004).

Bijlage A: Generatie van tonen en ruis in Matlab

In deze bijlage worden de codes meegegeven waarmee specifieke tonen en ruis gegenereerd zijn om de werking van de verschillende functies te verifiëren. Ze zijn als scriptfiles opgebouwd en heten respectievelijk **GenerateTone.m** en **GenerateNoise.m** maar zijn niet uitvoerbaar. Het is de bedoeling ze te openen in een editor en de gewenste code te kopiëren naar het Command Window. Het is zeker mogelijk er toch automatisch uitvoerbare scripts van te maken, maar dat zou een programma op zich vormen en valt te ver buiten het bestek van deze masterproef. De scripts zijn een samenstelling van eigen ontworpen en bestaande codes gevonden op Matlab Central (categorie Signal Processing), een online uitwisselplatform waarop gebruikers hun eigen codes kunnen plaatsen.

Tonen

```

1 % GenerateTone.m
2 % Algoritm to create pure, amplitude modulated and frequency
  modulated tones
3 % Change values and copy the needed code to the Matlab Command
  Window
4 % s is the final signal
5
6 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
7 % Generate pure tone
8
9 Fs = 44100; % sample frequency (Hz)
10 d = 10.0; % duration (s)
11 n = Fs * d; % number of samples
12
13 % set carrier
14 cf = 1000; % carrier frequency (Hz)
15 c = (1:n) / Fs; % carrier data preparation (time
  axis)
16 cc = sin(2 * pi * cf * c); % sinusoidal modulation
17 s=cc;
18
19 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
20 %Amplitude modulation
21
22 % set modulator
23 mf = 4; % modulator frequency (Hz)
24 mi = 1; % modulator index (on a scale of
  1)
25 m = (1:n) / Fs; % modulator data preparation
26 m = 1 + mi * sin(2 * pi * mf * m); % sinusoidal modulation
27
28 % amplitude modulation
29 s = m .* cc; % amplitude modulation
30

```

```
31 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
32 %Frequency modulation
33
34 % set modulator when modulator index is known
35 mf = 4; % modulator frequency (Hz)
36 mi = 200; % modulator index
37 % mi*mf gives peak frequency
   deviation
38 m = (1:n) / Fs; % modulator data preparation
39 m = mi * cos(2 * pi * mf * m); % sinusoidal modulation
40
41 % set modulator when frequency deviation is known
42 mf = 4; % modulator frequency (Hz)
43 fdev = 600; % frequency deviation
44 % mi*mf gives peak frequency
   deviation
45 m = (1:n) / Fs; % modulator data preparation
46 m = fdev/mf * cos(2 * pi * mf * m); % sinusoidal modulation
47
48 % frequency modulation
49 s = sin(c + m); % frequency modulation
50
51 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
52 % sound presentation
53 sound(s, Fs); % plays generated sound
```

Ruis

```

1 % GenerateNoise.m
2 % Algorithm to create filtered noise
3
4 % Generate white noise
5 d=10; % signal length
6 Fs=44100; % sample frequency
7 white=wgn(1,d*Fs,1,'linear','real'); % create noise
8 s=white;
9
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11 % Band-Pass Filter: Finite Impulse Response Filter, Hamming window
12
13 Fl=920; % lower border frequency of filter
14 Fu=1080; % upper border frequency of filter
15 % must be lower than nyquist frequency!!
16
17 n=10000; % filter order
18 ny=Fs/2; % nyquist frequency
19
20 freq(1)=0;
21 m(1)=0;
22 for i=1:Fl-1 % Stop-Band
23     freq(i+1)=i;
24     m(i+1)=0;
25 end
26 for i=Fl:Fu % Pass-Band
27     freq(i+1)=i;
28     m(i+1)=1;
29 end
30 for i=Fu+1:ny % Stop-Band
31     freq(i+1)=i;
32     m(i+1)=0;
33 end
34 f=freq/ny;
35
36 b=fir2(n,f,m); % filter
37 filt=fftfilt(b,white); % here the white noise is filtered.
38 % Off course this could be any signal.
39 s=filt;
40
41 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Bijlage B: Handleiding bij het SQEP

Inleiding

Het **Sound Quality Evaluation Program** (SQEP) is een computerprogramma dat enkel in Matlab uitvoerbaar is. Het is opgemaakt in Matlab 7 en de compatibiliteit met vroegere en nog uit te brengen versies kan niet gegarandeerd worden. Naast de werking van het SQEP wordt ook beschreven hoe eigen geluidsmetingen, geschikt voor analyse met het programma, kunnen uitgevoerd worden.

Van de gebruiker van het programma wordt in deze handleiding een basiskennis van de werking van Matlab verondersteld. Enige onderlegging op akoestisch gebied is eveneens vereist.

Inhoud van het programma

Het SQEP zal van een geluidssignaal de psychoakoestische parameterwaarden berekenen, om zo de Sound Quality van het product dat het geluid voortbrengt, objectief in kaart te brengen. Het geluidssignaal kan een WAV-bestand of een Matlab Data Vector zijn. De parameters luidheid, scherpte, fluctuatiesterkte, ruwheid en tonaalheid worden berekend. Hun inhoudelijke betekenis wordt in deze handleiding niet uiteengezet, net als hun wiskundige opbouw.

Captatie van het geluid

Om een geluid op te nemen, wordt de in het lab Industriële Fysica en Toegepaste Mechanica beschikbare Ono Sokki LA 220 decibelmeter gebruikt. Het AC outputkanaal op de decibelmeter wordt met de TRS-connector (*Tip Ring Sleeve*) verbonden met de microfooningang van de geluidskaart (ev. aanleggen via volumeregeling) van de computer. De captatie van het geluidssignaal gebeurt nu via de microfoon van de decibelmeter. De decibelmeter zelf wordt ingesteld op een ongewogen (L_p), equivalente (L_{eq}) geluidsdrukmeting. Ook de duur van de meting moet ingesteld worden. Daarvoor wordt best 10 seconden genomen. Zo zal het effectieve (RMS) geluidsdrukniveau van het 10 seconden durende signaal berekend en weergegeven worden na het verstrijken van de tien seconden. Die waarde is nodig om het signaal te kunnen kalibreren, waarover verder meer.

In Matlab wordt nu volgende code ingegeven:

```
Fs=44100;  
sound=wavrecord(10*Fs,Fs);
```

Waarbij F_s de bemonsteringsfrequentie (*Sample frequency*) is en `sound` de vector waar de amplitudes opgeslagen zullen worden. De namen van variabelen kunnen vrij gekozen worden! Met $10 \cdot F_s$ wordt ervoor gezorgd dat ook Matlab gedurende tien seconden het geluidssignaal zal capteren. Door nu gelijktijdig met het doorvoeren van dit commando op de startknop van de decibelmeter te drukken zullen zij hetzelfde geluid beoordelen.

Het opgenomen signaal kan bekeken worden door het te plotten: `plot(sound)` en kan opnieuw afgespeeld worden met `wavplay(sound, Fs)`. Opslaan van de variabelen kan via het menu `File – Save Workspace As...` Ze worden dan in één bestand gegroepeerd met de extensie `*.MAT`.

Tutorial

In deze tutorial zal de sound quality van een demo geluid bepaald worden. Het betreft een amplitudegemoduleerde smalbandruis. De amplitudemodulatie heeft een modulatiefrequentie van 4 Hz en een modulatiegraad van 1 (amplitude gaat naar nul). De smalbandruis is zo breed als de kritische band die rond 1 kHz gelegen is en bevat dus frequenties van 920 tot 1080 Hz. Deze demo is te vinden in de gelijknamige submap onder SQEP maar valt evengoed zelf te genereren (zie Bijlage A).

Na het opstarten van Matlab moet het SQEP programma –de eerste maal– ingeladen worden. Ga daarvoor naar **File – Set Path...** Klik op **Add Folder...** en selecteer daar de map waar het programma zich bevindt. De map zelf zal normaal gezien SQEP heten. Na bevestiging zal de map zich in het Matlab Search Path bevinden. **Save** en sluit het venster af.

Om vlot te kunnen werken, wordt de Current Directory best gewijzigd naar de map waarin het te analyseren signaal zich bevindt. Hier dus de map ‘demo’.

Als het te analyseren signaal een Matlab datavector is en deze zit nog niet in de Workspace, dan dient deze daar eerst geplaatst te worden. Dit moet hier gebeuren door het `demo.mat` bestand er in te slepen of door `load demo` in het Command Window te typen. De variabelen `noise` en `Fs` worden ingeladen. Eventueel kan de smalbandruis eens afgespeeld worden door het commando `soundsc(noise, Fs)` in te geven.

Om toch meer te weten over bepaalde functies, kan altijd `help naam van de functie` (bv. `help loudness`) in het Command Window ingegeven worden.

1. Inladen van het signaal

Om het programma te starten, dient het commando `sqep` ingegeven te worden. Volgende tekst verschijnt:

```
To load a WAV file, type W.
```

```
To load a workspace vector, type V:
```

```
Als w ingegeven wordt, komt dit erbij:
```

```
Give the name of the file to analyse:
```

```
If it is placed in the Current Directory, its name is sufficient.
```

```
If not, give full Address Path .
```

Als het te onderzoeken bestand dus *song.wav* zou heten en het bevindt zich in de Current Directory, dan volstaat het `song` in te geven.

In deze demo moet echter `v` ingegeven, want er wordt met een getallenrij gewerkt. Als dit gebeurd is, worden de naam van de vector (*noise*) en de sample frequency (*F_s*) gevraagd.

2. Kalibratie van het signaal

Als het signaal nog gekalibreerd dient te worden, moet `C` geantwoord worden op de vraag van het programma. Anders kan gewoon op Enter gedrukt worden. Kalibratie is nodig omdat de amplitudes van een signaal niet in Pascal uitgedrukt zijn. Er moet een referentiewaarde bekend zijn zodat de amplitudes aangepast kunnen worden naar de correcte geluidsdrukken. Deze referentiewaarde wordt gevonden in de *Leq*-waarde die de geluidsmeter berekend heeft. Door ook van het digitale signaal de effectieve waarde te berekenen en deze te vergelijken met *Leq*, kan de juiste factor aan de amplitudes toegevoegd worden.

Op de vraag

```
What is the unweighted RMS Lp-value [dB] of the sound?
```

kan voor deze demo 85 ingegeven worden. Kommagetallen dienen een punt als afscheiding tussen eenheden en tienden te hebben.

3. Geluidsveld

De volgende vraag die het programma stelt is

```
Is the field diffuse (type D) or free (=frontal sound) (type F)?
```

De ingave van het veldtype is nodig om de luidheid van het signaal te kunnen berekenen. De vraag is in welk soort veld het geluid opgenomen is. Een vrij veld bv. de openlucht of een dode kamer, waar het geluid niet weerkaatst wordt en de plaats van de bron duidelijk is aan de richting waaruit haar geluid komt. Een diffuus veld is net het tegenovergestelde; het lijkt er of het geluid van alle kanten komt.

Voor de demo veronderstellen we een vrij veld (F).

4. Modulatieperiode

Omdat het programma niet in staat is dat zelf te doen, moet de modulatiefrequentie manueel ingegeven worden. De modulatiefrequentie is de frequentie waarmee de amplitude of de frequentie van het originele signaal wijzigt. Dit wordt duidelijk op de grafiek die automatisch tevoorschijn komt en hieronder is weergegeven. Omdat de modulatiefrequentie niet rechtstreeks kan afgelezen worden, volstaat het de modulatieperiode in te geven. Dit na de volgende vraag:

```
This graph is visualized because the modulation period has to be estimated manually.
```

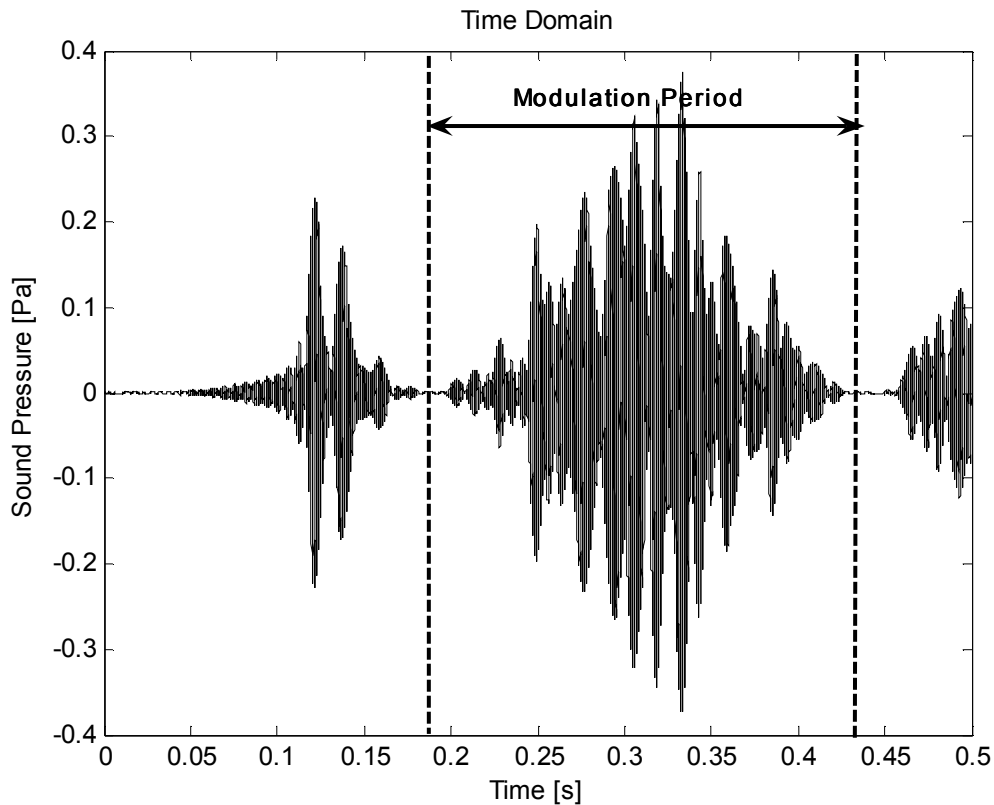
```
This should be done with the data cursor.
```

```
Zooming to improve estimation is possible by clicking on the magnifying glass.
```

```
What is the modulation period? (in seconds)
```

De modulatieperiode is nu het verschil tussen twee punten op de x-as. Deze bedraagt in de demo 0,25 seconden. Deze waarde dient dan ook ingegeven te worden in het Command Window. Let daarbij nogmaals op een punt als afscheiding tussen eenheden en tienden te gebruiken (0.25). Indien de modulatiefrequentie tussen de 0 en 20 Hz ligt, wordt de fluctuatiesterkte berekend. Voor ruwheid is dat tussen 20 en 300 Hz.

Is er geen modulatie in het signaal, geef dan 0 in als modulatieperiode. Automatisch zal het SQEP de berekening van fluctuatiesterkte of ruwheid overslaan.



Vervolgens wordt de volgende vraag gesteld:

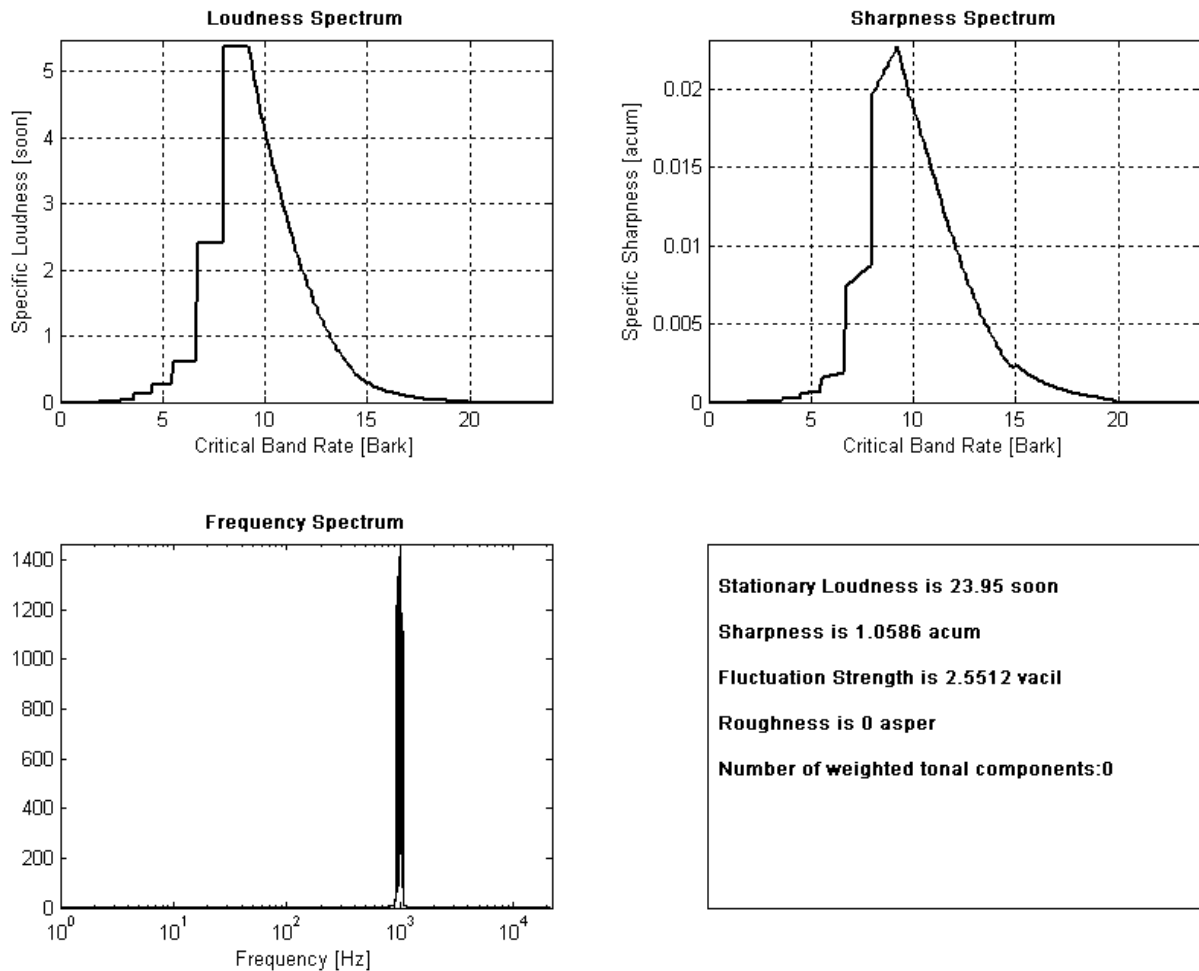
Where to start taking pieces? Default=1 second

Deze vraag gaat over de berekening van de minimale en maximale luidheid, nodig om de fluctuatiersterkte of de ruwheid te vinden. Daarvoor wordt 1 modulatieperiode in stukken gedeeld en berekend. Als die modulatieperiode echter in een bepaald deel van het signaal duidelijker is dan elders, dan kan het misschien lonen het tijdstip waarop de stukken uit een modulatieperiode geknipt worden, te verleggen.

Standaard is het starttijdstip 1 seconde. Aangezien het demosignaal kunstmatig gegenereerd is, doet dat tijdstip er niet toe. Er mag dan ook gewoon op Enter gedrukt worden.

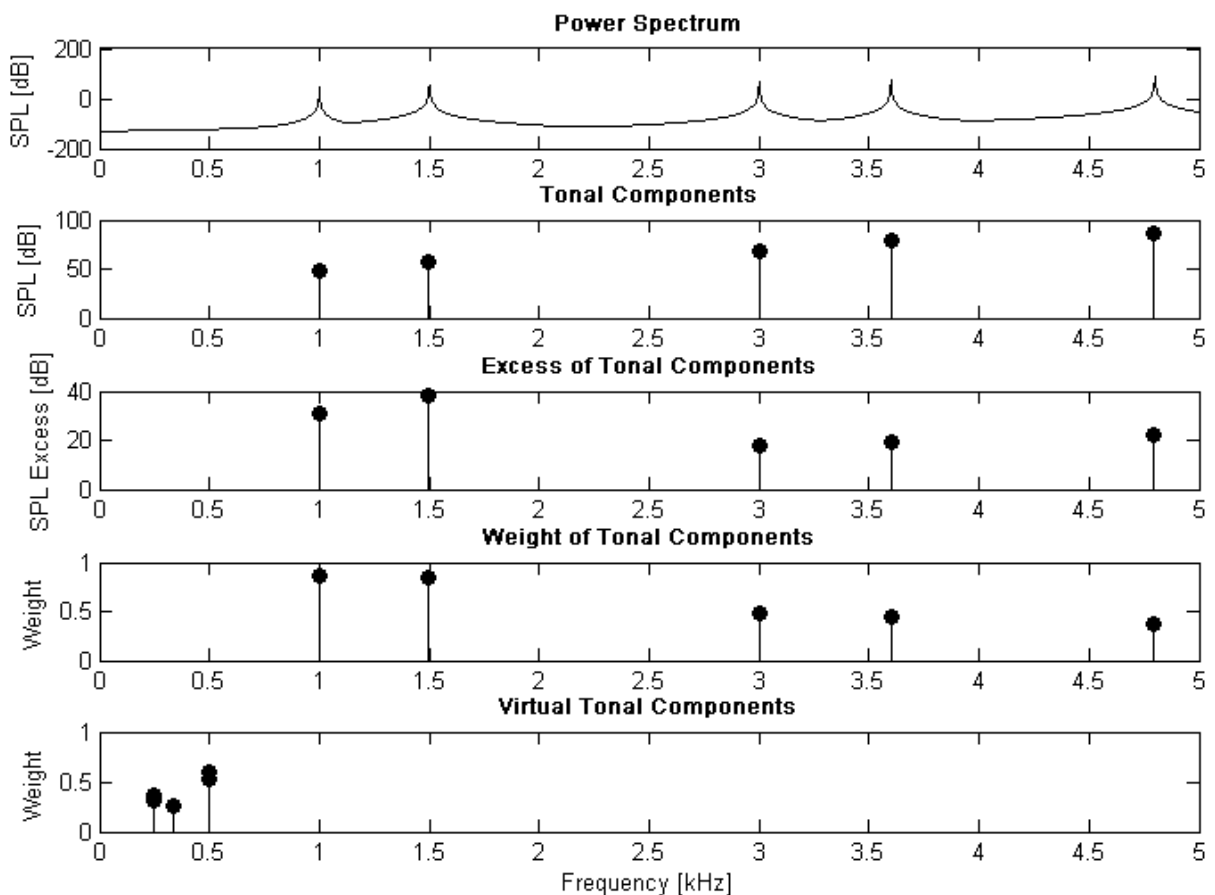
5. Resultaten

Na het nodige rekenwerk zullen twee vensters tevoorschijn komen. Deze visualiseren de resultaten van de berekening. In het geval van de amplitudegemoduleerde smalbandruis ziet de eerste figuur, **Sound Quality Evaluation Program pt. I**, er als volgt uit:



Linksboven is het luidheidsspectrum weergegeven, rechtboven het scherpte spectrum. Hun weergave per kritische band is nuttig omdat zo goed zichtbaar is welke banden/frequenties het luidst en het scherpst zijn. Linksonder is gewoon het frequentiespectrum van het signaal geplot. Dit kan nuttig zijn om te vergelijken met de andere spectra. In het laatste kadertje zijn dan de totale waarden van de parameters te lezen. De ruwheid bedraagt hier nul asper, eigenlijk is ze niet berekend. Wat wel berekend is, zijn de tonale componenten. Dat er geen in het signaal aanwezig zijn mag niet verwonderen want ruis is het tegengestelde van een tonaal geluid. Het heeft dan ook weinig zin nog de tweede figuur met de tonale componenten te bespreken.

Ter vervanging worden de grafieken getoond van een samengesteld signaal. Het is de samenstelling van een 1000, 1500, 3000, 3600 en 4800 Hz signaal. Deze willekeurig gekozen signalen kregen een L_p -waarde van respectievelijk 50, 60, 70, 80 en 90 dB. Deze tonale componenten en hun stijgende geluidsdruk zijn te zien op de eerste twee grafieken; het vermogenspectrum en de tonale componenten. Op de derde grafiek is de uitsprong gevisualiseerd. Dat is mate waarin de tonale component boven zijn omgeving uitsteekt. De vierde grafiek toont dat de ervaring van tonaalheid door het gehoor verschilt naargelang de frequentie van de toon. Zo is de 1 kHz component groter geworden dan de 1,5 kHz component. De vijfde en laatste grafiek toont de virtuele tonen, deze die men dus denkt te horen. Dat zijn tonen met frequentie 250, 333 en 500 Hz. Alle drie zijn het subharmonischen van de 1000, 1500 en 3000 Hz componenten.



6. Opslaan van de resultaten

Terwijl de twee vensters op de voorgrond kwamen, is in het Command Window de volgende tekst verschenen:

```
give a -short- name to this calculations (no spacing).
```

Typ hier dus een niet al te lang woord om de berekeningen onder te bewaren. Als geen woord ingegeven wordt, wordt het woord `test` gebruikt. Als nu `demo` als naam ingegeven wordt, wordt door het programma een *struct* variabele gegenereerd, `demo` genaamd, die alle belangrijke waarden bevat. Op de onderstaande schermafdruk zijn deze weergegeven. Een dergelijke variabele is dan toegankelijk door bv. `demo.Ns` in te geven. Deze manier van werken is nodig om later meerdere resultaten te kunnen vergelijken –bv. in één grafiek plotten– zonder dat eerst alle namen veranderd moeten worden omdat ze identiek zijn.

Name	Value	Class
Fs	44100	double
demo	<1x1 struct>	struct
noise	<1x441000 double>	double

Field	Value
Leq	85
sound_field	1
mod_freq	4
y	<1x441000 double>
Fs	44100
N	23.95
Ns	<1x240 double>
zz	<1x240 double>
S	1.0586
Ss	<1x240 double>
F	2.5512
Fsp	<1x21 double>
R	0
Rs	<1x21 double>
TC	[0 0]
WTC	[0 0]

Vooraleer de gebruiker dit te zien krijgt, zal echter eerst gevraagd worden de variabelen op te slaan als een *.MAT bestand. Op die manier gaat het rekenwerk niet verloren en kunnen de resultaten later nog vlotjes geraadpleegd worden.

Om de grafieken op te slaan, ga in het desbetreffende venster naar File – Save As... en sla ze op als een *.FIG bestand. Dit kan dan later heropend worden in Matlab. Om een afbeelding van de grafieken te verkrijgen, ga naar Edit – Copy Figure en plak deze waar gewenst.

Bijlage C: Matlab codes van het SQEP

sqep.m

```

1 % Sound Quality Evaluation Program
2 % sqep.m
3 %
4 % METHODOLOGY:
5 % Calibrate Signal
6 % Estimate Power Spectrum
7 % Filter
8 % Calculate Loudness according to DIN 45631
9 % Calculate Sharpness
10 % Estimate Modulation frequency
11 % Calculate minimum and maximum specific loudness
12 % Calculate Fluctuation Strength
13 % Calculate Roughness
14 % Calculate Tonal Components
15 % Plot
16 %
17 % VARIABLES:
18 % file           character to find out location of sound
19 % fname         file name of wav file to be loaded
20 % signal        Uncalibrated signal
21 % Fs           Sampling Frequency (Hz)
22 % calib        character to question need for calibration
23 % Lp           measured RMS sound pressure level
24 % y            Time Vector (Pascals), Calibrated
25 % df          Minimum desired resolution PowSpec will
                choose a power of 2
26 %             which gives at least this resolution
27 %             Further in the code, this becomes the true
                resolution
28 % Yxx          Power Spectrum (Pascals^2/Hz)
29 % f            Frequency (Hz)
30 % YdB          Power Spectrum Level
31 % H            Filter Frequency Response Function
32 % Lt          1/3 Octave Band Inputs to the Loudness
                Program (dB)
33 % SF          Sound Field (See DIN45631)
34 % N            Total Loudness [Soon]
35 % Ns          Specific Loudness [Soon/Bark]
36 % S            Total Sharpness [acum]
37 % Ss          Specific Sharpness [acum/Bark]
38 % Tm          Modulation Period
39 % Nsmax       Maximum loudness over one period
40 % Nsmin       Minimum loudness over one period
41 % F            Total Fluctuation Strength [vacil]
42 % Fsp         Specific Fluctuation Strength [vacil/Bark]
43 % R            Total Roughness [asper]
44 % Rs          Specific Roughness [asper/Bark]
45 % ps          Power Spectrum
46 % tc1         Tonal Components
47 % idx1        Index of tonal components
48 % tc2         Excess Level of Tonal Components
49 % idx2        Index of Excess Level of Tonal Components
50 % spec_pitch  Pitch Spectrum
51 % pitch_shift Pitch Shifts

```

```

52 % spec_weight      weight of Tonal Components
53 % vpp_input        spectral pitch pattern
54 % vpp_output       virtual pitch pattern
55 % screen           Variable to set the size of plot windows
56 % zz               Critical Band Rate per tenth
57 % yfft             fft of the signal vector
58 % yfft2            first half of yfft
59 % number           Number of Tonal Components
60 % name             name to save calculated values to
61 % savename         variable to be able to concatenate strings
62
63 % By Ward Jonckheere
64 % Last revision on 07/05/08
65 % based on the code of Aaron Hastings, Herrick Labs, Purdue
    University
66
67 close all
68 clc
69
70 %% Select file to analyse
71 file=input('To load a WAV file,type W.\nTo load a workspace vector,
    type V:\n    ','s');
72 if file=='W'|file=='w'
73     disp('Give the name of the file to analyse:');
74     disp('If it is placed in the Current Directory, its name is
    sufficient.');
```

```

75     fname=input('If not, give full Address Path .\n    ','s');
76     [signal,Fs,nbits]=wavread(fname);
77 elseif file=='V'|file=='v'
78     signal=input('Give the vector's name (from the Workspace).\n
    ');
79     Fs=input('Give the sample frequency (or its Workspace Name).\n
    ');
80 end
81
82 %% Calibrate signal if needed
83 calib=input('Does the signal still has to be calibrated? If so, type
    C:\n    ','s');
84 if calib=='C'|calib=='c'
85     Lp=input('What is the unweighted RMS Lp-value [dB] of the
    sound?\n    ');
86     [y]=CalibrateSignal(signal,Lp);
87 else
88     y=signal;
89     Lp=0;
90 end
91
92 %% Call function to perform PSD
93 df=2;
94 [Yxx,f]=PowSpec(y,Fs,df);
95 df=f(2)-f(1);
96
97 %% Convert to dB
98 [YdB]=Convert2dB(Yxx, 1);
99
100 %% Filter
101 [H]=GenerateFilters(f,Fs);
102
103 %% Scale so that the sum of the energy of all 1/3 OB for a given
    freq. is
104 %% amplified by 1. The sum of all filters for a given frequency
```



```

changes
105 %% depending on which freq you look at. To make sure that the total
energy
106 %% is not amplified to be greater than 1 times the original,
107 %% The average sum was used in calculating the correction.
108 %% e.g.
109 %%     for ink=1:length(H)
110 %%         energy(ink)=sum(abs(H(:,ink)));
111 %%     end
112 %%     AverageEnergyGain = mean(energy)
113 %%     Correction = 1 / AverageEnergyGain
114 %%
115 %% This would give a correction of 0.9639, but 0.9483 works better.
116
117 H=0.94833723551160*H;
118 for ink=1:28
119     Lt(ink)=10*log10(sum((10.^(YdB/10)).*(abs(H(ink,:).^2))));
120 end
121
122 %% Free Field/Diffuse Field
123 SF=input('Is the field diffuse (type D) or free (=frontal sound)
(type F)?\n ','s');
124
125 %% Calculate Loudness
126 [N, Ns]=Loudness(Lt, SF);
127
128 %% Calculate Sharpness
129 [S, Ss]=Sharpness(Ns, N);
130
131 %% Estimate modulation frequency/period
132 [Tm]=Tmod(y, Fs);
133
134 if Tm==0
135     F=0;Fsp(1:21)=0;R=0;Rs(1:21)=0;
136
137 elseif Tm>=0.05
138     %Calculate the max and min specific loudness over one period.
139     [Nsmax,Nsmin]=SpecLoud(y, Fs, SF, Tm);
140     % Calculate Fluctuation Strength
141     [F, Fsp]=FluctuationStrength(Nsmax, Nsmin, Tm);
142     R=0;Rs(1:21)=0;
143 elseif Tm<0.05 & Tm>=0.003333333333
144     %Calculate the max and min specific loudness over one period.
145     [Nsmax,Nsmin]=SpecLoud(y, Fs, SF, Tm);
146     %% Calculate Roughness
147     [R, Rs]=Roughness(Nsmax, Nsmin, Tm);
148     F=0;Fsp(1:21)=0;
149 else
150     F=0;Fsp(1:21)=0;
151     R=0; Rs(1:21)=0;
152 end
153
154 %% Calculate tonalness
155
156 %     Code based on Terhardt's algoritm
157 %     Matlab implementation taken from the master's thesis of Zhang
Z. and Shrestha M. at DTU
158 %     With thanks to mr. J.T. Kristensen from DTU
159 %     With thanks to mr. H. Haslev from Brüel & Kjaer
160

```

```

161     %make power spectrum matrix
162     ps=[f,YdB'];
163
164     %get tonal components
165     [tc1,idx1]=Get_tonal_comp(ps);
166     if isequal(tc1,0)==0
167         %get the SPL excess
168         [tc2,idx2]=Get_spl_excess(ps,tc1,idx1);
169
170         %calculate spectrum pitch, pitch shift, spectrum weight
171         [spec_pitch,pitch_shift]=Get_spectrum_pitch(tc1,idx2);
172         spec_weight=Get_sp_weight(tc2);
173
174         %calculate virtual pitch
175         vpp_input=Get_SP_Pattern(pitch_shift,spec_weight);
176         vpp_output=Get_VP_Pattern(vpp_input);
177     else
178         tc1=[0 0]; tc2=[0 0]; spec_weight=[0 0]; vpp_output=[0 0
179 0];
180     end
181     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
182     %% Plots pt. II: Tonal Components
183
184     screen=[20 50 950 700];%can be changed to screen size.
185
186     figure1=figure('Color',[1 1 1],'Name','Sound Quality Evaluation
187 Program pt.II: Tonal Components',...
188 'NumberTitle','off','Position',screen);
189
190     %show the power spectrum
191     subplot(5,1,1); plot(ps(:,1)/1000,ps(:,2))
192     ylabel('SPL [dB]')
193     xlim([0 5]) %x is limited to 5 kHz because higher tonal components
194     aren't relevant.
195     ylim([0 max(ps(:,2))*1.02])
196     title('Power Spectrum','FontWeight','demi')
197
198     %show the tonal components
199     subplot(5,1,2);stem(tc1(:,1)/1000,tc1(:,2),'fill')
200     ylabel('SPL [dB]')
201     xlim([0 5])
202     title('Tonal Components','FontWeight','demi')
203
204     %show the SPL excess value
205     subplot(5,1,3);stem(tc2(:,1)/1000,tc2(:,2),'fill')
206     ylabel('SPL Excess [dB]')
207     xlim([0 5])
208     title('Excess of Tonal Components','FontWeight','demi')
209
210     %show WS
211     subplot(5,1,4); stem(spec_weight(:,1)/1000,spec_weight(:,2),'fill')
212     ylabel('Weight')
213     xlim([0 5]);ylim([0 1])
214     title('Weight of Tonal Components','FontWeight','demi')
215
216     %show Virtual Pitch weight
217     subplot(5,1,5); stem(vpp_output(:,1)/1000,vpp_output(:,3),'fill');
218     xlabel('Frequency [kHz]')
219     ylabel('Weight')

```

```

218 xlim([0 5])
219 title('Virtual Tonal Components','FontWeight','demi')
220
221 annotation(figure1,'textbox','Position',[0.98 0.002 0.02 0.025],...
222         'LineStyle','none','FitHeightToText','off','FontSize',6,'String',{'W
J'});
223
224 %% Plots pt. I
225
226 figure2 = figure('Color',[1 1 1],'Name','Sound Quality Evaluation
Program pt.I',...
227         'NumberTitle','off','Position',screen);
228 subplot(221)
229 zz=0.1:0.1:24;
230 plot(zz,Ns,'LineWidth',2)
231 axis([0 24 0 max(Ns)*1.02])
232 grid
233 title('Loudness Spectrum','FontWeight','demi')
234 xlabel('Critical Band Rate [Bark]')
235 ylabel('Specific Loudness [soon]')
236
237 subplot(222)
238 plot(zz,Ss,'LineWidth',2)
239 axis([0 24 0 max(Ss)*1.02])
240 grid
241 title('Sharpness Spectrum','FontWeight','demi')
242 xlabel('Critical Band Rate [Bark]')
243 ylabel('Specific Sharpness [acum]')
244
245 subplot(223)
246 yfft=abs(fft(y,Fs));
247 yfft2=yfft(1:Fs/2);
248 semilogx(0:length(yfft2)-1,yfft2)
249 axis([0 Fs/2 0 max(yfft2)*1.02])
250 title('Frequency Spectrum','FontWeight','demi')
251 xlabel('Frequency [Hz]')
252 ylabel('')
253
254 % Because of disfunctionality of Fluctuation Strength and
Roughness
255 % calculation their spectrum has been omitted.
256
257 % if Tm>=0.05
258 %
259 %     subplot(224)
260 %     z=1:24;
261 %     stairs(z,Fsp)
262 %     axis tight
263 %     grid
264 %     title(['Fluctuation Spectrum', 10, 'Total Fluctuation strength
= ', num2str(F),' vacil'])
265 %     xlabel('Critical Band Rate [Bark]')
266 %     ylabel('Specific F.S. [vacil]')
267 % else
268 %
269 %     subplot(224)
270 %     z=1:24;
271 %     stairs(z,Rs)
272 %     axis tight
273 %     grid

```

```

274 %     title(['Roughness Spectrum', 10, 'Total Roughness = ',
num2str(R),' asper'])
275 %     xlabel('Critical Band Rate [Bark]')
276 %     ylabel('Specific roughness [asper]')
277 %
278 %     end
279
280 %% Give calculated values
281     if isequal(spec_weight(1,2),0)==1
282         number=0;
283     else
284         number=numel(spec_weight)/2;
285     end
286
287 annotation(figure2,'textbox','Position',[0.5703 0.11 0.3347
0.3412],...
288     'FitHeightToText','off','FontWeight','demi',...
289     'String',{' ',strcat(['Stationary Loudness is ' num2str(N) '
soon']),',',...
290     strcat(['Sharpness is ' num2str(S) ' acum']),',',...
291     strcat(['Fluctuation Strength is ' num2str(F) '
vacil']),',',...
292     strcat(['Roughness is ' num2str(R) ' asper']),',',...
293     strcat(['Number of weighted tonal components:'
num2str(number)])...
294     });
295
296 annotation(figure2,'textbox','Position',[0.98 0.002 0.02 0.025],...
297
'LineStyle','none','FitHeightToText','off','FontSize',6,'String',{'W
J'});
298
299 %% Save variables into one struct.
300
301 name=input('give a -short- name to this calculations (no
spacing).\n' , 's');
302
303 if isempty(name)==1
304     name='test';
305 end
306     savename= strcat(name, '.');
307
308     eval([savename 'Leq=Lp']);
309     eval([savename 'sound_field=SF']);
310     eval([savename 'mod_freq=1/Tm']);
311     eval([savename 'y=y']);
312     eval([savename 'Fs=Fs']);
313     eval([savename 'N=N']);
314     eval([savename 'Ns=Ns']);
315     eval([savename 'zz=zz']);
316     eval([savename 'S=S']);
317     eval([savename 'Ss=Ss']);
318     eval([savename 'F=F']);
319     eval([savename 'Fsp=Fsp']);
320     eval([savename 'R=R']);
321     eval([savename 'Rs=Rs']);
322 %     eval([savename 'z=z']);
323     eval([savename 'TC=tc1']);
324     eval([savename 'WTC=spec_weight']);
325
326 clear H Lp Lt YdB Yxx df f fname Fs ink calib signal file y SF N Ns

```

```

F Fsp...
327     R Rs S Ss z savename name zz Tm screen Nsmx Nsmin tc1 tc2 idx1
    idx2...
328     spec_weight pitch_shift spec_pitch vpp_input vpp_output ps yfft
    yfft2...
329     number ans figure1 figure2 nbits
330 clc
331
332 uisave

```

CalibrateSignal.m

```

1 % Calibration of Signal
2 %   [y]=CalibrateSignal(signal,Lp)
3 %
4 %   INPUT
5 %   signal           Time Vector, original
6 %   Lp               SPL, read on dB-meter
7 %
8 %   INTERNAL VARIABLES
9 %   Pref             Reference Pressure
10 %   RMS              RMS of Time Vector
11 %   Ly               SPL calculated
12 %   c                Calibration Coefficient
13 %
14 %   OUTPUT
15 %   y                calibrated time vector
16
17 %   by WJ
18 %   Last revision on 31/03/08
19
20 function [y]=CalibrateSignal(signal,Lp)
21
22 Pref = 20e-6;           %% ref. pressure is 20 µPa
23 RMS=sqrt(mean(signal.^2)); %% RMS value of the signal
24 Ly=20*log10(RMS/Pref);  %% dB-value of its RMS
25 c=10^((Lp-Ly)/20);     %% calculation of the calibration
    coefficient
26 y=c*signal;           %% adaptation o

```

PowSpec.m

```

1 % PowSpec.m
2 %
3 %   Syntax:
4 %   [Yxx,f]=PowSpec(y,Fs,df)
5 %
6 %   Variables:
7 %   INPUT
8 %   y           =   Time signal
9 %   Fs          =   Sampling frequency
10 %   df         =   DESIRED df
11 %
12 %   INTERNAL
13 %   NFFT       =   Nummer of points for a given fft
14 %   NOVERLAP   =   Number of Points to overlap
15 %
16 %   OUTPUT
17 %   Yxx        =   Power Spectrum Amplitude
18 %   f          =   Frequency vector of power spectrum
19 %
20 %   Author: Aaron Hastings, Herrick Labs, Purdue University
21 %   Date Started:  11 Nov 00
22 %   Last Revision: 11 Nov 00
23 %   Status: No Known Bugs
24 %
25 function [Yxx,f]=PowSpec(y,Fs,df);
26 %
27 %% Want to make sure that we have at least df resolution
28 NFFT=ceil(Fs/df);
29 %% Set NFFT to power of 2 again err towards higher resolution
30 NFFT=2^ceil(log2(NFFT));
31 %
32 NOVERLAP=0;
33 [Yxx,f] = psd(y,NFFT,Fs,NFFT,NOVERLAP);
34 Yxx=2*Yxx/NFFT; %% Scale to get the power spectrum correct
35 Yxx=Yxx';
36 %
37 return

```

Convert2dB.m

```

1 % Convert2dB.m
2 %   This program takes the complex fft amplitudes and converts them
   to dB
3 %   It uses the cal variable to produce calibrated levels
4 %
5 %   Syntax:
6 %   [YdB]=Convert2dB(Yxx, cal)
7 %
8 %   Variables:
9 %   INPUT
10 %   Yxx =   PSD of y
11 %   cal =   A calibration factor which multiplies each element of
   the wav file
12 %           to get the correct SPL. This will generally be set to

```

```
1 since
13 %           calibration is normally done elsewhere
14 %
15 %   OUTPUT
16 %   YdB   =       Amplitude spectrum in dB
17
18 %   Author: Aaron Hastings, Herrick Labs, Purdue University
19 %   Date Started:   8 July 00
20 %   Last Revision: 11 Nov 00
21 %   Status: No Known Bugs
22
23 function[YdB]=Convert2dB(Yxx, cal)
24
25 %%   Begin function
26
27 ref = 20e-6;
28 YdB=10*log10((cal^2)*Yxx/(ref^2));
```

GenerateFilters.m

```
1 % GenerateFilters.m
2 %
3 %   Methodology:
4 %   This program makes use of the Oct3dsign function written by
5 %   Christophe Couvreur
6 %   In order to get the poles to lie within the unit circle for
7 %   lower frequencies,
8 %   the filters were designed at sampling frequencies lower than
9 %   44100 Hz.
10 %   Therefore, lower frequency 1/3 Octave Bands have their
11 %   spectrums truncated
12 %   according to the fs/2 of the adjusted sampling frequencies.
13 %
14 %   Syntax:
15 %   [H]=GenerateFilters(f,Fs)
16 %
17 %   Variables:
18 %   INPUT
19 %   f           =   Frequency (Hz) of the spectrum to be filtered, used
20 %   to determine df
21 %   Fs          =   Assumed Sampling Frequency (Hz) of Signal
22 %
23 %   WORKING
24 %   df          =   Frequency resolution of spectrum (Hz)
25 %   FiltOrd     =   Order of filter to be designed
26 %   Fc          =   Center frequencies (Hz) for filters to be designed
27 %   q           =   Integer resampling coefficient
28 %   FsNew       =   New Sampling Frequency (Hz)
29 %   B#          =   Filter Polynomial Coefficient
30 %   A#          =   Filter Polynomial Coefficient
31 %   fo         =   Starting Frequency (Hz)
32 %   f2         =   Filter Frequency Vector (Hz)
33 %   H1         =   Temporary storage for Frequency Response of Filter
34 %   H           =   Filter Set
35 %
36 %   OUTPUT
37 %   H           =   Frequency Responses of the Filter Set
38 %
39 %   Author: Aaron Hastings, Herrick Labs, Purdue University
```

```

35 %   Date Started:   31 Oct 00
36 %   Last Revision: 20 March 01 (Added Fs input) +aanpassing WJ
37 %   Status: No Known Bugs
38
39 function[H]=GenerateFilters(f,Fs)
40
41 %%   Begin function
42
43 Fc=[25, 31.5, 40, 50, 63, 80, 100, 125, 160, 200, 250, 315, 400,
44     500, 630, ...
45     800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000, 6300,
46     8000, ...
47     10000, 12500, 16000];
48
49 %%   Filters with fc < 220 will be resampled and then zero padded
50 %%-----
51 %   q=4;
52 %   FsNew=Fs/q;
53 %   [B2,A2] = Oct3dsgn(Fc(1),FsNew);
54 %   df=f(2)-f(1);
55 %   fo=0;
56 %   f2=fo:df:FsWithNew/2;
57 %   H2=freqz(B2,A2,2*pi*f2/FsNew)';
58 %   H2sq=abs(H2).^2;
59 %
60 %   FiltLevel(1)=10*log10(sum((10.^(Yxx(1,1:4411)/10)).*(abs(H2.^2))'))
61 %   loglog(f2,abs(H2.^2))
62 %%-----
63 df=f(2)-f(1);
64 fo=0;
65 FiltOrd=3;
66
67 %%   Filter resampled for fc = 25 Hz.
68 ink=1;
69 q=16;
70 FsNew=Fs/q;
71 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
72 f2=fo:df:FsWithNew/2;
73 H1=freqz(B,A,2*pi*f2/FsNew)';
74 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
75
76 %%   Filter resampled for fc = 31.5 Hz.
77 ink=2;
78 q=8;
79 FsNew=Fs/q;
80 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
81 f2=fo:df:FsWithNew/2;
82 H1=freqz(B,A,2*pi*f2/FsNew)';
83 stop=length(H1);
84 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
85
86 %%   Filter resampled for fc = 40 Hz.
87 ink=3;
88 q=8;
89 FsNew=Fs/q;

```



```
90 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
91 f2=fo:df:FsNew/2;
92 H1=freqz(B,A,2*pi*f2/FsNew)';
93 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
94
95 %% Filter resampled for fc = 50 Hz.
96 ink=4;
97 q=8;
98 FsNew=Fs/q;
99 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
100 f2=fo:df:FsNew/2;
101 H1=freqz(B,A,2*pi*f2/FsNew)';
102 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
103
104 %% Filter resampled for fc = 63 Hz.
105 ink=5;
106 q=4;
107 FsNew=Fs/q;
108 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
109 f2=fo:df:FsNew/2;
110 H1=freqz(B,A,2*pi*f2/FsNew)';
111 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
112
113 %% Filter resampled for fc = 80 Hz.
114 ink=6;
115 q=4;
116 FsNew=Fs/q;
117 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
118 f2=fo:df:FsNew/2;
119 H1=freqz(B,A,2*pi*f2/FsNew)';
120 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
121
122 %% Filter resampled for fc = 100 Hz.
123 ink=7;
124 q=4;
125 FsNew=Fs/q;
126 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
127 f2=fo:df:FsNew/2;
128 H1=freqz(B,A,2*pi*f2/FsNew)';
129 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
130
131 %% Filter resampled for fc = 125 Hz.
132 ink=8;
133 q=2;
134 FsNew=Fs/q;
135 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
136 f2=fo:df:FsNew/2;
137 H1=freqz(B,A,2*pi*f2/FsNew)';
138 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
139
140 %% Filter resampled for fc = 160 Hz.
141 ink=9;
142 q=2;
143 FsNew=Fs/q;
144 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
145 f2=fo:df:FsNew/2;
146 H1=freqz(B,A,2*pi*f2/FsNew)';
147 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
148
149 %% Filter resampled for fc = 200 Hz.
150 ink=10;
```

```

151 q=2;
152 FsNew=Fs/q;
153 [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
154 f2=fo:df:FsNew/2;
155 H1=freqz(B,A,2*pi*f2/FsNew)';
156 H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
157
158 if Fs>=44100; %% minor change by WJ: Fs greater than 44100 Hz also
made possible.
159   for ink=11:29
160     q=1;
161     FsNew=Fs/q;
162     [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
163     f2=fo:df:FsNew/2;
164     H1=freqz(B,A,2*pi*f2/FsNew)';
165     H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
166   end
167
168   elseif Fs<44100; %% change by WJ: warning display when using low
Fs.
169     disp('Warning: You are using a low sample frequency. Sounds
should not have content above half the sampling frequency.')
170     for ink=11:26
171       q=1;
172       FsNew=Fs/q;
173       [B,A] = Oct3dsgn(Fc(ink),FsNew,FiltOrd);
174       f2=fo:df:FsNew/2;
175       H1=freqz(B,A,2*pi*f2/FsNew)';
176       H(ink,:)= [H1' (10e-37)*ones(1,length(f)-length(H1))];
177     end
178     H(ink+1,:)= [(10e-37)*ones(1,length(f))];
179     H(ink+2,:)= [(10e-37)*ones(1,length(f))];
180     H(ink+3,:)= [(10e-37)*ones(1,length(f))];
181   end

```

oct3dsgn.m

```

1 function [B,A] = oct3dsgn(Fc,Fs,N);
2 % OCT3DSGN Design of a one-third-octave filter.
3 % [B,A] = OCT3DSGN(Fc,Fs,N) designs a digital 1/3-octave filter
with
4 % center frequency Fc for sampling frequency Fs.
5 % The filter is designed according to the Order-N specification
6 % of the ANSI S1.1-1986 standard. Default value for N is 3.
7 % Warning: for meaningful design results, center frequency used
8 % should preferably be in range  $Fs/200 < Fc < Fs/5$ .
9 % Usage of the filter: Y = FILTER(B,A,X).
10 %
11 % Requires the Signal Processing Toolbox.
12 %
13 % See also OCT3SPEC, OCTDSGN, OCTSPEC.
14
15 % Author: Christophe Couvreur, Faculte Polytechnique de Mons
(Belgium)
16 % couvreur@thor.fpms.ac.be
17 % Last modification: Aug. 25, 1997, 2:00pm.
18
19 % References:

```

```

20 % [1] ANSI S1.1-1986 (ASA 65-1986): Specifications for
21 % Octave-Band and Fractional-Octave-Band Analog and
22 % Digital Filters, 1993.
23
24 if (nargin > 3) | (nargin < 2)
25     error('Invalid number of arguments.');
```

Loudness.m

```

1 % Program to calculate loudness based on DIN 45631
2 % Based on BASIC Program Published in J. Acoust. Soc. Jpn (E) 12,
3 % 1 (1991)
4 % by E. Zwicker, H. Fastl, U. Widmann, K. Kurakata, S. Kuwano,
5 % and S. Namba
6 % "Re-Author": Aaron Hastings, Herrick Labs, Purdue University
7 % Date Started: 29 October 00
8 % Last Modified: 29 Nov 01
9 % Licht aangepast door WJ
10 % Status: Program Correctly Calculates Loudness for a 70 dB 1000
11 % Hz sine
12 % filtered using 1/3 octave band filters
13 % [N, Ns]=loudness(Lt, MS)
14 %
15 % This is a loudness function which:
16 % Calculates loudness based on DIN 45631 / ISO 532 B
17 % (Zwicker)
18 % Accepts 1/3 octave band levels (SPL Linear Weighting)
19 % * This data must be calibrated using a separate calibration
20 % function
21 % Input Variables
22 % Lt(28) Field of 28 elements which represent the 1/3 OB
23 % levels in dB with
24 % fc = 25 Hz to 12.5 kHz
25 % SF String variable to distinguish the type of
```

Bijlagen

```
    sound field ( free / diffues )
24 %
25 %   Output Variables
26 %   N           Loudness in sone G
27 %   NS          Specific Loudness
28 %
29 %   Working Variables
30 %   FR(28)      Center frequencies of 1/3 OB
31 %   RAP(8)      Ranges of 1/3 OB levels for correction at low
frequencies according
32 %               to equal loudness contours
33 %   DLL(11,8)  Reduction of 1/3 OB levels at low frequencies
according to equal
34 %               loudness contours within the 8 ranges
defined by RAP
35 %   LHS(20)     Critical Band Rate level at absolute threshold
without taking into
36 %               account the transmission characteristics of
the ear
37 %   AO(20)     Correction of levels according to the
transmission characteristics
38 %               of the ear
39 %   DDF(20)    Level difference between free and diffuse sound
fields
40 %   DCB(20)    Adaptation of 1/3 OB levels to the
corresponding critical band level
41 %   ZUP(21)    Upper limits of approximated critical bands in
terms of critical
42 %               band rate
43 %   RNS(18)    Range of specific loudness for the
determination of the steepness of
44 %               the upper slopes in the specific loudness -
critical band rate pattern
45 %   USL(18,8)  Steepness of the upper slopes in the specific
loudness - critical
46 %               band rate pattern for the ranges RNS as a
function of the number of
47 %               the critical band
48 %
49 %   Working Variables (Uncertain of Definitions)
50 %   XP         Equal Loudness Contours
51 %   TI         Intensity of Lt
52 %   LCB        Lower Critical Band
53 %   LE         Level Excitation
54 %   NM         Critical Band Level
55 %   KORRY      Correction Factor
56 %   N          Loudness (in sones)
57 %   DZ         Separation in CBR
58 %   N2         Main Loudness
59 %   Z1         Critical Band Rate for Lower Limit
60 %   N1         Loudness of previous band
61 %   IZ         Center "Frequency" Counter, used with NS
62 %   Z          Critical band rate
63 %   J,IG       Counters used with USL
64
65 function [N,Ns]=Loudness(Lt,SF)
66
67 %% Begin initializing the working variables
68
69 FR=[25 31.5 40 50 63 80 100 125 160 200 250 315 400 500 630 800 1.0
1.25 1.6 ...
```

Bijlagen

```
70         2.0 2.5 3.15 4.0 5.0 6.3 8.0 10.0 12.5];
71
72 RAP=[45 55 65 71 80 90 100 120];
73
74 DLL=[-32 -24 -16 -10 -5 0 -7 -3 0 -2 0
75        -29 -22 -15 -10 -4 0 -7 -2 0 -2 0
76        -27 -19 -14 -9  -4 0 -6 -2 0 -2 0
77        -25 -17 -12 -9  -3 0 -5 -2 0 -2 0
78        -23 -16 -11 -7  -3 0 -4 -1 0 -1 0
79        -20 -14 -10 -6  -3 0 -4 -1 0 -1 0
80        -18 -12 -9  -6  -2 0 -3 -1 0 -1 0
81        -15 -10 -8  -4  -2 0 -3 -1 0 -1 0]'; %% BASIC code does this a
      oddly, hence the transpose
82
83 LHS=[30 18 12 8 7 6 5 4 3 3 3 3 3 3 3 3 3 3 3 3];%%hearing treshold
      level
84
85 AO=[0 0 0 0 0 0 0 0 0 0 -0.5 -1.6 -3.2 -5.4 -5.6 -4.0 -1.5 2.0 5.0
      12.0];
86
87 DDF=[0 0 0.5 0.9 1.2 1.6 2.3 2.8 3.0 2.0 0.0 -1.4 -2.0 -1.9 -1.0
      0.5 ...
88        3.0 4.0 4.3 4.0];
89
90 DCB=[-0.25 -0.6 -0.8 -0.8 -0.5 0.0 0.5 1.1 1.5 1.7 1.8 1.8 1.7 1.6
      1.4 ...
91        1.2 0.8 0.5 0.0 -0.5];
92
93 ZUP=[0.9 1.8 2.8 3.5 4.4 5.4 6.6 7.9 9.2 10.6 12.3 13.8 15.2 16.7
      18.1 ...
94        19.3 20.6 21.8 22.7 23.6 24.0];
95
96 RNS=[21.5 18.0 15.1 11.5 9.0 6.1 4.4 3.1 2.13 1.36 0.82 0.42 0.30
      0.22 ...
97        0.15 0.10 0.035 0.0];
98
99 USL=[13.00 8.20 6.30 5.50 5.50 5.50 5.50 5.50
100        9.00 7.50 6.00 5.10 4.50 4.50 4.50 4.50
101        7.80 6.70 5.60 4.90 4.40 3.90 3.90 3.90
102        6.20 5.40 4.60 4.00 3.50 3.20 3.20 3.20
103        4.50 3.80 3.60 3.20 2.90 2.70 2.70 2.70
104        3.70 3.00 2.80 2.35 2.20 2.20 2.20 2.20
105        2.90 2.30 2.10 1.90 1.80 1.70 1.70 1.70
106        2.40 1.70 1.50 1.35 1.30 1.30 1.30 1.30
107        1.95 1.45 1.30 1.15 1.10 1.10 1.10 1.10
108        1.50 1.20 0.94 0.86 0.82 0.82 0.82 0.82
109        0.72 0.67 0.64 0.63 0.62 0.62 0.62 0.62
110        0.59 0.53 0.51 0.50 0.42 0.42 0.42 0.42
111        0.40 0.33 0.26 0.24 0.22 0.22 0.22 0.22
112        0.27 0.21 0.20 0.18 0.17 0.17 0.17 0.17
113        0.16 0.15 0.14 0.12 0.11 0.11 0.11 0.11
114        0.12 0.11 0.10 0.08 0.08 0.08 0.08 0.08
115        0.09 0.08 0.07 0.06 0.06 0.06 0.06 0.05
116        0.06 0.05 0.03 0.02 0.02 0.02 0.02 0.02];
117
118 %% Begin Loudness Calculation
119
120 %% Correction of 1/3 OB levels according to equal loudness
      contours (XP) and
121 %% calculation of the intensities for 1/3 OB's up to 315 Hz
```

```

122
123 for I=1:11
124     J=1;
125     while J<8
126         if Lt(I)<=RAP(J)-DLL(I,J)
127             XP=Lt(I)+DLL(I,J);
128             TI(I)=10^(0.1*XP);
129             J=9;    %% To exit from the while loop
130         else
131             J=J+1;
132         end
133     end
134 end
135
136 %% Determination of Levels LCB(1), LCB(2), and LCB(3) within the
137 %% first three
138 %% critical bands
139 GI(1)=TI(1)+TI(2)+TI(3)+TI(4)+TI(5)+TI(6);
140 GI(2)=TI(7)+TI(8)+TI(9);
141 GI(3)=TI(10)+TI(11);
142
143 for I=1:3
144     if GI(I)>0
145         LCB(I)=10*log10(GI(I));
146         %% Note: The BASIC code uses a divide by "log(10)" to
147         %% gauruntee that
148         %% the log is base 10
149     end
150 end
151 %% Calculation of Main Loudness
152
153 for I=1:20
154     LE(I)=Lt(I+8);
155     if I<=3
156         LE(I)=LCB(I);
157     end
158     LE(I)=LE(I)-AO(I);
159     NM(I)=0;
160     if SF=='d' | SF=='D'
161         LE(I)=LE(I)+DDF(I);
162     end
163     if LE(I)>LHS(I)
164         LE(I)=LE(I)-DCB(I);
165         S=0.25;
166         MP1=0.0635*10^(0.025*LHS(I));
167         MP2=(1-S+S*10^(0.1*(LE(I)-LHS(I))))^0.25-1;
168         NM(I)=MP1*MP2;
169         if NM(I)<=0
170             NM(I)=0;
171         end
172     end
173 end
174
175 NM(21)=0;
176
177 %% Correction of specific loudness in the lowest critical band
178 %% taking
179 %% into account the dependence of absolute threshold within this
180 %% critical band

```

```

179
180 KORRY=0.4+0.32*NМ(1)^0.2;
181 if KORRY>1
182     KORRY=1;
183 end
184 NМ(1)=NМ(1)*KORRY;
185
186 %% Start Values
187
188 N=0;
189 Z1=0;
190 N1=0;
191 IZ=1;
192 Z=0.1;
193 short=0;
194
195 %% Step to first and subsequent critical bands
196
197 for I=1:21
198     ZUP(I)=ZUP(I)+0.0001;
199     IG=I-1;
200     if IG>8
201         IG=8;
202     end
203     while Z1<ZUP(I) %% Note, Z1 will always be < ZUP(I) when line
is first reached for each I
204         if N1>NМ(I)
205             %% Decide whether the critical band in question is
completely or
206             %% partly masked by accessory loudness
207             N2=RNS(J);
208             if N2<NМ(I)
209                 N2=NМ(I);
210             end
211             DZ=(N1-N2)/USL(J,IG);
212             Z2=Z1+DZ;
213             if Z2>ZUP(I)
214                 Z2=ZUP(I);
215                 DZ=Z2-Z1;
216                 N2=N1-DZ*USL(J,IG); % WJ: value is determined by the
shape of the masking curve
217             end
218             %% Contribution of accessory loudness to total loudness
219             N=N+DZ*(N1+N2)/2;
220             while Z<Z2
221                 Ns(IZ)=N1-(Z-Z1)*USL(J,IG);
222                 IZ=IZ+1;
223                 Z=Z+0.1;
224             end
225
226         elseif N1==NМ(I)
227             %% Contribution of unmasked main loudness to total loudness
and calculation
228             %% of values NS(IZ) with a spacing of Z=IZ*0.1 Bark
229             Z2=ZUP(I);
230             N2=NМ(I);
231             N=N+N2*(Z2-Z1);
232             while Z<Z2
233                 Ns(IZ)=N2;
234                 IZ=IZ+1;
235                 Z=Z+0.1;

```

```
236         end
237     else
238         %% Determination of the number J corresponding to the
range of specific
239         %% loudness
240         for J=1:18
241             if RNS(J)<NM(I)
242                 break
243             end
244         end
245         %% Contribution of unmasked main loudness to total loudness
and calculation
246         %% of values NS(IZ) with a spacing of Z=IZ*0.1 Bark
247         Z2=ZUP(I);
248         N2=NM(I);
249         N=N+N2*(Z2-Z1);
250         while Z<Z2
251             Ns(IZ)=N2;
252             IZ=IZ+1;
253             Z=Z+0.1;
254         end
255     end
256     %% Step to next segment
257     while J<18
258         if N2<=RNS(J)
259             J=J+1;
260         else
261             break
262         end
263     end
264     if N2<=RNS(J) & J>=18
265         J=18;
266     end
267     Z1=Z2;
268     N1=N2;
269 end
270 end
271 %% Now apply some sort of correction
272
273 if N<0
274     N=0;
275 elseif N<=16
276     N=floor(N*1000+0.5)/1000;
277 else
278     N=floor(N*100+0.5)/100;
279 end
```


Sharpness.m

```
1 % Calculation of Sharpness
2 % The used formula is Zwicker & Fastl's one.
3 %
4 %INPUT
5 %   Ns    specific loudness
6 %   N     total loudness
7 %
8 %INTERNAL VARIABLES
9 %   g(z)  weighting function
10 %   z     Critical band rate per tenth.
11 %
12 %OUTPUT
13 %   Ss    specific sharpness [acum/Bark]
14 %   S     total sharpness [acum]
15
16 % by WJ
17 % last revision on 29/03/08
18
19 function [S,Ss]= Sharpness(Ns,N)
20
21 %define vector z
22 z=(1:length(Ns));
23
24 %define values of weighting factor g(z)
25 for k=1:max(z)
26     g(k)=0.2*exp(1)^(0.308*(k/10-15))+1;
27 end
28     %under 15 Bark: g(z)=1
29     g(1:149)=1;
30
31 % calculation of specific sharpness
32 Ss=0.11*((Ns.*g)/10).*(z./10)/N;
33
34 % calculation of total sharpness
35 S=sum(Ss);
```

Tmod.m

```
1 % Visual estimation of the modulation period
2 % (inverse of the modulation frequency)
3 % display of signal in time domain.
4 % Needed for calculation of fluctuation strength and roughness.
5 %
6 %INPUT
7 %   y      signal to analyse
8 %   Fs     sample frequency
9 %
10 %INTERNAL VARIABLES
11 %   dt     sample resolution
12 %   t     vector containing sampling times
13 %
14 %OUTPUT
15 %   Tm     modulation period
16
17 % by WJ
18 % last revised on 03/04/08
19
20 function [Tm]=Tmod(y,Fs)
21
22     disp('This graph is visualized because the modulation period has
23 to be estimated manually.');
```

```
23     disp('This should be done with the data cursor.');
```

```
24     disp('Zooming to improve estimation is possible by clicking on the
25 magnifying glass.');
```

```
26 dt=1/Fs;
27 t=dt:dt:length(y)/Fs;
28
29 % plot
30     figure('Name','Estimation of Modulation Period','color',[1 1
31 1], 'NumberTitle','off')
32     plot(t(1:Fs/2),y(1:Fs/2))
33     grid
34     zoom xon
35     title('Time Domain')
36     xlabel('Time [s]')
37     ylabel('Sound Pressure [Pa]')
38
39 %ask for user's estimation
40 Tm=input('What is the modulation period? (in seconds)\n');
```

```
41 close all
```

SpecLoud.m

```

1 % Function to calculate max and min specific loudness of 8 equal-
  length
2 % pieces of one modulation period.
3 %
4 % INPUT
5 %   x      signal to be analysed
6 %   Fs     sample frequency
7 %   SF     Sound Field
8 %   Tm     modulation period
9 %
10 % INTERNAL VARIABLES
11 %   start   point of time to start fragmentation on
12 %   ZUP     Upper limits of approximated critical bands in terms
  of critical band rate
13 %   y      amplified signal
14 %   sample  sample corresponding with point of time
15 %   frag    fragment of signal, to be analysed
16 %   df     frequency resolution
17 %   YdB    Sound Pressure Level
18 %   H      Filter Frequency Response Function
19 %   Lt     1/3 Octave Band Inputs to the Loudness Program (dB)
20 %   N      Total Loudness (Sones)
21 %   Nsmean  mean value of specific loudness within one band.
22 %
23 % OUTPUT
24 %   Nsmax   maximum specific loudness of all fragments
25 %   Nsmin   minimum specific loudness of all fragments
26
27 %   by WJ
28 %   last revision on: 30/04/08
29
30 function [Nsmax,Nsmin]=SpecLoud(x,Fs,SF,Tm);
31
32 %ask where to start taking pieces
33 start=input('Where to start taking pieces? Default=1 second\n');
34 if isempty(start)
35     start=1;
36 end
37
38 % Upper limits of approximated critical bands in terms of critical
  band rate
39 ZUP=[0.9 1.8 2.8 3.5 4.4 5.4 6.6 7.9 9.2 10.6 12.3 13.8 15.2 16.7
  18.1 ...
40     19.3 20.6 21.8 22.7 23.6 24.0];
41
42 % amplification of x
43 y=10*x;
44
45 % four sections loop
46 for i=1:4
47     %define start values
48     start(i+1)=start(i)+Tm/4;
49     %change seconds to equal sample number
50     sample(i)=Fs*start(i);
51     % separate the four sections of y
52     frag =y(round(sample(i)):round(sample(i)+Tm/4*Fs-1));
53

```

```

54 %      %trial to allow roughness calculation
55 %      len=length(frag);
56 %      for l=1:10
57 %          frag2(len*(l-1)+1:len*l)=frag;
58 %      end
59 %      frag=frag2;
60
61 %      %plot four fragments
62 %      subplot(1,4,i)
63 %      plot(frag2)
64 %      axis([0 length(frag) -1 1])
65
66 %      % define power spectrum
67 %      df=2;
68 %      [Yxx,f]=PowSpec(frag,Fs,df);
69 %      df=f(2)-f(1);
70
71 %      % Convert to dB
72 %      [YdB]=Convert2dB(Yxx, 1);
73
74 %      % Filter
75 %      [H]=GenerateFilters(f,Fs);
76
77 %      H=0.94833723551160*H;
78 %      for ink=1:28
79 %          Lt(ink)=10*log10(sum((10.^(YdB/10)).*(abs(H(ink,:).^2))));
80 %      end
81 %      % Calculate Loudness
82 %      [N, Ns]=LoudnessNoMask(Lt, SF);
83 %      [N, Ns]=Loudness(Lt, SF);
84
85 %      % reduce 240 Ns values to 21 and put all eight fragments in one
86 %      matrix.
87 %      for j=1:21
88 %          if j==1
89 %              Nsmean(i,j)= mean(Ns(1:ZUP(j)*10));
90 %          else
91 %              Nsmean(i,j)= mean(Ns(ZUP(j-1)*10+1:ZUP(j)*10));
92 %          end
93 %      end
94
95 %      %Only the highest and the lowest of the eight min & max specific
96 %      loudness have to be kept.
97 %      Nsmax=max(Nsmean());
98 %      Nsmin=min(Nsmean());

```

FluctuationStrength.m

```

1 %Calculation of Fluctuation Strength
2 %
3 %INPUT
4 % Nsmax      maximal specific loudness values
5 % Nsmin      minimal specific loudness values
6 % Tm         modulation period (= inversion of modulation
7 %            frequency)

```

```

 8  %INTERNAL VARIABLES
 9  %   dL      masking depth of 1 band
10  %   DL      integrated masking depth of 1 band
11  %
12  %OUTPUT
13  %   Fsp     specific fluctuation strength
14  %   F       total fluctuation strength
15
16  % by WJ
17  % last revision on 29/04/08
18
19  function [F,Fsp]=FluctuationStrength(Nsmax,Nsmin,Tm)
20
21  % Calculation of masking depth
22  for j=1:21
23
24      % Avoid division by zero error.
25      % Do not count Ns if too small
26      if Nsmax(j)==0 | Nsmin(j)==0 | Nsmin(j)<=max(Nsmin)/10
27          dL(j)=0;
28      else
29          dL(j)=log10(Nsmax(j)/Nsmin(j));
30      end
31
32      %integration: multiply each element with its z-value
33      DL(j)=dL(j)*j;
34  end
35
36  % specific FS.
37  % Fsp=0.008/((Tm/0.25)+(0.25/Tm)).*DL;
38  Fsp=0.26/((Tm/0.25)+(0.25/Tm)).*DL;
39
40  %total FS
41  F=sum(Fsp);

```

Roughness.m

```

1  % Calculation of Roughness
2  %
3  % INPUT
4  %   Nsmax      maximal specific loudness values
5  %   Nsmin      minimal specific loudness values
6  %   Tm         modulation period (= inversion of modulation
   frequency)
7  %
8  %INTERNAL VARIABLES
9  %   dL         masking depth of 1 band
10 %   DL         integrated masking depth of 1 band
11 %
12 %OUTPUT
13 %   Rs         specific roughness
14 %   R         total roughness
15
16 % by WJ
17 % last revision on 29/04/08
18
19 function [R,Rs]=Roughness(Nsmax,Nsmin,Tm)
20
21 % masking depth calculation
22 for j=1:21
23
24     % Avoid division by zero error.
25     % Do not count Ns if too small
26     if Nsmax(j)==0 | Nsmin(j)==0 | Nsmin(j)<=max(Nsmin)/10
27         dL(j)=0;
28     else
29         dL(j)=log10(Nsmax(j)/Nsmin(j));
30     end
31     %integration: multiply each element with its z-value
32     DL(j)=dL(j)*j;
33 end
34
35 %specific roughness
36 %convert modulation period [s] to mod.freq. [kHz].
37 Rs=0.3/(Tm*1000).*DL;
38
39 %total roughness
40 R=sum(Rs);

```

Get_tonal_comp.m

```

1  %Extraction of tonal components.
2  %Input the power spectrum
3  %Output the tonal components.
4
5  function [y,z] = Get_tonal_comp(x)
6
7  %Check whether the input matrix validate.
8  [m,n] = size(x);
9  if ~(n == 2) | (m == 1 & n == 1)

```

```

10     error('Input must be a two column vector!');
11 end
12 z=0;
13 y=0;
14 k = 1;
15 for i = 4:(length(x)-3)
16
17     %Look for local maximal samples.
18     %if x(i,2)>=x(i+1,2) & x(i,2)>x(i-1,2) & x(i,2)-x(i-3,2)>=3 &
x(i,2)-x(i-2,2)>=3 & x(i,2)-x(i+2,2)>=3 & x(i,2)-x(i+3,2)>=3
19     if x(i,2)>=x(i+1,2) & x(i,2)>x(i-1,2) & x(i,2)-x(i-3,2)>=7 &
x(i,2)-x(i-2,2)>=7 & x(i,2)-x(i+2,2)>=7 & x(i,2)-x(i+3,2)>=7
20
21         %Calculate and save the frequency and the SPL values for
tonal components.
22         y(k,1)=x(i,1)+0.46*(x(i+1,2)-x(i-1,2));
23         y(k,2)=x(i,2);
24
25         %Check whether you want to remember the power spectrum
indices of the local maximal samples.
26         if (nargout == 2)
27             z(k,1)=i;
28         end
29         k=k+1;
30     end
31 end

```

Get_spl_excess

```

1 function [y,idx]=Get_spl_excess(ps,x,z)
2 %SPL Excess: The extent to which a tonal component is 'aurally
relevant'.
3
4 %ps: the sample matrix which has the power spectrum(Frequency&SPL) of
a signal
5 %x: the first result matrix from Get_tonal_comp(ps)
6 %z: the second result matrix from Get_tonal_comp(ps)
7 %idx: is used to remember the row index of x whose SPL Excess >0
8
9 y=[0 0];
10 idx=1;
11 k=1;
12 for i = 1:numel(x)/2
13     fu=x(i,1);
14     Lu=x(i,2);
15     j=1;
16     sum=0;
17     while j<=numel(x)/2
18         fv=x(j,1);
19         Lv=x(j,2);
20         if j~=i
21             sum=sum+power(10,Lev(fu,fv,Lv)/20);
22         end
23         j=j+1;
24     end
25
26     %calculate the SPL excess for the uth component,save if larger

```

```

    than 0
27     LXu=Lx-10*log10(sum^2+IN(fu,ps,z)+power(10,LTH(fu)/10));
28     if (LXu>0)
29         y(k,2)=LXu;
30         y(k,1)=fu;
31         if (nargout == 2)
32             idx(k,1)=i;
33         end
34         k=k+1;
35     end
36 end
37
38 %-----
39
40 function y=z(x)
41 %Get the critical-band rate(Bark) of a frequency(Hz)
42 y=13*atan(0.76*x/1000)+3.5*atan((x/7500)^2);
43
44 %-----
45
46 function y=IN(f,x,idx)
47 %Noise Intensity, present in the critical band (z(f)-0.5, z(f)+0.5)
48 %around the considered tonal component.
49 %The sound intensity level and the sound pressure level are
50 %approximately numerically equal.
51 %f: c tonal component's frequency;
52 %x: power spectrum matrix;
53 %idx: [y,idx]=Get_tonal_comp(a), idx has the Index value of local
54 %maximal samples
55
56 m=x;
57 for i=1:length(idx)
58     if idx(i)>=1 & idx(i)<=length(x)
59         m(idx(i),:)= [0,0];
60     end
61     if (idx(i)-1)>=1 & (idx(i)-1)<=length(x)
62         m(idx(i)-1,:)= [0,0];
63     end
64     if (idx(i)-2)>=1 & (idx(i)-2)<=length(x)
65         m(idx(i)-2,:)= [0,0];
66     end
67     if (idx(i)+1)>=1 & (idx(i)+1)<=length(x)
68         m(idx(i)+1,:)= [0,0];
69     end
70     if (idx(i)+2)>=1 & (idx(i)+2)<=length(x)
71         m(idx(i)+2,:)= [0,0];
72     end
73 end
74 i=1;
75 while i<=length(m)
76     if m(i,:)==[0,0]
77         m(i,:)=[];
78     end
79     i=i+1;
80 end
81
82 %need to use z.m to get the critical-band rate(Bark) of a
83 %frequency(Hz)
84 %z.m have defined outside.
85 Zu=z(f);

```



```

83 Zu_min=Zu-0.5;
84 Zu_max=Zu+0.5;
85
86 %sum will count the summary of dB values
87 sum=0;
88 for i=1:length(x)
89     if z(x(i,1))<=Zu_max & z(x(i,1))>=Zu_min
90         %The sum of all the dB values.
91         %Lp (Sound Pressure Level) is approximatly equal to Li
          (Intensity Level)
92         Lp=x(i,2);
93         sum=20*(log10(power(10,Lp/20)+power(10,sum/20)));
94     end
95 end
96
97 y=sum;
98
99 %-----
100
101 function y=LTH(x)
102 %The hearing threshold at frequency x
103
104 y=3.64*(x/1000)^(-0.8)-6.5*exp(-0.6*(x/1000-3.3)^2)+10^-3*(x/1000)^4;

```

Lev.m

```

1 % This function is used by both Get_spl_excess and
  Get_spectrum_pitch function
2
3 function y=Lev(fu,fv,Lv)
4 %The excitation level which produced at frequency fu by the vth
  tonal component
5
6 y=Lv-s(fu,fv,Lv)*(z(fv)-z(fu));
7
8 %-----
9
10 function y=s(fu,fv,Lv)
11 %The steepness of the slopes
12
13 if (fu<=fv)
14     y=27;
15 else
16     y=-24-(0.23*1000/fv)+(0.2*Lv);
17 end
18
19 %-----
20
21 function y=z(x)
22 %Get the critical-band rate(Bark) of a frequency(Hz)
23
24 y=13*atan(0.76*x/1000)+3.5*atan((x/7500)^2);

```

Get_spectrum_pitch.m

```

1 function [y,z] = Get_spectrum_pitch(x,idx2)
2 % [y,z] = Get_spectrum_pitch(x,idx2)
3 % Calculate the spectrum pitch (unit: pu)
4 % x: tonal components, (x,idx1)=Get_tonal_comp(sp_x), [Frequency,SPL]
5 % idx2:                (SPL_excess,idx2)=Get_spl_excess(sp_x,x,idx1)
6 % y: spectrum pitch,   [Frequency,Pitch(pu)]
7 % z: spectrum pitch shift.
8
9 for i=1:length(idx2)
10     u=idx2(i);
11     fu=x(u,1);
12     Lu=x(u,2);
13     %The calculation of Vu is only required for those tonal
components whose SPL excess is greater than 0.
14     %Trace the relevant tonal components using idx2. It contains the
row index of the relevant tonal componets.
15     v=Vu(x,u);
16     y(i,2)=fu*(1+v) ;
17     y(i,1)=fu;
18     if (nargout == 2)
19         z(i,2)= v;
20         z(i,1)= fu ;
21     end
22 end
23
24 %-----
25
26 function y=Vu(x,u)
27 %The induced pitch shifts(a quantity of maximally a few percent)
28 %The calculation of Vu is only required for those tonal components
whose SPL excess is greater than 0.
29
30 fu=x(u,1);
31 Lu=x(u,2);
32 [LXu1,LXu2]=specific_LXu(x,u);
33 y=2*power(10,-4)*(Lu-60)*(fu/1000-2)+1.5*power(10,-2)*exp(-
LXu1/20)*(3-log(fu/1000))+3*power(10,-2)*exp(-
LXu2/20)*(0.36+log(fu/1000));
34
35 %-----
36
37 function [LXu1,LXu2]=specific_LXu(x,u)
38 %Specific representations of the SPL excess(for the uth component).
39 %Pertaining to the interference of the considered components
40 %with the lower(LXu1) and the higher (LXu2) ones.
41 %when using this function, u should be the relevant tonal component
whose SPL excess > 0
42 %x:tonal components, (x,idx1)=Get_tonal_comp(sp_x), [Frequency,SPL]
43
44 fu=x(u,1);
45 Lu=x(u,2);
46
47 j=1;
48 sum1=0;% use to calculate LXu1
49 sum2=0;% use to calculate LXu2
50 while j<=numel(x)/2
51     fv=x(j,1);

```

```

52     Lv=x(j,2);
53     if j<u
54         lev1=Lev(fu,fv,Lv);
55         sum1=sum1+power(10,lev1/20);
56     elseif j>u
57         lev2=Lev(fu,fv,Lv);
58         sum2=sum2+power(10,lev2/20);
59     end
60     j=j+1;
61 end
62
63 if sum1==0
64     LXu1=Lu;
65 else
66     LXu1=Lu-20*log10(sum1);
67 end
68 if sum2==0
69     LXu2=Lu;
70 else
71     LXu2=Lu-20*log10(sum2);
72 end

```

Get_sp_weight.m

```

1 function y = Get_sp_weight(x)
2 %Spectrum pitch weight
3 %The Extend to which each tonal component contributes to the entire
4 %tonal percept is considered to depend on
5 % (1). the SPL excess and (2). the frequency of the component.
6 %y = Get_sp_weight(x)
7 %x: the SPL excess matrix, we get it through:
8 [x,idx2]=Get_spl_excess(spectrum_sample,tonal_component,idx1)
9 %y: spectrum pitch weight
10
11 %can not use length(x) here, because
12 %length([1,2])=2
13 %length([1,2;1,2])=2
14 %so when x is only 1 row 2 column, there will be a problem.
15
16 k= 1;
17 [m,n]=size(x);
18 for i=1:m
19     fu=x(i,1);
20     LXu=x(i,2);
21     if LXu >= 0
22         y(k,2) = (1-exp(-LXu/15))*(1+ 0.07*(fu/(0.7*1000)-
23         0.7*1000/fu)^2)^(-1/2);
24         y(k,1) = fu;
25         k = k + 1;
26     end
27 end

```

Get_SP_pattern

```

1 function sp = Get_SP_Pattern(A,B)
2
3 %A=[f,v]   B=[f,WS]
4 %They have the same length
5
6 [m,n]=size(A);
7 for i=1:m
8     sp(i,:)= [A(i,:),B(i,2)];
9 end

```

Get_VP_pattern

```

1 function y = Get_VP_Pattern(sp)
2 %Virtual Pitch Pattern
3
4 %sp:spectrum pitch pattern [f,v,WS]
5
6 %sort rows in ascending order
7 sp_sorted0= sortrows(sp,3);
8
9 %sp_pattern : the result of sorting sp in terms of descending WS
10 %change the ascending order into descending order
11 [s1,s2]=size(sp_sorted0);
12 k=s1;
13 for i=1:s1
14     sp_sorted(k,:)=sp_sorted0(i,:);
15     k=k-1;
16 end
17
18 %get Imax
19 [s1,s2]=size(sp_sorted);
20 Imax=1;
21 WSmag=sp_sorted(1,3);
22 if s1>=2
23     WSk=sp_sorted(Imax+1,3);
24     while (WSk>=0.7*WSmag) & (Imax+1<=length(sp_sorted))
25         Imax=Imax+1;
26         if Imax < s1
27             WSk=sp_sorted(Imax+1,3);
28         else
29             break
30         end
31     end
32 end
33 end
34
35 %generate the result for virtual pitch pattern [f,H,W]
36 [s1,s2]=size(sp);
37 t=1;
38 for i=1:Imax
39     fi=sp_sorted(i,1);
40     for m=1:12
41         W=0;

```

```

42         j=1;
43         while (j<=s1)
44             if (j~=i)
45                 Cij = C(i,j,m,sp_sorted);
46                 W = W + Cij;
47             end
48             j=j+1;
49         end
50         sub_fi=fi/m;
51         Him=H(i,m,sp_sorted);
52         B=Beta(i,m,sp_sorted);
53         Wim=B*W;
54         %save result in the output matrix
55         y(t,:)=[sub_fi,Him,Wim];
56         t=t+1;
57     end
58 end
59
60 %sort the result in terms of increasing virtual pitch weight
61 y=sortrows(y,3);
62
63 % get the last five ones which has maximal weight
64 k=1;
65 for i=length(y)-4:length(y)
66     temp(k,:)=y(i,:);
67     k=k+1;
68 end
69 y=temp;
70
71 %sort the result in terms of increasing subharmonic frequency.
72 y=sortrows(y,1);
73
74 %-----
75 function y = Beta(i,m,sp_sorted)
76 %It is a factor, called the 'fundamental frequency weight'
77
78 %i,m: entrance to the for loop
79 %sp_sorted: is the sorted matrix according to the WS value, it
80 contains [f,v,WS]
81
82 k=H(i,m,sp_sorted);
83 y=(1+(k/800)^4)^(-1);
84
85 function y = H(i,m,sp_sorted)
86 %true Virtual pitch
87
88 %sp_sorted: is the sorted matrix according to the WS value, it
89 contains [f,v,WS]
90
91 fi = sp_sorted(i,1);
92 vi = sp_sorted(i,2);
93 t1 = sign(m - 1)*(10^(-3))*(18+2.5*m - (50 - 7* m)*(fi / 1000)*(m^(-
94 1)) + 0.1*((m^(-1))*(fi / 1000))^(-2));
95 t = (m^(-1))*fi*(1+vi-t1);
96 y=t;
97
98 %-----
99 function y = C(i,j,m,sp_sorted)
100 %coincidence coefficient

```

```
99
100 %sp_sorted: is the sorted matrix according to the WS value, it
    contains [f,v,WS]
101
102 WSi=sp_sorted(i,3);
103 WSj=sp_sorted(j,3);
104 g = gamma(i,j,m,sp_sorted);
105 n = n(i,j,m,sp_sorted);
106 if ((g <= 0.08)&(n<=20))
107     t = (((WSi*WSj)/(m*n))^0.5)*(1-(g/0.08));
108 else
109     t=0;
110 end
111
112 y=t;
113
114 function y = gamma(i,j,m,sp_sorted)
115 %g:The coefficient gama represents the degree of inharmonicity of
    the tested component pair
116
117 %sp_sorted: is the sorted matrix according to the WS value, it
    contains [f,v,WS]
118 fi = sp_sorted(i,1);
119 fj = sp_sorted(j,1);
120 Nijm=n(i,j,m,sp_sorted);
121 y1 = (Nijm * fi) / (m * fj) - 1 ;
122 y = abs(y1);
123
124 function y = n(i,j,m,sp_sorted)
125 %It is integer which is nearest to the quantity mfj/fi
126
127 %sp_sorted: is the sorted matrix according to the WS value, it
    contains [f,v,WS]
128 fi = sp_sorted(i,1);
129 fj = sp_sorted(j,1);
130 y = Int((m * fj / fi) + 0.5);
131
132 function y= Int(x)
133 % y is the interger part of the quantity x
134 % used only when x >0
135
136 y=fix(x);
```

Bijlage D: Diapresentatie

Op de bijgevoegde CD-ROM staat een diapresentatie over Sound Quality, geschikt als addendum bij de cursus Mechanische Trillingen & Geluid.

