



Einfluss unterschiedlicher
Turbinenübergangskanäle auf die
Schallemissionen in einem zweistufigen
Turbinenprüfstand

Diplomarbeit

zur Erlangung des akademischen Titels Diplomingenieur
eingereicht an der
Fakultät für Maschinenbau der Technischen Universität Graz

von
Sabine Bauinger, BSc

Institut für Thermische Turbomaschinen und Maschinendynamik
Vorstand: o.Univ.-Prof. Dr.-Ing. Franz Heitmeir

Begutachter:
Dipl.-Ing. Dr.techn. Andreas Marn

Betreuer:
Dipl.-Ing. Christian Faustmann

Graz, Februar 2013

Wer glaubt etwas zu sein, hat aufgehört etwas zu werden.

Sokrates

Danksagung

Herzlich bedanken möchte ich mich bei meinem Betreuer Dipl.-Ing. Christian Faustmann für die ausgesprochen gute Zusammenarbeit, sein offenes Ohr für all meine Anliegen und die tatkräftige Unterstützung, die weit über das Verfassen der Diplomarbeit hinausging. Auch Herr Dipl.-Ing. Dr. techn. Andreas Marn hat als Begutachter und Beurteiler dieser Arbeit mit seinen guten Ideen und konstruktiven Verbesserungsvorschlägen einen unverzichtbaren Beitrag geleistet, wofür ich ihm sehr dankbar bin.

Dank gebührt auch unserem Institutsvorstand Herrn Univ.-Prof. Dr.-Ing. Franz Heitmeir, der nicht nur mein Interesse für das Gebiet der Thermischen Turbomaschinen weckte, sondern mich auch von Anfang an freundlich am Institut aufgenommen und mich darüber hinaus zum Bleiben motiviert hat.

Hinter dem Gelingen dieser Arbeit stecken so viele Menschen, deren Namen eigentlich dort stehen müssten, wo jetzt der meine steht, allen voran der meiner Eltern Renate und Friedrich Bauinger. Sie haben mir mein Studium nicht nur finanziell ermöglicht, sondern mich zu jeder Zeit sowohl fachlich (danke Papa) wie auch moralisch (danke Mama) unterstützt und mich während aller Höhen und Tiefen begleitet ohne dabei jemals an mir zu zweifeln.

Bedanken möchte ich mich außerdem bei meinem guten Freund Sebastian, der im wahrsten Sinne des Wortes vom ersten Tag meines Studiums an ein unverzichtbarer Bestandteil desselben war, und bei meinem Freund Karl für seine unendliche Geduld und die zahlreichen motivierenden Worte und Taten.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am 2.2.2013

Sabine Bauinger, BSc

Abkürzungen und Symbole

Abkürzungen

TMTF	Turning Mid Turbine Frame
HDT	Hochdruckturbine
NDT	Niederdruckturbine
HP	High Pressure
LP	Low Pressure
DREAM	Validation for Radical Engine Architecture Systems
BPF	Blade Passing Frequency
DFT	Diskrete Fourier Transformation
FFT	Fast Fourier Transformation
IDFT	Inverse DFT
RSA	Rotor Synchronic Averaging
DLR	Deutsches Zentrum für Luft- und Raumfahrt
AMA	Azimutalmodenanalyse
RMA	Radialmodenanalyse
SPL	Sound Pressure Level, Schalldruckpegel

Symbole

Griechische Symbole

ρ	Dichte bzw. Schalldichte
φ	Winkel
ω	Winkelgeschwindigkeit
Ω	Winkelgeschwindigkeit
$\Delta\varphi$	Schrittweite der Traversierung
φ_k	Winkelposition des Mikrofons k
δ	Grenzschichtdicke
λ	Wellenlänge
θ	Winkel, Zylinderkoordinate

Lateinische Symbole

p	Druck bzw. Schalldruck
\vec{v}	Schallschnelle
q	Schallfluss
I	Schallintensität
P	Schalleistung
L_p	Schalldruckpegel
L_v	Schallschnellepegel
L_I	Schallintensitätspegel
L_W	Schalleistungspegel
c	Wellenausbreitungsgeschwindigkeit
r	Radius, Zylinderkoordinate
R	Außenradius des Strömungskanals
x	Koordinate
y	Koordinate
z	Koordinate, Zylinderkoordinate
t	Zeit
m	Azimutale Modenordnung
n	Radiale Modenordnung, Drehzahl
f	Frequenz
k	Wellenzahl
u	Geschwindigkeit in x-Richtung
Δm	Massendifferenz
S	Querschnittsfläche des Strömungskanals
M	Machzahl
M_x	Machzahl in x-Richtung
h	harmonischer Index
B	Anzahl der Laufradschaufeln
V	Anzahl der Leitrad-schaufeln
N	Anzahl der Abtastwerte im Zeitbereich
T	Periodendauer
N_φ	Anzahl der azimutalen Messpunkte
W_N	Drehoperator
Δh	Enthalpiedifferenz
f_A	Abtastfrequenz
nn	Anzahl der Neuabtastungen
n_{period}	Anzahl der Umdrehungen während der Messzeit
A_m	Komplexe Amplitude
\tilde{p}	Komplexer Schalldruck
f_{mn}	Zylinderfunktion
F_{mn}	Skalierungsfunktion
Y_m	Neumannfunktion
J_m	Besselfunktion

K	Anzahl der Mikrofone
p_k	Komplexer Schalldruck am Mikrofon k
p_0	Komplexer Schalldruck am Referenzmikrofon
p_0^*	Konjugiert komplexer Schalldruck am Referenzmikrofon
f_c	Kritische Grenzfrequenz
W_m	Matrix mit akustischen Modellgleichungen
W_m'	Adjungierte Matrix von W_m
W_m^+	Pseudoinverse von W_m

Inhaltsverzeichnis

1	Einleitung und Aufgabenstellung	1
1.1	Einleitung	1
1.2	Aufgabenstellung	2
2	Versuchseinrichtung	3
2.1	Verdichteranlage	3
2.2	Beschreibung des Prüfstands	5
2.3	Untersuchte Konfigurationen des TMTF	9
2.3.1	MTU Aero Engines	10
2.3.2	Volvo Aero	11
2.3.3	TTM Graz	12
3	Grundlagen	14
3.1	Definition Schall und Schallentstehung	14
3.1.1	Grundgrößen und Pegelmaße	15
3.1.2	Schallentstehung	17
3.2	Schallausbreitung	20
3.2.1	Lösung der Wellengleichung für zylindrische Rohre	22
3.2.2	Die cut-off Bedingung für den dünnen ringförmigen Kanal	28
3.2.3	Wellenausbreitung bei überlagerter Strömung	29

3.3	Problem des erweiterten Kanals	30
3.4	Problem des Druckgradienten und der drallbehafteten Abströmung	33
3.5	Modenordnungen nach Tyler und Sofrin	34
3.6	Mathematische Grundlagen der Fast Fourier Transformation (FFT)	36
4	Schallmessung	44
4.1	Verwendete Messtechnik	44
4.2	Anordnung der Mikrofone	45
4.3	Datenerfassung	48
5	Auswertemethoden	49
5.1	Auswertung des Rohsignals	49
5.2	Adaptive Neuabtastung	51
5.3	Phase Averaging	52
5.4	Rotor Synchronic Averaging (RSA)	56
5.5	Azimutalmodenanalyse (AMA)	60
5.6	Radialmodenanalyse (RMA)	63
6	Diskussion der Messergebnisse	66
6.1	Vergleich zwischen Messungen an verschiedenen Tagen und mit verschiedenen Referenzmikrofonen	66
6.2	Vergleich der FFT des Rohsignals und der FFT nach Datenaufbe- reitung	68
6.3	Verwendung verschiedener Trigger	70
6.3.1	Hochdrucktrigger	70
6.3.2	Niederdrucktrigger	73
6.3.3	Rotor Synchronic Averaging	76
6.4	Vergleich der drei Konfigurationen	78

7 Zusammenfassung und Ausblick

90

Kapitel 1

Einleitung und Aufgabenstellung

1.1 Einleitung

Alleine in Österreich hat sich die Anzahl der Flugbewegungen, die den kommerziellen Luftverkehr betreffen, in den letzten 20 Jahren beinahe verdreifacht. Das bedeutet nicht nur, dass sich die Anzahl der Starts und Landungen von rund 125.000 im Jahr 1990 auf etwa 350.000 im Jahr 2008 erhöht hat, sondern auch, dass der Flugverkehr in weitaus höherem Maß anwächst als der Straßenverkehr und somit nicht nur Maßnahmen zur Reduktion der CO₂ Emissionen unbedingt erforderlich sind, sondern auch ein besonderes Augenmerk auf die Senkung der Lärmemissionen eines Flugzeuges gelegt werden muss. Besonders der Start, der Landeanflug und die Imissionen in den Bereichen seitlich der Landebahn stehen hierbei im Mittelpunkt der Betrachtungen, da die Lärmbelästigung für Anrainer von Flughäfen nicht nur eine Verminderung der Lebensqualität darstellt, sondern im schlimmsten Fall auch zu massiven gesundheitlichen Problemen der Betroffenen führen kann. Aus diesem Grund wurde 2009 vom Umweltbundesamt für Österreich eine Luftverkehr-Immissionschutzverordnung (LuIV) veröffentlicht, die Immissionsgrenzwerte für Lärm festlegt, genaueres dazu ist unter [7] nachzulesen. Was die Schallemissionen von Flugzeugen betrifft, so sind diese im Abkommen über die internationale Zivilluftfahrt geregelt, wobei der Trend selbstverständlich zu immer leiseren Flugzeugen hin geht. Eine Lärmquelle stellen hierbei die Triebwerke eines Flugzeuges dar, auf die auch ausschließlich in der folgenden Arbeit eingegangen werden soll. Insbesondere wird hier die Schallentwicklung hinter der Niederdruckturbinen und der Einfluss des sogenannten Turning Mid Turbine Frame (TMTF) betrachtet, da das Flugzeug selbst und auch der Fan immer "leiser" werden und dadurch der Anteil der Turbinen, speziell die Niederdruckturbinen, an den gesamten Lärmemissionen immer größer wird. Zu diesem Zweck wurden drei verschiedene TMTFs in den bestehenden zweistufigen Zweiwellen-Turbinenprüfstand am Institut für Thermische Turbomaschinen und Maschinendynamik an der TU Graz eingebaut, die entsprechenden Akustikmessungen und

deren Auswertung durchgeführt und die drei Konfigurationen hinsichtlich ihrer Schallemissionen untersucht und verglichen. Die Details und Ergebnisse dazu sind in der nachfolgenden Arbeit zu finden.

Abbildung 1.1 zeigt die Lärmquellen an einem anfliegenden Verkehrsflugzeug.



Abbildung 1.1: Lärmquellen an einem anfliegenden Verkehrsflugzeug [14]

1.2 Aufgabenstellung

Ziel dieser Diplomarbeit war es, am bestehenden transsonischen Zwei-Wellen-Turbinenprüfstand des Instituts für Thermische Turbomaschinen und Maschinendynamik drei verschiedene beschaufelte Turbinenübergangskanäle (TMTFs), die hier den Stator der Niederdruckturbinen ersetzen, hinsichtlich ihres Einflusses auf die Schallentwicklung hinter dem Niederdruckrotor zu untersuchen und die drei Konfigurationen miteinander zu vergleichen. Die entsprechenden Umbauten am Prüfstand sollen vorgenommen und die Akustikmessungen durchgeführt werden. Anschließend werden die erhaltenen Messdaten anhand von bestimmten Auswerterroutinen ausgewertet, um die Unterschiede zwischen den verschiedenen Komponenten bezüglich ihres Einflusses auf die Schallemissionen des Triebwerks aufzuzeigen. Des Weiteren war die Aufgabenstellung dieser Arbeit, neben der Durchführung der Messungen und der Aufbereitung der Messdaten, auch eine angemessene Dokumentation für die verwendeten Auswerterroutinen zu erstellen, um in Zukunft die Arbeit mit diesen Programmen zu erleichtern und ein entsprechendes Nachschlagewerk für nachfolgende Messauswertungen zur Verfügung zu stellen. Die Details dazu sind im Anhang, der als eine Art Handbuch dienen soll, zu finden.

Kapitel 2

Versuchseinrichtung

In den nachfolgenden Kapiteln soll nun etwas vertiefender auf die Versuchseinrichtung, an der die Akustikmessungen durchgeführt wurden, eingegangen werden. Im Detail werden hier die Verdichteranlage beschrieben, der Aufbau des Turbinenprüfstands - im Speziellen die Niederdruckturbine - erklärt und die drei untersuchten Konfigurationen des Turbinenübergangskanals erläutert.

2.1 Verdichteranlage

Wie aus [9] und [10] entnommen, besteht die Verdichteranlage im zweiten Untergeschoß des Instituts aus zwei Turbokompressoren und einem Schraubenkompressor, genauer gesagt einem parallel arbeitenden Doppel-Schraubenverdichter, dessen Komponenten in einem Gehäuse untergebracht sind und somit in weiterer Folge von drei und nicht von vier Kompressoren die Rede ist. Diese drei Verdichter können nun in Reihe oder in Serie zusammengeschalten werden, zusätzlich kann jeder von ihnen einzeln betrieben werden und dadurch sind insgesamt 13 Schaltungen bzw. Fahrweisen möglich, wobei vier davon beim Betrieb der Versuchsturbine zum Einsatz kommen. Steuerung und Überwachung der Anlage erfolgen von den Beobachtungsräumen im Erdgeschoß und dem ersten Untergeschoß der Institutsräumlichkeiten aus, auch die unterschiedlichen Fahrweisen lassen sich von hier aus vollautomatisch schalten.

Die Leistungsdaten der einzelnen Verdichter sind nachfolgender Tabelle zu entnehmen.

Tabelle 2.1: Technische Daten der einzelnen Verdichter [11]

Verdichter	Ansaug- volumen- strom [m ³ /h]	Massen- strom [kg/s]	Druck- verhältnis	Verdichter- drehzahl [U/min]	Motor- drehzahl [U/min]	Motor- leistung [kW]
SC20	27500	9.0	2.9	15780	2978	1450
SC14	15500	5.1	2.9	20930	996-1482	900
E1+E2	8000	2.6	3.1	3730	1485	400

Die maximal aufgenommene elektrische Leistung für die Verdichteranlage beträgt insgesamt etwa 3 MW und wird von Dreiphasen-Asynchronmotoren, die als Einzelantriebe für jeden der drei Verdichter ausgelegt sind, zur Verfügung gestellt. Wichtig ist, dass in keiner der Schaltungen/Fahrweisen die zur Verfügung stehende Maximalleistung überschritten werden darf. Der hohe Stromverbrauch ist auch der Grund, warum die Anlage in anderen Fahrweisen als den Einzelbetrieben fast ausschließlich in der Nacht in Betrieb genommen wird.

Zur Regelung der Temperatur gibt es zwei Luftkühler, die sowohl als Zwischen- als auch als Nachkühler in das System geschaltet werden können. Der vorwiegende Nachtbetrieb der Anlage hat hier den Vorteil, dass die Außentemperaturen meist niedriger sind als untermittags und dadurch der Betrieb der Kühler begünstigt wird. Außerdem sind die Temperaturschwankungen in der Nacht nicht so hoch wie tagsüber.

Die nachstehende Abbildung 2.1 zeigt nun schematisch die gesamte Verdichteranlage, des Weiteren werden in Abbildung 2.2 die Schaltungen für vier der 13 möglichen Fahrweisen dargestellt.

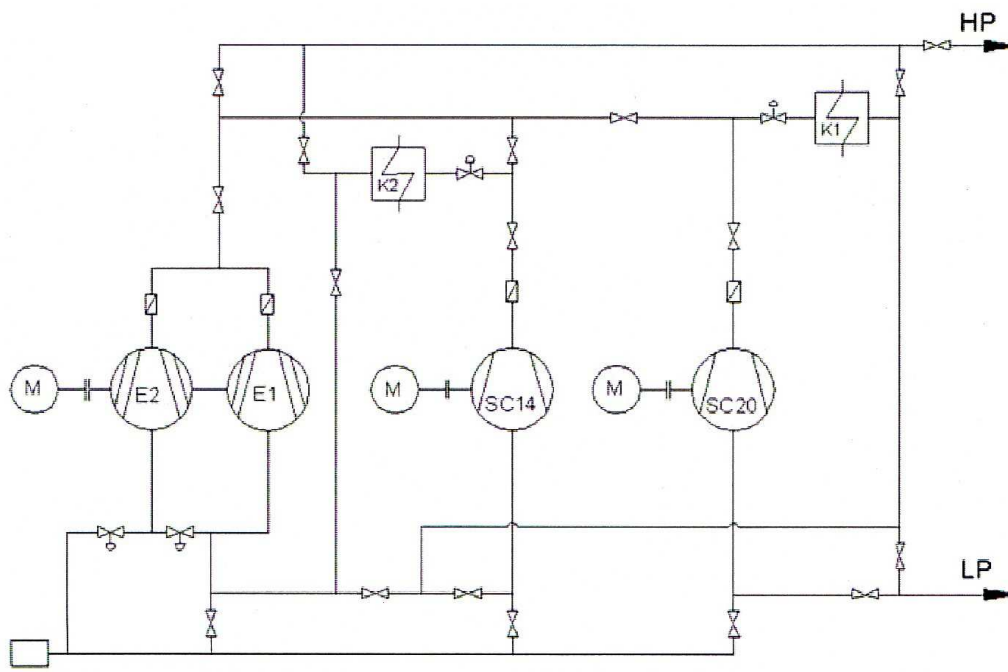


Abbildung 2.1: Schematische Darstellung der Verdichteranlage [10]

Folgende technische Eckdaten lassen sich nun für die gesamte Verdichteranlage zusammenfassen: Für den Prüfstand kann komprimierte Luft bis zu einem Druckverhältnis von 10 im Druckbetrieb bzw. 25 im Saugbetrieb zur Verfügung gestellt werden, wobei hier noch anzumerken sei, dass für einen Großteil der Versuche tatsächlich das Druckverhältnis und nicht die absoluten Drücke von Bedeutung sind. Was die Fördermengen betrifft so sind hier zwischen 2,6 und 16,7 kg/s möglich.

Weitere Details zu Planung, Auslegung und Betrieb der Verdichteranlage können in [10] nachgelesen werden.

2.2 Beschreibung des Prüfstands

Durchgeführt wurden die in dieser Diplomarbeit diskutierten Akustikmessungen an einem zweistufigen, transsonischen Zwei-Wellen-Turbinenprüfstand am Institut für Thermische Turbomaschinen und Maschinendynamik, der aus einer Hochdruckstufe und einer entgegengesetzt rotierenden Niederdruckstufe besteht. Die verdichtete Luft, die für die Versuchsfahrten benötigt wird, liefert die 3 MW Verdichteranlage im Keller des Instituts, die bereits in Kapitel 2.1 beschrieben wurde. Durch die Zwischenkühlung, wie sie ebenfalls bereits erwähnt wurde, kann die Eintrittstemperatur in die Turbine von 40°C bis 185°C variiert werden. Die Wellenleistung der Hochdruckturbinen, deren Drehzahl auf maximal 11550 rpm limitiert ist, wird verwendet, um einen dreistufigen, radialen Bremsverdich-

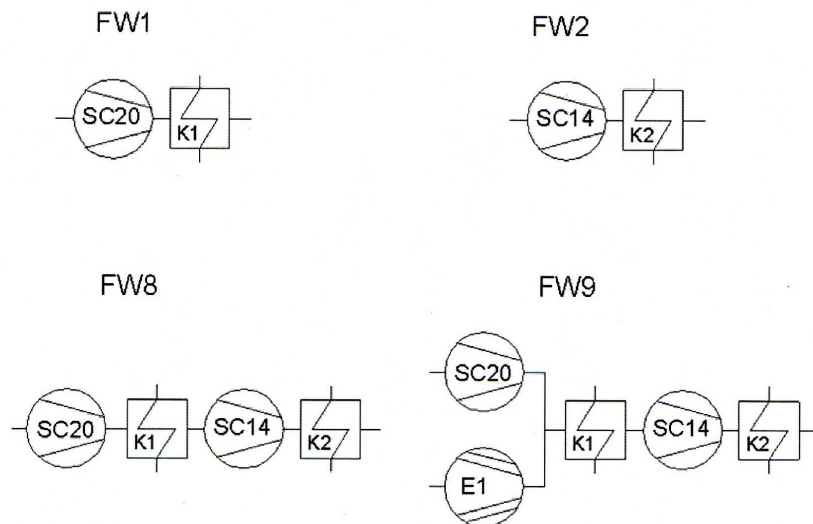


Abbildung 2.2: Schaltungen für vier ausgesuchte Fahrweisen [10]

ter zu betreiben, der zusätzlich Umgebungsluft ansaugt, welche anschließend in der Mischkammer mit der komprimierten Luft aus der Verdichteranlage vermischt wird und dadurch zu einer Vergrößerung des Luft-Massenstromes führt. Was die Niederdruckturbine betrifft, so wird ihre Leistung von einer Wasserbremse, deren maximale Leistung 700 kW beträgt, aufgenommen. Die Drehzahl des Niederdruckrotors betrug in den durchgeführten Versuchen 3550 rpm.

Bemerkenswert ist, dass die Niederdruckstufe axial verschoben werden kann, und es dadurch möglich ist, Niederdruckstatoren bzw. Turbinenübergangskanäle mit unterschiedlicher axialer Länge in den Prüfstand einzubauen und außerdem ein deutlich einfacherer Umbau der Versuchsanlage bewerkstelligt werden kann.

Betrieben wurde die Anlage im sogenannten Aero Design Point, bei dem das Gesamt-Druckverhältnis in etwa 4 beträgt, das der Hochdruckstufe allein ungefähr 3 und das der Niederdruckstufe ca. 1,3. Die absolute Machzahl am Eintritt des TMTFs beträgt hier 0,5 und ist damit repräsentativ für Zweikreis-Triebwerke mit einer einstufigen Hochdruckturbine. Generell gilt, dass für die Vergleichbarkeit mit Ergebnissen anderer Messungen bzw. mit den tatsächlichen Verhältnissen in Triebwerken vor allem relative Kennzahlen, wie etwa das Druckverhältnis oder die Machzahl, herangezogen werden. Grund dafür ist die bei Thermischen Turbomaschinen zum Einsatz kommende Machzahl-Ähnlichkeit, was bedeutet, dass eine Strömung physikalisch dann ähnlich ist, wenn für die Umströmung geometrisch ähnlicher Körper die relevanten Kennzahlen dieselben sind. Basierend auf der Mach'schen Ähnlichkeit können nun Größen wie der reduzierte Massenstrom oder die reduzierte Drehzahl berechnet werden und dadurch nicht nur von einer Messung zur nächsten unabhängig von den Umgebungsbedingungen wieder nahezu dieselben Strömungsgrößen eingestellt, sondern im Vergleich zu geometrisch ähnlichen Maschinen auch noch ähnliche Geschwindigkeitsdreiecke erzielt werden.

Die folgenden zwei Abbildungen zeigen verschiedene Ansichten der Versuchsanlage.

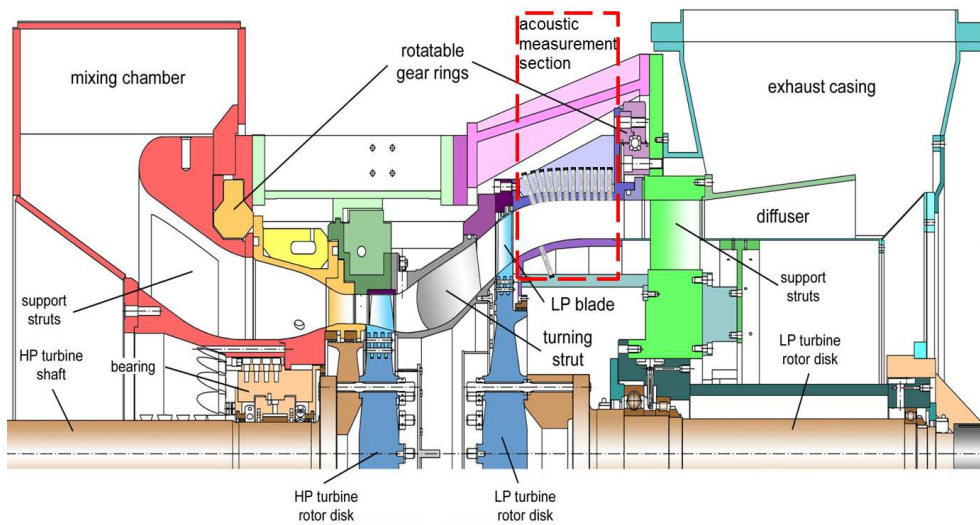


Abbildung 2.3: Achsenschnitt durch die Versuchsanlage [8]

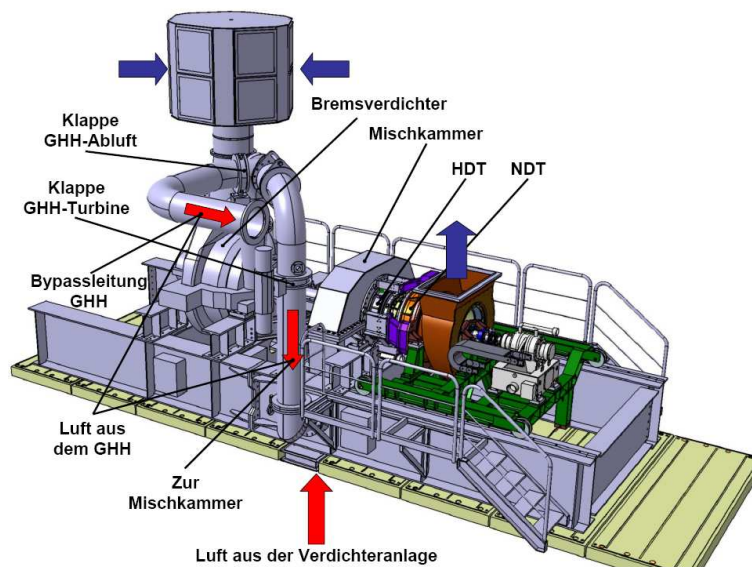


Abbildung 2.4: Darstellung der gesamten Versuchsanlage

Die komprimierte Luft strömt aus der Mischkammer zuerst zum Hochdruckstator, der mit 24 Leitschaufeln bestückt und um 360° drehbar ist, wird dort in Umfangsrichtung beschleunigt und trifft dann auf den Hochdruckrotor mit 36 Laufschaufeln, der in Abbildung 2.3 blau dargestellt ist. Nach der Hochdruckstufe folgt anstelle des Stators der Niederdruckstufe ein beschaufelter S-förmiger Turbinenübergangskanal, in weiterer Folge auch als TMTF (Turning Mid Turbine Frame) bezeichnet, der aus 16 sehr massiven Schaufeln besteht. Die Robustheit dieses Bauteils liegt darin begründet, dass er unter anderem auch dafür vorgesehen sein wird, sowohl die Lagerung des HP Rotors als auch der rotierenden Teile der nachfolgenden Niederdruckturbinen zu übernehmen. Folgende zwei Schnitte durch ein Triebwerk sollen dies veranschaulichen.

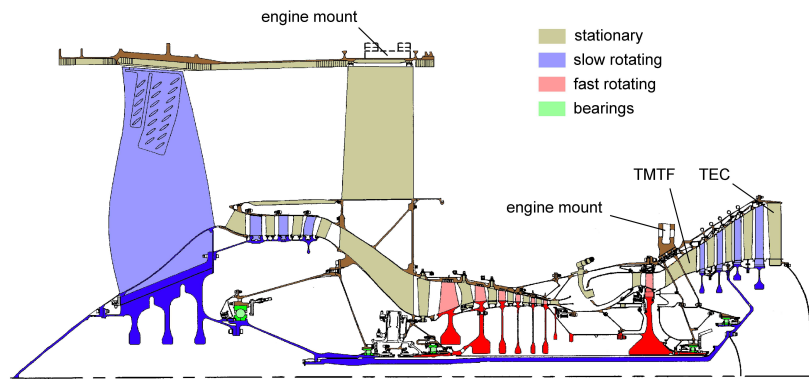


Abbildung 2.5: Triebwerk mit Darstellung der Lager

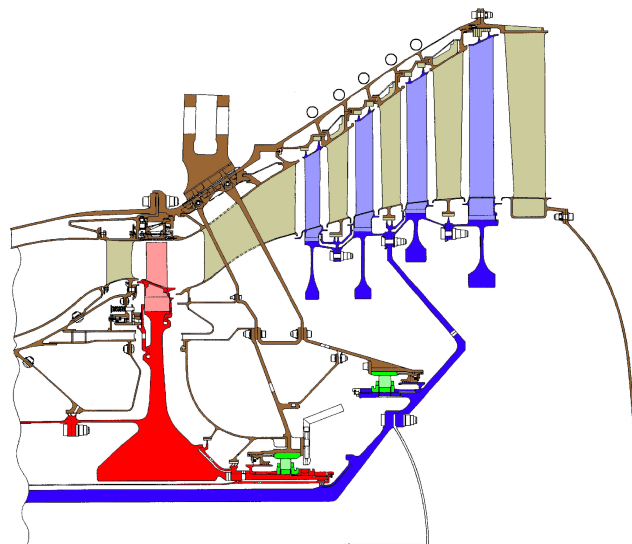


Abbildung 2.6: Detail des TMTFs

S-förmig ist der Übergangskanal wegen dem radialen Offset zwischen HPT und LPT, der sich aus der Forderung ergibt, dass die Leistungsziffer $\frac{\Delta h}{u^2}$ von HPT und LPT, vorrangig jedoch LPT, annähernd gleich bleiben soll. Da die Drehzahl der Niederdruckturbine abnimmt, lässt sich dies nur durch einen größeren Radius bewerkstelligen. Die Drehzahl muss kleiner werden, da die LPT mit dem Fan gekoppelt ist und dieser nicht zu schnell drehen darf, da ansonsten die Schaufelfußbelastung zu groß werden und es außerdem zu einer Überschallströmung mit zu hoher Machzahl an den Schaufelspitzen kommen würde. Senkt man die Drehzahl, so kann der Fandurchmesser und somit auch das Bypass-Verhältnis vergrößert werden, ohne dass es zu Festigkeitsproblemen und Schwierigkeiten aufgrund von großen Mach-Zahlen kommt. Details zu Konstruktion und Auslegung des Übergangskanals, dessen Funktion in die des Niederdruckstators integriert wurde, können in [16] nachgelesen werden.

Durch die Struts des TMTFs wird die einströmende Luft in - relativ zur Drehrichtung des HP-Rotors betrachtet - negative Richtung abgelenkt und tritt anschließend in den entgegengesetzt drehenden Rotor der Niederdruckstufe, bestehend aus 72 Schaufeln, ein. Der Niederdruckrotor dreht sich deshalb entgegengesetzt zum Hochdruckrotor, weil bei der Abströmung mit Gegendrall in der HPT das Maximum an Energie gewonnen werden kann und damit die Strömung danach nicht so stark umgelenkt werden muss, lässt man den LP Rotor in entgegengesetzter Richtung rotieren. Hinter der Niederdruckturbine folgt die Messstrecke, entlang derer die Akustikmessungen durchgeführt wurden und die ebenfalls um 360° traversiert werden kann. Zum besseren Verständnis sind die Mikrofone in der obigen Darstellung (Abbildung 2.3) eingezeichnet. Abgeführt wird die Luft letztendlich über einen Abluftkanal mit integrierten Schalldämpfern, der im Anschluss an die Akustikmessstrecke angebracht ist und die Abluft ins Freie leitet. Zu erwähnen sei hier auch noch das Wellenüberwachungssystem von Bently Nevada, das nicht nur als Maschinenüberwachung dient und somit die Anlagensicherheit gewährleistet, sondern auch die Triggersignale vom Hochdruck- bzw. Niederdruckrotor liefert (siehe auch Kapitel 5).

Details zum Aufbau und der Funktionsweise des Prüfstandes sind unter [16] und [17] zu finden.

2.3 Untersuchte Konfigurationen des TMTF

Hauptaugenmerk lag bei den im Rahmen dieser Arbeit durchgeführten Akustikmessungen auf der Niederdruckturbine, speziell auf dem TMTF. Zu diesem Zweck wurden drei verschiedene Konfigurationen des TMTFs eingebaut und für jedes der drei die jeweiligen Messungen durchgeführt und diese miteinander verglichen, alle anderen Komponenten der Turbine blieben dabei unverändert, außerdem wurde die Relativposition aller Schaufeln zueinander bei allen drei Setups konstant gehalten. Genauer zu den Messergebnissen und deren Diskussion ist in Kapitel

6 zu finden. Abbildung 2.7 zeigt zusammenfassend die Entwicklung des TMTFs, die im Prinzip alle drei untersuchten Konfigurationen beinhaltet.

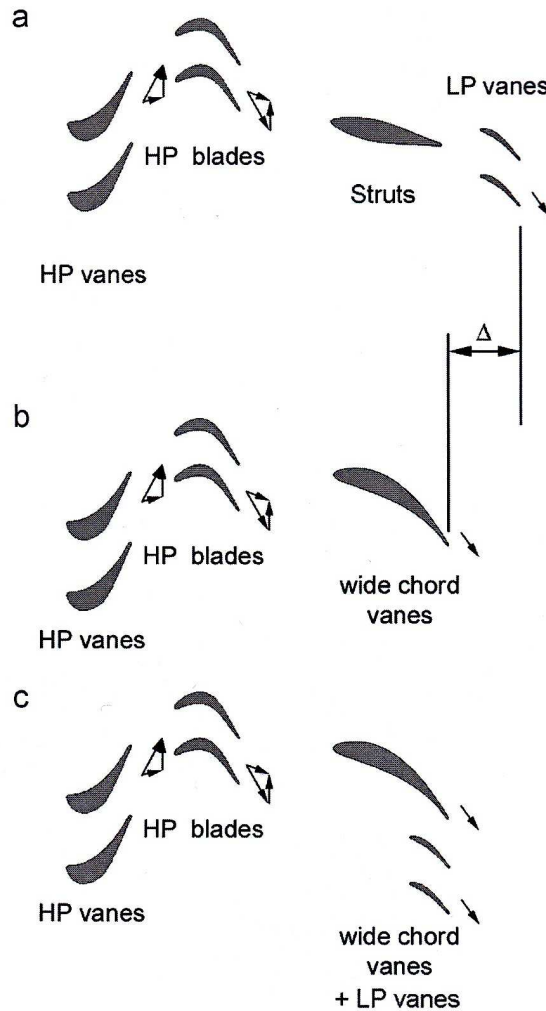


Abbildung 2.7: (a) Diffusor mit Struts, (b) TMTF, (c) TMTF mit Splittern, vgl. Setup TTM [22]

Auf die konstruktiven Unterschiede zwischen den Setups, basierend auf den Inhalten von [15], soll nun in den folgenden Abschnitten hingewiesen werden.

2.3.1 MTU Aero Engines

Das erste der drei Designs für das TMTF stammt von der Firma MTU Aero Engines und wurde, wie auch das Setup von Volvo Aero, im Rahmen des EU-Projektes DREAM (validation for Radical Engine Architecture systems) entworfen, konstruiert und untersucht.

Die dimensionslose Länge dieser Konfiguration beträgt 3.5 ($= \frac{L_{out}}{h_{in}}$) und die Grö-

ßen, die zu deren Berechnung herangezogen wurden, sind in folgendem Längsschnitt der Versuchsanlage, der eigentlich das Setup von Volvo zeigt, jedoch für die Darstellung der geometrischen Kenngrößen herangezogen wurde, eingezeichnet.

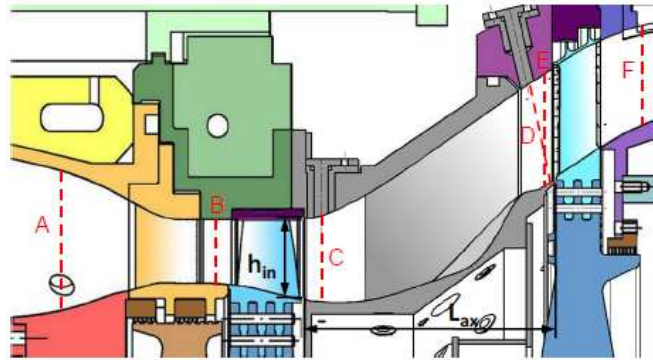


Abbildung 2.8: Schnitt durch die Anlage mit Darstellung der Größen L_{ax} und h_{in} [15]

Wie bereits in Kapitel 2.2 erklärt, ist das TMTF Teil der Lagerung sowohl der Hochdruck- als auch der Niederdruckturbinen und somit ein Strukturbauteil, was die geringe Anzahl jedoch die sehr massive Ausführung der Struts erklärt. Die dicken Schaufeln haben aber den Nachteil, dass sie starke Sekundärströmungen hervorrufen, was durch ein entsprechendes Schaufeldesign etwas vermindert werden soll. Die Firma MTU versuchte dies bei ihrer Konfiguration durch eine dreidimensionale Gestaltung der Schaufeln zu erreichen, ohne jedoch dabei die Durchführbarkeit für Serviceleitungen, als Antrieb für Hilfsaggregate oder Ähnliches zu verringern. Zusätzlich sind die Struts in Zukunft für die Durchführung der Triebwerksaufhängung vorgesehen (siehe Abbildung 2.6).

2.3.2 Volvo Aero

Der Entwurf des zweiten Setups wurde von Volvo Aero durchgeführt und weist nicht nur eine um zehn Prozent kürzere Kanallänge auf, sondern auch die axiale Ausdehnung der Struts wurde im Vergleich zum MTU-Design um zehn Prozent verkürzt. Die daraus resultierende Gewichtseinsparung verringert auf der einen Seite den Treibstoffverbrauch und führt auch gleichzeitig zur Einsparung von Kosten. Indem die Kontur der Seitenwände am Hub nicht achsensymmetrisch gestaltet worden ist, wurden trotz kürzerer Einbaulänge im Vergleich zum ersten Design zusätzliche Verluste vermieden und gleiche Einströmbedingungen in den Rotor der Niederdruckturbinen erreicht. Genauer bedeutet das, dass ein sogenanntes Endwall Contouring am Hub durchgeführt wurde, wodurch der Druckunterschied zwischen Druck- und Saugseite verkleinert und somit Sekundärströmungen vermindert werden. Aufgrund des Druckgradienten zwischen den beiden Schaufelseiten kommt

es nämlich zu einer Strömung vom hohen zum niedrigeren Druck. Um das zu verringern, versucht man einerseits durch Verkleinerung der Querschnittsfläche des Kanals auf der Druckseite die Strömungsgeschwindigkeit zu erhöhen und dadurch den Druck zu senken, andererseits durch Vergrößerung der Querschnittsfläche auf der Saugseite die Geschwindigkeit zu senken und gleichzeitig den Druck zu erhöhen. Unverändert blieben die Anzahl der Struts, die insgesamt nach wie vor 16 beträgt, und der radiale Offset, allerdings sind die Schaufeln im Vergleich zum MTU-Setup nicht mehr dreidimensional gestaltet und daher bei gleichbleibender Durchführbarkeit und Festigkeit nicht mehr ganz so dick.

Die Unterschiede zwischen dem ersten und dem zweiten Design soll die Skizze in Abbildung 2.9 veranschaulichen. Das MTU Setup wurde hier blau eingezeichnet, die Konfiguration von Volvo in rot.

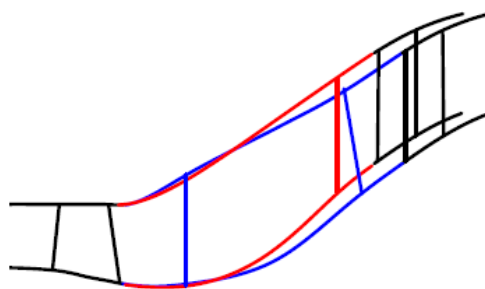


Abbildung 2.9: Vergleich zwischen den Designs von MTU und Volvo [15]

2.3.3 TTM Graz

Der Entwurf des dritten Setups entstand im Rahmen einer Dissertation am Institut für Thermische Turbomaschinen und Maschinendynamik an der TU Graz und basiert im Wesentlichen auf dem Design der MTU. Das heißt, die Struts, die Kanalgeometrie und die axiale Länge sind identisch mit denen der ursprünglichen Konstruktion, allerdings wurden zwischen jeweils zwei Struts zusätzlich zwei Splitter eingefügt. Ziel dieser Umgestaltung des TMTFs war einerseits die Reduktion von Sekundärströmungen, andererseits die Dämpfung der Rotor-Rotor-Interaktionen. Anzumerken ist jedoch, dass diese Ausführung nicht ganz optimal ist, da zwar die Splitter eingefügt, die Kanalgeometrie jedoch nicht dementsprechend angepasst wurde. Aus diesem Grund musste auch der Betriebspunkt verändert werden, damit die Mach-Zahlen am Austritt im Vergleich zu den anderen beiden Konfigurationen in etwa gleich blieben. Abbildung 2.10 zeigt schematisch einen Schnitt durch die Versuchsanlage mit diesem Setup. Weiterführende Informationen zu diesem Thema sind auch unter [18] zu finden.

Tabelle 2.2 soll nun noch einmal die wichtigsten Parameter aller drei Konfigurationen zusammenfassen und beinhaltet zusätzlich die Betriebsbedingungen.

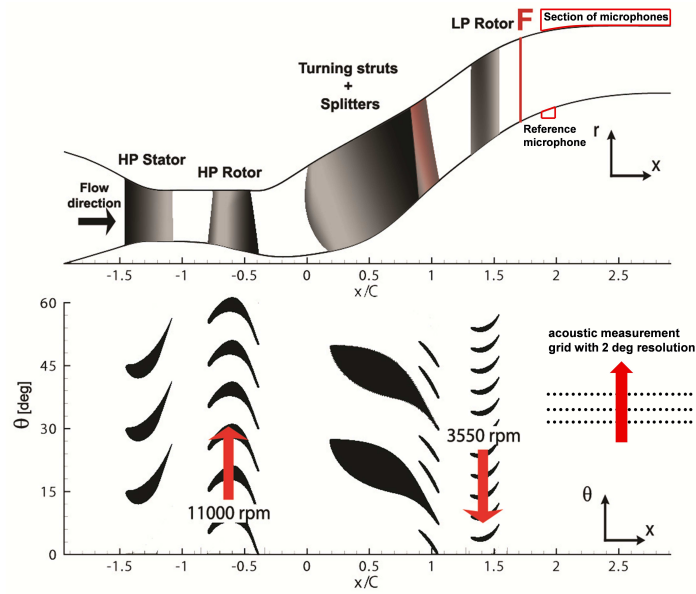


Abbildung 2.10: Institutsinternes Design des TMTFs [21]

Tabelle 2.2: Parameter der drei Setups und Betriebsbedingungen [21]

Blading parameters					
	HP vane	HP blade	Struts	Splitters	LP blade
Vane/ blade no.	24	36	16	24	72
h/c_{ax}	1.15	1.37	0.53	3.5	2.94
$Re(10^{-6})$	2.38	1.1	1.86	0.4	0.46
Tip gap	-	unshrouded	-	-	shrouded
Operating conditions					
	HP stage		LP stage		
$n_{r,in}[rpm/\sqrt{k}]$	524.4		195.3		
$m_{r,in}[kg/s \cdot \sqrt{k}/bar]$	81.2		214.6		
Baseline	Stage p_t ratio		3	1.3	
Baseline	Power [kW]		1710	340	
with Splitters	Stage p_t ratio		2.83	1.36	
with Splitters	Power [kW]		1425	353	

Kapitel 3

Grundlagen

In diesem Abschnitt sollen nun die für das Verständnis dieser Arbeit notwendigen Grundlagen vermittelt werden. Dazu wird zuerst auf die in der Akustik verwendeten Grundgrößen bzw. -gleichungen und Pegelmaße eingegangen und Allgemeines zur Schallentstehung behandelt. Anschließend wird in Kapitel 3.2 die Schallausbreitung erklärt und zu diesem Zweck die Wellengleichung hergeleitet, sowie für den Fall des zylindrischen Rohres gelöst. Neben eher allgemein gehaltener Theorie werden auch speziell den Turbinenprüfstand betreffende Probleme der Akustik angesprochen, wie etwa das des erweiterten Kanals oder der drallbehafteten Abströmung, des Weiteren sind im letzten Abschnitt dieses Kapitels auch Informationen bezüglich der Modenordnungen nach Tyler und Sofrin zu finden.

Als Quellen für die Erklärungen, Definitionen und Herleitungen wurden vorwiegend [2], [12] und [13] verwendet.

3.1 Definition Schall und Schallentstehung

Im allgemeinen Sprachgebrauch bezeichnet man als Schall Töne, Klänge und Geräusche, die vom menschlichen Ohr wahrgenommen werden können, physikalisch betrachtet handelt es sich jedoch um Störungen bzw. Schwankungen, die sich wellenförmig in gasförmigen, flüssigen oder festen Medien ausbreiten können. Im Folgenden soll nun ausschließlich auf Schall und seine Ausbreitung in Gasen eingegangen werden, da nur dieser Fall für die vorliegende Arbeit von Relevanz ist.

3.1.1 Grundgrößen und Pegelmaße

Wie bereits erwähnt, handelt es sich bei Schall um mechanische Wellen, die in einem Medium transportiert werden, genauer um Druck- bzw. Dichteschwankungen gegenüber den statischen Werten, wie sie ohne Schall in dem entsprechenden Medium vorzufinden wären.

Aus diesem Grund werden folgende Schwankungsgrößen definiert:

Schalldruck p' in Pa

Schalldicht ρ' in kg/m^3

Schallschnelle \vec{v}' in m/s

Hier sei angemerkt, dass in weiterer Folge dort wo Missverständnisse auszuschließen sind, auf die Apostrophe in den oben genannten Größen verzichtet wird.

Nimmt man an, dass der Schalldruck - ein dem Gleichdruck des Mediums überlagerter Wechseldruck - eine harmonische Schwingung ist, dann kann man ihn folgendermaßen darstellen:

$$p' = \hat{p} \cos(\omega t + \varphi_P) \quad (3.1)$$

\hat{p} ist hier die Amplitude, ω die Winkelgeschwindigkeit und φ_P die Phasenverschiebung, außerdem kann die Druckschwankung auch komplex dargestellt werden, wobei bei einer kosinusförmigen Schwingung nur der Realteil und bei einer sinusförmigen Schwingung nur der Imaginärteil betrachtet wird.

$$\underline{p}' = \hat{p} e^{i(\omega t + \varphi_P)} \quad (3.2)$$

Analoges gilt für die Schallschnelle, die jene Geschwindigkeit darstellt, mit der die Teilchen des betrachteten Mediums um ihre Ruhelage schwingen

$$v' = \hat{v} \cos(\omega t + \varphi_v) \quad (3.3)$$

bzw. in komplexer Darstellung:

$$\underline{v}' = \hat{v} e^{i(\omega t + \varphi_v)} \quad (3.4)$$

Weitere in der Akustik verwendete Kenngrößen sind der Schallfluss

$$q = \vec{v} \cdot \vec{A} \quad (3.5)$$

der als das Produkt aus Schallschnelle und einer gerichteten Fläche berechnet wird, und die Schallintensität (Einheit: W/m^2 bzw. Nm/m^2s)

$$\vec{I} = p \cdot \vec{v}, \quad (3.6)$$

welche die Schallenergie darstellt, die in der Zeiteinheit durch eine Flächeneinheit strömt.

Aus ihr kann nun durch Integration die Schalleistung in Watt berechnet werden:

$$P = \int_A \vec{I} \cdot d\vec{A} \quad (3.7)$$

Aufgrund der extrem großen Wertebereiche aller angegebenen akustischen Kenngrößen, hat es sich als zweckmäßig erwiesen, diese nicht in ihren eigentlichen Einheiten darzustellen, sondern in Pegelgrößen umzurechnen und so durch Wahl einer logarithmischen Skala deutlich kleinere und besser darstellbare Bereiche zu erzielen. Außerdem erhält man dabei auch noch eine Skala, die etwa der menschlichen Hörempfindung entspricht. Man bezeichnet die logarithmierten Werte dann als Schalldruckpegel, Schallschnellepegel, Schallintensitätspegel und Schalleistungspegel, deren Einheit Dezibel (dB) ist.

Schalldruckpegel

$$L_p = 10 \cdot \log \left(\frac{p^2}{p_0^2} \right) = 20 \cdot \log \left(\frac{p}{p_0} \right) \quad (3.8)$$

$p_0 = 2 \cdot 10^{-5}$ Pa ist nicht nur der Bezugsdruck, sondern stellt auch die Hörschwelle dar.

Schallschnellepegel

$$L_v = 10 \cdot \log \left(\frac{v^2}{v_0^2} \right) = 20 \cdot \log \left(\frac{v}{v_0} \right) \quad (3.9)$$

mit $v_0 = \frac{p_0}{\rho c} = 5 \cdot 10^{-8}$ m/s für Luft. $\rho \cdot c$ beträgt in etwa $400 \frac{Ns}{m^3}$ bei 1013 mbar und $20^\circ C$ und stellt genau genommen nur einen gerundeten Wert dar.

Schallintensitätspegel

$$L_I = 10 \cdot \log \left(\frac{I}{I_0} \right) \quad (3.10)$$

Der Bezugswert für die Schallintensität berechnet sich aus $I_0 = p_0 \cdot v_0$ und hat den Wert $10^{12} \frac{W}{m^2}$.

Schalleistungspegel

$$L_W = 10 \cdot \log \left(\frac{P}{P_0} \right) \quad (3.11)$$

Bezugsleistung ist $P_0 = 10^{-12}$ W mit $P_0 = I_0 \cdot A_0$ und $A_0 = 1m^2$.

3.1.2 Schallentstehung

Grundsätzlich unterscheidet man vier Arten der Schallentstehung:

- mechanische Schallentstehung (durch Stöße, Reibung, Unwucht etc.)
- Schallentstehung durch Strömung, thermodynamische Schallentstehung (strömende Medien, Verbrennung etc.)
- elektrische Schallentstehung (z.B. in Elektromotoren, Generatoren, Transformatoren etc.)
- magnetische Schallentstehung (siehe elektrische Schallentstehung)

Speziell für den in dieser Arbeit untersuchten Fall - die Flugzeugturbine - sind jedoch nur die ersten beiden Punkte relevant. Wie groß der Einfluss der Triebwerke auf die gesamte Geräuschemission eines Flugzeuges ist, hängt in erster Linie von der Flugphase, das heißt dem Betriebspunkt, ab. Während bei der Landung die Umströmungsgeräusche am Fahrwerk, den Hochauftriebshilfen oder hervorgerufen durch Kavitäten am Rumpf und den Flügeln einen sehr großen Beitrag leisten, dominiert beim Start der Anteil der Triebwerke das Gesamtgeräusch.

Welchen Beitrag Turbine, Verdichter, Brennkammer und der Abgasstrahl zur gesamten Schallemission eines Triebwerks leisten, soll folgendes Bild sowohl für Einkreis- als auch für Zweikreis-Triebwerke veranschaulichen.

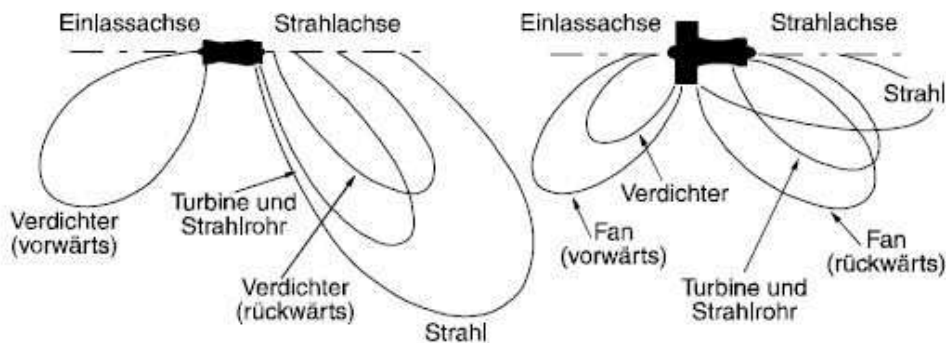


Abbildung 3.1: Schallemissionen von Einkreis- (links) und Zweikreis-Triebwerken (rechts) [1]

Auch wenn selbstverständlich sowohl alle Hauptkomponenten eines Triebwerks, als auch der Strahl maßgeblich das Gesamtgeräusch beeinflussen, so steht hier in weiterer Folge nur die Turbine, im Speziellen deren Niederdruckstufe, im Mittelpunkt der Betrachtungen. Ihre Geräuschemissionen setzen sich aus einem breitbandigen und einem tonalen Anteil zusammen, der durch diskrete Töne im Spektrum gekennzeichnet ist und außerdem in einer Turbine die wichtigere Rolle spielt. Ihren Ursprung haben die Emissionen in einer Turbine unter anderem in Schall, der durch fluktuierende Profildruckverteilungen bzw. schwankende Schaufelkräfte hervorgerufen wird. Diese wiederum haben ihre Ursache in ungleichförmiger Zu- und Abströmung aufgrund von Nachlaufzellen, wodurch es zur Bildung von Wirbeln und Turbulenzen kommt.

Eine weitere Lärmquelle stellt ein nicht konstanter Inzidenzwinkel dar, der sich aus dem Staffelungswinkel der Beschaukelung und dem effektiven Zuströmwinkel ergibt und somit maßgeblich für die Dicke des Profelnachlaufs verantwortlich ist. Wird nun das Triebwerk nicht am optimalen Betriebspunkt, also seinem Auslegungspunkt, betrieben, so weicht der Zuströmwinkel von seinem optimalen Wert ab und es kommt zur Bildung von dickeren Totwassergebieten und zur erhöhten Lärmgenerierung.

Weiters spielen Wechselwirkungen zwischen Schaufelnachläufen mit den nächstgelegenen Leit- und Laufschaufeln und Druckstörungen in Form von Verdichtungsstößen an der Beschaukelung eine maßgebliche Rolle bei der Entwicklung von unerwünschtem Schall. Bei Letzterem kommt es zu einer Störung des homogenen Strömungsfeldes, wie das auch bei einer im Strömungsfeld rotierenden Laufschaufel der Fall ist. Das Laufrad erzeugt dabei einen diskreten Ton bei der Frequenz, mit der die Schaufeln einen ruhenden Beobachter passieren und die von diesem Beobachter hinter dem Rotor wahrgenommen wird, auch Blade Passing Frequency (BPF) genannt. Diese Frequenz hängt also von der Drehzahl des Rotors und der Schaufelzahl ab und lässt sich folgendermaßen berechnen:

$$f = \frac{n}{60} \cdot i_s \quad (3.12)$$

Die Drehzahl in $\frac{U}{min}$ wurde hier mit n abgekürzt, i_s steht für die Schaufelzahl des Rotors und f ist die Grundfrequenz, die durch den Rotor angeregt wird. Im Spektrum überlagern sich die Grundfrequenz und ihre ganzzahligen Vielfachen dem Breitbandlärm in einem bestimmten, sehr begrenzten Frequenzbereich. Oft werden die höher Harmonischen wegen ihrer hohen Frequenzen von der Atmosphäre gedämpft oder liegen schon in Bereichen, die vom menschlichen Ohr nicht mehr wahrgenommen werden können. Zusätzlicher Lärm wird in einer Turbine, wie schon erwähnt, durch die Interaktionen zwischen Rotoren und Statoren, aber auch durch Wechselwirkungen zwischen zwei Rotoren generiert. Diese Phänomene sind jedoch derart komplex, dass sie nur noch schwer beschrieben werden können. Beispielsweise kann es durch Interferenzeffekte zu Auslöschungen bei bestimmten Frequenzen kommen, jedoch ist auch eine Verstärkung der Störungen möglich. Sichtbar werden diese Wechselwirkungen im Spektrum durch sogenannte Summentöne, die sich aus Grundtönen und höher Harmonischen zweier rotierender Bauteile zusammensetzen, jedoch können auch Differenztöne sichtbar werden. Diese sind typisch für den Fall, dass stehende und rotierende Bauteile in geringem axialen Abstand aufeinander folgen. Das Spektrum in Abbildung 3.2 zeigt sowohl Summen- als auch Differenztöne.

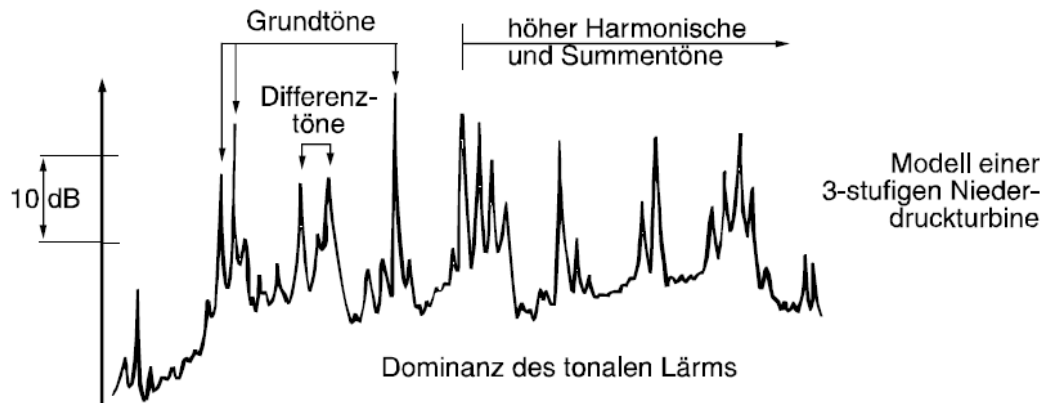


Abbildung 3.2: Spektrum einer dreistufigen Niederdruckturbinen [14]

Genauer zum Thema der Schallentwicklung in Flugzeugtriebwerken kann unter [1], [2] und [14] nachgelesen werden.

3.2 Schallausbreitung

Um die Phänomene der Ausbreitung des Schalls, der ja in einem der vorhergehende Kapitel als eine sich wellenförmig ausbreitende Störung definiert wurde, besser zu verstehen, wird im Folgenden zuerst ganz allgemein die Wellengleichung hergeleitet, um diese dann mit den entsprechenden Randbedingungen für den zylindrischen Strömungskanal zu lösen.

Ausgangspunkt für die Herleitung sind die Kontinuitätsgleichung, die Eulergleichung und eine Beziehung zwischen Druck und Dichte, die im allgemeinsten Fall $p = p(\rho)$ lautet.

Kontinuitätsgleichung:

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \vec{v}) = 0 \quad (3.13)$$

Eulergleichung:

$$\rho \frac{D\vec{v}}{Dt} + \nabla p = 0 \quad (3.14)$$

$\frac{D\vec{v}}{Dt}$ ist hier die substantielle Ableitung mit $\frac{D\vec{v}}{Dt} = \frac{\partial \vec{v}}{\partial t} + u \frac{\partial \vec{v}}{\partial x} + v \frac{\partial \vec{v}}{\partial y} + w \frac{\partial \vec{v}}{\partial z}$.

Da es sich, wie bereits erwähnt, bei Schall um eine Störung der Fluidbewegung handelt, können die akustischen Größen mittels des sogenannten Störungsansatzes, bei dem kleine von der Strömung unabhängige Schwankungen der Fluidbewegung überlagert sind, beschrieben werden, was eigentlich auch schon einer Linearisierung entspricht.

$$\rho = \rho_0 + \rho' \quad (3.15)$$

$$\vec{v} = \vec{v}_0 + \vec{v}' \quad (3.16)$$

$$p = p_0 + p' \quad (3.17)$$

ρ_0 , \vec{v}_0 und p_0 sind hier die Strömungsgrößen, wobei ein ruhendes Fluid angenommen wird, das heißt $\vec{v}_0 = 0$.

Diese Störungsansätze werden nun in die Kontinuitäts- bzw. in die Eulergleichung eingesetzt und man erhält folgende Ausdrücke:

$$\frac{\partial \rho'}{\partial t} + \nabla(\rho_0 \vec{v}_0 + \rho' \vec{v}_0 + \rho_0 \vec{v}' + \rho' \vec{v}') = 0 \quad (3.18)$$

$$\rho_0 \frac{\partial \vec{v}'}{\partial t} + \rho' \frac{\partial \vec{v}'}{\partial t} + (\rho_0 \vec{v}_0 + \rho' \vec{v}_0 + \rho_0 \vec{v}' + \rho' \vec{v}') \nabla(\vec{v}_0 + \vec{v}') + \nabla p' = 0 \quad (3.19)$$

Wie bereits diskutiert, sind die Störungsgrößen praktisch unabhängig von der Fluidbewegung, weshalb auch diese Linearisierung der Gleichungen 3.18 und 3.19 vorgenommen werden darf. Das bedeutet in diesem Fall, dass alle Terme, die sich aus mindestens zwei miteinander multiplizierten Störungsgrößen zusammensetzen, vernachlässigt werden dürfen, außerdem wird $\vec{v}_0 = 0$ eingesetzt. Daraus ergibt sich

$$\frac{\partial \rho'}{\partial t} + \nabla(\rho_0 \vec{v}') = 0 \quad (3.20)$$

$$\rho_0 \frac{\partial \vec{v}'}{\partial t} + \nabla p' = 0 \quad (3.21)$$

Das Stoffgesetz wird weiterhin ganz allgemein gehalten und in eine Taylorreihe entwickelt, die nach dem zweiten Glied abgebrochen wird.

$$p(\rho) = p(\rho_0) + (\rho - \rho_0) \frac{dp}{d\rho} + \dots \quad (3.22)$$

Es gilt jedoch $\rho - \rho_0 = \rho'$ und $p - p_0 = p'$ mit $p_0 = p(\rho_0)$, weiters ist $\frac{dp}{d\rho}$ unter Annahme konstanter Entropie gleich dem Quadrat der Schallgeschwindigkeit c^2 . Daraus folgt die linearisierte Stoffbeziehung

$$p' = \rho' \frac{dp}{d\rho} = \rho' c^2 \quad (3.23)$$

Es sei an dieser Stelle angemerkt, dass für die Schallgeschwindigkeit eigentlich der Wert im Ausgangszustand (c_0) herangezogen werden müsste, außerdem stellen die linearisierten Gleichungen nur unter bestimmten Voraussetzungen eine gute Annäherung an die eigentlich nichtlinearen Zusammenhänge dar.

Formt man nun Gleichung 3.23 nach ρ' um und setzt dies in Gleichung 3.20 ein, so erhält man

$$\frac{1}{c^2} \frac{\partial p'}{\partial t} + \nabla(\rho_0 \vec{v}') = 0 \quad (3.24)$$

Einmal nach der Zeit abgeleitet ergibt sich:

$$\frac{1}{c^2} \frac{\partial^2 p'}{\partial t^2} + \nabla(\rho_0 \frac{\partial \vec{v}'}{\partial t}) = 0 \quad (3.25)$$

Nach Umformen der Gleichung 3.21

$$\rho_0 \frac{\partial \vec{v}'}{\partial t} = -\nabla p' \quad (3.26)$$

und Einsetzen in Gleichung 3.25 folgt

$$\frac{1}{c^2} \frac{\partial^2 p'}{\partial t^2} - \nabla \nabla p' = 0 \quad (3.27)$$

Die Wellengleichung für den Schalldruck lautet also

$$\frac{1}{c^2} \frac{\partial^2 p'}{\partial t^2} - \Delta p' = 0 \quad (3.28)$$

mit $\Delta p = \frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2}$.

Auf ähnliche Art und Weise lässt sich eine entsprechende Wellengleichung sowohl für die Dichteänderung, als auch für das Geschwindigkeitsfeld herleiten.

3.2.1 Lösung der Wellengleichung für zylindrische Rohre

In weiterer Folge wird zum leichteren Verständnis die Schallausbreitung im Eindimensionalen betrachtet. Man geht nun also von einer ebenen Welle, das heißt alle akustischen Größen hängen nurmehr von einer einzigen Variablen ab, aus und die Wellengleichung lässt sich vereinfachen und folgendermaßen anschreiben:

$$\frac{1}{c^2} \frac{\partial^2 p'}{\partial t^2} = \frac{\partial^2 p'}{\partial x^2} \quad (3.29)$$

Ausbreitungsrichtung ist hier die x-Richtung, das bedeutet alle durch die Schallwelle verursachten Bewegungen erfolgen in dieser Koordinatenrichtung.

Nach d'Alembert lautet die Lösung dieser eindimensionalen Differentialgleichung

$$p' = F(ct - x) + G(ct + x) \quad (3.30)$$

F und G sind zwei beliebige Funktionen, die mindestens zweimal ableitbar sein müssen. G beschreibt dabei eine nach links und F eine nach recht laufende Welle. Das bedeutet, F ist eine Welle, die sich auf der Kurve $ct - x = \textit{konstant}$ unverzerrt fortpflanzt, während dies für die Welle G auf der Kurve $ct + x = \textit{konstant}$ der Fall ist. c ist die Geschwindigkeit, mit der sich die Änderung der betreffenden Schallgröße ausbreitet. Die sich ausbreitenden Wellen werden zwar als eben bezeichnet, was jedoch lediglich bedeutet, dass die Wellenfronten ebene Flächen sind, und nicht, dass es sich dabei um eine Lösung im Zweidimensionalen handelt. Die Lösung der Wellengleichung lässt sich nicht nur für den Schalldruck angeben, sondern kann mithilfe der Kontinuitätsgleichung und dem Zusammenhang 3.23 auch für die Dichte und die Schallschnelle ermittelt werden. Dies soll hier jedoch nicht weiter ausgeführt werden, kann aber beispielsweise in [2] und [12] nachgelesen werden.

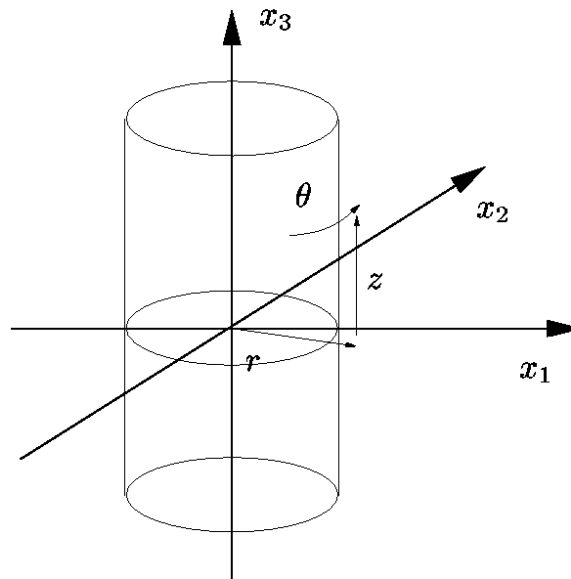


Abbildung 3.3: Zylinderkoordinaten [29]

Um die Wellengleichung für zylindrische Strömungskanäle zu lösen, ist es zweckmäßig, die Gleichung zuerst in Zylinderkoordinaten darzustellen, die in Abbildung 3.3 noch einmal veranschaulicht wurden.

$$\frac{1}{c^2} \frac{\partial^2 p'}{\partial t^2} - \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial p'}{\partial r} \right) - \frac{1}{r^2} \frac{\partial^2 p'}{\partial \theta^2} - \frac{\partial^2 p'}{\partial z^2} = 0 \quad (3.31)$$

Als Lösungsansatz wählt man hier einen sogenannten Separationsansatz mit harmonischer Zeitabhängigkeit

$$p' = f(z)g(r)h(\theta)e^{i\omega t} \quad (3.32)$$

Nach Ableiten und Einsetzen in die Wellengleichung ergibt sich

$$\left(\frac{\omega}{c}\right)^2 + \frac{1}{f} \frac{d^2 f}{dz^2} + \frac{1}{g} \left(\frac{1}{r} \frac{d}{dr} \left(r \frac{dg}{dr} \right) \right) + \frac{1}{h} \frac{1}{r^2} \frac{d^2 h}{d\theta^2} = 0 \quad (3.33)$$

Um $f(z)$, $g(r)$ und $h(\theta)$ zu bestimmen, werden jeweils Terme mit gleichen Variablen auf eine Seite der Gleichung gebracht. Damit die Gleichheit beider Seiten für alle Kombinationen von r , z und θ gegeben ist, bleibt nur eine Möglichkeit, und zwar, dass beide Seiten konstant sein müssen. Daraus folgt

$$\frac{1}{h} \frac{d^2 h}{d\theta^2} = -\sigma^2 \quad (3.34)$$

$$\frac{1}{f} \frac{d^2 f}{dz^2} = -\alpha^2 \quad (3.35)$$

Ein Problem stellen nur die Terme dar, die die Variable r beinhalten, da sie nicht vom Rest der Gleichung isoliert werden können. Das bedeutet, dass $g(r)$ nur dann bestimmt werden kann, wenn bereits eine der oben verwendeten Konstanten, σ oder α , bekannt ist. Um zuerst θ zu eliminieren, könnte man beispielsweise σ als gegeben annehmen und anschließend alle Terme, die von r abhängen auf eine Seite bringen.

$$\frac{1}{h} \frac{1}{r^2} \frac{d^2 h}{d\theta^2} = -\frac{\sigma^2}{r^2} \quad (3.36)$$

$$\frac{1}{g} \left(\frac{1}{r} \frac{d}{dr} \left(r \frac{dg}{dr} \right) \right) - \frac{\sigma^2}{r^2} = -\frac{1}{f} \frac{d^2 f}{dz^2} - \left(\frac{\omega}{c}\right)^2 \quad (3.37)$$

Die andere Seite der Gleichung 3.37 hängt dann nurmehr von z ab und muss, damit die Differentialgleichung erfüllt ist, ebenfalls konstant sein. Diese Konstante wird hier mit β bezeichnet.

$$\frac{1}{g} \left(\frac{1}{r} \frac{d}{dr} \left(r \frac{dg}{dr} \right) \right) - \frac{\sigma^2}{r^2} = -\beta^2 \quad (3.38)$$

Aus den Gleichungen 3.38 und 3.35 folgt dann

$$\left(\frac{\omega}{c}\right)^2 = \alpha^2 + \beta^2 \quad (3.39)$$

Das bedeutet also, wenn für $h(\theta)$ eine Lösung gefunden wurde, kann daraus die Konstante σ berechnet werden. Wenn man diese kennt, können in weiterer Folge auch $g(r)$ und β ermittelt werden. Aus der Beziehung 3.35 ergibt sich dann letztendlich auch α . Zur Bestimmung von $h(\theta)$ wählt man den Ansatz $h(\theta) = A_3 e^{-i\sigma\theta} + B_3 e^{i\sigma\theta}$, wobei die Konstanten A_3, B_3 und σ so gewählt werden müssen, dass die Randbedingungen erfüllt sind. Für θ werden als Ränder $\theta = 0$ und $\theta = 2\pi$ festgelegt, wo genau und in welche Richtung man $\theta = 0$ festlegt, sollte jedoch keinen Einfluss auf die Lösung der Wellengleichung haben. Um dies zu garantieren, wählt man eine periodische Randbedingung $h(\theta) = h(\theta + 2\pi)$. Diese Randbedingung wird erfüllt, wenn $\sigma = \sigma_m = m = 0, 1, 2, \dots$ ist, was bedeutet, dass es für diese Konstante bei der periodischen Lösung, die man aufgrund des Ansatzes für $h(\theta)$ erhält, keinen eindeutigen Wert gibt. Dasselbe gilt auch für die Konstanten A_3 und B_3 , die wegen der periodischen Randbedingung ebenfalls unabhängig voneinander frei gewählt werden. Für die Berechnung von $g(r)$ wird in der Gleichung 3.38 $\beta r = s$, $\sigma = \nu$ und $g = a(s) = s(\beta r)$ substituiert und man erhält die sogenannte Bessel'sche Differentialgleichung

$$s^2 \frac{d^2 a}{ds^2} + s \frac{da}{ds} + (s^2 - \nu^2) a = 0 \quad (3.40)$$

deren Lösung mit

$$a = A J_\nu + B N_\nu \quad (3.41)$$

angegeben werden kann. J_ν ist hier die Bessel-Funktion und N_ν die Neumann-Funktion. Für ν gilt: $\nu = \sigma = m$.

Nach Rücksubstitution ergibt sich schließlich

$$g(r) = A_2 J_m(\beta r) + B_2 N_m(\beta r) \quad (3.42)$$

Um die Konstanten zu bestimmen, verwendet man einerseits die Randbedingungen einer schallharten Rohrwand, andererseits muss eine zweite Randbedingung bei $r=0$ erfüllt werden, da sich ansonsten bei $r=0$ nach dem Ansatz eine Singularität ergeben würde. Aus diesem Grund legt man formal fest, dass die Lösung bei $r=0$ endlich sein muss, was wiederum bedeutet, dass der Faktor vor der Neumann-Funktion Null sein muss und sich daher für $g(r)$ folgende Lösung ergibt:

$$g(r) = A_2 J_m(\beta r) \quad (3.43)$$

β dient hier als Skalierungsfaktor und legt den Klammerausdruck derart fest, dass die Funktion $J_m(\beta r)$ genau bei $r=R$ ein Maximum oder ein Minimum aufweist. Betrachtet man die Bessel-Funktion in Abbildung 3.4, so erkennt man, dass

diese unendlich viele Extrema besitzt und somit auch für β unendlich viele Werte gewählt werden können. Es wird nun eine zweite Ordnungszahl n eingeführt und s_{mn} bezeichnet somit die Stelle des n -ten lokalen Extremums der Bessel-Funktion m -ter Ordnung.

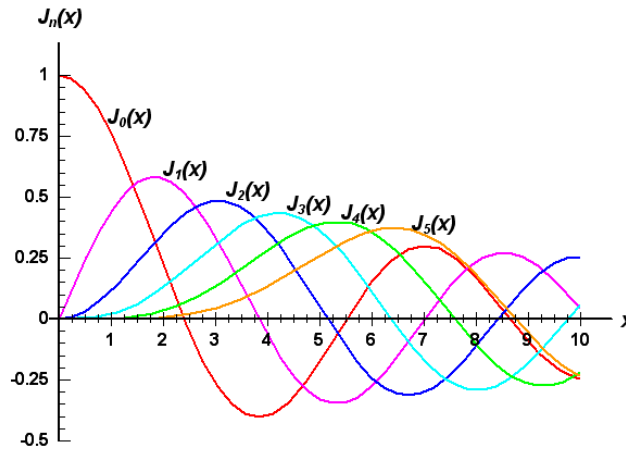


Abbildung 3.4: Besselfunktion [23]

Mit $\beta = \frac{s_{mn}}{R} = \beta_{mn}$ ergibt sich Gleichung 3.43 somit zu

$$g(r) = A_2 J_m(\beta_{mn} r) \quad (3.44)$$

Analog zum Lösungsansatz für $h(\theta)$ folgt für $f(z)$ mit $\alpha = \sqrt{\left(\frac{\omega}{c}\right)^2 - \beta_{mn}^2} = \alpha_{mn}$:

$$f(z) = A_1 e^{-i\alpha z} + B_1 e^{i\alpha z} \quad (3.45)$$

Für den Schalldruck ergibt sich dann schlussendlich folgender Ausdruck:

$$p'(z, r, \theta, t) = (A_3 e^{-im\theta} + B_3 e^{im\theta}) A_2 J_m(\beta_{mn} r) (A_1 e^{-i\alpha_{mn} z} + B_1 e^{i\alpha_{mn} z}) e^{i\omega t} \quad (3.46)$$

Abbildung 3.5 zeigt die momentane Druckverteilung im Querschnitt des zylindrischen Kanals. Die Ordnungszahl n steht hier für die Anzahl der Knotenringe um die Mitte, während m die Anzahl der Perioden in Umfangsrichtung angibt.

Wichtig ist, dass sich eine Welle im zylindrischen Strömungskanal nur dann in axialer Richtung ausbreiten kann, wenn eine bestimmte cut-on Bedingung erfüllt ist. In diesem Fall muss α_{mn} rein reell sein, damit eine Ausbreitung möglich ist.

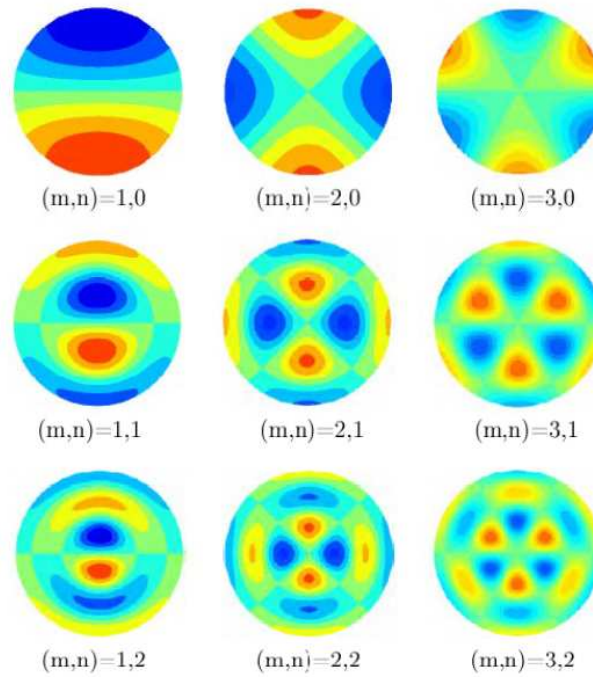


Abbildung 3.5: Druckverteilung im Querschnitt eines zylindrischen Rohres [2]

Wird bei einem konstanten ω die Modenordnung m immer weiter erhöht, so wird der Ausdruck unter der Wurzel bei der Berechnung von α_{mn} irgendwann negativ und α_{mn} somit imaginär. Das bedeutet, es ist keine reguläre Wellenausbreitung mehr möglich und die Lösung klingt entweder ab oder wächst an. Es gilt also

$$\frac{\omega}{c} > \beta_{mn} = \frac{s_{mn}}{R} \quad (3.47)$$

Die cut-off Frequenz, also die Grenzfrequenz, ab der eine Ausbreitung der Welle möglich ist, lautet daher $\omega_{C,mn} = c \frac{s_{mn}}{R}$ bzw. ergibt sich nach dem Umformen $\omega_{C,mn} R > c s_{mn}$ auf der rechten Seite dieser Ungleichung eine Geschwindigkeit, die genau der Umlaufgeschwindigkeit eines Knotenpunkts an der Wand für $m=1$ ("Spinning Mode") entspricht. Da sich das Druckfeld im Querschnitt des Strömungskanals während einer Periode um 360° dreht, legt der Knotenpunkt an der Wand eine Strecke von $2\pi R$ zurück und zwar mit einer Bahngeschwindigkeit von ωR . Laut cut-off Bedingung muss die Geschwindigkeit des Knotens für $m=1$ also größer als $c s_{1n}$ sein, damit sich die entsprechende Mode überhaupt ausbreiten kann.

3.2.2 Die cut-off Bedingung für den dünnen ringförmigen Kanal

Geht man davon aus, dass der Umfang deutlich größer als die Differenz von äußerem und innerem Radius ist, so spricht man von einem dünnen Strömungskanal wie er in folgender Abbildung dargestellt ist (entnommen wurde dieser Abschnitt aus [1]).

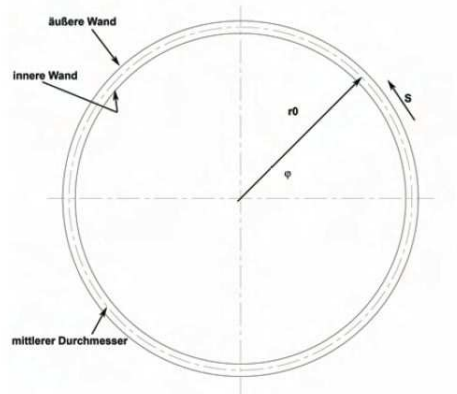


Abbildung 3.6: Dünnere ringförmiger Kanal [1]

In diesem Kanal gilt eine periodische Randbedingung, und zwar muss sowohl der Druck, als auch der Druckgradient an der Stelle $s = 0$ und $s = 2\pi r_0$ (Umfang) gleich sein.

$$p(s) = p(s + 2\pi r_0) \quad (3.48)$$

$$\left. \frac{dp}{ds} \right|_s = \left. \frac{dp}{ds} \right|_{s+2\pi r_0} \quad (3.49)$$

Geht man nun weiters davon aus, dass die Krümmung des Kanals sehr klein ist, so weist das Schallfeld darin ein ähnliches Verhalten auf wie in einem dünnen rechteckigen Kanal, in dem für die cut-off Frequenz gilt:

$$f_c = \frac{c}{\lambda(s)} \quad (3.50)$$

Mit der Wellenlänge in azimuthaler Richtung für den ringförmigen Kanal $\lambda(s) = \frac{2\pi r_0}{m}$ (m ist die Modenordnung) ergibt sich daher die Grenzfrequenz für die azimuthale Schwingung zu

$$f_c = m \frac{c}{2\pi r_0} \quad (3.51)$$

3.2.3 Wellenausbreitung bei überlagerter Strömung

Im vorhergehenden Abschnitt wurde angenommen, dass die Strömungsgeschwindigkeit des Fluids gleich Null ist, um basierend auf dieser Annahme die Wellengleichung herzuleiten. Für die Behandlung der Wellenausbreitung bei überlagerter Strömung gibt es nun zwei Möglichkeiten: Entweder die Wellengleichung wird um die Strömungsgeschwindigkeit des Fluids erweitert und anschließend gelöst, oder man betrachtet die Situation zuerst in einem mit der Strömung mitbewegten Bezugssystem und transformiert die Ergebnisse dann auf ein ruhendes Koordinatensystem. Letzteres soll nun unter Voraussetzung konstanter Strömungsgeschwindigkeit und unter Vernachlässigung von Reibungseffekten und Haften des Fluids an der Kanalwand für eindimensionale Schallwellen gezeigt werden.

Im mitbewegten Koordinatensystem ist die Geschwindigkeit des Fluids gleich Null, daher gilt die Wellengleichung in der bereits gezeigten Form (Gleichung 3.28). Mit dem Index B sollen nun im Folgenden alle Größen im bewegten System, mit R alle Größen im ruhenden System gekennzeichnet werden. Die Schwankungsgrößen p' , ρ' und u' bleiben ohne Index, da sie unabhängig vom Bezugssystem sind. Für die Geschwindigkeiten gilt somit

$$u_B = u' \quad (3.52)$$

$$u_R = u_0 + u' \quad (3.53)$$

Für die Transformation zwischen den Koordinatensystemen werden folgende Zusammenhänge verwendet:

$$x_R = x_B + u_0 t \quad (3.54)$$

$$x_B = x_R - u_0 t \quad (3.55)$$

Die Lösung im mitbewegten System ergibt sich laut Gleichung 3.30 zu

$$p'_B(x_B, t) = F(x_B - ct) + G(x_B + ct) \quad (3.56)$$

Da p' unabhängig vom Koordinatensystem ist, kann auch geschrieben werden:

$$p'_B(x_B, t) = p'_R(x_R, t) = p'_B(x_R - u_0 t, t) \quad (3.57)$$

Damit ergibt sich für das ruhende System:

$$p'_R(x_R, t) = F(x_R - (c + u_0)t) + G(x_B + (c + u_0)t) \quad (3.58)$$

In diesem Fall beschreibt die Funktion F eine Welle, die sich mit der Geschwindigkeit $c - u_0$ in positive x -Richtung ausbreitet und G eine, die sich mit der Geschwindigkeit $c + u_0$ in negative x -Richtung ausbreitet.

Da $p'_B(x_B, t) = p'_B(x_R - u_0 t, t)$ gilt unter der Verwendung der Wellenzahl $k = \frac{\omega}{c} = \frac{2\pi}{\lambda}$, die ebenfalls unabhängig vom Bezugssystem ist:

$$kx_B + \omega_B t = -k(x_B + u_0 t) - \omega_R t \quad (3.59)$$

Nach Einführung der Machzahl $Ma = \frac{u_0}{c}$ folgt aus (3.59) der Zusammenhang zwischen der Frequenz im ruhenden und im mitbewegten System, wobei die Frequenz im ruhenden Bezugssystem im Vergleich zu der im bewegten um den sogenannten Dopplereffektor $(1 + Ma)$ verschoben ist.

$$\omega_R = \omega_B(1 + ku_0) = \omega_B(1 + Ma) \quad (3.60)$$

Weiteres zum Thema Schallausbreitung in Kanälen ist in [24] zu finden.

3.3 Problem des erweiterten Kanals

Bisher wurde immer von einem zylindrischen Kanal mit konstantem Querschnitt ausgegangen, was jedoch gerade in Turbomaschinen nicht immer der Realität entspricht. Aus diesem Grund soll hier auch die Berechnung eines Kanals mit veränderlichem Querschnitt, wie er in Abbildung 3.7 dargestellt ist, gezeigt werden.

Es wird angenommen, dass die Länge der Schallwellen sehr viel größer als die Abmessungen des Rohres ist, also $\lambda \gg d_1$ und $\lambda \gg d_2$, und damit die Frequenz des Schalls deutlich unter den cut-off Frequenzen liegt. Das bedeutet, dass sich nur ebene Wellen im Rohr ausbreiten. Weiters wird angenommen, dass der Übergangsbereich zwischen den verschiedenen Querschnitten sehr viel kürzer als die Wellenlänge ist, das heißt $\lambda \gg l$. Nach dem Eintreten in das Rohr wird die Schallwelle sowohl transmittiert als auch reflektiert, außerdem kommt es zu

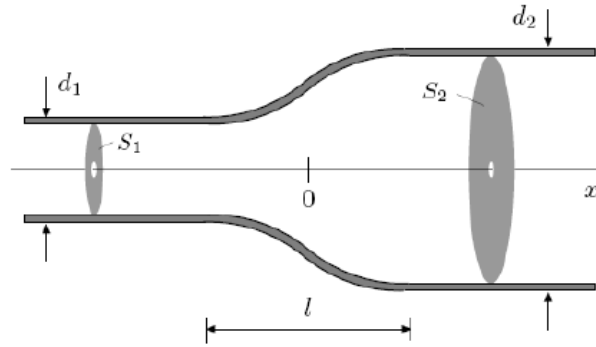


Abbildung 3.7: Strömungskanal mit veränderlichem Querschnitt [2]

überaus komplexen Vorgängen im Übergangsbereich. Das bedeutet, im Übergangskanal werden durch die Reflexion höhere Moden angeregt, die jedoch nicht ausbreitungsfähig sind, des Weiteren spielen in diesem Bereich dreidimensionale Effekte eine wichtige Rolle.

Wie in Abbildung 3.7 eingezeichnet, wird eine neue x-Koordinate eingeführt, so dass gilt:

$$p'(x, t) = \begin{cases} p'_1(x, t) & \text{bei } x < 0 \\ p'_2(x, t) & \text{bei } x > 0 \end{cases} \quad (3.61)$$

Es existiert folgende bekannte Lösung der Wellengleichung für ebene Wellen bei harmonischer Störung und für den Fall, dass sich die Welle 1 in beide und die Welle 2 nur in die positive x-Richtung ausbreitet:

$$p'_1(x, t) = A_1 e^{i(\omega t - kx)} + B_1 e^{i(\omega t + kx)} \quad (3.62)$$

$$p'_2(x, t) = A_2 e^{i(\omega t - kx)} \quad (3.63)$$

k ist in den obigen Gleichungen die Wellenzahl ($k = \frac{\omega}{c}$).

Wenn die eintreffende Schallwelle bekannt bzw. vorgegeben ist, so kennt man auch die Konstante A_1 , A_2 und B_1 müssen jedoch durch vereinfachende Annahmen aus der Energie- und der Bilanzgleichung bestimmt werden, was im nachfolgenden Abschnitt kurz erklärt wird.

Ein großes Problem stellt beim Rohr mit Querschnittsänderung, wie bereits weiter oben erwähnt, der Übergangskanal aufgrund des dreidimensionalen Druck- und Schnellefeldes in diesem Bereich dar. Um die Verhältnisse in diesem Abschnitt

des Kanals wiederzugeben, wird zuerst ein Kontrollvolumen eingeführt wie es in Abbildung 3.8 ersichtlich ist.

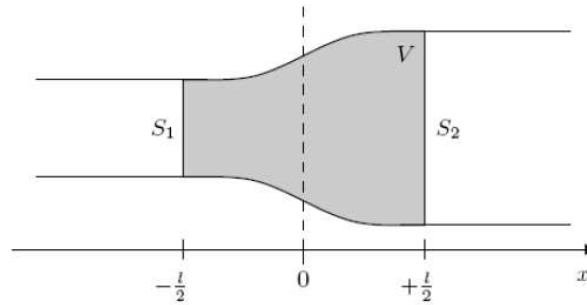


Abbildung 3.8: Kontrollvolumen im Übergangsbereich des Kanals [2]

Angenommen wird hier, dass außerhalb des Kontrollvolumens und an seinen Rändern an den Stellen $+\frac{l}{2}$ und $-\frac{l}{2}$ das Schallfeld aus ebenen Wellen besteht. Selbstverständlich muss hier auch die Massenerhaltung erfüllt sein, was durch folgende Gleichung, die die zeitliche Änderung der Masse, die sich im Kontrollvolumen befindet, wiedergegeben wird.

$$\frac{\Delta m}{\Delta t} = \rho_0 u'_1 S_1 - \rho_0 u'_2 S_2 \quad (3.64)$$

Dieser Ausdruck ist jedoch schon so komplex, dass die Massenänderung im Übergangsbereich nurmehr numerisch berechnet werden kann. Zu diesem Zweck sind zahlreiche Annahmen und Vereinfachungen notwendig, die wiederum die Genauigkeit des Ergebnisses beeinflussen. So kann zum Beispiel angenommen werden, dass $u'_1(0, t) = u'_1(-\frac{l}{2}, t)$ und $u'_2(0, t) = u'_2(+\frac{l}{2}, t)$. Es wird außerdem vereinfachend davon ausgegangen, dass Druck und Dichte im Übergangskanal annähernd konstant sind und auch nur zeitliche und keine räumlichen Schwankungen der Strömungsgrößen auftreten. Eine weitere Vereinfachung gilt für den Druck im Kontrollvolumen, nämlich $p'_1(0, t) = p'_2(0, t)$.

Anhand dieser Annahmen lassen sich die Massen- und die Energiebilanz ebenfalls vereinfacht darstellen, um aus diesen dann letztendlich die beiden unbekanntenen Konstanten A_2 und B_1 zu berechnen. In weiterer Folge lassen sich auch der Reflexions- und der Transmissionsfaktor ermitteln, die angeben wie groß der Anteil an reflektiertem bzw. transmittiertem Schall im Strömungskanal ist.

Basierend auf den obigen Ausführungen müssten genau genommen alle Übergangsbereiche in einem Strömungskanal, wo eine Querschnittsänderung stattfindet, gesondert betrachtet, der tatsächliche Zustand in diesem Gebiet durch bestimmte Annahmen vereinfacht und anschließend numerisch berechnet werden. Auch vom Standpunkt der Messtechnik für Akustikmessungen aus betrachtet,

stellt dieser Abschnitt im Kanal mit Sicherheit eine Herausforderung dar bzw. besteht hier bestimmt noch großes Verbesserungspotential.

3.4 Problem des Druckgradienten und der drallbehafteten Abströmung

Wie in Kapitel 3.2 beschrieben, wurde bisher die Wellengleichung ohne Berücksichtigung viskoser Reibungsterme behandelt. Werden diese nun in die Betrachtungen mit einbezogen, erhält man aus den Navier-Stoke'schen Gleichungen nach Einführung der Abkürzung $\phi = \frac{\partial v_r}{\partial r} + \frac{1}{r} \frac{\partial v_\theta}{\partial \theta} + \frac{\partial v_z}{\partial z}$ und unter Verwendung der Machzahl folgende Form der Wellengleichung:

$$\begin{aligned} \frac{1}{c^2} \frac{\partial^2 p}{\partial t^2} + M^2 \frac{\partial^2 p}{\partial z^2} + 2 \frac{M}{c} \frac{\partial^2 p}{\partial z \partial t} - 2c \frac{\partial M}{\partial r} \frac{\partial v_r}{\partial z} = \Delta p - \eta_0 a c \frac{\partial p}{\partial z} \left(\frac{1}{p_0} - \frac{1}{\rho_0 c} \right) \\ \left(\frac{\partial^2 M}{\partial r^2} + r \frac{\partial M}{\partial r} \right) + \eta_0 \left(\frac{2}{r^2} \frac{\partial v_r}{\partial r} - \frac{1}{r^3} v_\theta \theta + \frac{2}{r^2} \frac{\partial^2 v_\theta}{\partial \theta \partial r} \right) \end{aligned} \quad (3.65)$$

Man erkennt, dass auf der rechten Seite sowohl der Druckgradient $\frac{\partial p}{\partial z}$, als auch Ableitungen der Machzahl und des radialen Schallschnelleprofils nach dem Radius enthalten sind und daher das radiale Strömungsprofil, entgegen der bisher getroffenen Annahmen, einen Einfluss auf das Ergebnis hat.

Erschwert wird diese Vorgehensweise nicht nur dadurch, dass auf der rechten Seite kompliziert zu bestimmende Ableitungen der azimuthalen und radialen Schallschnelle auftreten, sondern auch die unbekannte thermische Wärmeleitfähigkeit a und die dynamische Viskosität η_0 stellen bei der Berechnung große Probleme dar. Ganz allgemein lässt sich sagen, dass aufgrund der fehlenden Vorgaben für die viskosen Scherterme deren Behandlung sehr kritisch ist und daher eine reibungsfreie Betrachtung vorgezogen wird.

Damit man also die Wellengleichung wie bereits in einem der vorhergehenden Kapitel beschrieben, lösen kann, wird die rechte Seite des Ausdrucks 3.65 zu Null gesetzt und somit die viskose Reibung vernachlässigt. Dies bedeutet in weiterer Folge jedoch, dass auch das Vorhandensein von Grenzschichten im Strömungskanal gänzlich außer Acht gelassen wird, denn ohne Reibung auch kein fluiddynamisches Grenzschichtprofil.

Genau genommen ist diese Einschränkung für Akustikmessungen aber nur dann korrekt, wenn die Grenzschichtdicke δ deutlich kleiner als die momentan betrachtete Wellenlänge des Schalls ist.

$$\delta \ll \lambda \tag{3.66}$$

Möchte man trotz allem eine Grenzschicht in die Betrachtungen miteinbeziehen, so gibt es dafür spezielle Ansätze, die beispielsweise in der Arbeit [33], auf der auch diese Zusammenfassung beruht, im Detail beschrieben werden.

Eine ähnliche Problemstellung ergibt sich bei der Betrachtung einer drallbehalteten Abströmung, da diese nur dann zustande kommt, wenn viskose Reibung und damit auch eine Grenzschicht mit der Dicke δ vorhanden ist. Das wiederum bedeutet, dass die Lösung des vorliegenden Problems, wenn überhaupt möglich, bei weitem komplizierter als die bisher durchgeführten Betrachtungen wird.

3.5 Modenordnungen nach Tyler und Sofrin

Basierend auf der Arbeit von Tyler und Sofrin (siehe dazu [25], [2] und [1]) wird in diesem Kapitel näher auf die Interaktionen zwischen Stator und Laufrad und die dabei angeregten Moden eingegangen.

Das rotierende Druckfeld eines Rotors wirkt sich sowohl auf die stromab als auch die stromauf stehenden Statorschaufeln aus. Das Laufrad trennt nämlich einerseits die Nachläufe der davor liegenden Statorschaufeln, andererseits schlagen die Nachläufe des Rotors auf die Vorderkanten der nachfolgenden Statorschaufeln auf. Aufgrund der Nachläufe kommt es zu Geschwindigkeitsschwankungen, die wiederum zu einer periodischen Änderung der Statorzuströmung führen. Zusätzlich zu den genannten Phänomenen verschiebt sich der Staupunkt an der Vorderkante der Schaufeln, wodurch sich eine Druckstörung mit einem zeitlichen Versatz zwischen jeweils zwei Schaufeln, die sich im Idealfall kreisförmig ausbreitet, bildet. Dies führt zu einem in Umfangsrichtung rotierenden Druckmuster, wobei sich die Druckstörungen der einzelnen Schaufelinteraktionen zu einer Wellenfront überlagern und ausbreiten, vorausgesetzt natürlich, dass die Geschwindigkeit der Störung in Umfangsrichtung größer ist als die Ausbreitungsgeschwindigkeit der Druckstörung selbst (cut-on, siehe dazu auch Kapitel 3.2).

Die Umfangsmodenordnungen lassen sich mit V als Anzahl der Statorschaufeln und B als Anzahl der Rotorschaufeln nach Gleichung 3.67 berechnen. k steht hier für eine positive oder negative ganze Zahl und h für den sogenannten harmonischen Index ($h=1,2,3\dots$).

$$m = h \cdot B + k \cdot V \tag{3.67}$$

Ergibt sich für eine Mode m ein negatives Vorzeichen, so bedeutet dies, dass

sich diese entgegengesetzt der Rotordrehrichtung ausbreitet. Oben stehender Aus-
druck kann auch für mehrere Statoren und Rotoren erweitert werden.

Für die Berechnung der Drehgeschwindigkeit des oben genannten, auf den In-
teraktionen zwischen Stator und Rotor basierenden Druckmusters, wird folgende
Formel angewendet:

$$\Omega_{Muster} = \Omega \frac{h \cdot B}{m} = \Omega \frac{h \cdot B}{h \cdot B + k \cdot V} \quad (3.68)$$

Zum besseren Verständnis der Rotor-Stator-Interaktionen soll nun ein Beispiel
gebracht werden.

In der nachfolgenden Abbildung wird eine Kombination aus sechs Stator- und
acht Rotorschaukeln dargestellt, wobei die schwarzen Pfeile für den Stator und
die blauen Pfeile für den Rotor stehen. Dort wo es zu Interaktionen zwischen den
beiden Komponenten kommt, sind rote Pfeile eingezeichnet. Eine der Rotorschau-
keln wurde mit einer Markierung versehen, um die Drehrichtung in den Bildern
in Abbildung 3.10 zu verdeutlichen.

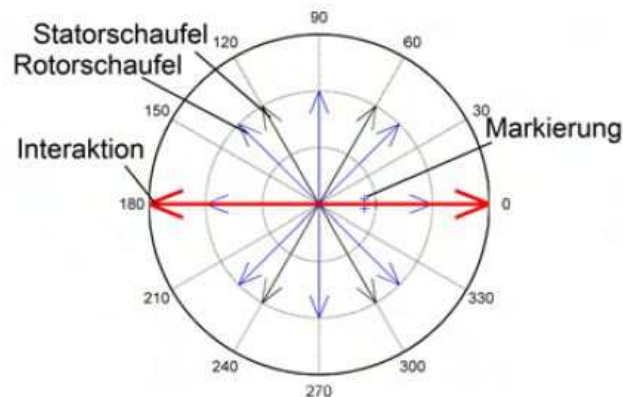


Abbildung 3.9: Schematische Darstellung von Rotor und Stator [1]

Die Ordnung der niedrigsten Schwingungsmode kann man nun mittels Gleichung
3.67 zu $m = 8 - 6 = 2$ berechnen und die Drehgeschwindigkeit des Druckmusters
ergibt sich wegen Gleichung 3.68 zu $\Omega_{Muster} = \Omega \frac{8}{8-6} = \Omega \frac{8}{2} = 4\Omega$. Das bedeutet,
dass die Interaktion vier mal so schnell rotiert wie das Laufrad und aufgrund des
positiven Vorzeichens auch denselben Drehsinn hat.

Abbildung 3.10 zeigt ein Laufrad, das sich gegen den Uhrzeigersinn dreht, wo-
durch es zu Interaktionen zwischen Rotor und Stator kommt. Diese liegen genau
dort, wo sich eine Stator- und eine Rotorschaukel in der Skizze überlagern und
sind, wie bereits erwähnt, mit einem roten Pfeil markiert. Bewegt sich nun der
Rotor aus der Position (a) nur um ein kleines Stück weiter, dann finden so lan-

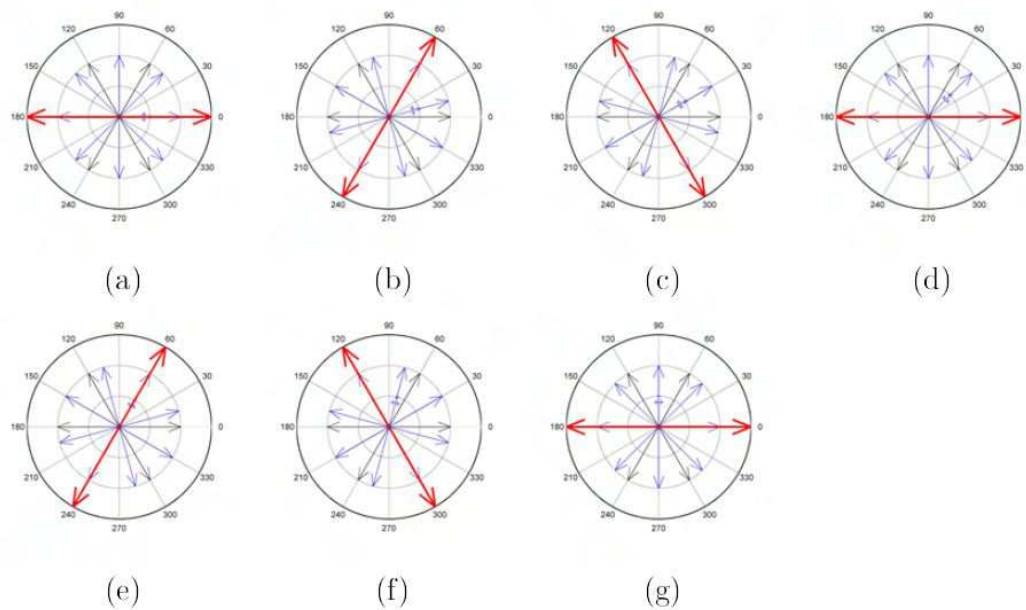


Abbildung 3.10: Schematische Darstellung der Rotor-Stator-Interaktionen [1]

ge keine Interaktionen statt, bis der Rotor Position (b) erreicht hat. Dies gilt selbstverständlich bis zum Bild (g).

3.6 Mathematische Grundlagen der Fast Fourier Transformation (FFT)

Die Fourier Transformation im Allgemeinen basiert auf der Tatsache, dass Prozesse und Signale sowohl im Zeit- als auch im Frequenzbereich dargestellt werden können. Die mathematischen Grundlagen der Fourier Transformation sollen nun, aufbauend auf den Inhalten von [25] und [26], kurz erläutert werden, im Speziellen werden hier diskrete Folgen im Zeit- und Frequenzbereich miteinander verknüpft, da dies auch für die vorliegenden Akustikmessungen dem konkreten Anwendungsfall entspricht.

Wird eine Größe im Zeitbereich angegeben, so bedeutet das, dass sie als Funktion x in Abhängigkeit von der Zeit dargestellt wird. Im Frequenzbereich wiederum wird ein Signal durch eine Amplitude X als Funktion der Frequenz beschrieben, die im Allgemeinen eine komplexe Zahl ist, und somit auch eine Phase hat.

$x(t)$ und $X(f)$ sind also zwei verschiedene Darstellungen von ein- und derselben Funktion. Die Umrechnung zwischen diesen beiden Darstellungen erfolgt über das Fourier-Transformationspaar, wie es in den beiden folgenden Gleichungen angegeben ist ($\Omega = \omega T$).

$$X(e^{j\Omega}) = \sum_{k=-\infty}^{\infty} x(k) e^{-j\Omega k} \quad (3.69)$$

$$x(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\Omega}) e^{j\Omega k} d\Omega \quad (3.70)$$

Die im Zeitbereich vorliegende Folge von Messwerten entspricht $x(k)$ und $X(e^{j\Omega})$ steht für die kontinuierliche Spektralfunktion.

In einem digitalen System gibt es aufgrund des begrenzten Speicherplatzes nur eine endliche Anzahl an Werten $x(k)$, außerdem müssen sowohl die Zeit- als auch die Frequenzvariablen diskretisiert werden, da der Rechner nur diskrete Werte verarbeiten kann. Aus den Gleichungen 3.69 und 3.70 muss deshalb ein Transformationspaar abgeleitet werden, das auf einem digitalen Rechner ausgewertet werden kann, die sogenannte Diskrete Fourier Transformation (DFT) bzw. ein schnellerer Algorithmus der Fourier Transformation, nämlich die Fast Fourier Transformation (FFT), auf die weiter hinten in diesem Kapitel noch etwas genauer eingegangen wird. Für die Umwandlung in ein digital auswertbares Transformationspaar lässt man in Gleichung 3.69 k von $k = 0$ bis $k = N - 1$ laufen, wobei N die Anzahl an Abtastwerten ist.

Um auch Gleichung 3.70 umzurechnen geht man davon aus, dass das Spektrum eines zeitdiskreten Signals periodisch ist, also

$$X(e^{j\Omega}) = X(e^{j\omega T}) = X(e^{j(\omega T \pm 2\pi i)}) = X(e^{j(\Omega \pm 2\pi i)}) \quad (3.71)$$

Das bedeutet, man muss für die Darstellung des Signals nur eine einzige Periode des Spektrums betrachten. Für die Diskretisierung des Spektrums muss nun im betrachteten Frequenzintervall eine endliche Anzahl an Spektrallinien festgesetzt werden. Diese Zahl ist im Prinzip beliebig wählbar, es ist aber oft von Vorteil, wenn es dieselbe ist wie die Anzahl der Abtastwerte im Zeitbereich (N).

Es wird nun die spektrale Auflösung Δf mit Hilfe der Abtastfrequenz f_A definiert:

$$f_A = \frac{1}{T} = N \cdot \Delta f \quad (3.72)$$

$$\Delta f = \frac{1}{N \cdot T} \quad (3.73)$$

Des Weiteren gelten folgende Bedingungen:

$$0 \leq k \leq N - 1 \quad (3.74)$$

$$\omega_n = 2\pi n \Delta f = 2\pi \frac{n}{N \cdot T} \quad (3.75)$$

bzw.

$$\Omega_n = 2\pi n \Delta f = 2\pi \frac{n}{N} \quad (3.76)$$

Diese Voraussetzungen ergeben dann zusammen mit Gleichung 3.69 den Zusammenhang

$$X(e^{j2\pi n/N}) =: X(n) \sum_{k=0}^{N-1} x(k) e^{-jkn2\pi/N} \quad (3.77)$$

Für die Rücktransformation, die sogenannte inverse DFT, wird nun die Zeitfolge zu einem bestimmten Zeitpunkt $k = l$, die man durch Multiplikation der Gleichung 3.77 mit dem Ausdruck $e^{j2\pi nl/N}$ und Summierung über alle N Frequenzwerte erhält, benötigt.

$$\begin{aligned} \sum_{n=0}^{N-1} X(n) e^{j2\pi nl/N} &= \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} x(k) e^{-j2\pi(k-l)n/N} \\ &= \sum_{k=0}^{N-1} x(k) \sum_{n=0}^{N-1} e^{-j2\pi(k-l)n/N} \end{aligned} \quad (3.78)$$

Bildet man die Summe über alle n , so verschwindet diese für alle $k \neq l$ und man kann mit Hilfe des Kronecker-Deltas schreiben

$$\sum_{n=0}^{N-1} e^{j2\pi(k-l)n/N} = N \cdot \delta(k-l) \quad (3.79)$$

Der Ausdruck 3.78 lässt sich somit folgendermaßen umformen:

$$\sum_{n=0}^{N-1} X(n) e^{j2\pi nl/N} = \sum_{k=0}^{N-1} x(k) N \cdot \delta(k-l) \quad (3.80)$$

Wird l dann ganz allgemein wieder durch k ersetzt, so folgt mit Gleichung 3.77 und dem komplexen Drehoperator $W_N = e^{-j2\pi/N}$ für das Fourier-Transformationspaar

$$X(n) = DFT\{x(k)\} = \sum_{k=0}^{N-1} x(k)W_N^{kn} \quad (3.81)$$

$$x(k) = IDFT\{X(n)\} = \sum_{n=0}^{N-1} X(n)W_N^{-kn} \quad (3.82)$$

Wie bereits weiter oben erwähnt, ist es für die Durchführung der DFT völlig ausreichend, wenn man eine Periode von $x(k)$ bzw. von $X(n)$ kennt.

Um nun im Konkreten auf die FFT einzugehen: Hierbei handelt es sich um eine spezielle Form der Diskreten Fourier Transformation, die durch einen besonderen Algorithmus schneller gemacht wird. Sehr oft kommt dabei der sogenannte Radix-2-Algorithmus zur Anwendung, der einfach zu implementieren und aus diesem Grund sehr weit verbereitet ist und außerdem ausgesprochen effizient arbeitet.

Während also bei der DFT für N zu transformierende Werte N^2 komplexe Multiplikationen und Additionen benötigt werden, so sind es bei der FFT - vorausgesetzt, N ist eine Potenz von 2 - nur $N \cdot \lg(N)$ (binärer Logarithmus mit der Basis 2) Rechenoperationen. Der Optimalfall liegt eben vor, wenn N so gewählt wird, dass es eine Potenz von 2 ist. Ist dies nicht der Fall, so werden die restlichen Stellen bis zur nächsten Potenz mit Nullen aufgefüllt.

Verkürzt wird die Rechenzeit bei der FFT dadurch, indem Symmetrien und Periodizitäten, die in W_N enthalten sind, in der Auswertung ausgenutzt werden. $x(k)$ wird dabei durch stetige Halbierung in immer kürzere Teilfolgen aufgespalten, bis man Folgen der Länge 2 erreicht. Zwei gleichwertige Methoden kommen hier zum Einsatz, nämlich die Reduktion im Zeitbereich und die Reduktion im Frequenzbereich. Die Reduktion im Zeitbereich soll hier beispielhaft ganz kurz erläutert werden, das Vorgehen bei der Reduktion im Frequenzbereich erfolgt im Prinzip analog.

Unter Voraussetzung, dass N eine Potenz von 2 und daher geradzahlig ist, kann $x(k)$ in zwei Teilfolgen der Länge $N/2$ unterteilt werden, wobei eine der Folgen alle geraden Indizes k enthält und die andere alle ungeraden k .

$$u(k) = x(2k), k = 0 \dots \frac{N}{2} - 1 \quad (3.83)$$

$$v(k) = x(2k + 1), k = 0 \dots \frac{N}{2} - 1 \quad (3.84)$$

Mit diesen Zusammenhängen lässt sich auch $X(n)$ in zwei Teilsummen zerlegen:

$$\begin{aligned}
 X(n) &= \sum_{k=0}^{N-1} x(k)W_N^{kn} \\
 &= \sum_{k=0}^{N/2-1} x(2k)W_N^{2kn} + \sum_{k=0}^{N/2-1} x(2k+1)W_N^{(2k+1)n} \\
 &= \sum_{k=0}^{N/2-1} u(k)(W_N^2)^{kn} + W_N^n \sum_{k=0}^{N/2-1} v(k)(W_N^2)^{kn}
 \end{aligned} \tag{3.85}$$

Mit der Definition für den Drehoperator W_N folgt

$$W_N^2 = \left(e^{-j2\pi/N}\right)^2 = e^{-j2\cdot 2\pi/N} = e^{-j2\pi/(N/2)} = W_{N/2} \tag{3.86}$$

und damit ergibt sich zusammen mit Gleichung 3.85 $X(n)$ zu

$$X(n) = \sum_{k=0}^{N/2-1} u(k)W_{N/2}^{kn} + W_N^n \sum_{k=0}^{N/2-1} v(k)W_{N/2}^{kn} \tag{3.87}$$

Der Vorteil dieser Umformung liegt also darin, dass aus einer N-Punkte-DFT zwei DFTs mit jeweils $N/2$ Punkten gemacht wurden und dadurch die Rechengeschwindigkeit deutlich verkürzt wird.

Einige Phänomene, die im Zusammenhang mit der FFT immer wieder in Erscheinung treten und auch die Ergebnisse derselben maßgeblich mit beeinflussen, sollen im Folgenden kurz beschrieben werden.

Unbedingt zu erfüllen ist dabei auch das bereits mehrmals erwähnte Nyquist-Kriterium, das auch Einfluss auf das nachfolgend erklärte Aliasing hat. Um dieses Kriterium zu erfüllen, muss die Frequenz des Eingangssignals kleiner sein als eine kritische Grenzfrequenz, die sich nach unten stehender Gleichung berechnet.

$$f_c = \frac{1}{2\Delta} \tag{3.88}$$

Δ ist das Intervall zwischen zwei Samples bei der Abtastung des Signals. Nimmt man als Beispiel eine Sinuskurve, so entspricht die kritische Sampling-Rate zwei Samples pro Periode. Wird etwa bei einem positiven Peak zum ersten Mal abgetastet, dann muss der darauf folgende Sample-Punkt beim nächsten negativen Peak liegen usw. um das Signal korrekt abzutasten und es zu keiner Verfälschung der Sinuskurve kommt.

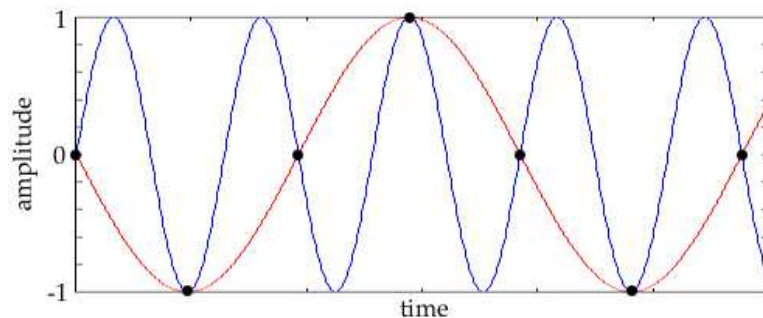


Abbildung 3.11: Ursprüngliches Signal (blau) und aufgrund zu geringer Abtastrate Verfälschung des Signals (rot) [27]

Aliasing

Aliasing tritt immer dann auf, wenn die Anzahl der Samples, mit der ein Signal abgetastet wird, nicht ausreichend groß ist, das heißt die Frequenz des Input-Signals liegt außerhalb der Nyquist-Grenzen ($-f_c; f_c$) (siehe dazu auch Abschnitt über Nyquist-Kriterium und Abbildung 3.11). Die Frequenzen, die außerhalb dieser Grenzen liegen, werden bei der Fourier-Transformation falsch in den begrenzten Bereich übertragen und dadurch das Ergebnis der FFT beeinträchtigt. Dies zeigt auch Abbildung 3.12.

Vermieden werden kann Aliasing, indem man entweder genau die Bandbreite eines Signals kennt und mit dem Nyquist-Theorem abstimmt bzw. gegebenenfalls die Sampling-Rate anpasst, damit das Nyquist-Kriterium erfüllt ist, oder durch Erzwingen von Grenzen mit Hilfe von entsprechenden Filtern.

Leakage-Effekt

Wird die Fensterlänge der FFT unpassend - also zu kurz oder zu lang - gewählt, kann es aufgrund der periodischen Fortsetzung eines Signals zu Sprungstellen im Spektrum und zusätzlichen Spektrallinien bzw. Frequenzanteilen kommen.

Im oberen Graphen der Abbildung 3.13 ist die Periodendauer ein ganzzahliges Vielfaches der Fensterlänge, was sich bei der FFT dahingehend auswirkt, dass nur eine einzige Spektralkomponente bei der Frequenz der harmonischen Schwingung auftritt und alle anderen Komponenten Null sind.

Das untere Bild zeigt, dass die Periodendauer kein ganzzahliges Vielfaches der Fensterlänge ist und es dadurch zum Leakage-Effekt kommt. Es gibt hier im Spektrum mehrere Komponenten bei verschiedenen Frequenzen, die ungleich Null sind, daher wird auch nicht die tatsächliche, maximale Amplitude dargestellt, sondern der höchste Peak ist niedriger als beim Spektrum ohne Leakage. Dieser Effekt tritt jedoch nicht nur bei rein harmonischen Signalen auf, sondern auch

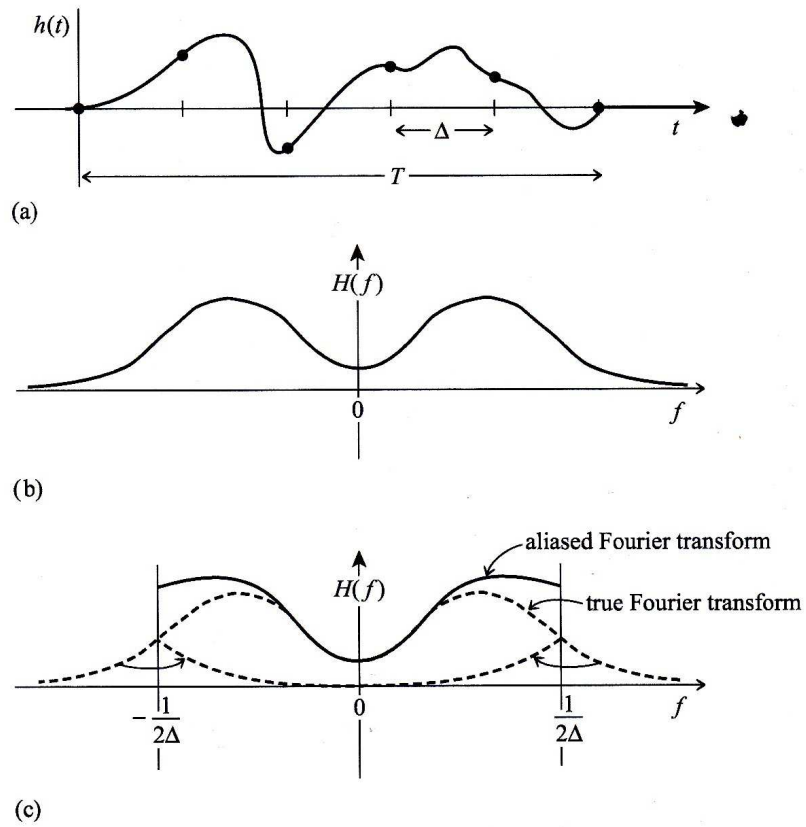


Abbildung 3.12: Aliasing - (a) Kontinuierliches Signal in Abhängigkeit der Zeit (b)FFT des Originalsignals (c) FFT nach Sampling des Originalsignals mit dem Intervall Δ und den Nyquist-Grenzen

bei Signalen, die sich aus mehreren harmonischen Anteilen zusammensetzen. Um Leakage zu vermeiden, kann das Signal vor dem Durchführen der FFT mit einer Fensterfunktion multipliziert werden.

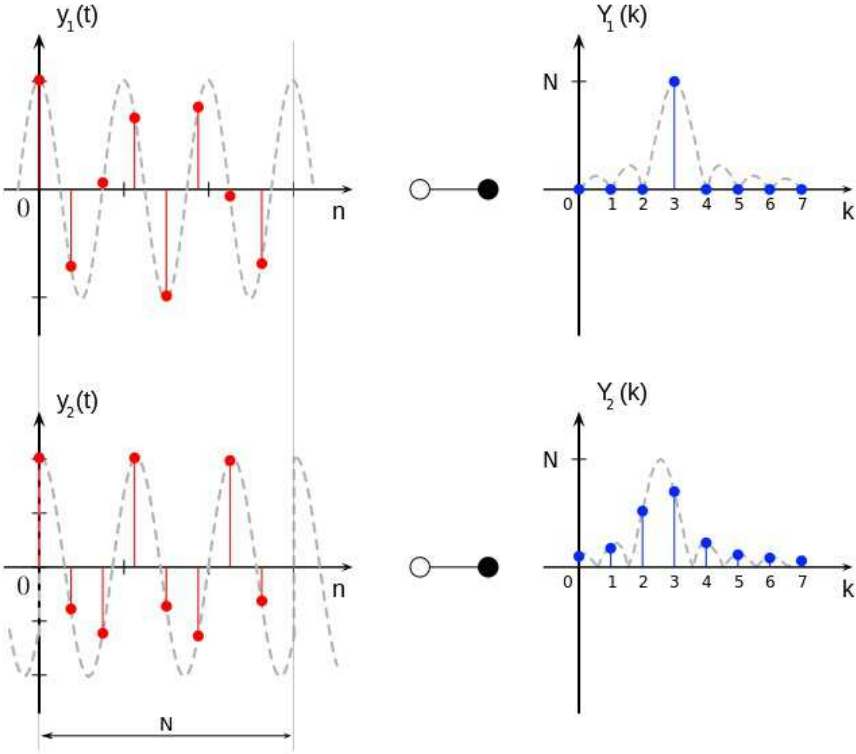


Abbildung 3.13: Leakage-Effekt bei harmonischem Signal [28]

Kapitel 4

Schallmessung

4.1 Verwendete Messtechnik

Wie auch in Kapitel 5.1 genauer erläutert, muss vor der eigentlichen Schallmessung jedes der Mikrofone mittels eines Pistonphons der Firma G.R.A.S., Typ 42AA kalibriert werden. Dabei sendet das Pistonphon einen Ton bei einer Frequenz von 250 Hz mit einer Amplitude von 114 dB aus, und ein LabView-Programm ermittelt dann für jedes einzelne Mikrofon den Unterschied zwischen dem ausgesendeten und dem vom Mikrofon aufgezeichneten Signal. Aus diesem Unterschied wird dann ein Kalibrationsfaktor berechnet, der einerseits die Differenz kompensiert, andererseits auch als Umrechnungsfaktor (siehe 5.1) dient.

Eine wichtige Hardware-Komponente für die Datenerfassung stellt das PXI-Chassis von National Instruments mit insgesamt 32 verfügbaren Kanälen dar, das außerdem zwei PXI-Messkarten vom Typ PXIe-4498 beinhaltet. Von diesen 32 Kanälen werden 24 für die Mikrofone selbst und zwei für die beiden an der Nabe der Turbine angebrachten Referenzmikrofone, auf die im nachfolgenden Kapitel noch näher eingegangen wird, verwendet. Kanal 27 und Kanal 28 sind den beiden Triggern für den Hochdruck- und für den Niederdruckrotor zugeteilt. Sie erlauben es, eine Beziehung zwischen dem Schallfeld und der Position des jeweiligen Rotors herzustellen (siehe auch Kapitel 5.2 bzw. 5.3). Ein Fiberglaskabel überträgt die Daten danach vom Chassis, das an der Stelle, wo das Kabel angeschlossen wird, mit einer PXIe-Messkarte Typ NI PXIe 8375 bestückt ist, zu einem in der Messwarte aufgebauten Standrechner, der ebenfalls mit einer entsprechenden Karte vom Typ NI PCIE-8375 ausgestattet ist und auf dem die Messergebnisse für jeden einzelnen Messpunkt auch abgespeichert werden, übertragen.

Was die Sensoren selbst betrifft, so kommen hier Mikrofone der Firma G.R.A.S. vom Typ 40DB zum Einsatz. Es handelt sich dabei um vorpolarisierte, 1/4-inch Druckmikrofone, die aufgrund ihrer geringen Empfindlichkeit für Schalldruckpe-

gel bis zu 170 dB verwendet werden können, des Weiteren beträgt ihr Frequenzbereich bis zu 70 kHz. Jedes der Mikrofone ist zusätzlich mit einem Vorverstärker der gleichen Firma vom Typ 26AC ausgestattet, der sich durch geringes Eigenrauschen, hohe Temperaturbeständigkeit (kann bis zu 120°C Umgebungstemperatur verwendet werden) und geringe Empfindlichkeit gegenüber Vibrationen auszeichnet. Die Druckmikrofone verfügen über eine sogenannte Kugelcharakteristik, was bedeutet, dass es egal ist, in welchem Winkel die Schallwelle auf die Membran des Mikrofons trifft, die Intensität ist immer annähernd dieselbe.

4.2 Anordnung der Mikrofone

Wie bereits erwähnt, wurden für die Akustikmessungen insgesamt 26 Mikrofone verwendet, 24 davon wurden zueinander versetzt und wandbündig ins Außengehäuse hinter der Niederdruckturbine eingebaut, zwei dienen als Referenzmikrofone und befinden sich an der Nabe, ebenfalls im Kanal stromabwärts hinter der Niederdruckturbine. Bereits Abbildung 2.3 zeigt in dem rot markierten Rahmen die Akustikmessstrecke und mit ihr die Position der Mikrofone bzw. der beiden Referenzmikrofone im Strömungskanal, außerdem ist ihre Anordnung ein Abbildung 4.2 ersichtlich.

Die 24 Mikrofone sind in eine Platte eingebaut, die sehr einfach in das Gehäuse ein- und auch wieder ausgebaut werden kann, und zueinander versetzt angeordnet. Das bedeutet, dass die Mikrofone einerseits in zwei leicht versetzten Reihen zu jeweils 12 Sensoren eingebaut werden, andererseits die beiden Reihen zueinander einen Winkelversatz von 6° aufweisen, was durch das nachfolgende Bild veranschaulicht wird.

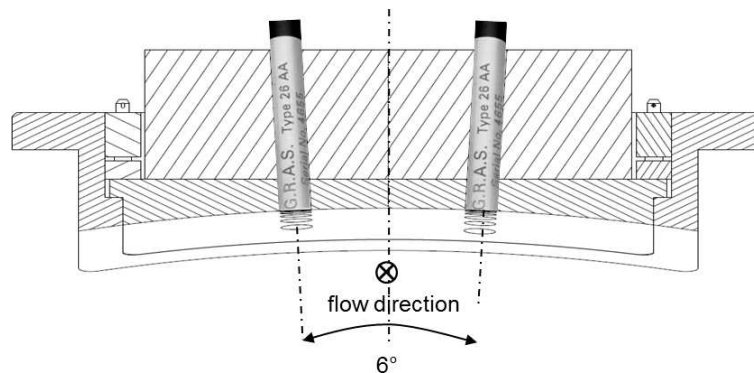


Abbildung 4.1: Winkelversatz zwischen den beiden Mikrofonreihen [21]

Die ersten 12 Mikrofone befinden sich noch in dem Teil des Strömungskanals, der nicht zylindrisch ist, ihr axialer Abstand beträgt vom Mittelpunkt der Membran aus gemessen 6 mm. Die hinteren 12 Mikrofone liegen schon im vollständig zylindrischen Abschnitt des Kanals, und zwar mit nur 5 mm axialem Abstand, da hier

Rücksicht auf die begrenzten Platzverhältnisse genommen werden musste. Der axiale Abstand der Mikrofone wurde so gewählt, dass auf alle Fälle das Nyquist-Kriterium erfüllt ist. Das bedeutet, die halbe Wellenlänge der Schallwelle muss auf alle Fälle größer sein als der axiale Abstand der Sensoren, damit es zu keiner Verfälschung der Messergebnisse kommt.

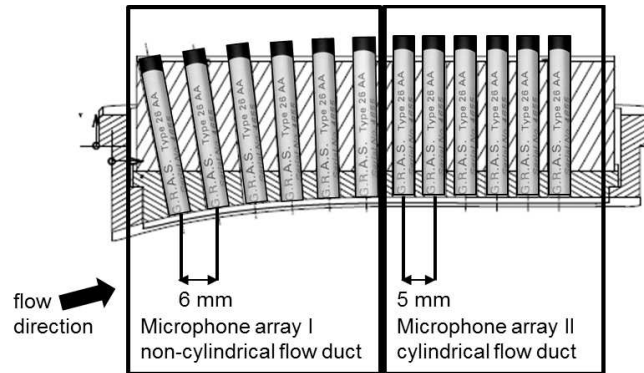


Abbildung 4.2: Mikrofonplatte mit axialen Abständen der Mikrofone [21]

Zu erwähnen ist an dieser Stelle, dass es noch nicht geklärt wurde, inwiefern sich die Geometrie des nicht-zylindrischen Teils des Strömungskanals auf die Messergebnisse auswirkt. Die vereinfachenden Annahmen für die Auswerterroutinen gehen von einem zylindrischen Kanal aus und berücksichtigen dadurch nicht den Einfluss der ersten 12 Mikrofone, die sich in dem betroffenen Abschnitt befinden, auf die Ergebnisse der Auswertung. Abhilfe hat man sich in diesem Fall dahingehend verschafft, indem das Post-Processing - genauer gesagt die Azimutalmodenanalyse - sowohl für die vorderen 12, als auch für die hinteren 12 getrennt, aber ebenso für alle 24 Mikrofone gemeinsam betrachtet werden kann.

Die nachfolgende Begründung, warum die Mikrofone im Strömungskanal genau in der Art und Weise angeordnet sind, wie es weiter oben beschrieben wurde, basiert auf Untersuchungen des Deutschen Zentrums für Luft- und Raumfahrt (DLR) zu diesem Thema und kann im Detail in [20] nachgelesen werden. Diese Arbeit beinhaltet eine experimentelle Analyse von hochfrequentem tonalem Schall, der sich in einem Strömungskanal mit konstanter Querschnittsfläche ausbreitet, mittels einer Radialmodenanalyse (RMA) und den Einfluss der Sensoranordnung auf die Ergebnisse. Untersucht wurden dabei die folgenden vier Mikrofonanordnungen mit einer insgesamt konstanten Anzahl von 24 Mikrofonen je untersuchter Konfiguration:

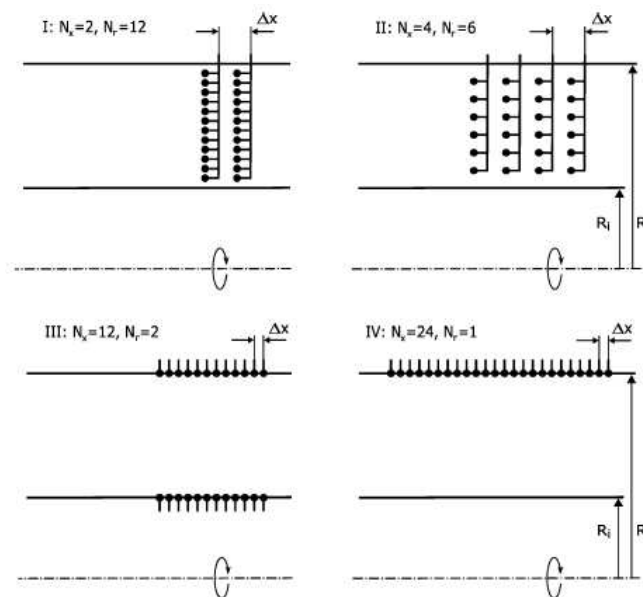


Abbildung 4.3: Varianten für Mikrofonanordnung mit konstanter Sensorzahl [20]

Wie die obige Abbildung schon zeigt, wurde bei den untersuchten Anordnungen entweder die Anzahl der axialen oder der radialen Positionen variiert, die Gesamtanzahl an Sensoren wurde jedoch mit 24 Mikrofonen immer konstant gehalten. Wichtig ist, dass bei den Untersuchungen des DLR der Schwerpunkt auf der Auswertung der Messdaten mittels einer Radialmodenanalyse lag, für die eine radiale Sensoranordnung mit Sicherheit von Vorteil ist. In den hier durchgeführten Analysen ist jedoch auch die Azimutalmodenanalyse von großer Bedeutung, was auch die Wahl der Mikrofonanordnung beeinflusste.

Mathematisch betrachtet haben die radial angeordneten Sensorrechen gegenüber der axialen Anordnung einen Vorteil, da die Eigenfunktionen der radialen Moden im Vergleich zu den axialen Wellenzahlen orthogonal sind und daher zu höherer numerischer Stabilität führen. Außerdem sind sie für die Analyse radialer Moden besser geeignet als axial montierte Sensoren. Ein großer Nachteil ist jedoch, dass der Nachlauf, der durch die radialen Rakes hervorgerufen wird, mit der Schallquelle interagieren und dadurch das eigentliche Schallfeld verändern kann, das heißt zusätzlicher Lärm könnte erzeugt werden. Des Weiteren kann es durch die Sensorrechen zu Blockagen in der Strömung kommen, die wiederum zu einer Veränderung der Betriebsbedingungen der Maschine führen, außerdem werden abgestrahlte und reflektierte Moden von den Sensoren in radialer Anordnung nicht unterdrückt. Und obwohl die radialen Rechen deutlich unempfindlicher auf Mikrofonfehler reagieren, entschied man sich letzten Endes doch für die Variante mit den wandbündigen Mikrofonen, die allesamt an der äußeren Kontur angebracht werden. Dies garantiert eine vergleichsweise einfache Montage und Demontage und ist überdies die preisgünstigere Lösung bei der außerdem die Strömung im Kanal nicht gestört wird. Die Sensoren am Hub wurden auf die beiden Refe-

renzmikrofone reduziert, da sich hier das Aus- und Einbauen als ausgesprochen schwierig erweist. Diese werden benötigt, da sie einerseits nach Durchführung der FFT einen Vergleichswert für das Ergebnis der restlichen Mikrofone darstellen, andererseits aber auch in die Berechnung der Azimutal- und der Radialmoden einfließen, um dadurch das Rauschen und damit den Fehler aufgrund der digitalen Signale zu reduzieren. Warum die RMA trotz der axialen Anordnung der Sensoren durchgeführt werden kann, wird in Kapitel 5.6 beschrieben.

4.3 Datenerfassung

Wie schon in Kapitel 2.2 erwähnt, ist der Teil des Gehäuses, in dem sich die Mikrofonplatte befindet, um 360° traversierbar, was bedeutet, dass die Akustikmessungen an jeder beliebigen Position in azimuthaler Richtung durchgeführt werden können. In diesem Fall wurde die Platte in 2° -Schritten traversiert und in jedem Messpunkt wurden die Daten von jedem einzelnen Kanal gleichzeitig aufgenommen und gespeichert. Als Messdauer wurde an jeder der insgesamt 180 Messpositionen 20 s gewählt, während die Abtastfrequenz der Mikrofone 60 kHz betrug. An dieser Stelle soll noch angemerkt werden, dass auch die Anzahl der Messpunkte in azimuthaler Richtung unbedingt dem Nyquist-Theorem zu folgen hat und demnach folgende Gleichung erfüllen muss:

$$\frac{1}{2} \cdot \frac{2\pi}{m_{max}} > \Delta\varphi = \frac{2\pi}{N_\varphi} \quad (4.1)$$

m_{max} ist hier die maximale Modenordnung, N_φ die Anzahl der azimuthalen Messpositionen und $\Delta\varphi$ die Schrittweite der Traversierung.

Die Auswertung der aus den Akustikmessungen stammenden Daten und deren Darstellung im Spektrum erfolgte bis zu einer Frequenz von 15 kHz, da sich oberhalb dieser Grenze aufgrund der kleinen Peaks nur noch wenig an Information gewinnen lässt.

Kapitel 5

Auswertemethoden

Für die Auswertung der Messdaten wurde eine Vielzahl an Auswerteroutinen herangezogen, die schon vor den hier behandelten Messungen zum Einsatz kamen und im Rahmen zahlreicher wissenschaftlicher Arbeiten und Dissertationen am Institut entstanden sind. Es handelt sich hier in erster Linie um Matlab bzw. Fortran-Programme, die während dieser Diplomarbeit um ein Tool für die Auswertung des Rohsignals erweitert wurden und deren grundlegende theoretische Zusammenhänge in den folgenden Kapiteln erläutert werden. Die Details zu Handhabung und Ausführung der Routinen hingegen, sowie Informationen zu den notwendigen Eingabeparametern und -dateien sind im Anhang dieser Arbeit nachzulesen.

5.1 Auswertung des Rohsignals

Bevor die Rohdaten der Akustikmessungen anhand der Auswerteroutinen nachbearbeitet und aufbereitet wurden, wurde eine Auswertung des reinen Rohsignals mittels einer sehr einfachen Matlab-Routine durchgeführt, um erste Tendenzen bei den Ergebnissen erkennen zu können, ohne vorher aufwändige und zeitintensive Rechenschritte durchlaufen zu müssen. Des Weiteren konnte mit Hilfe dieser Methode sehr schnell eruiert werden, ob eines der Mikrofone während des Versuches beschädigt wurde, da die Auswertung in diesem Fall für jeden einzelnen der über den Umfang verteilten 180 Messpunkte erfolgte, sowie für jedes einzelne der insgesamt 24 Mikrofone. Das heißt, es wurde anfangs weder eine Mittelung über die 180 Umfangspositionen im Abstand von jeweils zwei Grad, noch über die 24 Mikrofone durchgeführt, so wie das in den weiteren Schritten der Messdatenauswertung der Fall sein wird (siehe nachfolgende Kapitel). In weiterer Folge war es dann auch möglich, die Rohdaten über alle Mikrofone bzw. alle Umfangspositionen zu mitteln.

Unabhängig von den Auswerterroutinen muss vor der Durchführung der Messungen jedes einzelne der Mikrofone kalibriert werden, und zwar mit Hilfe eines Pistophon. Dieses erzeugt, wie bereits im vorhergehenden Kapitel erwähnt, einen Ton bei einer Frequenz von 250 Hz und einer Amplitude von 114 dB. Für jedes Mikrofon wird dabei mittels eines LabView Programms ein Kalibrationsfaktor ermittelt, der einerseits die Abweichung der tatsächlich aufgezeichneten Amplitude von den 114 dB ausgleicht, andererseits aber auch als Umrechnungsfaktor dient. Das bedeutet, dass das Signal, das von den Mikrofonen an den Rechner weitergegeben wird, ein Spannungssignal in Millivolt ist und daher zuerst in Pascal, da ja eigentlich der Schalldruck gemessen wird (siehe dazu auch Abschnitt 3.1), bzw. in weiterer Folge dann in dB umgerechnet werden muss.

Das Rohsignal wird also für jedes Mikrofon an einem bestimmten Messpunkt, z.B. Punkt 4, was einer Traversierung des TMTFs um acht Grad entspricht, eingelesen und noch vor der Multiplikation mit dem Kalibrationsfaktor und einer damit einhergehenden Umrechnung in Pascal grafisch dargestellt. Der Plot ist in Abbildung 5.1 zu sehen.

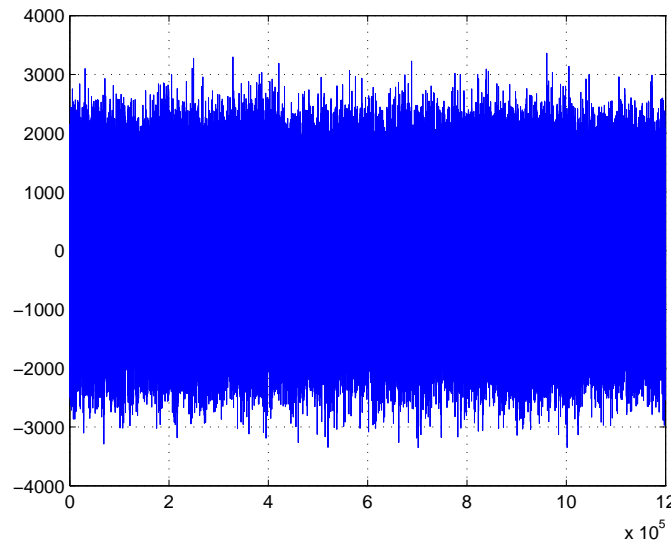


Abbildung 5.1: Rohsignal von Mikrofon 3 am Messpunkt 4 (entspricht 8°)

Nach dem Einlesen folgt, wie auch später bei der Auswertung mit den gemittelten und aufbereiteten Daten, eine Fast Fourier Transformation (FFT) des Signals und eine Umrechnung der Amplitude von Pascal in Dezibel. Abbildung 5.2 und Abbildung 5.3 zeigen das Resultat für Mikrofon 1 sowohl am Messpunkt 1 (0° Traversierung) als auch am Messpunkt 4 (8° Traversierung).

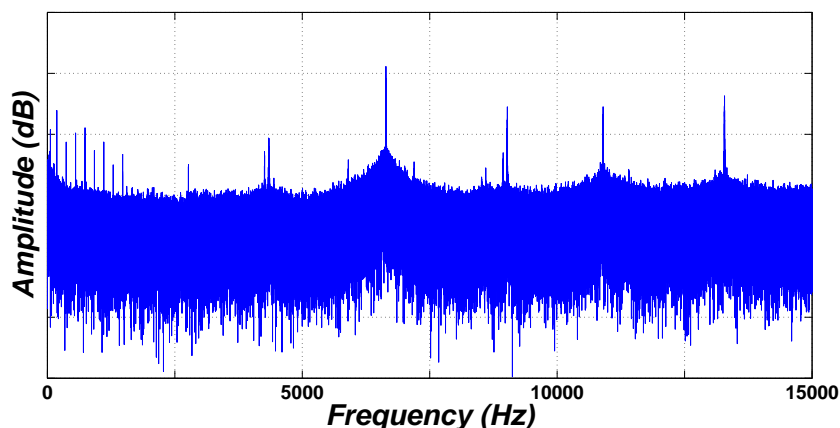


Abbildung 5.2: FFT des Rohsignals für Mikrofon 1, Messpunkt 1

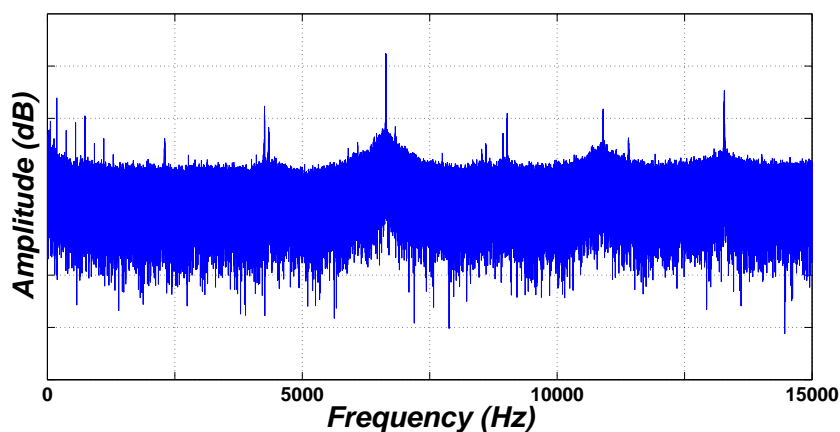


Abbildung 5.3: FFT des Rohsignals für Mikrofon 1, Messpunkt 4

5.2 Adaptive Neuabtastung

Bei den vorliegenden Akustikmessungen betrug die Abtastfrequenz mit der die Druckschwankungen aufgenommen wurden 60 kHz bei einer Messzeit von 20 s pro Messposition, was zum Beispiel beim Niederdruckrotor mit ungefähr 3550 Umdrehungen pro Minute in etwa 1000 abgetasteten Einzelmesswerten pro Umdrehung entspricht. Das Problem ist, dass die tatsächliche Anzahl der Einzelmesswerte (Samples) pro Umdrehung nicht konstant ist, da die Drehzahl der Welle minimalen Schwankungen unterliegt. Aus diesem Grund muss die Sampling-Frequenz mit Hilfe der Analyse eines Wellen-Encoder-Systems (Trigger) korrigiert werden. Das heißt, ein Trigger - sowohl für den HP als auch den LP Rotor vorhanden - stellt Anfang und Ende einer Umdrehung der Welle fest, anschließend findet ein Vorgang statt, der als Adaptive Resampling oder Adaptive Neuabtastung bezeichnet wird und der eine Rotorumdrehung in eine konstante Anzahl von Samples unabhängig von der momentanen Drehzahl unterteilt. Wie oft während einer

Umdrehung abgetastet wird, kann im Prinzip in den verwendeten Routinen frei gewählt werden, folgende Vorschrift sollte für das Programm jedoch gelten:

$$nn = \frac{f_s \cdot t}{nperiod} > 2^n \quad (5.1)$$

In dieser Gleichung stellt f_A die Abtastfrequenz, also 60 kHz, dar, t steht für die Messzeit von 20 Sekunden und nn für die Anzahl der Neuabtastungen. $nperiod$ bezeichnet die Anzahl der Umdrehungen während der Messzeit von 20 s pro Messpunkt - entweder für den Hochdruck- oder für den Niederdruckrotor - und wird folgendermaßen berechnet:

$$nperiod = t \cdot \frac{rpm}{60} \quad (5.2)$$

rpm ist hier für die Umdrehungen eines Rotors pro Minute, t wiederum die Messzeit von 20 s.

Aus Gleichung 5.1 wird deutlich, dass das Ergebnis für nn größer ist, als eine Zahl, die sich aus dem Zusammenhang 2^n berechnet, was bedeutet, dass für die tatsächliche Anzahl an Resamples eine Zahl gewählt werden muss, die sowohl kleiner ist als das Ergebnis für nn , als auch aus 2^n berechnet werden kann. Der Exponent n muss hier also derart ausgesucht werden, dass die Ungleichung in Formel 5.1 erfüllt ist. Dass die Anzahl der Abtastungen genau eine Potenz zur Basis 2 sein soll, hat jedoch keine mathematischen Gründe, sondern beruht auf dem in den Fortran-Routinen verwendeten Radix-2-Algorithmus zur Durchführung der FFT. Die Anwendung dieses Algorithmus für die digitale Signalverarbeitung spart einen großen Teil der Rechenoperationen ein und führt dadurch zu geringerem Speicherbedarf und kürzeren Rechenzeiten (siehe auch Kapitel 3.6)

Auch wenn die Drehzahlschwankungen bei Turbinenwellen weniger als $\pm 1\%$ betragen, würden diese Abweichungen von der Sollzahl ohne eine Korrektur durch die Adaptive Neuabtastung dazu führen, dass zwar genau bei der rechnerischen Blattfolgefrequenz (Blade Passing Frequency BPF), die ja mit der Sollzahl berechnet wird, ein Ton mit einer bestimmten Amplitude vorherrscht, jedoch bei einer minimalen Drehzahlreduktion auch ein Ton bei einer kleineren Frequenz angeregt wird bzw. umgekehrt bei einer minimalen Drehzahlsteigerung ein Ton bei einer höheren Frequenz als der BPF angeregt wird.

5.3 Phase Averaging

Ein weiterer Schritt im Post Processing ist das sogenannte Phase Averaging, bei dem - zunächst ganz einfach formuliert - an einem beliebigen Punkt im Raum die

Werte einer Strömungsgröße p bei einer bestimmten Phase ϕ gemittelt werden. Folgendes Bild soll dies ganz allgemein für zwei Signale, ein harmonisches und ein stochastisches, veranschaulichen.

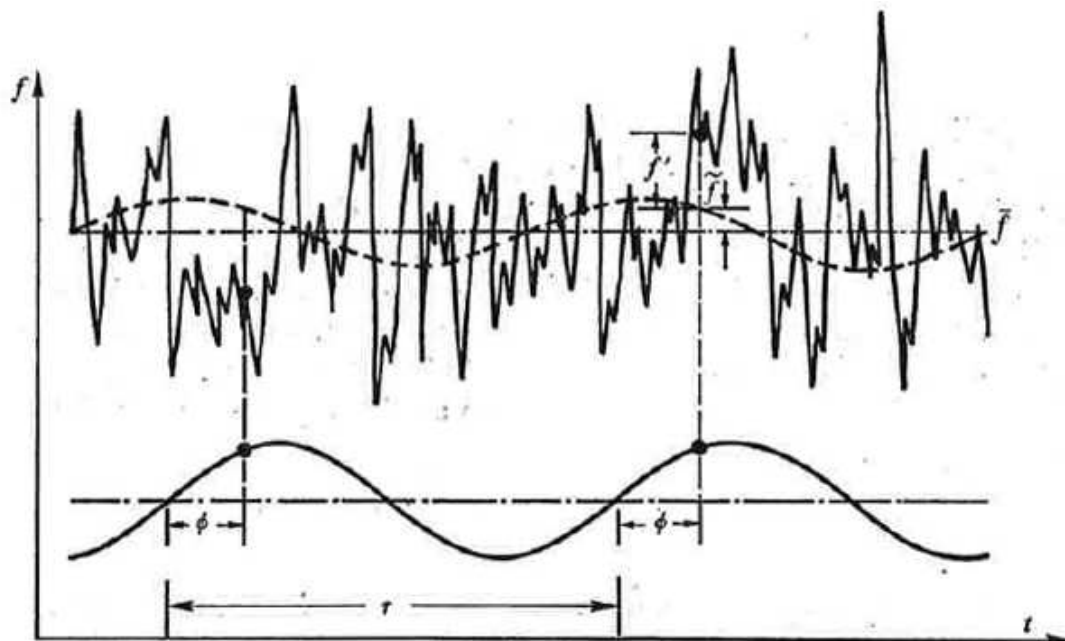


Abbildung 5.4: Phase Averaging

Für den Fall, dass eine periodische Schwankung eine Strömung charakterisiert, kann eine Größe im Strömungsfeld dreifach aufgespalten werden, was auch als “triple decomposition procedure” bezeichnet wird.

$$p(t) = \bar{p} + \langle p(t) \rangle + p'(t) \quad (5.3)$$

In Gleichung 5.3 ist $p(t)$ die Größe im Strömungsfeld, \bar{p} der zeitliche Mittelwert, $\langle p(t) \rangle$ der rein periodische Anteil der Größe und $p'(t)$ steht für zufällige Fluktuationen, die in erster Linie auf Turbulenzen im Strömungskanal zurückzuführen sind.

Für zwei Rotoren, deren Frequenzen nicht korrelieren, wie das in der betrachteten Versuchsanlage der Fall ist, lässt sich die Gleichung folgendermaßen umformulieren:

$$p(t) = \bar{p} + \langle p(t) \rangle_{HP} + \langle p(t) \rangle_{LP} + \langle p(t) \rangle_{HP,LP} + p'(t) \quad (5.4)$$

$\langle p(t) \rangle_{HP}$ und $\langle p(t) \rangle_{LP}$ sind jeweils die rein periodischen Anteile des Hochdruck- bzw. des Niederdruck-Rotors, die jedoch unterschiedliche Periodendauern und

Frequenzen haben. $\langle p(t) \rangle_{HP,LP}$ bildet die periodischen Anteile ab, die sich aufgrund der Interaktionen zwischen Hochdruck- und Niederdruckrotor ergeben und erzeugt Peaks bei Frequenzen, die sich aus $i \cdot BPF_{LP} + k \cdot BPF_{HP}$ berechnen lassen ($i, k \neq 0$, Genaueres dazu ist in Kapitel 3.5 zu finden).

Bei den durchgeführten Messungen wurde die Strömung getriggert, das heißt ein Winkelkodierer (shaft encoder) gibt bei jeder Umdrehung des Rotors, entweder des HP oder des LP Rotors, einen Impuls ab und ermöglicht damit die Auflösung der Strömung - unter Berücksichtigung der Phasenlage der Welle - von einem der beiden Rotoren. Durch dieses Vorgehen können die Einflüsse der beiden Rotoren auf die Strömung separiert und die Effekte von jeweils einem Rotor isoliert werden. Beim Rotor Synchronic Averaging hingegen werden die Einflüsse der beiden Rotoren miteinander kombiniert und die Rotor-Rotor-Interaktionen ebenfalls berücksichtigt (siehe Kapitel 5.4).

Die phasengemittelten Werte berechnen sich also aus dem Mittelwert aller Samples bei ein- und derselben Phase, basierend auf der Winkelkodierung eines Rotors. Das erlaubt die Identifizierung von Strukturen, die mit der Drehgeschwindigkeit dieses Rotors korrelieren, was bedeutet, dass beispielsweise in einem Frequenzspektrum die Peaks in erster Linie bei den Frequenzen auftreten, die in Zusammenhang mit der Drehzahl des betrachteten Rotors stehen.

Da die beiden Rotoren (HP und LP) entgegengesetzt rotieren und somit ihre Drehgeschwindigkeiten nicht miteinander in Beziehung stehen, führt ein Phase Averaging basierend auf der Phase einer der beiden Rotoren dazu, dass die Fluktuationen der Strömungsgrößen, die durch den anderen Rotor hervorgerufen werden, beinahe vollständig unterdrückt werden. Je nachdem welchen Trigger man also wählt, dominieren im Spektrum entweder die Amplituden bei den Blade Passing Frequencies (BPF) des Hochdruck- oder des Niederdruckrotors, wie das auch in den Abbildungen 5.5 und 5.6 veranschaulicht wird.

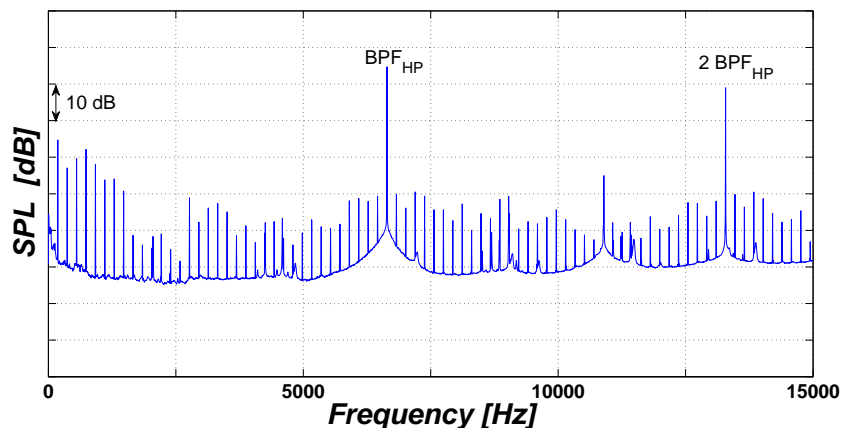


Abbildung 5.5: Frequenzspektrum nach Adaptive Resampling mit Hochdruck-Trigger

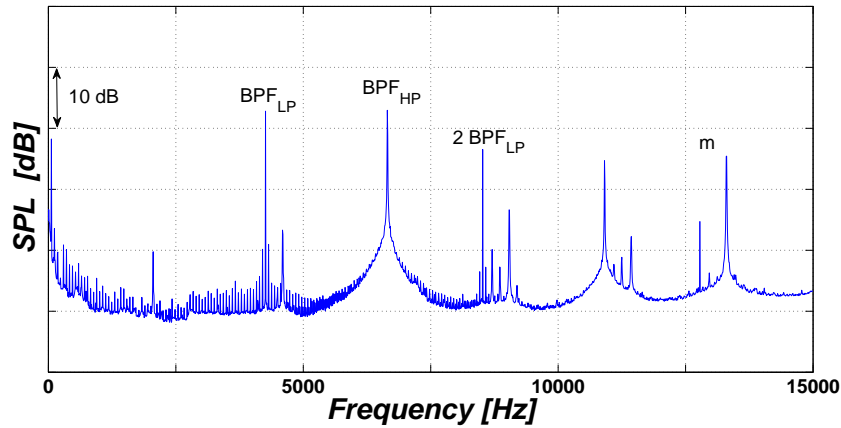


Abbildung 5.6: Frequenzspektrum nach Adaptive Resampling mit Niederdruck-Trigger

Um noch einmal auf Gleichung 5.4 zurückzukommen: Wenn das Phasen-Mittel nun über eine ausreichend große Anzahl an Perioden eines Rotors (z.B. des HP Rotors) gebildet wird, verschwinden sowohl die Effekte $\langle p(t) \rangle_{LP}$ de LP-Rotors, als auch die Schwankungen aufgrund der Turbulenz $p'(t)$ und die kombinierten Interaktionen $\langle p(t) \rangle_{HP,LP}$. Die betrachtete Größe im Strömungsfeld ergibt sich also nach der Mittelung zu

$$\tilde{p}(t)_{HP} = \bar{p} + \langle p(t) \rangle_{HP} \quad (5.5)$$

Anzumerken ist, dass eine reine Zeitmittelung die Effekte der beiden Rotoren und deren Einfluss auf die Strömung komplett entfernen würde.

Wie genau das Phase Averaging in den für die Auswertung verwendeten Routinen durchgeführt wurde, kann in der Programmdokumentation im Anhang nachgelesen werden, bzw. sind weitere Informationen zu diesem Thema in [3] und [5] zu finden, woraus auch die hier ausgeführten Grundlagen entnommen wurden.

5.4 Rotor Synchronic Averaging (RSA)

Die nachfolgenden Ausführungen basieren auf der Arbeit von Lengani et al., nachzulesen in [3].

Mehrmals angemerkt wurden bereits die Effekte beim Rotor Synchronic Averaging, wo nicht der Einfluss jedes einzelnen Rotors auf die Schallentwicklung betrachtet wird, sondern die Interaktionen zwischen den beiden Rotoren im Mittelpunkt stehen. Das bedeutet, im Spektrum treten nicht nur Peaks bei den Blade Passing Frequencies (BPF) des betrachteten Rotors auf, sondern auch bei Frequenzen, die sich nach Gleichung 5.6 berechnen lassen, wobei i und k ganze Zahlen ungleich 0 sind.

$$f_{HP,LP} = i \cdot BPF_{LP} + k \cdot BPF_{HP} \quad (5.6)$$

Abbildung 5.7 soll nun helfen, das Prinzip der RSA zu verdeutlichen.

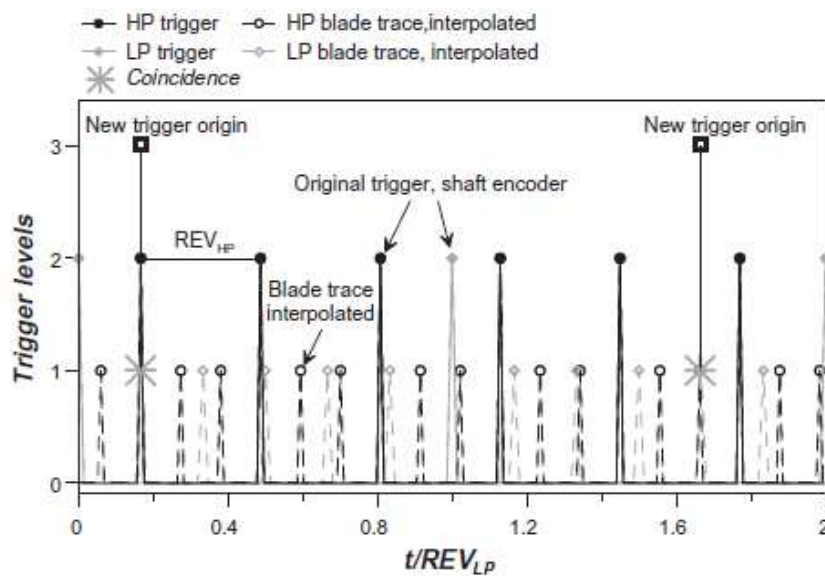


Abbildung 5.7: Skizze zur Erklärung des Rotor Synchronic Averaging [3]

Auf der Abszisse aufgetragen ist die dimensionslose Zeit, die durch die Umlaufzeit des Niederdruck-Rotors dividiert wurde. In dieser Skizze ist das Triggerlevel nur eine symbolische Zahl und soll die verschiedenen Trigger und ihren Ursprung voneinander unterscheiden. Der Trigger mit dem Wert 2 steht für die Wellenencoder-Signale der Hochdruck- und der Niederdruckturbine und kennzeichnet jeweils den Anfang bzw. das Ende einer Umdrehung. Das graue Signal steht hier für den Trigger der LP Turbine, das schwarze für die HP Turbine. Das Triggerlevel mit der Größe 1 dient dazu, die jeweiligen Schaufeln innerhalb einer Umdrehung

eines Rotors zu identifizieren, wodurch eine Umdrehung in äquidistante Intervalle unterteilt wird. Der Einfachheit halber besteht der HP Rotor hier aus drei (Triggersignale mit schwarzen Kreisen) und der LP Rotor aus sechs Schaufeln (Triggersignale mit grauen Vierecken), das heißt trotz verringerter Anzahl an Schaufeln bleibt das ursprüngliche Schaufelzahlenverhältnis erhalten. Wichtig zu erwähnen ist noch, dass die Schaufel-Trigger im zugehörigen Programm interpoliert werden. Ein entsprechender Algorithmus identifiziert nun die Positionen, wo es zu einer Übereinstimmung zwischen den Triggern der beiden Rotoren kommt (auf Triggerlevel 1 und/oder 2). Diese Koinzidenz ist Ausgangspunkt für ein neues Triggersignal mit dem Level 3. Bei diesem Signal befinden sich also die Schaufeln der beiden Rotoren immer in derselben relativen Position zueinander. Wie man auch aus dem Bild erkennen kann, ist das Zeitintervall Δt zwischen zwei Triggersignalen mit dem Wert 3 ein Vielfaches der Blade Passing Period (BPP) des HP Rotors bzw. des LP Rotors.

$$\Delta t = m \cdot T_{HP} = n \cdot T_{LP} \quad (5.7)$$

Auch hier sind m und n ganze Zahlen verschieden von Null. Formt man nun Gleichung 5.6 unter Verwendung von T_{LP} , T_{HP} und des Zusammenhangs $T_{LP} = \frac{m}{n} \cdot T_{HP}$ um, ergibt sich

$$f_{HP,LP} = \frac{i}{T_{LP}} + \frac{k}{T_{HP}} = \frac{i}{\frac{m}{n} \cdot T_{HP}} + \frac{k}{T_{HP}} \quad (5.8)$$

$$f_{HP,LP} = \frac{i \cdot n + k \cdot m}{m \cdot T_{HP}} = \frac{i \cdot n + k \cdot m}{\Delta t} \quad (5.9)$$

Die zweite Gleichung zeigt, dass alle Frequenzen, bei der die Interaktionen stattfinden, ein Vielfaches der Frequenz $\frac{1}{\Delta t}$ des neuen Triggers ist.

In der Realität ist das Ganze noch etwas komplizierter, da bei beiden Rotoren zwölf Mal so viele Schaufeln und damit auch viel mehr Koinzidenz-Positionen vorhanden sind. Weiters kann die Drehzahl der Rotoren nur innerhalb gewisser Grenzen konstant gehalten werden, was auch im Post Processing berücksichtigt werden muss, da die beiden Wellensignale bei sehr hohen Frequenzen erfasst werden. Das bedeutet, die kleinen Drehzahlschwankungen müssen daher ausgeglichen werden, weil ansonsten die Intervalle zwischen den Koinzidenz-Positionen nicht konstant wären. Ein weiterer Punkt ist, dass die Schaufeln geometrisch nicht vollkommen identisch sind, was sich in Form von Peaks bei niedrigen Frequenzen bemerkbar macht. Wird ein RSA jedoch über eine genügend große Anzahl an Perioden durchgeführt, werden die Einflüsse der minimal abweichenden Schaufelgeometrie durch die Mittelung entfernt.

Wie schon beim Phase Averaging gezeigt, kann eine Größe im Strömungsfeld folgendermaßen aufgespalten werden:

$$p(t) = \bar{p} + \langle p(t) \rangle_{HP} + \langle p(t) \rangle_{LP} + \langle p(t) \rangle_{HP,LP} + p'(t) \quad (5.10)$$

Nach der Auswertung mittels Rotor Synchronic Averaging, setzt sich diese wie folgt zusammen:

$$\tilde{p}_{RSA} = \bar{p} + \langle p(t) \rangle_{HP} + \langle p(t) \rangle_{LP} + \langle p(t) \rangle_{HP,LP} \quad (5.11)$$

Mit Hilfe des RSA kann somit der Frequenzinhalt der zusammenhängenden Strukturen vollständig dargestellt werden und nur das Spektrum, das nach Durchführung des Rotor Synchronic Averaging erstellt wurde, enthält, abgesehen von den stochastischen Anteilen, die ja weggemittelt werden, die gesamte Energie des ursprünglichen Signals.

Zusammenfassend sind in Abbildung 5.8 noch einmal alle drei Auswertemethoden des Post Processings gegenüber gestellt.

(a) zeigt das Spektrum nach dem Phase Averaging mit LP Trigger, welches die Harmonischen der BPF der Niederdruckturbine beinhaltet, gleiches gilt für (b), jedoch für die Hochdruckturbine. Bei dem Spektrum (c), das auf dem RSA basiert, sind sowohl die Harmonischen der BPFs beider Rotoren zu sehen, als auch Peaks bei $BPF_{LP} + BPF_{HP}$, $BPF_{HP} + BPF_{LP}$ und $2 \cdot BPF_{LP} - BPF_{HP}$ und anderen linearen Kombinationen.

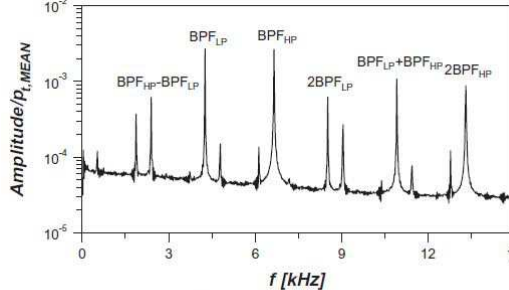
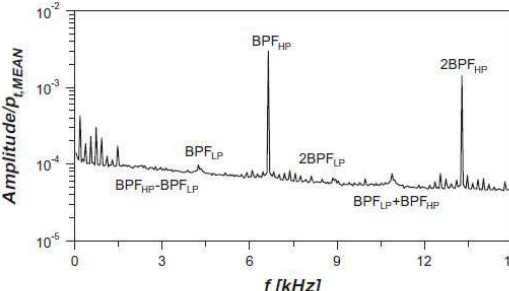
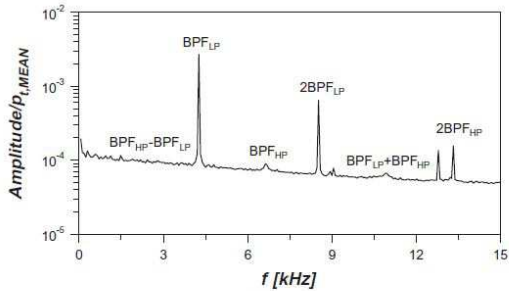


Abbildung 5.8: Dimensionslose Amplituden-Spektren nach Auswertung [3]

5.5 Azimutalmodenanalyse (AMA)

Die folgenden Abschnitte wurden aus den Veröffentlichungen [30], [31] und [32] entnommen, welche auch für eine intensivere Auseinandersetzung mit diesem Thema herangezogen werden können.

Ganz allgemein breiten sich im ringförmigen Strömungskanal die Schallwellen in Form von Moden, also Eigenlösungen, aus, die nach Lösen der Wellengleichung die Druckverteilung im Kanal darstellen. Als Modenordnung wird in weiterer Folge die Anzahl der Nullstellen des Druckverlaufs in azimutaler bzw. in radialer Richtung bezeichnet. Um die einzelnen Moden darstellen zu können, muss die azimutale bzw. die radiale Verteilung sowohl der Amplitude als auch der Phase von jeder spektralen Komponente bekannt sein.

Der komplexe Schalldruck kann nach Lösung der Wellengleichung in die einzelnen Azimutalmoden aufgelöst und danach folgendermaßen dargestellt werden:

$$\tilde{p}(x, r, \varphi, \omega) = \sum_{m=-\infty}^{m=\infty} A_m(x, r, \omega) e^{-im\varphi} \quad (5.12)$$

Mit A_m wird in obigem Zusammenhang die komplexe Amplitude der azimutalen Moden der Ordnung m , die in eine Vielzahl radialer Moden mit den Amplituden A_{mn}^+ und A_{mn}^- mit der radialen Ordnung n aufgeteilt werden kann, bezeichnet. Genauer gesagt wird mit folgender Gleichung die Amplitude der Modenausbreitung in einem zylindrischen Kanal für eine azimutale Modenordnung m über alle Radialmoden n zum Ausdruck gebracht:

$$A_m(x, r, \omega) = \sum_{n=0}^{n_{max}} f_{mn} \left(\sigma_{mn} \frac{r}{R} \right) \left[A_{mn}^+(\omega) e^{ik_{mn}^+ x} + A_{mn}^-(\omega) e^{-ik_{mn}^- x} \right] \quad (5.13)$$

Das + bedeutet hier eine Ausbreitung des Schalls in Strömungsrichtung, das – kennzeichnet eine Ausbreitung entgegen der Strömungsrichtung. k_{mn}^+ und k_{mn}^- stehen für die Wellenzahlen in x-Richtung und mit R wird der äußere Kanalradius bezeichnet. f_{mn} ist eine Zylinderfunktion und Lösung der Bessel'schen Differentialgleichung und beschreibt das radiale Schalldruckfeld für akustisch harte Kanalwände.

$$f_{mn} \left(\sigma_{mn} \frac{r}{R} \right) = \frac{1}{\sqrt{F_{mn}}} \left[J_m \left(\sigma_{mn} \frac{r}{R} \right) + Q_{mn} Y_m \left(\sigma_{mn} \frac{r}{R} \right) \right] \quad (5.14)$$

In diesem Ausdruck taucht neben der Bessel-Funktion der Ordnung m J_m auch die Neumann-Funktion der Ordnung m Y_m auf (siehe dazu Kap. 3.2), weiters

stehen σ_{mn} und Q_{mn} für die Eigenwerte, die sich aus den Randbedingungen an den als schallhart angenommenen Wänden und der Bedingung, dass die radiale Geschwindigkeit an den Kanalwänden Null ist, ergeben. Die Größe Q_{mn} ist Null für Kanäle, die nicht ringförmig sind. F_{mn} stellt eine Normalisierungsfunktion dar, mit der das für akustisch harte Wände vorliegende orthogonale in ein orthonormales Eigensystem transformiert wird.

$$F_{mn} = \begin{cases} \frac{1}{2} \left[1 - \left(\frac{r}{R} \right)^2 \right] & \text{bei } m = n = 0 \\ \frac{1}{2} \left[\left(1 - \frac{m^2}{\sigma_{mn}^2} \right) \hat{f}_{mn}^2(R) - \left(\frac{r^2}{R^2} - \frac{m^2}{\sigma_{mn}^2} \right) \hat{f}_{mn}^2(r) \right] & \text{sonst} \end{cases} \quad (5.15)$$

$$\hat{f}_{mn} \left(\sigma_{mn} \frac{r}{R} \right) = J_m \left(\sigma_{mn} \frac{r}{R} \right) + Q_{mn} Y_m \left(\sigma_{mn} \frac{r}{R} \right) \quad (5.16)$$

Die axialen Komponenten der Wellenzahl können mit M_x als Machzahl in x-Richtung und k als Wellenzahl folgendermaßen berechnet werden:

$$k_{mn}^{\pm} = \frac{k}{1 - M_x^2} \left(-M_x \pm \sqrt{1 - (1 - M_x^2) \frac{\sigma_{mn}^2}{(kR)^2}} \right) \quad (5.17)$$

Mit $\xi = \frac{\omega}{\omega_{cutoff}}$ und der Mach-Zahl M kann die vorige Gleichung auch anders dargestellt werden:

$$k_{mn}^{\pm} = \frac{-\xi M \pm \sqrt{\xi^2 - 1} \sigma_{mn}}{\sqrt{1 - M^2}} \quad (5.18)$$

Ist k_{mn} reell, dann kann sich eine Mode ausbreiten, ist sie komplex, so klingt die Mode exponentiell ab. Besonders deutlich wird dies in Gleichung 5.18: Ist $\xi < 1$, also die Frequenz kleiner als die Grenzfrequenz für cut-off bzw. cut-on, dann ist k_{mn} imaginär. Diese Eigenschaft der Wellenzahl lässt sich mit den Vorgängen bei einer gedämpften mechanischen Schwingung vergleichen: Ist die Dämpfung sehr groß, dann klingt die Schwingung sofort ab, es kommt zu einer so genannten Kriechbewegung. Bei kleinerer Dämpfung ist das System schwingfähig und die Amplitude der Schwingung nimmt exponentiell ab.

Der Bereich der azimuthalen Moden, der aus den experimentellen Daten bestimmt werden kann, ist durch das Nyquist-Theorem beschränkt. Das heißt eine Mode kann nur dann korrekt bestimmt werden, wenn ihre azimuthale Wellenlänge $\frac{2\pi}{m}$ größer als der doppelte Abstand zwischen zwei Messpunkten $\Delta\varphi$ ist. Daraus folgt für die maximale Modenordnung

$$\frac{2\pi}{m_{max}} > 2\Delta\varphi = 2 \left(\frac{2\pi}{N_\varphi} \right) \rightarrow m_{max} < \frac{N_\varphi}{2} \quad (5.19)$$

N_φ ist die Anzahl der Messpunkte in azimuthaler Richtung.

Um nun A_m in Gleichung 5.12 zu ermitteln, gibt es mehrere Möglichkeiten für die Durchführung der Azimutalmodenanalyse, die allesamt in [31] untersucht und bewertet wurden. Laut dieser Veröffentlichung von Sijtsma und Zillmann eignet sich das Kreuzkorrelationsverfahren mit Referenzsignal nach Gleichung 5.20 für den in dieser Arbeit vorliegenden Fall am besten zur Ermittlung der Moden.

$$a_m = \frac{1}{K} \sum_{k=1}^K \frac{p_k p_0^* e^{im\varphi_k}}{\sqrt{p_0 p_0^*}} \quad (5.20)$$

φ_k bezeichnet hier die Winkelposition des momentan betrachteten Mikrofons k , K steht für die Anzahl der Mikrofone und p_k stellt den komplexen Schalldruck dar, den man durch Auswerten der Messergebnisse mittels einer DFT für einen Block einer konstanten Anzahl an Samples erhält. p_0 ist der am Referenzmikrofon gemessene Druck.

Daraus lässt sich dann A_m wie folgt berechnen:

$$A_m = \frac{1}{2} |a_m|^2 \quad (5.21)$$

Dieses Verfahren beruht auf dem Zusammenhang zwischen dem Referenzsignal und den gemessenen Moden in azimuthaler Richtung, weshalb in Gleichung 5.20 auch der am Referenzkanal gemessene Schalldruck p_0 vorkommt. Das Referenzmikrofon ist in diesem Verfahren, wie auch im Prüfstand, fix an der Nabe montiert und wird nicht mit den anderen Mikrofonen traversiert.

Da beim Kreuzkorrelationsverfahren ein Referenzmikrofon als Bezug herangezogen wird, kann diese Methode auch für Frequenzen angewendet werden, die nicht mit der Rotordrehzahl korreliert sind. Würde man eine Vorgehensweise wählen, bei der beispielsweise der Trigger das Bezugssignal darstellt, so könnten nur diejenigen Moden festgestellt werden, deren Frequenzen mit der Drehzahl des entsprechenden Rotors in Zusammenhang stehen. Der Vorteil dieses Vorgehens wäre, dass störende Einflüsse, wie etwa turbulente Druckfluktuationen, die nicht an die Drehzahl gekoppelt sind, unterdrückt werden könnten.

Mit den vorliegenden Gleichungen und Zusammenhängen kann nun für jede Modenordnung in dem vom Nyquist-Kriterium eingeschränkten Bereich eine zugehörige Amplitude A_m bestimmt und zur Grafik 5.9 zusammengesetzt werden.

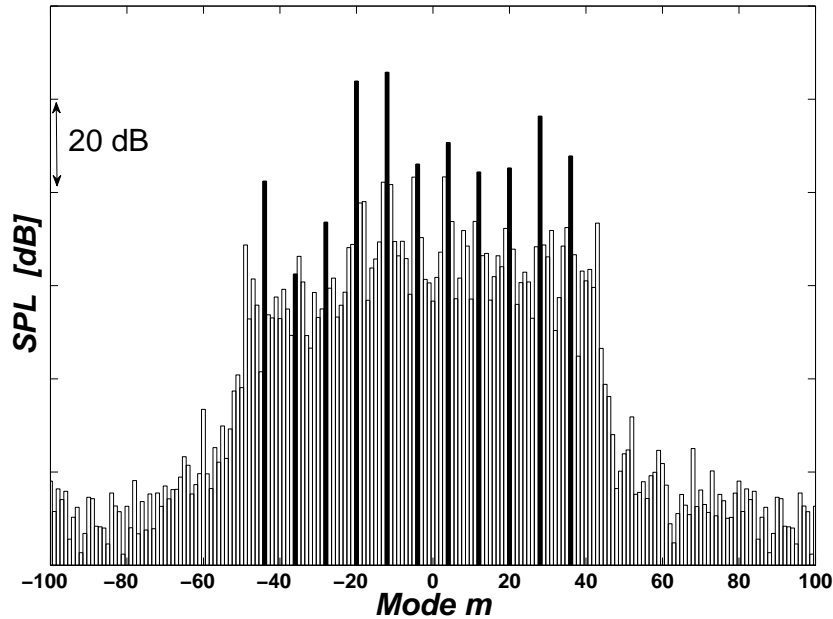


Abbildung 5.9: Azimutalmodenanalyse mit HP-Trigger

5.6 Radialmodenanalyse (RMA)

Basierend auf Gleichung 5.13, die nun für jede azimutale und jede radiale Position aufgestellt wird, erhält man ein Gleichungssystem für jede Azimutalordnung m :

$$\mathbf{A}_m = \mathbf{W}_m \cdot \mathbf{A} \quad (5.22)$$

\mathbf{A}_m bezeichnet hier den Vektor der gemessenen azimutalen modalen Amplituden (siehe Kapitel 5.5) mit der Länge $2 \cdot x_i$, \mathbf{W}_m die Matrix mit den akustischen Modellgleichungen mit der Dimension $2 \cdot x_i \times 2 \cdot n_{max}$ und \mathbf{A} steht für den Vektor mit den modalen Amplituden in radialer Richtung, die hier die gesuchten Unbekannten darstellen. Es kann sein, dass im Gleichungssystem 5.22 mehr Unbekannte als Gleichungen auftreten, weshalb man für die Lösung dann die Methode der kleinsten Fehlerquadrate verwendet.

$$\mathbf{A}_m = \begin{pmatrix} A_m(x_1 r_1) \\ \vdots \\ A_m(x_i r_1) \\ A_m(x_1 r_2) \\ \vdots \\ A_m(x_i r_2) \end{pmatrix} \quad (5.23)$$

$$\mathbf{W}_m = \begin{pmatrix} W_m^+(x_1 r_1 n_1) & W_m^-(x_1 r_1 n_1) & \cdot & \cdot & W_m^+(x_1 r_1 n_{max}) & W_m^-(x_1 r_1 n_{max}) \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ W_m^+(x_i r_1 n_1) & W_m^-(x_i r_1 n_1) & \cdot & \cdot & W_m^+(x_i r_1 n_{max}) & W_m^-(x_i r_1 n_{max}) \\ W_m^+(x_1 r_2 n_1) & W_m^-(x_1 r_2 n_1) & \cdot & \cdot & W_m^+(x_1 r_2 n_{max}) & W_m^-(x_1 r_2 n_{max}) \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ W_m^+(x_i r_2 n_1) & W_m^-(x_i r_2 n_1) & \cdot & \cdot & W_m^+(x_i r_2 n_{max}) & W_m^-(x_i r_2 n_{max}) \end{pmatrix} \quad (5.24)$$

$$\mathbf{A} = \begin{pmatrix} A_{mn}^+(n_1) \\ A_{mn}^-(n_1) \\ \cdot \\ \cdot \\ A_{mn}^+(n_{max}) \\ A_{mn}^-(n_{max}) \end{pmatrix} \quad (5.25)$$

Um nun den Vektor \mathbf{A} zu ermitteln, verwendet man, wie bereits erwähnt, die Methode der kleinsten Fehlerquadrate. Zu diesem Zweck wird die Pseudoinverse von \mathbf{W}_m benötigt, welche wie im Folgenden beschrieben, für die Lösung des Problems verwendet wird.

Nach Einführen der adjungierten Matrix von \mathbf{W}_m , die in weiterer Folge mit \mathbf{W}_m' bezeichnet wird, erhält man aus dem Zusammenhang 5.22 folgenden Ausdruck für \mathbf{A} :

$$\mathbf{A} = [\mathbf{W}_m' \mathbf{W}_m]^{-1} \mathbf{W}_m' \mathbf{A}_m \quad (5.26)$$

Die Multiplikation von \mathbf{W}_m und ihrer Adjunkten \mathbf{W}_m' ist jedoch nur dann möglich, wenn es sich um eine quadratische Matrix handelt und somit die Zeilen- und Spaltenzahlen zusammen passen, außerdem kann ihr Produkt nur dann invertiert werden, wenn die entsprechende Determinante ungleich Null ist. Anschließend lässt sich mit Hilfe der Pseudoinversen \mathbf{W}_m^+ der Vektor \mathbf{A} wie folgt berechnen:

$$\mathbf{A} = \mathbf{W}_m^+ \mathbf{A}_m \quad (5.27)$$

Die Schalleistung in und entgegen der Strömungsrichtung lautet dann

$$P_{mn}^\pm = \frac{\pi R^2}{\rho c} \frac{\alpha_{mn} (1 - M_x^2)^2}{(1 \mp \alpha_{mn} M_x)^2} |A_{mn}^\pm|^2 \quad (5.28)$$

mit

$$\alpha_{mn} = \sqrt{1 - (1 - M_x^2) \frac{\sigma_{mn}^2}{(kR)^2}} \quad (5.29)$$

Für die Pseudoinverse gilt also $\mathbf{W}_m^+ = [\mathbf{W}_m' \mathbf{W}_m]^{-1} \mathbf{W}_m'$. Im Gegensatz zu der oben beschriebenen Vorgehensweise der RMA beruhte die ursprünglich verwendete, konventionelle Methode, die zur Analyse der radialen Modenverteilung zum Einsatz kam, darauf, dass die azimutale Modenverteilung an einer einzigen axialen Position, jedoch an einer Vielzahl radialer Positionen bestimmt und daraus dann der Vektor \mathbf{A} ermittelt wurde. Das bedeutet auch, dass anstelle der im Rahmen dieser Arbeit verwendeten, axial und wandbündig monitierten Messsensoren, radiale Sensorrechen in Verwendung waren (siehe dazu auch Kapitel 4.2).

Warum auch die hier gezeigte Analysetechnik, bei der Mikrofone an verschiedenen axialen Positionen an einer Stelle mit konstantem radialen Abstand im Strömungskanal montiert wurden, für die RMA angewendet werden kann und dabei ebenfalls gute Ergebnisse erzielt werden, wird in [32] genauer erläutert. Wie aus dieser Arbeit von Enghardt, Zhang und Neise entnommen werden kann, würden im Prinzip zwei axial angeordnete Mikrofone reichen, um die RMA durchführen zu können, allerdings würde es in diesem Fall viel mehr Unbekannte als Gleichungen geben. Um nun Gleichung 5.27 lösen und die radiale Modenverteilung bestimmen zu können, verwendet man deshalb weitaus mehr als zwei Mikrofone mit unterschiedlichem axialen Abstand zur Schallquelle und kann somit das Gleichungssystem lösen. Sollen auch transmittierte und reflektierte Schallwellen in der Analyse berücksichtigt werden, muss die Anzahl der Messpositionen selbstverständlich dementsprechend angepasst werden.

Kapitel 6

Diskussion der Messergebnisse

Bevor nun in diesem Kapitel die Messergebnisse der drei verschiedenen Konfigurationen des TMTFs miteinander verglichen werden, sollen zunächst auch Messungen an verschiedenen Tagen hinsichtlich ihrer Vergleichbarkeit untersucht und die Resultate unter Verwendung verschiedener Referenzmikrofone gegenübergestellt werden. Außerdem wurden die Messdaten mittels verschiedener Auswerterroutinen nachbearbeitet und die Unterschiede der daraus resultierenden Spektren, Azimutal- und Radialmodenanalysen dargestellt. Details zu den einzelnen Untersuchungen sind in den jeweiligen Kapiteln zu finden.

6.1 Vergleich zwischen Messungen an verschiedenen Tagen und mit verschiedenen Referenzmikrofonen

Da es sich bei dem Prüfstand, an dem die Messungen durchgeführt wurde, um einen offenen Kreislauf handelt, bei dem Umgebungsluft angesaugt und später durch den Abluftkanal wieder abgegeben wird, musste überprüft werden, ob der Einfluss der Umgebungsbedingungen auf die Messergebnisse tatsächlich vernachlässigbar klein ist bzw. ob und wie groß eventuelle Abweichungen sind. Im Verlauf der Versuche hat sich gezeigt, dass ein- und dieselbe Messung mit derselben Konfiguration des TMTFs, jedoch an zwei verschiedenen Tagen durchgeführt, Resultate liefert, die im Bereich der Messungenauigkeit identisch sind. Das bedeutet einerseits, dass die einzelnen Messergebnisse reproduzierbar sind, auf der anderen Seite ist somit sichergestellt, dass die Daten von verschiedenen Versuchsfahrten auch tatsächlich miteinander verglichen werden können.

Die Abbildungen 6.1 und 6.2 zeigen Spektren nach Datenaufbereitung mittels Adaptiver Neuabtastung und Phase Averaging unter Verwendung des Hochdruck-Triggers. Vergleicht man nun die beiden Bilder, so sind zwischen den Peaks an

der Stelle der ersten Blade Passing Frequency des HP Rotors (6642 Hz) und ihrer ersten Harmonischen (13284 Hz) kaum Unterschiede festzustellen.

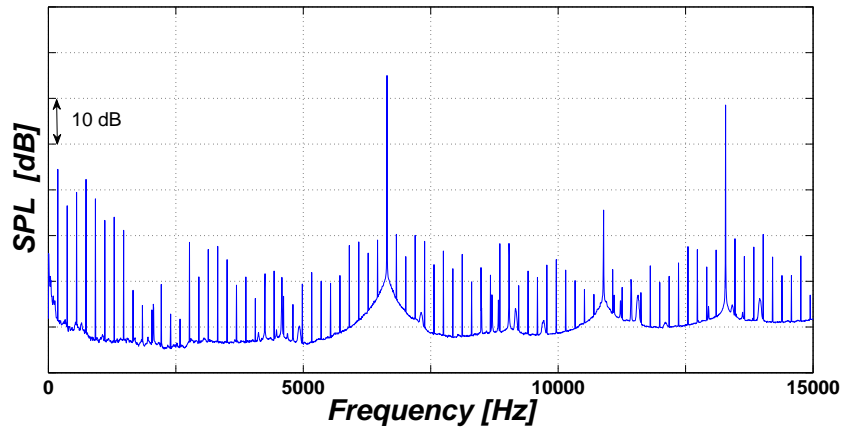


Abbildung 6.1: FFT mit HP Trigger, Tag 1

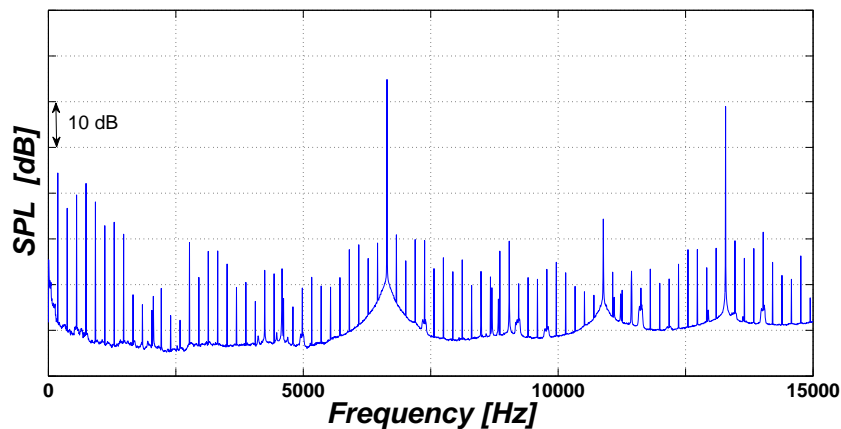


Abbildung 6.2: FFT mit HP Trigger, Tag 2

Beim Vergleich der Frequenzspektren, die mit zwei verschiedenen Referenzmikrofonen mit HP Trigger ermittelt wurden, hat sich herausgestellt, dass die Abweichungen bei der ersten und zweiten BPF des HP Rotors minimal sind und es daher keinen Unterschied macht, welches der beiden man als Referenz heranzieht. Da die Referenzmikrofone an der Nabe angebracht sind und es deshalb sehr schwierig ist sie auszutauschen, wurden bei der Inbetriebnahme des Prüfstands gleich zwei Sensoren montiert, falls einer davon beschädigt wird. Aus diesem Grund wurde hier nachgewiesen, dass man für den Fall, dass eines der beiden Mikrofone im Betrieb zerstört wird, bei der Messdaten-Auswertung problemlos und ohne Beeinträchtigung der Ergebnisse auf das andere umsteigen kann.

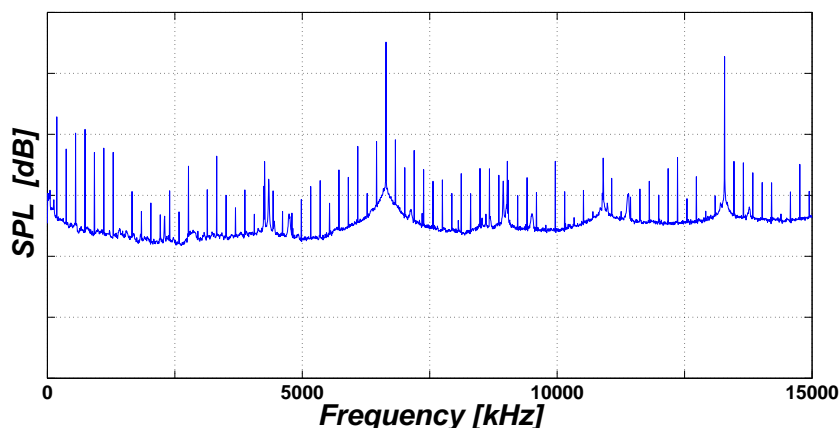


Abbildung 6.3: FFT des Referenzmikrofons am Kanal 25

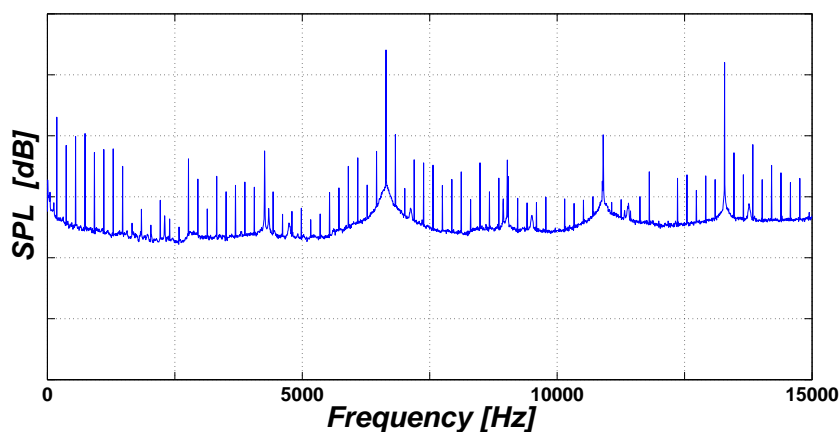


Abbildung 6.4: FFT des Referenzmikrofons am Kanal 26

6.2 Vergleich der FFT des Rohsignals und der FFT nach Datenaufbereitung

Um einzelne Mikrofone auf ihre Funktionalität hin überprüfen zu können bzw. um eine Vergleichsansicht zur FFT nach der Datenaufbereitung durch Adaptive Resampling etc. zur Verfügung zu haben, wurde vor Durchführung der Messungen ein Matlab-Programm erstellt, mit dem die reinen Rohdaten ausgewertet und eine FFT derselben durchgeführt und dargestellt werden kann.

Dies wurde zuerst für jedes einzelne Mikrofon an jeder Messposition in Umfangsrichtung gemacht - Abbildung 6.5 zeigt beispielsweise das Spektrum für Mikrofon 1 für Messpunkt 4 - und anschließend für das arithmetische Mittel im Zeitbereich aus allen 24 Mikrofonen und allen 180 Messpunkten (Abbildung 6.6). Diese Vorgehensweise wurde zuvor auch schon bei der Beschreibung der Auswerterroutinen in Kapitel 5.1 dargestellt.

Während sowohl in Abbildung 6.5, als auch in Abbildung 6.6 bereits die stark

ausgeprägten Grundfrequenzen sichtbar sind, zum Beispiel bei der ersten BPF des HP Rotors (6642 Hz), wird durch die zuvor erwähnte Mittelung der Rauschanteil in Abbildung 6.6 bereits deutlich verringert und die stochastischen Anteile des Signals nehmen ab. Je größer die Anzahl der Mikrofone bzw. die Anzahl der Umfangspositionen, über die die Mittelung durchgeführt wird, umso geringer wäre das Rauschen.

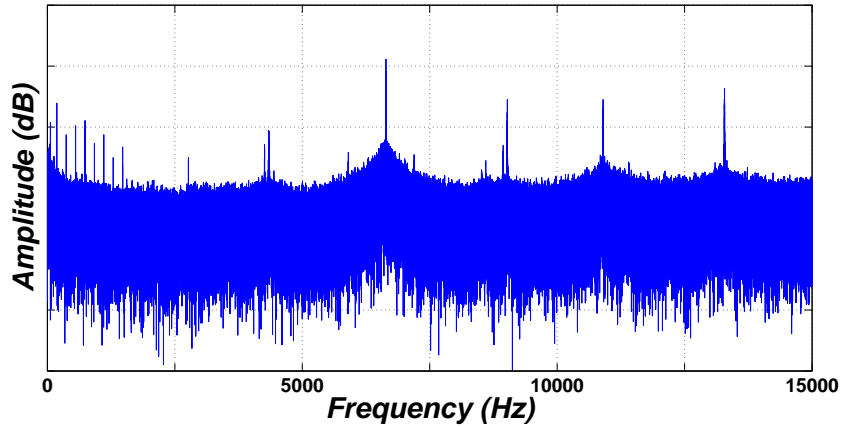


Abbildung 6.5: FFT des Rohsignals für ein Mikrofon und eine Messposition

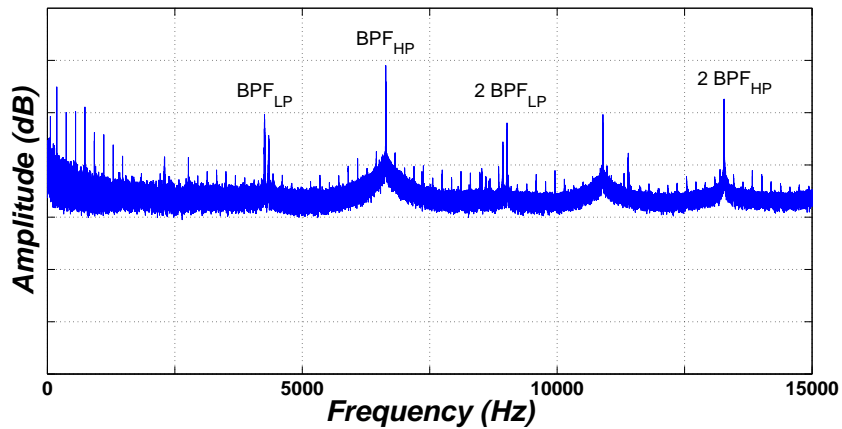


Abbildung 6.6: FFT des Rohsignals gemittelt über alle 24 Mikrofone und alle 180 Messpunkte

Auch ohne Aufbereitung der Messdaten bzw. nach einfacher Mittelung sind, wie schon beschrieben, in den zwei oberen Bildern deutlich die Peaks bei erster und zweiter Blade Passing Frequency sowohl des Hochdruck- als auch des Niederdruck-Rotors zu erkennen, welche auch in Abbildung 6.6 kenntlich gemacht wurden.

Abbildung 6.7 zeigt die FFT nach Datenaufbereitung und zwar mit HP Trigger. Man erkennt hier im Vergleich zur FFT des Rohsignals noch deutlicher die Peaks bei der ersten bzw. zweiten BPF des Hochdruck-Rotors, die Spitzen hingegen, die mit der Drehzahl des Niederdruck-Rotors korreliert sind, werden hier ganz eindeutig unterdrückt. Umgekehrtes gilt natürlich für die Auswertung mit dem LP Trigger.

Neben den Effekten des Niederdruck-Rotors verschwinden im Spektrum aufgrund der Mittelung auch die in Abbildung 6.5 noch deutlich sichtbaren Rauschanteile.

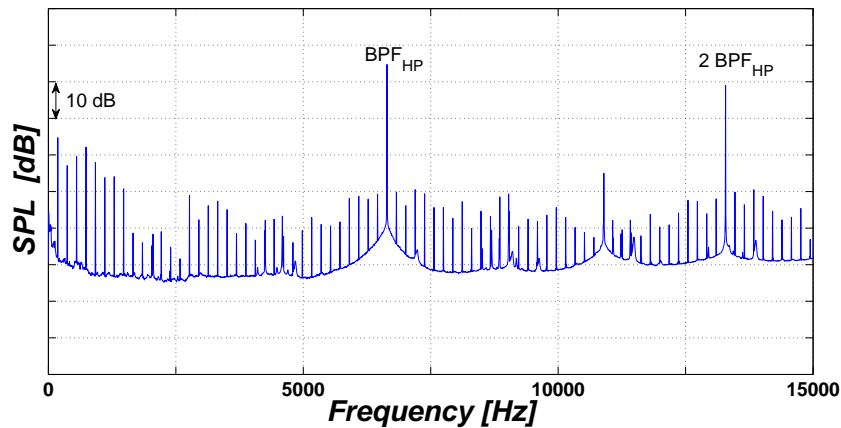


Abbildung 6.7: FFT nach Datenaufbereitung durch Adaptive Resampling und Phase Averaging mit HP Trigger

6.3 Verwendung verschiedener Trigger

Um die Effekte der einzelnen Rotoren zu separieren, bzw. um eine Auswertung durchzuführen, die auch die Interaktionen zwischen den Rotoren berücksichtigt, wurden verschiedene Trigger - einer für den HP und einer für den LP Rotor - bzw. ein Verfahren namens Rotor Synchronic Sveraging verwendet (die Theorie dazu ist in den Kapiteln 5.2, 5.3 und 5.4 zu finden). Die folgenden Abschnitte zeigen die Unterschiede zwischen den drei genannten Auswertemethoden.

6.3.1 Hochdrucktrigger

Bei der Verwendung verschiedener Trigger sind die Unterschiede unter anderem im Frequenzspektrum sehr gut zu erkennen. Wie bereits im Abschnitt 6.2 er-

läutert, verschwinden bei Verwendung des HP Triggers die Effekte des LP Rotors und die Peaks treten fast ausschließlich bei den mit dem Hochdruck-Rotor korrelierten Frequenzen auf. Nach Aufbereitung der Messdaten mittels Adaptive Resampling und Phase Averaging wurde unter Anwendung des HP Triggers das Spektrum in Abbildung 6.8 erstellt und die Peaks bei der ersten bzw. zweiten BPF gekennzeichnet.

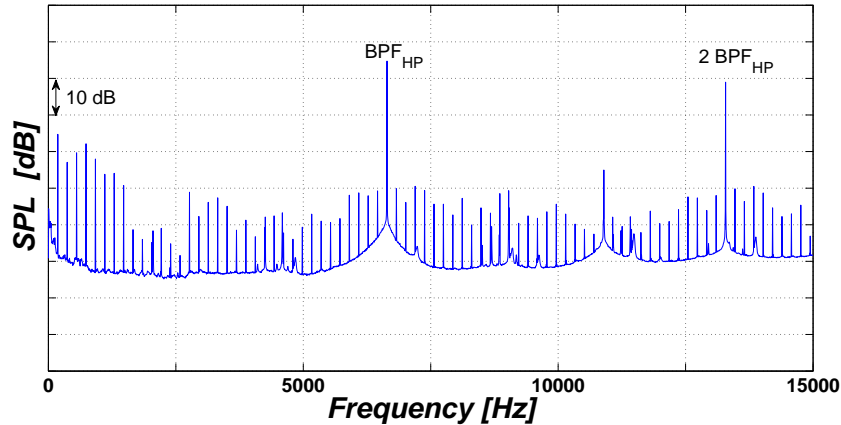


Abbildung 6.8: FFT mit Hochdruck-Trigger

Die nachfolgende Azimutalmodenanalyse (siehe Kapitel 5.5) wurde nur für die hinteren 12 Mikrofone durchgeführt, also diejenigen, die sich schon im geraden Teil des Strömungskanals befinden. Die vorderen 12 Mikrofone, die noch im nicht-zylindrischen Abschnitt des Kanals liegen, wurden hier nicht berücksichtigt, da in den Auswerteroutinen vereinfachend von einem zylindrischen Strömungskanal ausgegangen wird und daher der Einfluss der vorderen 12 Sensoren auf die Messergebnisse nicht abgeschätzt werden konnte.

Welche Moden ausbreitungsfähig sind, bzw. welche davon abklingen, kann bereits vor der Auswertung nach Tyler und Sofrin (Gleichung 3.67) und unter Berücksichtigung der cut-off Bedingung theoretisch vorhergesagt werden.

Für die Interaktionen zwischen HP Stator und HP Rotor gilt beispielsweise laut Tyler und Sofrin:

$$m = 36 + k \cdot 24 = \dots - 36, -12, 12, 36, \dots \quad (6.1)$$

Die angeregten Moden aufgrund der Interaktionen zwischen HP Stator, HP Rotor und LP Stator lassen sich demnach, basierend auf der Arbeit von Holste und Neise (siehe [30]) als Erweiterung der Theorie von Tyler und Sofrin, folgendermaßen berechnen:

$$m = 36 + k_1 \cdot 24 + k_2 \cdot 16 = \dots -44, -36, -28, -20, -12, -4, 4, 12, 20, 28, 36, \dots \quad (6.2)$$

Grundsätzlich würde m von $-\infty$ bis $+\infty$ reichen, die größte ausbreitungsfähige Modenordnung wird jedoch durch die cut-off Bedingung festgelegt. Die in Gleichung 6.2 berechneten Moden sind auch in Abbildung 6.9, in der alle ausbreitungsfähigen Moden schwarz markiert wurden und die modale Zerlegung für die erste BPF über der azimuthalen Modenordnung aufgetragen ist, deutlich zu erkennen. Die ausbreitungsfähigen Moden reichen eindeutig von -44 bis 36, alle anderen Moden weisen deutlich kleinere Amplituden auf und werden laut cut-off Bedingung im Strömungskanal abgedämpft. Hier sei auf die Berechnung der cut-off Frequenz in Gleichung 3.47 verwiesen. Zusammenfassend bedeutet das, dass der vom Hochdruck-Rotor kommende Schall nicht alleine betrachtet werden darf, sondern auch seine Wechselwirkungen mit den Statorschaufeln stromab- und stromaufwärts miteinbezogen werden müssen.

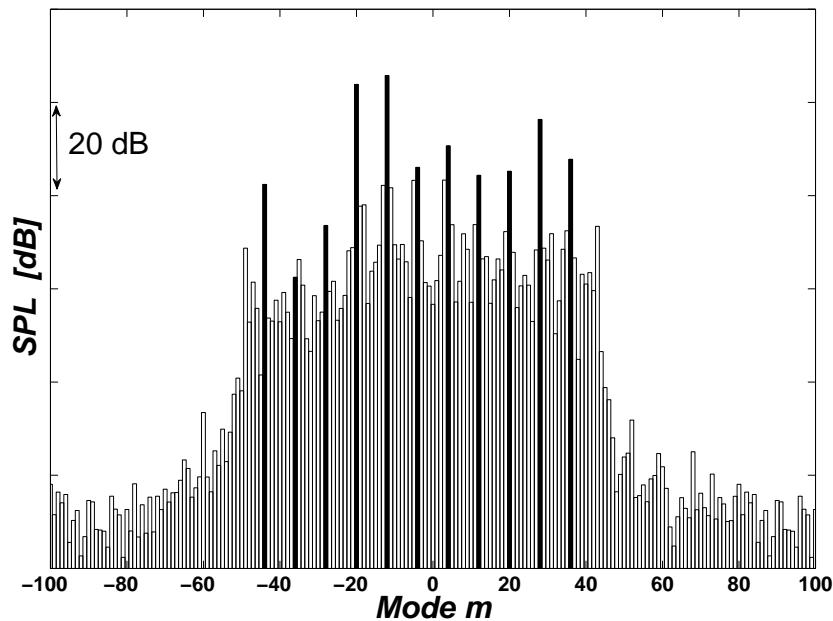


Abbildung 6.9: Azimutalmodenanalyse mit Hochdruck-Trigger

Welche der Moden ausbreitungsfähig sind, ist außerdem aus Abbildung 6.10 ersichtlich, da für alle nicht ausbreitungsfähigen Moden, also alle deren Wellenzahl imaginär ist, die Schalleistung Null gesetzt wurde (siehe dazu auch die Programmdokumentation im Anhang). Aufgetragen wurde hier die Schalleistung - sowohl in (P_m^+) als auch entgegen (P_m^-) der Strömungsrichtung - über der azimuthalen Modenordnung. Zusätzlich angegeben sind auch die Summenwerte für die Schalleistung, die unter der Grafik zu sehen sind. Was die Darstellung der Schalleistung betrifft, so sind die Werte bei Anwendung des HP Triggers deutlich größer als bei der Auswertung mit LP Trigger. Diese Tendenz, dass ganz allgemein die Werte der Messdaten-Analyse mit HP Trigger über denen mit LP Trigger liegen, war auch schon bei der FFT und der AMA ersichtlich und wird

nachfolgend noch erläutert. Die Ergebnisse mit HP Trigger, die mit bis zu 140 dB im Bereich der Schalleistungen nach RSA (siehe Abb. 6.16) liegen, sprechen für den großen Einfluss des HP Rotors und seiner Interaktionen auf das Gesamtergebnis.

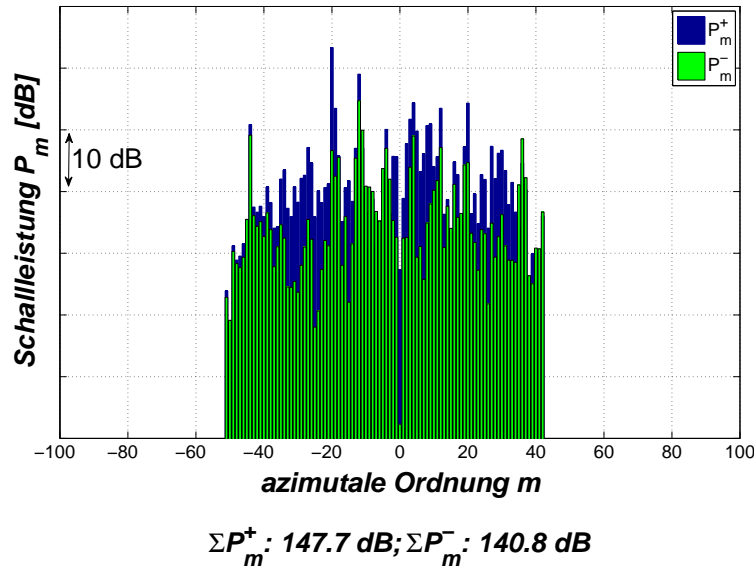


Abbildung 6.10: Schalleistung für 1. BPF mit Hochdruck-Trigger

6.3.2 Niederdrucktrigger

Vergleicht man das nachfolgende Spektrum mit Niederdruck-Trigger mit dem aus dem vorhergehenden Abschnitt, so erkennt man sofort die Peaks bei erster und zweiter BPF des Niederdruck-Rotors, die in der Abbildung auch deutlich gekennzeichnet wurden. Nach wie vor sichtbar sind auch die Spitzen bei der ersten und zweiten BPF des Hochdruck-Rotors, allerdings ist ihre Amplitude im Vergleich zum Spektrum mit HP Trigger deutlich reduziert. Eine mögliche Erklärung dafür, dass die Peaks des HP Rotors trotz LP Trigger sichtbar sind, ist die Periodizität zwischen HP und LP Rotor. Das bedeutet, zwischen zwei Triggersignalen des LP Rotors kommen immer wieder Anteile des HP Rotors durch. Da der Einfluss des Hochdruck-Rotors sehr stark ist (siehe letzter Absatz in Kapitel 6.3.1), werden diese Anteile bei der Mittelung über eine gewisse Anzahl an Blöcken - detailliert beschrieben wird dieses Vorgehen im Anhang beim Abschnitt über das Phase Averaging - nur abgedämpft und nicht vollständig entfernt, wie das beim Auswerten mit HP Trigger für die Peaks des LP Rotors der Fall ist.

Das Bild für die AMA ist im Prinzip gleich aufgebaut wie bei der Auswertung mit HP Trigger, ebenso lassen sich die angeregten Modenordnungen, beispielsweise

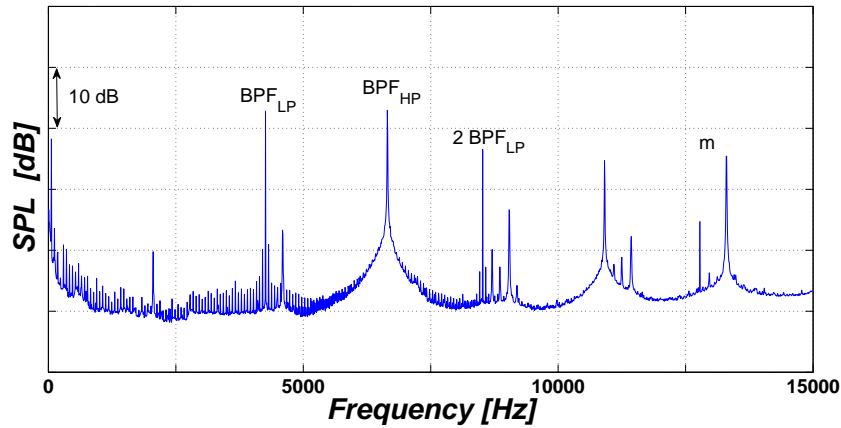


Abbildung 6.11: FFT mit Niederdruck-Trigger

für das Zusammenspiel von LP Stator und LP Rotor, nach Tyler und Sofrin berechnen (Gleichung 6.3).

$$m = -72 + k \cdot 16 = \dots - 72, -56, -40, -24, -8, 8, 24, \dots \quad (6.3)$$

Die Modenordnungen für die Interaktionen zwischen LP Stator, LP Rotor und HP Stator ergeben sich schließlich nach Holste und Neise zu

$$m = -72 + k_1 \cdot 16 + k_2 \cdot 24 \quad (6.4)$$

Abbildung 6.12 zeigt die Azimutalmodenanalyse nach Signalaufbereitung und unter Verwendung des LP Triggers, analog zum vorigen Abschnitt wird auch hier der Schalldruckpegel über der azimutalen Modenordnung aufgetragen und in Form eines Balkendiagrammes dargestellt. Die ausbreitungsfähigen Moden, die aus den Interaktionen zwischen LP Stator und LP Rotor sowie zwischen LP Stator, LP Rotor und HP Stator stammen, wurden mit schwarzen Balken gekennzeichnet. Die Moden -72, -56 und -40 sind in dieser Abbildung zwar noch deutlich sichtbar, erfüllen jedoch die cut-on Bedingung nicht und liegen somit außerhalb des ausbreitungsfähigen Bereichs. Zu sehen ist diese Tatsache auch, wie bereits bei der Auswertung mit HP Trigger erklärt, in der Darstellung der Schalleistung in und entgegen der Strömungsrichtung in Abbildung 6.13, in der bei allen Moden kleiner -32 und größer 28 die Schalleistung Null ist.

Die Schalleistung weist hier im Vergleich zur Auswertung mit HP Trigger im Allgemeinen ein deutlich niedrigeres Niveau auf (Werte bis maximal 100 dB), die signifikanten, ausbreitungsfähigen Moden aus Abbildung 6.12, wie z.B. bei $m=-24$ oder $m=-8$, sind auch in der Darstellung der Schalleistung (Abbildung 6.13) deutlich sichtbar. Gezeigt wird auch hier die Schalleistung in und entgegen der

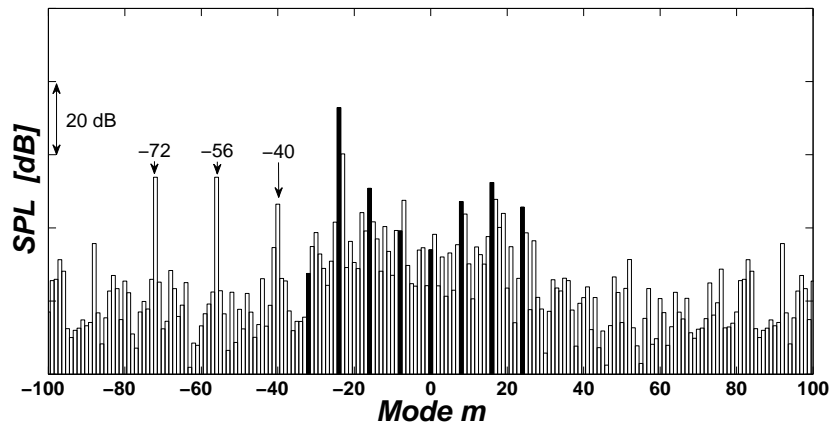


Abbildung 6.12: Azimutalmodenanalyse mit Niederdruck-Trigger

Strömungsrichtung, aufgetragen über der Modenordnung in azimuthaler Richtung, und ihre zugehörigen Summenwerte ($\sum P_m^+$ und $\sum P_m^-$).

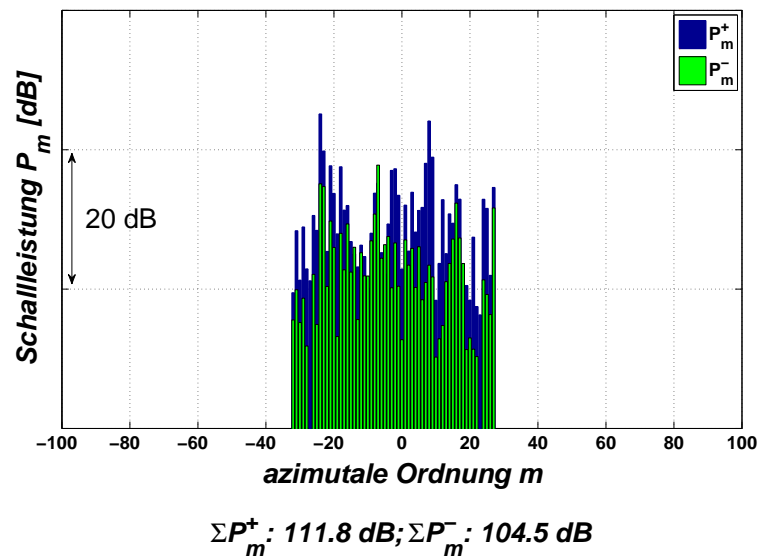


Abbildung 6.13: Schalleistung für 1. BPF mit Niederdruck-Trigger

6.3.3 Rotor Synchronic Averaging

Im Spektrum nach dem Rotor Synchronic Averaging ist, wie in Kapitel 5.4 erklärt, abgesehen von den stochastischen Schwankungen, die gesamte Energie des ursprünglichen Signals enthalten und somit Ausschläge bei allen Frequenzen, die den Drehzahlen der beiden Rotoren zugeordnet werden können, vorhanden. Neben den beiden Grundfrequenzen und ihren ersten Harmonischen sind außerdem die Interaktionen zwischen HP und LP Stufe in Form von Summen bzw. Differenzen zwischen den korrelierten Frequenzen sichtbar (1.BPF HP + 1.BPF LP bzw. 2.BPF HP - 1.BPF LP).

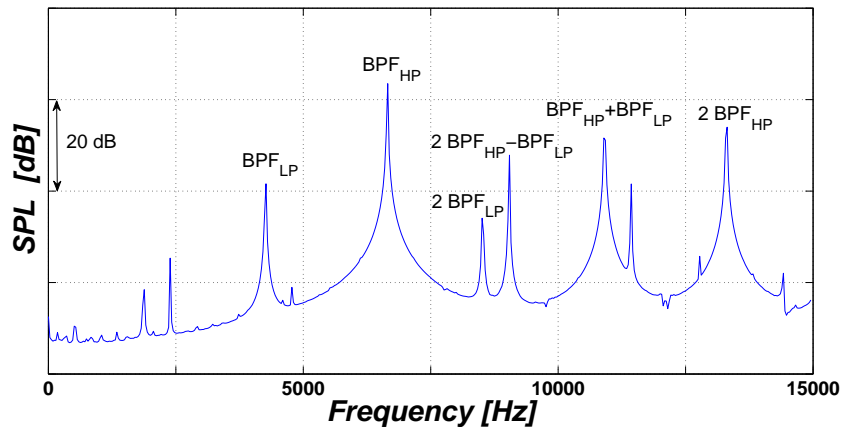


Abbildung 6.14: FFT nach Rotor Synchronic Averaging

Es fällt auf, dass sich die Amplitude bei der ersten und zweiten BPF des Hochdruck-Rotors auf fast gleichem Niveau befindet wie bei der Auswertung mit dem Hochdruck-Trigger. Der Schalldruckpegel (SPL) zeigt mit einem Wert von etwa 140 dB bei der ersten BPF also, dass der HP Rotor den größten Einfluss auf die Schallemissionen hat, was bereits Abbildung 6.8 erahnen lässt.

Was die Blade Passing Frequencies des Niederdruck-Rotors betrifft, so stimmt nur die zweite in etwa mit den Ergebnissen nach Verwendung des LP Triggers überein (knapp unter 120 dB), die erste ist nach Auswertung mittels RSA deutlich höher und liegt ungefähr bei 130 dB.

Im Vergleich zur Auswertung mit HP bzw. LP Trigger weisen die Interaktionen ein deutlich höheres Niveau auf (ca. 130 bzw. 140 dB), außerdem ist der Schalldruckpegel bei der Frequenz 2.BPF HP - 1.BPF LP in etwa vergleichbar mit dem Level bei der Summe der Frequenzen.

Abbildung 6.15 zeigt nun die Azimutalmodenanalyse nach RSA und aufgrund der Betrachtung beider Rotoren und ihrer Wechselwirkungen somit deutlich mehr ausbreitungsfähige Moden als die vorhergehenden Bilder der AMA. Diese große Bandbreite an ausbreitungsfähigen Moden kommt auch in der Darstellung der Schalleistung in Abbildung 6.16 zum Ausdruck.

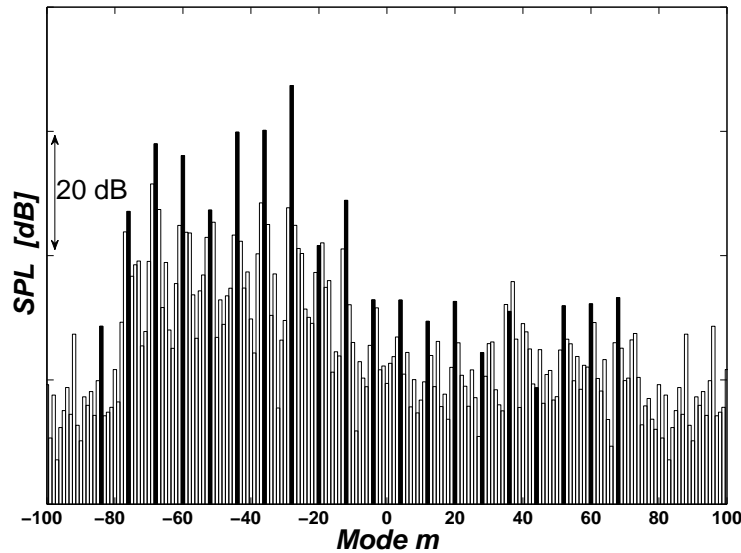


Abbildung 6.15: Azimutalmodenanalyse nach Rotor Synchronic Averaging

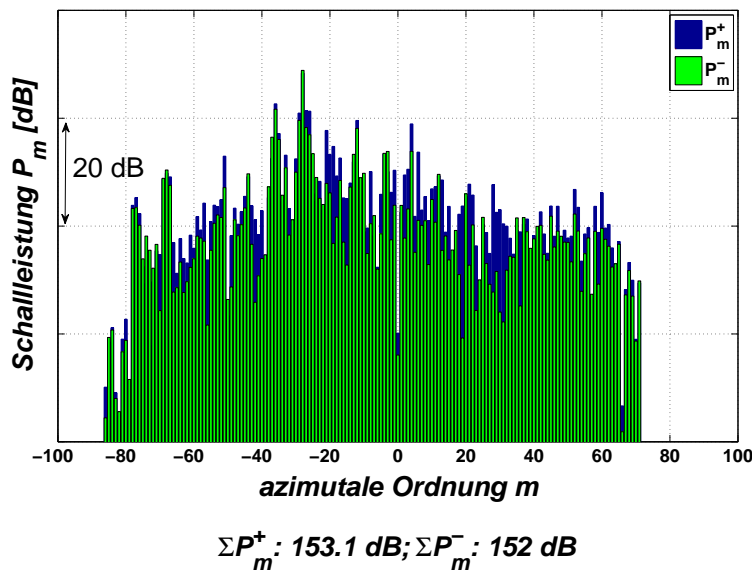


Abbildung 6.16: Schalleistung nach Rotor Synchronic Averaging für Summe der 1. BPFs

Welche der drei Möglichkeiten der Signalaufbereitung - Auswertung mit HP Trigger, mit LP Trigger oder nach Rotor Synchronic Averaging - hängt nun davon ab, welche Informationen man erhalten möchte. Wird eine Darstellung, die bis auf die stochastischen Schwankungen die gesamte Energie des ursprünglichen Signals enthält und auch die Interaktionen zwischen den Rotoren darstellt, gewünscht, so sollte eine auf der RSA basierende Analyse durchgeführt werden. Verwendet man an Stelle der RSA einen der beiden Trigger, werden weder Interaktionen

noch die gesamte Energie des Signals dargestellt, jedoch können die Effekte der einzelnen Rotoren separiert und unabhängig voneinander betrachtet werden. Um das Maximum an Information gewinnen zu können, wurden aus diesem Grund im Rahmen der hier durchgeführten Akustikmessungen jeweils alle drei Methoden angewandt und deren Ergebnisse gemeinsam betrachtet.

6.4 Vergleich der drei Konfigurationen

In den nachfolgenden Spektren werden nun jeweils die Graphen aller drei Konfigurationen in einem Bild dargestellt, die aufgrund der besseren Lesbarkeit absichtlich gegeneinander verschoben wurden.

Vergleicht man das MTU Setup (C1) mit dem TMTF von Volvo (C2), so unterscheiden sich bei der Auswertung mit HP Trigger ihre Amplituden im Spektrum bei der ersten BPF um etwa 5 dB, während der Wert für C3 (MTU Setup mit Splitter) genau dazwischen liegt. Bei der zweiten BPF sind die Unterschiede marginal, vor allem zwischen C2 und C3.

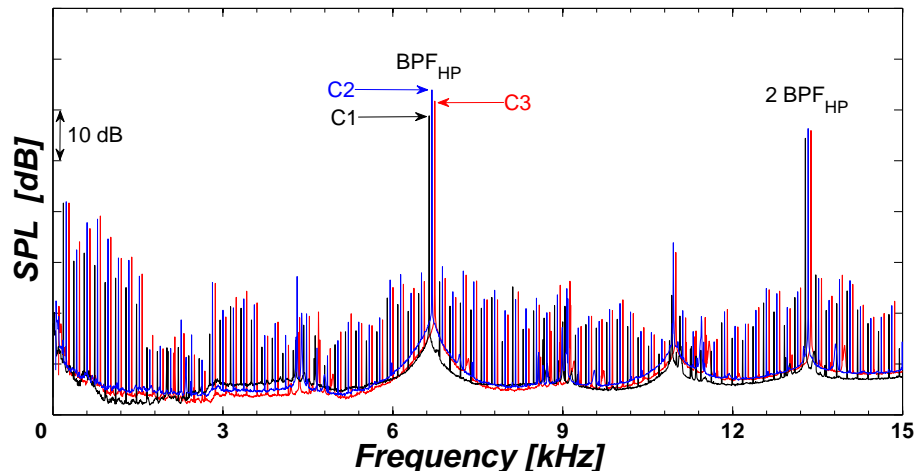


Abbildung 6.17: Vergleich der Frequenzspektren von C1, C2 und C3 mit HP Trigger

Bei der Auswertung mittels LP Trigger (Abbildung 6.18) sind die Abweichungen zwischen den drei Konfigurationen schon sichtbar größer, hier liegt der Schalldruckpegel von C2 etwa 10 dB über dem von C1. Die Setups C1 und C3 hingegen liegen nur wenige dB auseinander. Bei der zweiten BPF sind wiederum kaum Unterschiede festzustellen.

Abgesehen von einem vertikalen Offset von C2 nach der Auswertung mittels RSA, zeigt das Spektrum in Abbildung 6.19 eine sehr gute Übereinstimmung bei den Peaks von zweiter BPF des LP Rotors sowie beim Differenzton bei etwa 9 kHz. Große Abweichungen zeigen sich bei der ersten BPF des LP Rotors, bei der der

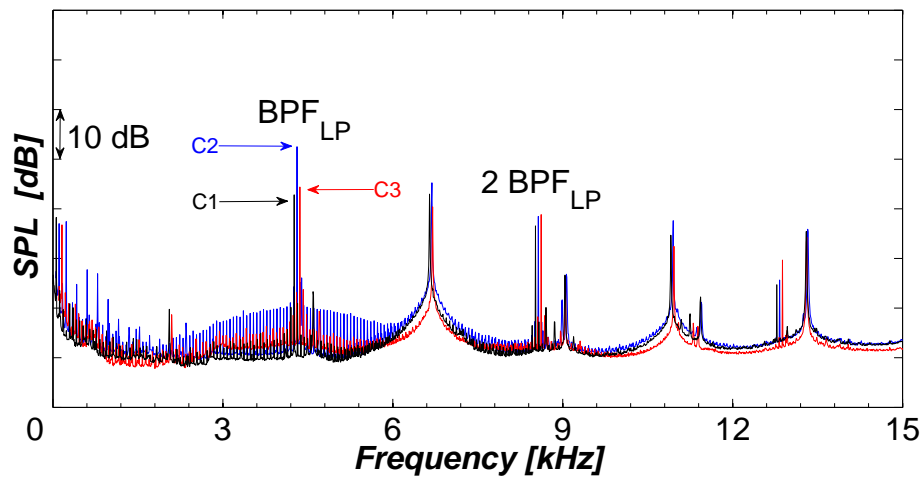


Abbildung 6.18: Vergleich der Frequenzspektren von C1, C2 und C3 mit LP Trigger

Wert des Volvo Setups (blauer Graph) etwa 10 dB über dem des MTU Setups (schwarzer Graph) liegt, während C1 und C3 sehr nah beieinander liegen. Bei erster bzw. zweiter BPF des HP Rotors liegt C3 jeweils wenige dB unterhalb des Wertes der beiden anderen Konfigurationen, beim Summenton bei circa 11 kHz jedoch übersteigt C2 die anderen zwei Setups um beinahe 5 dB.

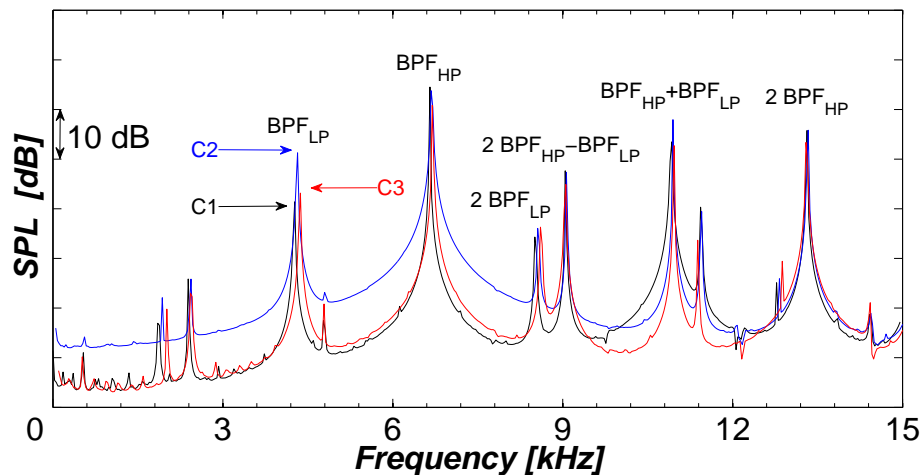


Abbildung 6.19: Vergleich der Frequenzspektren von C1, C2 und C3 nach RSA

Die nach Betrachtung der Frequenzspektren zu erkennende Tendenz, nämlich die der deutlich größeren Lärmentwicklung der Konfiguration C2 in Vergleich zu C1 und C3, lässt sich unter anderem auf die verringerte axiale Ausdehnung dieses Setups zurückführen, aufgrund derer die Vermischung der Schaufel-Nachläufe weniger fortgeschritten, die Nachlaufdellen also noch stärker ausgeprägt sind als bei C1 bzw. C3 und dadurch die Lärmgenerierung durch Wirbel gefördert wird.

Bei der experimentellen Untersuchung der Setups von MTU und Volvo in [15] wurden bei C2 außerdem schärfere und geradere Nachläufe festgestellt als bei C1 und C3, deren Auftreffen auf den nachfolgenden Schaufeln des Niederdruck-Rotors Ursache für den erhöhten Schalldruckpegel sein könnten. Bei diesen Analysen wurde im Nachlauf von C2 zusätzlich ein Bereich mit niedrigen Mach-Zahlen bedingt durch einen Wirbel gefunden, was ebenfalls einen Einfluss auf die Schallentwicklung haben könnte.

Vergleicht man nun die Ergebnisse der Azimutalmodenanalyse mit HP Trigger, so fallen bei den Moden $m=-44$, $m=-28$ und $m=28$, die nach Gleichung 6.2 aus den Interaktionen zwischen Stator, Rotor und Stator stammen und in den Abbildungen 6.20, 6.21 und 6.22 markiert sind, große Unterschiede auf. Die stärksten Abweichungen sind bei $m=-44$ zu bemerken, wo die Amplitude von C1 in etwa 20 dB über der von C2 liegt. Ganz allgemein betrachtet liegt jedoch der Summenschalldruckpegel beim Setup von Volvo (C2) im Bereich der ausbreitungsfähigen Moden über dem des MTU und des TTM Setups, was sich auch in einem um 8 dB höheren Moden-Mittelwert zeigt, der in diesen Abbildungen jedoch nicht explizit angegeben wurde.

Für das Setup C3 ergibt sich nun das Problem, welche Schaufelzahl man für den LP Stator zur Modenberechnung nach Tyler und Sofrin bzw. Holste und Neise verwendet, da zusätzlich zu den 16 Struts 32 Splitter eingefügt wurden, die jedoch über eine andere Geometrie verfügen als die Struts. Als erste Näherung wurde für den LP Stator der Idealfall von 48 Stator-Schaukeln angenommen und die angeregten Moden für die Interaktionen zwischen HP Stator, HP Rotor und LP Stator folgendermaßen berechnet:

$$m = 36 + k_1 \cdot 24 + k_2 \cdot 48 = \dots - 36, -12, 12, 36, \dots \quad (6.5)$$

Eine andere Möglichkeit der näherungsweise Modenberechnung stellt folgender Zusammenhang dar, bei dem die Anzahl der Stator-Schaukeln als ein Vielfaches der Anzahl der Splitter angenommen wird:

$$\begin{aligned} m &= 36 + k_1 \cdot 24 + k_2 \cdot 48 + k_3 \cdot 16 \\ &= \dots - 44, -36, -28, -20, -12, -4, 4, 12, 20, 28, 36, \dots \end{aligned} \quad (6.6)$$

Interessant ist hier, dass die aussagekräftigen Moden wie z.B. an der Stelle $m=-44$

oder $m=-28$ bei der Berechnung mittels der idealen Schaufelzahl des LP Stators (48 Schaufeln, Gleichung 6.5) nicht auftreten, aus diesem Grund wurde auch die Berechnungsmethode in Gleichung 6.6 vorgezogen.

Vergleicht man nun die AMA von C3 mit der von C1, so ist der Schalldruckpegel an der Stelle $m=-44$ bei C3 etwa um 20 dB niedriger als bei C1, ebenso ist die Amplitude bei $m=-28$ und $m=28$ um ungefähr 10 dB reduziert. Der außerdem um ca. 5 dB niedrigere Moden-Mittelwert des Setups C3 lässt sich auf die Splitter zurückführen, die hier offensichtlich eine dämpfende Wirkung haben und somit zur Senkung des mittleren Schalldruckpegels beitragen.

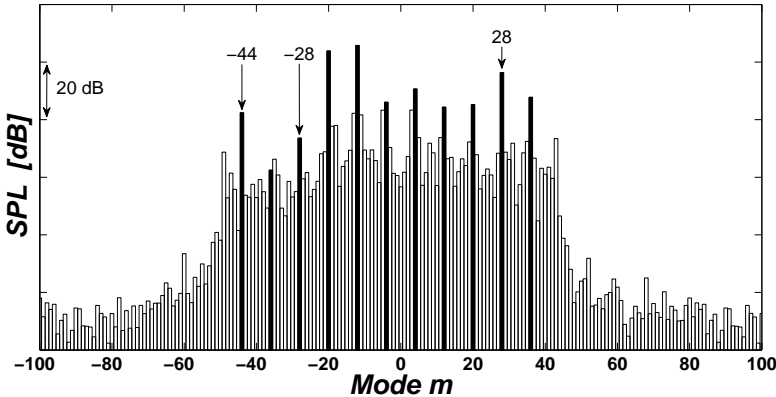


Abbildung 6.20: Azimutalmodenanalyse mit HP Trigger vom Setup C1

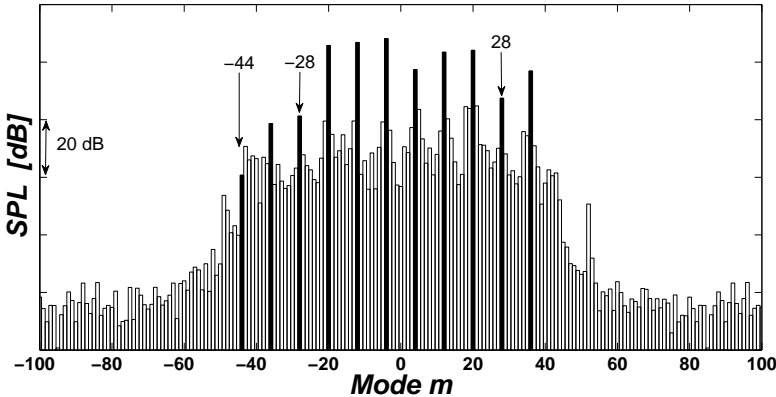


Abbildung 6.21: Azimutalmodenanalyse mit HP Trigger vom Setup C2

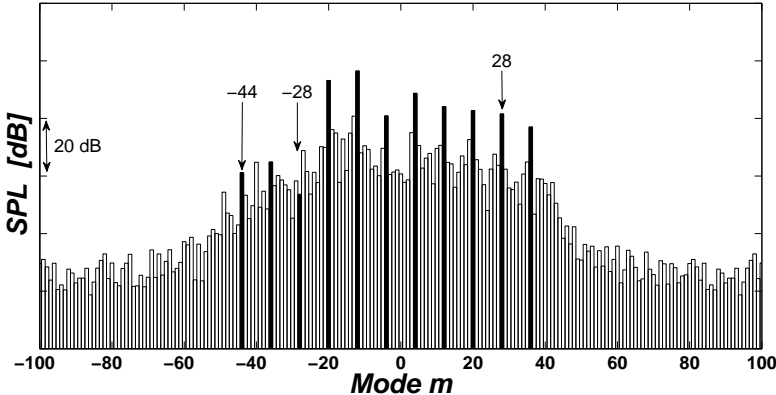


Abbildung 6.22: Azimutalmodenanalyse mit HP Trigger vom Setup C3

Auffallend bei der AMA mit LP Trigger (Abbildungen 6.23, 6.24 und 6.25) ist die Mode $m=-24$, die nach Gleichung 6.3 ihren Ursprung in den Wechselwirkungen zwischen LP Stator und LP Rotor hat. An dieser Stelle übersteigt der Wert von C2 den von C1 um mehr als 20 dB, zwischen C2 und C3 ist die Differenz sogar noch größer. Grund dafür könnte sein, dass beim Setup C2 der LP Stator näher am Rotor ist als bei C1 und C3 und daher die Vermischung der Schaufel-Nachläufe beim Auftreffen auf den Stator weniger ausgeprägt ist.

Besonders herausragend sind auch noch die Modenordnungen -8 und 0, die beim Setup C1 verglichen mit den anderen beiden Setups einen sehr niedrigen Wert aufweisen, der Schalldruckpegel bei Mode $m=16$ hingegen, die aus den Interaktionen zwischen LP Stator, LP Rotor und HP Stator stammt, ist bei C3 deutlich kleiner, und zwar um etwa 20 dB im Vergleich zu C1 und um etwa 25 dB im Vergleich zu C2. Auch bei der Auswertung mit LP Trigger ist, wie zuvor schon bei der Anwendung des HP Triggers, der Moden-Mittelwert für den Schalldruckpegel von C2 um mehrere dB größer als bei den anderen beiden Konfigurationen. Dasselbe gilt auch für die AMA nach RSA (Abbildungen 6.26, 6.27 und 6.28), wobei hier der Moden-Mittelwert von C2 ungefähr 14 dB über dem von C1 und etwa 12 dB über dem von C3 liegt.

Nach Durchführung der RSA stechen besonders die Moden im Bereich $m=-68$ bis $m=-28$ hervor, deren Amplituden bei C2 sichtbar größer sind als bei C1 und C3, des Weiteren ist die Mode $m=-20$ sehr markant, deren Wert des Setups C3 den von C2 zwar nur um wenige Dezibel, den von C1 jedoch um beinahe 20 dB übersteigt. Selbiges gilt auch für $m=-4$.

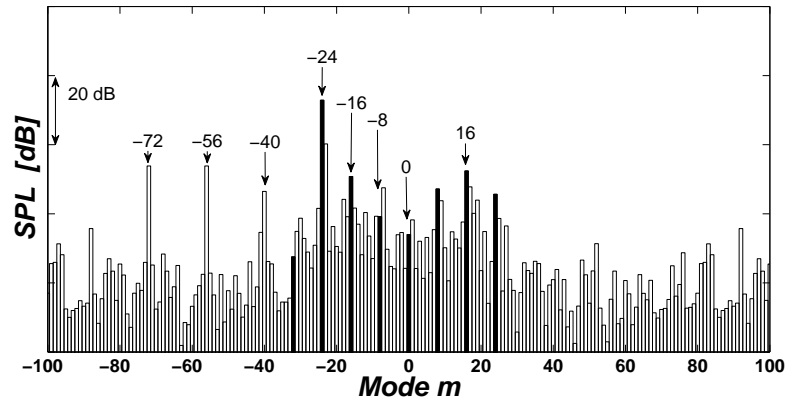


Abbildung 6.23: Azimutalmodenanalyse mit LP Trigger vom Setup C1

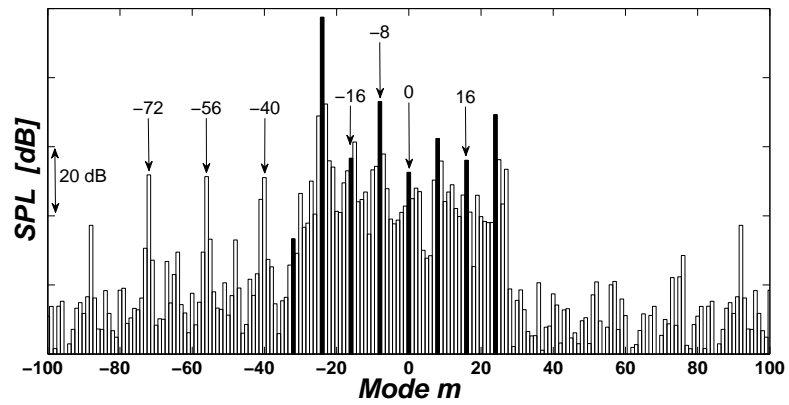


Abbildung 6.24: Azimutalmodenanalyse mit LP Trigger vom Setup C2

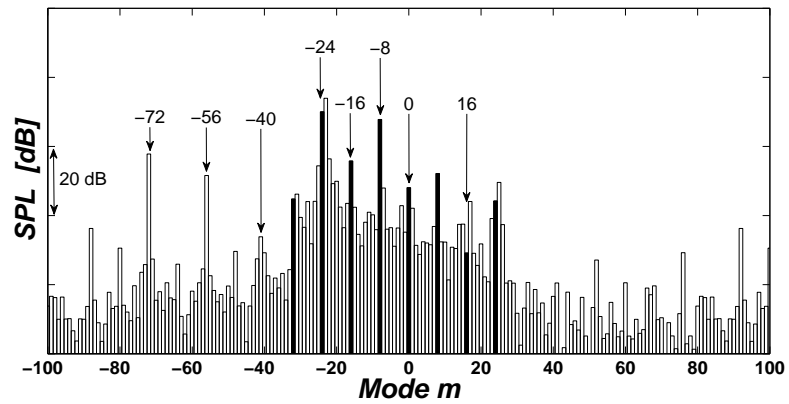


Abbildung 6.25: Azimutalmodenanalyse mit LP Trigger vom Setup C3

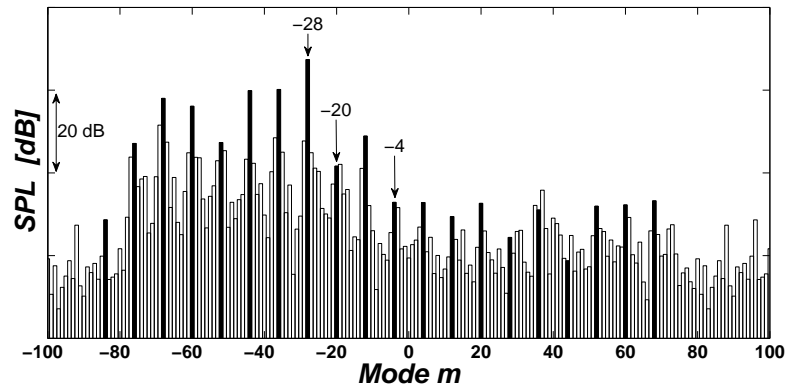


Abbildung 6.26: Azimutalmodenanalyse nach RSA vom Setup C1

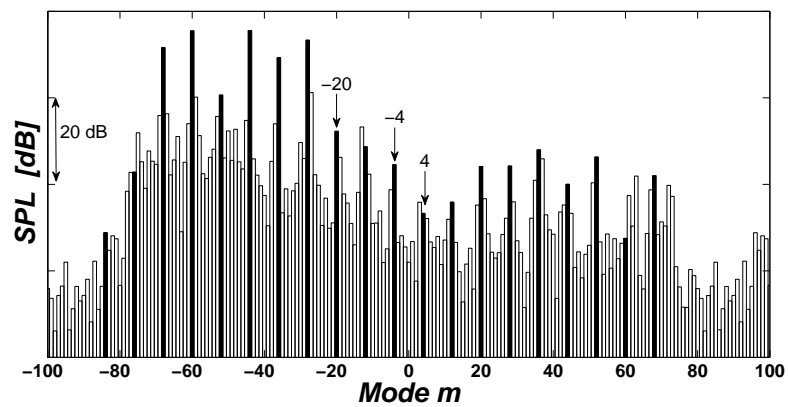


Abbildung 6.27: Azimutalmodenanalyse nach RSA vom Setup C2

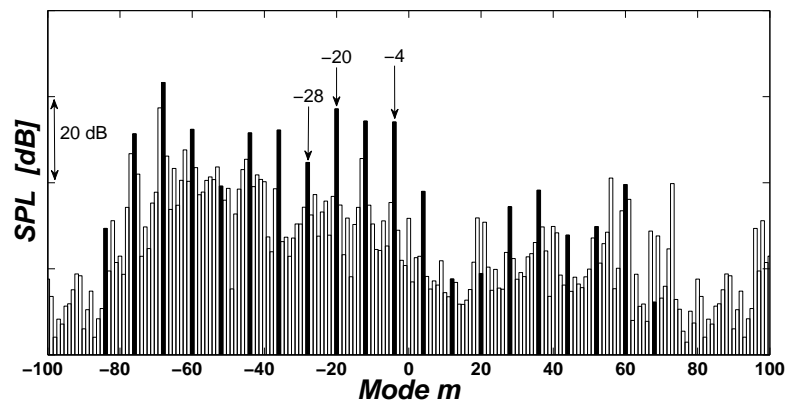


Abbildung 6.28: Azimutalmodenanalyse nach RSA vom Setup C3

In den nachfolgenden Abbildungen wird nun die Schalleistung als Summe aller Radialmoden über den azimuthalen Modenordnungen in und entgegen der Strömungsrichtung aufgetragen und außerdem ein Mittelwert über alle Azimutalmoden bestimmt. Auch hier zeigt sich Ähnliches, was anhand der vorhergehenden Bildern schon festgestellt wurde, nämlich dass die Werte von C2 deutlich über denen von C1 und C3 liegen. Sehr klar zu erkennen ist dies vor allem an den Summenpegeln unterhalb der Grafiken: Sowohl für die Auswertung mit HP Trigger als auch mit LP Trigger und nach der RSA liegen die gemittelten Werte der Konfiguration C2 bis zu 10 dB über den Ergebnissen der beiden anderen Setups. C3 hingegen schneidet hier bei allen drei Auswertemethoden am besten ab, was sich sowohl in den Summenpegeln, als auch an den kaum vorhandenen, herausragenden Peaks bei bestimmten Modenordnungen bemerkbar macht. Grund dafür könnte, wie bereits angemerkt, die dämpfende Wirkung der zusätzlich eingefügten Splitter sein.

Besonders hervorzuheben sind bei der Verwendung des HP Triggers die Mode $m=20$, die bei der Auswertung des Volvo Setups sowohl in als entgegen der Strömungsrichtung 10 dB über den Ergebnissen von C1 und C3 liegt, und die Mode $m=-4$ in Strömungsrichtung (blauer Balken), die insbesondere bei der Konfiguration C2 hervorsteht. Stammen könnten diese beiden Moden aus den Interaktionen zwischen HP Stator, HP Rotor und LP Stator und lassen sich nach Gleichung 6.2 folgendermaßen berechnen: $-4 = 36 - 3 \cdot 24 + 2 \cdot 16$ und $20 = 36 - 2 \cdot 24 + 2 \cdot 16$. Die Mode mit der Ordnung 12, die bei C2 sehr gut sichtbar ist, könnte hingegen aus den Wechselwirkungen zwischen HP Rotor und HP Stator nach Gleichung 6.1 stammen ($36 - 24 = 12$).

Ähnliches gilt nach der Auswertung mittels Niederdruck-Trigger (Abbildung 6.32, 6.33 und 6.34), hier sind es die Moden $m=-24$ und $m=-8$, die bei C2 besonders signifikant sind, während beim Setup C1 die Mode $m=8$ eine größere Amplitude als die beiden anderen Konfigurationen aufweist. Nach Gleichung 6.3 spielen LP Rotor und LP Stator bei diesen Moden eine Rolle, ein Einfluss der Hochdruckstufe ist jedoch kaum zu sehen ($-24 = 3 \cdot 16 - 1 \cdot 72$ und $-8 = 4 \cdot 16 - 1 \cdot 72$). Auffallend ist außerdem, dass der Bereich der ausbreitungsfähigen Moden bei der Analyse der Messdaten mit LP Trigger um einiges kleiner ist als bei der Auswertung mit HP Trigger.

Nach dem Rotor Synchronic Averaging (Abbildung 6.35, 6.36 und 6.37), fallen beim zweiten TMTF (C2) vor allem die Moden $m=-44$ und $m=-28$ auf, deren Amplituden in beiden Ausbreitungsrichtungen mehr als 10 dB über denen von C1 liegen, im Vergleich zu C3 sind die Unterschiede sogar noch größer. Berechnet werden könnten diese Moden wie folgt (Gleichung 6.2): $-44 = 36 + 0 \cdot 24 - 5 \cdot 15$ bzw. $-28 = 36 + 0 \cdot 24 - 4 \cdot 15$. Eine mögliche Erklärung dafür wären bei C2 starke Interaktionen zwischen dem Hochdruck-Rotor und den Struts des TMTFs.

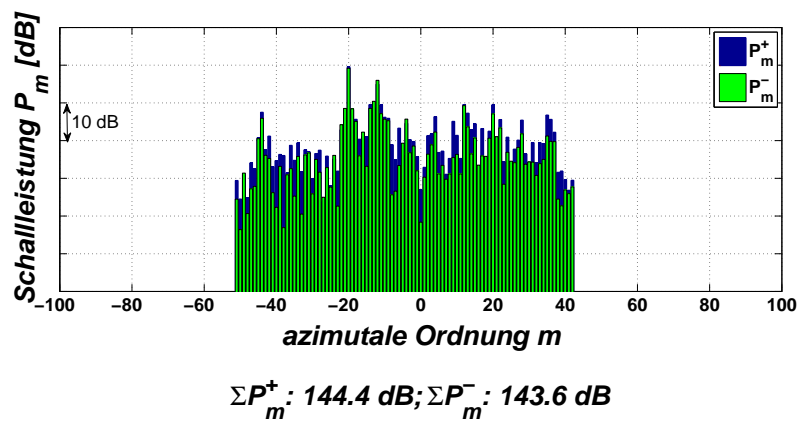


Abbildung 6.29: Schalleistung mit Hochdruck-Trigger für das Setup C1

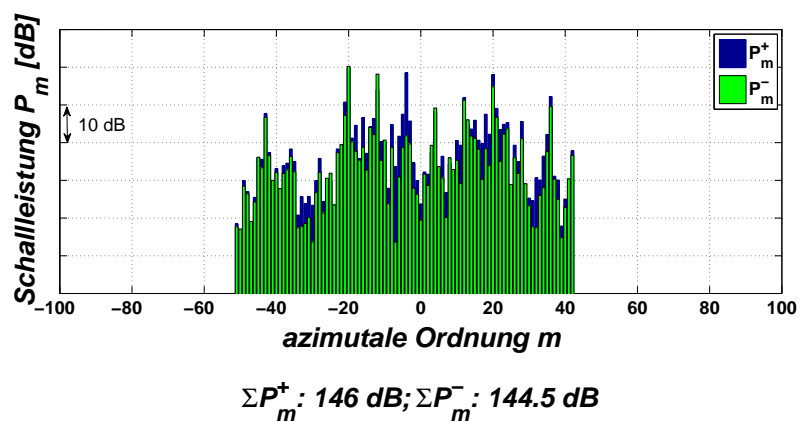


Abbildung 6.30: Schalleistung mit Hochdruck-Trigger für das Setup C2

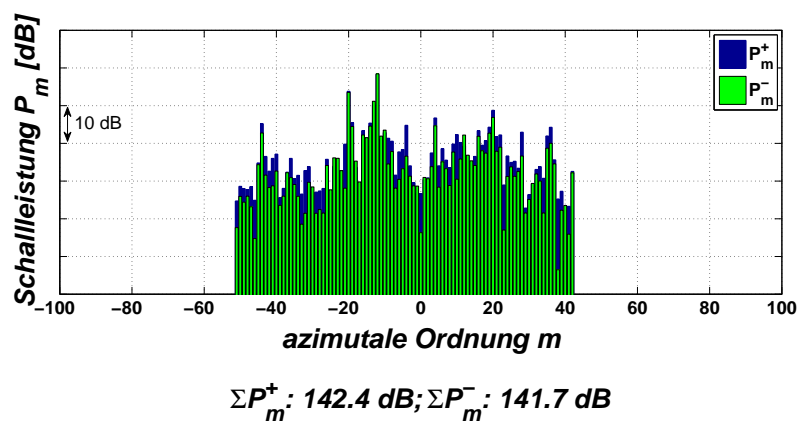


Abbildung 6.31: Schalleistung mit Hochdruck-Trigger für das Setup C3

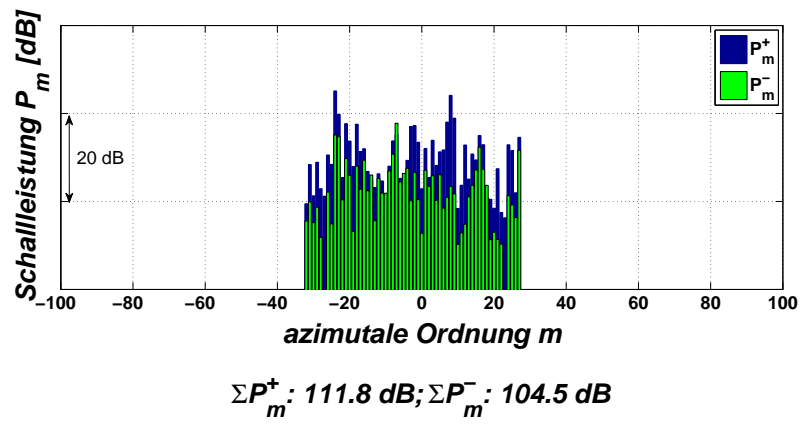


Abbildung 6.32: Schalleistung mit Niederdruck-Trigger für das Setup C1

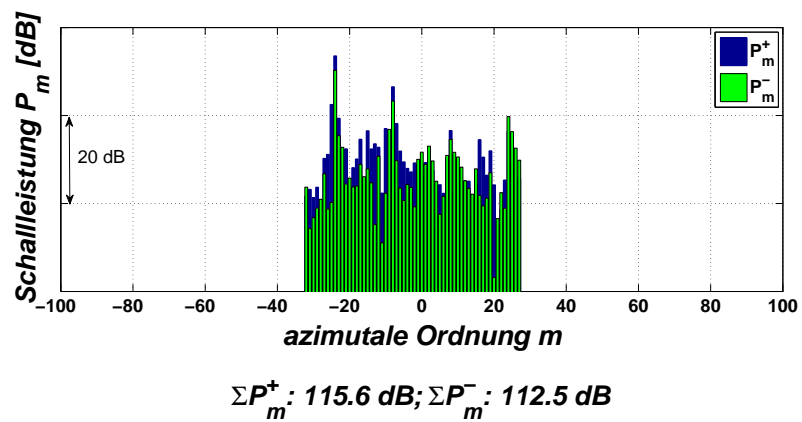


Abbildung 6.33: Schalleistung mit Niederdruck-Trigger für das Setup C2

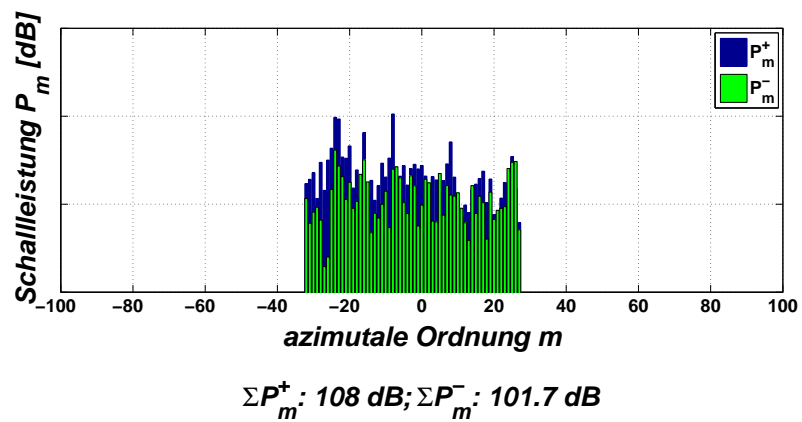


Abbildung 6.34: Schalleistung mit Niederdruck-Trigger für das Setup C3

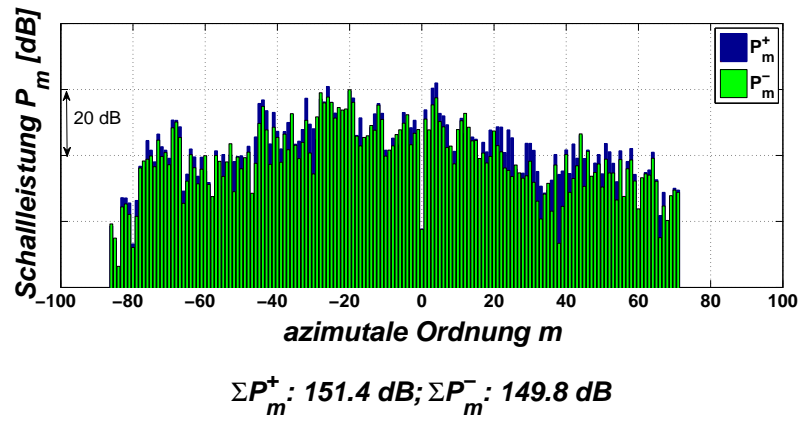


Abbildung 6.35: Schalleistung nach RSA für das Setup C1

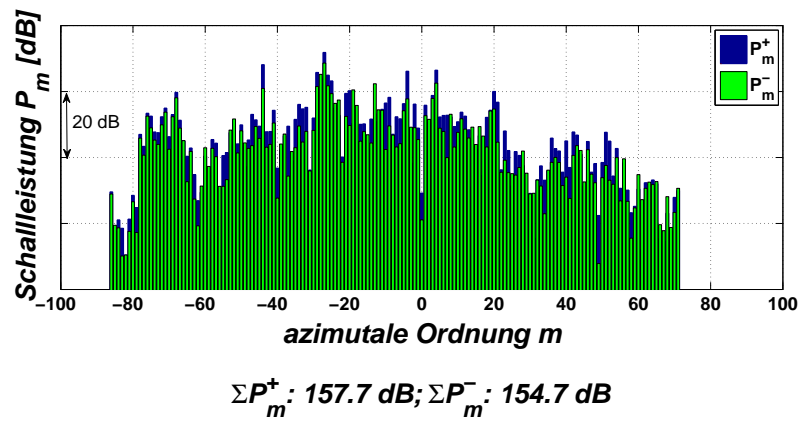


Abbildung 6.36: Schalleistung nach RSA für das Setup C2

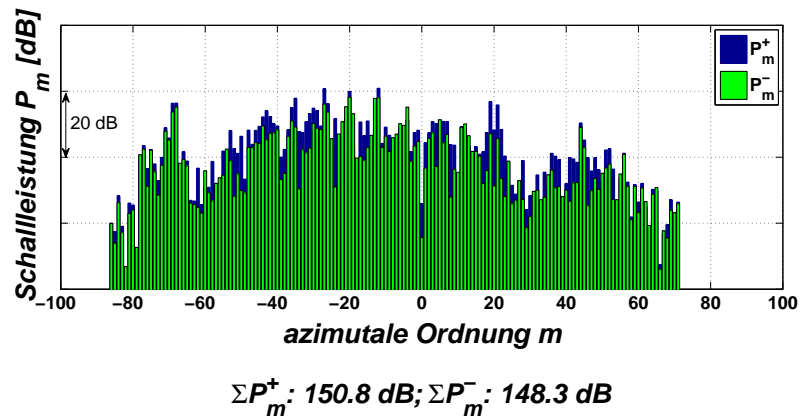


Abbildung 6.37: Schalleistung nach RSA für das Setup C3

Kapitel 7

Zusammenfassung und Ausblick

Nach der Durchführung der Akustikmessungen am Turbinenprüfstand des Instituts für Thermische Turbomaschinen und Maschinendynamik, die den Vergleich von drei verschiedenen Turbinenübergangskanälen (TMTF) zum Ziel hatte, nach Erarbeitung der dazu notwendigen theoretischen Grundlagen und nach Auswertung der Messergebnisse mittels spezieller Auswerteroutinen, lassen sich die Resultate nun folgendermaßen kurz zusammenfassen: Die Konfiguration von Volvo steigt, was die Lärmgenerierung betrifft, im Vergleich zu den beiden anderen Setups deutlich schlechter aus, die möglichen Ursachen dafür werden in Kapitel 6.4 kurz erläutert. Das institutsinterne Setup (C3) schließt hier zwar besser ab als C2, könnte jedoch durch die Anpassung der Struts an den Strömungskanal sogar noch weiter optimiert werden und bietet somit Spielraum für zusätzliche Untersuchungen und Analysen.

Für die Zukunft wären instationäre Messungen im Strömungskanal eine Möglichkeit, den Ursprung bestimmter akustischer Phänomene zu begründen, weiters wäre eine messtechnische Überprüfung der Auswerteroutinen, bei der beispielsweise der LP Rotor ausgebaut und nur der HP Rotor vermessen wird, wünschenswert. Auf diese Art und Weise könnten die Ergebnisse nach Auswertung mit HP Trigger bzw. LP Trigger auch anhand einer Messung verifiziert bzw. die Routine durch die gewonnenen Erkenntnisse zusätzlich verbessert werden.

Eine weitere Möglichkeit wäre die Einführung einer neuen traversierbaren Messebene vor dem TMTF, die beispielsweise mit vier Mikrofonen bestückt wird und damit sowohl vor als auch hinter dem TMTF Akustikmessungen durchgeführt und anschließend für jede Konfiguration Übertragungsfunktionen ermittelt werden können.

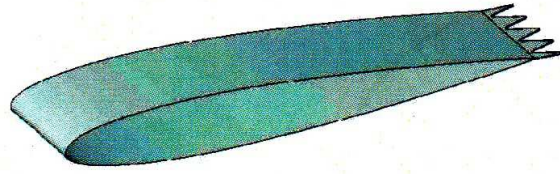


Abbildung 7.1: Schaufel mit gezackter Hinterkante

Ein akustisch optimiertes TMTF, dessen Schaufeln mit gezackten Hinterkanten, wie sie in Abbildung 7.1 dargestellt sind, ausgestattet ist, könnte durch die schnellere und bessere Vermischung der Schaufelnachläufe zur Vermeidung von größeren Wirbeln führen und dadurch die Lärmentwicklung deutlich reduziert werden. Details zu diesem Thema können in der Arbeit [34] nachgelesen werden.

Vorstellbar wären zukünftig auch schallabsorbierende Schaufelmaterialien, ausgeführt als $\lambda/4$ Absorber.

Literaturverzeichnis

- [1] Moser M.: "Untersuchung der Schallemissionen von Flugtriebwerks- Niederdruckturbinen in einem neuen Akustikprüfstand", Dissertation TU Graz, 2009
- [2] Marn A.: Skript zur Vorlesung Technische Akustik, TU Graz, 2011
- [3] Lengani D., Santner C., Spataro R., Göttlich E.: "Analysis tools for the unsteady interactions in a counter-rotating two -spool turbine rig", *Experimental Thermal and Fluid Science*, 42, pp. 248-257 (2012)
- [4] Gehrke W.: "Fortran 90 Referenzhandbuch", ISBN 3-446-16321-2, Carl Hanser Verlag München Wien, 1. Auflage 1991
- [5] Hussain A., Reynolds W.C.: "The mechanics of an organized wave in turbulent shear flow", *Journal of Fluid Mechanics*, Vol. 41, Part 2, pp. 241-258 (1970)
- [6] Lengani D., Paradiso B., Marn A.: "A Method for the Determination of Turbulence Intensity by Means of a Fast Response Pressure Probe and its Application in a LP Turbine", *Journal of Thermal Science*, Vol. 21, pp. 21-31 (2012)
- [7] <http://www.umweltbundesamt.at/umweltsituation/laerm/verursacher/fluglaerm/>, Abfrage: Oktober 2012
- [8] Faustmann C., Lengani D., Broszat D., Marn A., Göttlich E., Heitmeir F.: "Investigation on noise generation and propagation for different vane-vane positions in a two-stage two-spool test turbine" submitted for the 10th European Conference of Turbomachinery Fluid Dynamics and Thermodynamics
- [9] Neumayer F., Kulhanek G., Pirker H.-P., Jericha H., Seyr A., Sanz W.: "Operational behaviour of a complex transonic test turbine facility", ASME Paper 2001-GT-04891, Proceedings of the ASME Turbo Expo, New Orleans, USA (2001)
- [10] Pirker H.-P., Jericha H., Zhuber-Okrog G.: "Auslegung und Betriebsverhalten einer Verdichteranlage für die Luftversorgung wissenschaftlicher Versuchseinrichtungen", *VDI Berichte Nr 1208*, pp. 331-347

- [11] Göttlich E.: “Experimentelle Untersuchung von Kühlung und instationärer Strömung transsonischer Hochtemperaturgasturbinenstufen”, Dissertation TU Graz, 2004
- [12] Lerch R., Sessler G., Wolf D.: “Technische Akustik - Grundlagen und Anwendungen”, ISBN 978-3-540-23430-2, Springer, 2009
- [13] Möser M.: “Technische Akustik”, ISBN 978-3-540-89817-7, Springer, 8. Auflage 2009
- [14] Bräunling W.: “Flugzeugtriebwerke - Grundlagen, Aero-Thermodynamik, ideale und reale Kreisprozesse, Thermische Turbomaschinen, Komponenten, Emissionen und Systeme”, ISBN 978-3-540-76370-3, Springer, 3. Auflage 2009
- [15] Santner C., Göttlich E., Wallin F., Hoeger M.: “Experimental Investigation of Turning Mid Turbine Frame Designs”, ISABE Paper, ISABE-2011-1710 (2011)
- [16] Hubinka J., Santner C., Paradiso B., Malzacher F., Göttlich E., Heitmeir F.: “Design and Construction of a Two Shaft Test Turbine for Investigation of Mid Turbine Frame Flows”, ISABE Paper, ISABE-2009-1293 (2009)
- [17] Hubinka J., Paradiso B., Santner C., Göttlich E., Heitmeir F.: “Design and Operation of a Two Spool High Pressure Test Turbine Facility”, 9. Europäische Turbomaschinen-Konferenz, Istanbul, Türkei (2011)
- [18] Spataro R., Göttlich E., Lengani D., Faustmann C., Heitmeir F.: “Development of a Turning Mid Turbine Frame with Embedded Design - Part I: Design and steady measurements”, Proceedings of the ASME Turbo Expo, June 3-7, San Antonio, Texas (2013)
- [19] Kammeyer K.-D., Kroschel K.: “Digitale Signalverarbeitung”, ISBN 3-519-46122-6, Teubner, 5. Auflage 2002
- [20] Tapken U., Enghardt L.: “Optimization of Sensor Arrays for Radial Mode Analysis in Flow Ducts”, American Institute of Aeronautics and Astronautics, Collection of Technical Papers - 12th AIAA/CEAS Aeroacoustics Conference 5 , pp. 3073-3092 (2006)
- [21] Faustmann C., Lengani D., Spataro R., Marn A., Göttlich E., Heitmeir F.: “Experimental Investigation of the Noise Generation and Propagation for Different Turning Mid Turbine Frame Setups in a Two-Stage Two-Spool Test Turbine”, ASME Paper GT 2013-95698, Proceedings of the ASME Turbo Expo, June 3-7, San Antonio, Texas (2013)
- [22] Göttlich E.: “Research on the aerodynamics of intermediate turbine diffusers”, Progress in Aerospace Sciences 47, 249-279 (2011)

- [23] <http://www.dplot.com/fctbesselj.htm>, Abfrage: Dezember 2012
- [24] Ehrenfried K.: Skriptum zur Vorlesung Strömungsakustik I, Technische Universität Berlin, WS 02/03
- [25] Tyler, J. and Sofrin, T., "Axial Flow Compressor Noise Studies", SAE Technical Paper 620532, doi:10.4271/620532 (1962)
- [25] Kammeyer K.-D., Kroschel K.: "Digitale Signalverarbeitung", ISBN 3-519-46122-6, Teubner, 5. Auflage 2002
- [26] Press W. H., Teukolsky S. A., Vetterling W. T., Flannery B. P.: "Numerical Recipes in Fortran 77: The Art of Scientific Computing", ISBN 0-521-43064-X, Cambridge University Press, 2. Auflage 1992
- [27] <http://music.columbia.edu/cmc/musicandcomputers/chapter2/02.php>, Abfrage: Dezember 2012
- [28] <http://de.wikipedia.org/wiki/Leck-Effekt>, Abfrage: Dezember 2012
- [29] <http://www.iam.uni-bonn.de/alt/ws2001/EN/analysis3-hyp55.html>, Abfrage: Dezember 2012
- [30] Holste F., Neise W.: "Noise Source Identification in a Propfan Model by Means of Acoustical Near Field Measurements", Journal of Sound and Vibration, 203(4), pp. 641-665 (1997)
- [31] Sijtsma P., Zillmann J.: "In-Duct and Far-Field Mode Detection Techniques", 13th AIAA/CEAS Aeroacoustics Conference, AIAA 2007-3439 (2007)
- [32] Enghardt L., Zhang Y., Neise W.: "Experimental Verification of a Radial Mode Analysis Technique Using Wall-Flush Mounted Sensors", 137th Meeting of the Acoustical Society of America, Berlin, March 15-19 (1999)
- [33] Kornow O., Enghardt L.: "Der Einfluss des radialen Strömungsprofils auf die Schallausbreitung von höheren Moden in zylindrischen und koaxialen Strömungskanälen", Deutsches Zentrum für Luft- und Raumfahrt e.V., IB 92517-05/B1
- [34] Jones L.E., Sandberg R. D. : "Numerical Investigation of Airfoil Self-Noise Reduction by Addition of Trailing-Edge Serrations", 16th AIAA/CEAS Aeroacoustics Conference, AIAA 2010-3703 (2010)

Abbildungsverzeichnis

1.1	Lärmquellen an einem anfliegenden Verkehrsflugzeug [14]	2
2.1	Schematische Darstellung der Verdichteranlage [10]	5
2.2	Schaltungen für vier ausgesuchte Fahrweisen [10]	6
2.3	Achsenschnitt durch die Versuchsanlage [8]	7
2.4	Darstellung der gesamten Versuchsanlage	7
2.5	Triebwerk mit Darstellung der Lager	8
2.6	Detail des TMTFs	8
2.7	(a) Diffusor mit Struts, (b) TMTF, (c) TMTF mit Splittern, vgl. Setup TTM [22]	10
2.8	Schnitt durch die Anlage mit Darstellung der Größen L_{ax} und h_{in} [15]	11
2.9	Vergleich zwischen den Designs von MTU und Volvo [15]	12
2.10	Institutsinternes Design des TMTFs [21]	13
3.1	Schallemissionen von Einkreis- (links) und Zweikreis-Triebwerken (rechts) [1]	18
3.2	Spektrum einer dreistufigen Niederdruckturbine [14]	19
3.3	Zylinderkoordinaten [29]	23
3.4	Besselfunktion [23]	26
3.5	Druckverteilung im Querschnitt eines zylindrischen Rohres [2]	27

3.6	Dünnere ringförmiger Kanal [1]	28
3.7	Strömungskanal mit veränderlichem Querschnitt [2]	31
3.8	Kontrollvolumen im Übergangsbereich des Kanals [2]	32
3.9	Schematische Darstellung von Rotor und Stator [1]	35
3.10	Schematische Darstellung der Rotor-Stator-Interaktionen [1] . . .	36
3.11	Ursprüngliches Signal (blau) und aufgrund zu geringer Abtastrate Verfälschung des Signals (rot) [27]	41
3.12	Aliasing - (a) Kontinuierliches Signal in Abhängigkeit der Zeit (b)FFT des Originalsignals (c) FFT nach Sampling des Original- signals mit dem Intervall Δ und den Nyquist-Grenzen	42
3.13	Leakage-Effekt bei harmonischem Signal [28]	43
4.1	Winkelversatz zwischen den beiden Mikrofonreihen [21]	45
4.2	Mikrofonplatte mit axialen Abständen der Mikrofone [21]	46
4.3	Varianten für Mikrofonanordnung mit konstanter Sensorzahl [20] .	47
5.1	Rohsignal von Mikrofon 3 am Messpunkt 4 (entspricht 8°)	50
5.2	FFT des Rohsignals für Mikrofon 1, Messpunkt 1	51
5.3	FFT des Rohsignals für Mikrofon 1, Messpunkt 4	51
5.4	Phase Averaging	53
5.5	Frequenzspektrum nach Adaptive Resampling mit Hochdruck- Trig- ger	54
5.6	Frequenzspektrum nach Adaptive Resampling mit Niederdruck- Trigger	55
5.7	Skizze zur Erklärung des Rotor Synchronic Averaging [3]	56
5.8	Dimensionslose Amplituden-Spektren nach Auswertung [3]	59
5.9	Azimutalmodenanalyse mit HP-Trigger	63
6.1	FFT mit HP Trigger, Tag 1	67

6.2	FFT mit HP Trigger, Tag 2	67
6.3	FFT des Referenzmikrofons am Kanal 25	68
6.4	FFT des Referenzmikrofons am Kanal 26	68
6.5	FFT des Rohsignals für ein Mikrofon und eine Messposition	69
6.6	FFT des Rohsignals gemittelt über alle 24 Mikrofone und alle 180 Messpunkte	69
6.7	FFT nach Datenaufbereitung durch Adaptive Resampling und Pha- se Averaging mit HP Trigger	70
6.8	FFT mit Hochdruck-Trigger	71
6.9	Azimutalmodenanalyse mit Hochdruck-Trigger	72
6.10	Schalleistung für 1. BPF mit Hochdruck-Trigger	73
6.11	FFT mit Niederdruck-Trigger	74
6.12	Azimutalmodenanalyse mit Niederdruck-Trigger	75
6.13	Schalleistung für 1. BPF mit Niederdruck-Trigger	75
6.14	FFT nach Rotor Synchronic Averaging	76
6.15	Azimutalmodenanalyse nach Rotor Synchronic Averaging	77
6.16	Schalleistung nach Rotor Synchronic Averaging für Summe der 1. BPFs	77
6.17	Vergleich der Frequenzspektren von C1, C2 und C3 mit HP Trigger	78
6.18	Vergleich der Frequenzspektren von C1, C2 und C3 mit LP Trigger	79
6.19	Vergleich der Frequenzspektren von C1, C2 und C3 nach RSA	79
6.20	Azimutalmodenanalyse mit HP Trigger vom Setup C1	82
6.21	Azimutalmodenanalyse mit HP Trigger vom Setup C2	82
6.22	Azimutalmodenanalyse mit HP Trigger vom Setup C3	82
6.23	Azimutalmodenanalyse mit LP Trigger vom Setup C1	84
6.24	Azimutalmodenanalyse mit LP Trigger vom Setup C2	84

6.25	Azimutalmodenanalyse mit LP Trigger vom Setup C3	84
6.26	Azimutalmodenanalyse nach RSA vom Setup C1	85
6.27	Azimutalmodenanalyse nach RSA vom Setup C2	85
6.28	Azimutalmodenanalyse nach RSA vom Setup C3	85
6.29	Schalleistung mit Hochdruck-Trigger für das Setup C1	87
6.30	Schalleistung mit Hochdruck-Trigger für das Setup C2	87
6.31	Schalleistung mit Hochdruck-Trigger für das Setup C3	87
6.32	Schalleistung mit Niederdruck-Trigger für das Setup C1	88
6.33	Schalleistung mit Niederdruck-Trigger für das Setup C2	88
6.34	Schalleistung mit Niederdruck-Trigger für das Setup C3	88
6.35	Schalleistung nach RSA für das Setup C1	89
6.36	Schalleistung nach RSA für das Setup C2	89
6.37	Schalleistung nach RSA für das Setup C3	89
7.1	Schaufel mit gezackter Hinterkante	91

Tabellenverzeichnis

2.1	Technische Daten der einzelnen Verdichter [11]	4
2.2	Parameter der drei Setups und Betriebsbedingungen [21]	13

Index

- Abtastfrequenz, 37, 48, 51
- Adaptive Neuabtastung, 51
- adjungierten Matrix, 64
- Aliasing, 41
- AMA, 72, 76
- Azimutalmodenanalyse, 60, 62, 71, 76, 80

- Bessel'sche Differentialgleichung, 25
- Bessel-Funktion, 25
- Blade Passing Frequency, 18, 52, 70
- BPF, 18, 52, 54, 56, 58, 70, 72, 73, 76, 78

- cut-off, 27, 28, 30, 61, 71
- cut-on, 26, 34, 61, 74

- d'Alembert, 22
- DFT, 39
- Diskrete Fourier Transformation, 37

- Endwall Contouring, 11
- Eulergleichung, 20

- Fast Fourier Transformation, 37, 50
- FFT, 39, 40, 68, 70

- Kontinuitätsgleichung, 20, 23

- Leakage-Effekt, 41

- Modenordnung, 27, 28, 60

- Neumann-Funktion, 25
- Normalisierungsfunktion, 61
- Nyquist-Kriterium, 40, 41, 46, 62
- Nyquist-Theorem, 48, 61

- Phase Averaging, 52, 54, 55, 58
- Pseudoinverse, 64

- Radialmodenanalyse, 63

- RMA, 65
- Rotor Synchronic Averaging, 54, 56, 76, 86
- Rotor-Rotor-Interaktionen, 54
- Rotor-Stator-Interaktionen, 35

- Schalldichte, 15
- Schalldruck, 15
- Schalldruckpegel, 16
- Schallintensitätspegel, 17
- Schalleistungspegel, 17
- Schallschnelle, 15
- Schallschnellepegel, 16
- Schaufel-Trigger, 57
- Sensoranordnung, 46, 47
- Splitter, 12

- TMTF, 1, 6, 8, 9
- Turbinenübergangskanal, 8

- Wellenüberwachungssystem, 9
- Wellengleichung, 14, 20, 22, 23, 31, 33
- Wellenzahl, 30, 31, 61

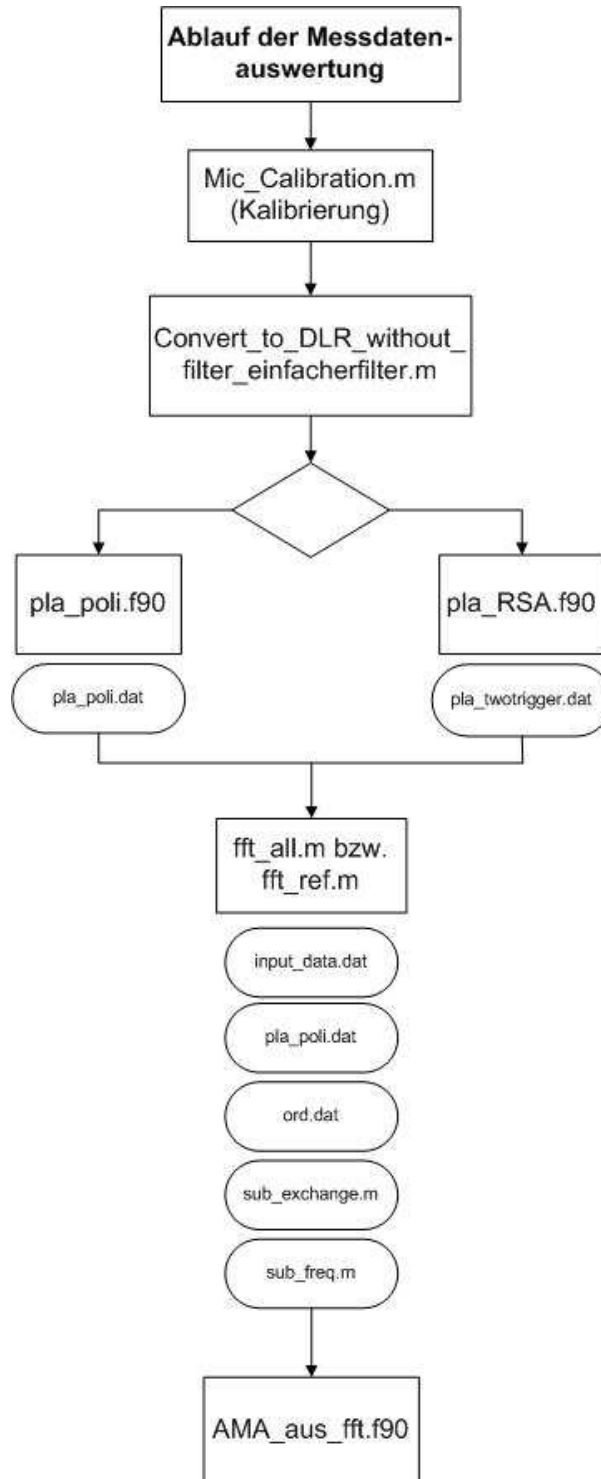
- Zylinderfunktion, 60

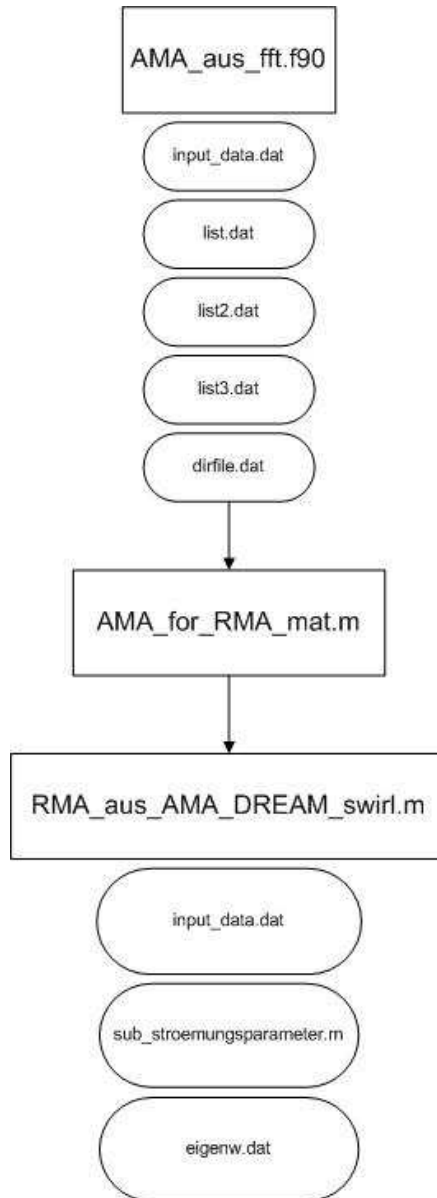
Anhang

Beschreibung der für die Auswertung verwendeten Programme

Der Übersichtlichkeit halber sollten für die nachfolgenden Auswertungen alle Programme im selben Ordner liegen, die Pfade, die auf die Messdaten nach der Konvertierung führen, können aber prinzipiell in den jeweiligen Input-Files individuell festgelegt werden. Wichtig ist nur, dass mit Hilfe der Input-Dateien (insbesondere “input_data.dat” und “pla_poli.dat” auf den richtigen Ordner “Data_of_traverse_steps”, der die Messdaten nach Konvertierung enthält, zugegriffen wird.

Folgendes Flussdiagramm soll den Ablauf der Auswerteroutinen zusammenfassen:





Kalibrierung der Mikrofone

Dateiname: Mic_Calibration.m

Für die Kalibrierung werden neben dem hier beschriebenen Matlab-File das LabView Programm mit dem Namen “Data Acquisition Acoustic AMICA.vi” zur Datenaufzeichnung und ein Pistonphon benötigt.

```
clear all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
MK='05';                %ID of the measurement camp.
db_corr                = -0.5;        %dB; Umgebungsdruckkorrektur

max_ampl_lin          = 114+db_corr;   %dB
f_cal                 = 250;          %Hz

channel               = 26;           %total number of channels
acq_channel           = 26;           %number of channels in use
circumf_num           = 181;         %Umfangpositionen
fs                    = 60000;       %Hz; sampling rate
t                     = 5;           %s; measurement time
start                 = 0;           %Position des Pointers beim File einlesen
p0                    = 2*10^-5;      %Pa
p                     = p0*10^(max_ampl_lin/20); %Pa
```

- Mit “dB_corr” wird der Umgebungsdruck korrigiert; muss vor dem Kalibrieren von einem Barometer, das dem Pistonphon beigelegt ist, abgelesen und in diesem m-File eingetragen werden
- “max_ampl_lin” und “f_cal” stehen für Amplitude und Frequenz des Tones, der vom Pistonphon ausgesendet wird
- Abgetastet wird hier mit 60000 Hz und die Messzeit beträgt 5 Sekunden, dies muss vor der Kalibrierung auch in dem zugehörigen LabView Programm eingegeben werden
- p_0 stellt den Referenzdruck dar, der für die Umrechnung zwischen Pascal und Dezibel benötigt wird

```
fpfad=['Z:\Amica\Messungen\C2\0\calibration'];
savepad='Z:\Amica\Messungen\C2\0\rawData';
```

- “fpfad” öffnet den Ordner mit dem Namen “calibration”, in dem dem Kalibrationsfiles für jedes einzelne Mikrofon zu finden sind; Files haben den Namen “Calxx_1_1.dat” - xx steht für den Kanal des Mikrofons - und werden bei der Kalibrierung vom oben genannten LabView Programm erstellt und müssen dann in den gewünschten Ordner kopiert werden
- “savepad” legt den Ordner fest, in den am Ende das dat.-File mit den berechneten Kalibrationsfunktionen abgespeichert wird; dies sollte, wie bereits oben zu sehen ist, der Ordner sein, der die Rohdaten enthält

```
for k=1:acq_channel %channel
    K=num2str(k);
    fname      =['Cal' K '_1_1.dat'];
    name       =[fpfad '\ ' fname]

fid2=fopen(name,'r','l');
fseek(fid2,start,'bof');
B=fread(fid2,'float32');
fclose(fid2);

cal=reshape(B,fs*t,acq_channel);
clear B;
cal=cal';
cal_micro=cal(k,:);

clear cal;
```

- Mit “for” beginnt eine Schleife über alle Channels bzw. alle Mikrofone, danach wird der Pfad der einzelnen Kalibrationsfiles festgelegt und mit “fopen” wird das .dat-File des entsprechenden Mikrofons geöffnet; das “r” bedeutet, dass die Datei nur zum Lesen geöffnet wird
- Während “fseek” den Anfang des Files sucht, werden mit “fread” die Inhalte der Datei eingelesen, deren Form und Größe im letzten Eintrag innerhalb der Klammer festgelegt werden (hier “float32”); Werte werden anschließend in die Matrix B geschrieben
- “fclose” schließt das Kalibrationsfile
- Danach wird mit dem Befehl “reshape” die Dimension der Matrix B verändert (die Matrix cal hat demnach $f_S \cdot t$ Zeilen und die Spaltenzahl ist die Anzahl der Channels), die Matrix wird transponiert und anschließend wird genau diejenige Zeile herausgelesen, in der die Werte des momentan betrachteten Mikrofons k stehen

```

offset=mean(cal_micro)
cal_micro=cal_micro-offset;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% FFT %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

NFFT=2^13;    %Fensterlänge bestimmen;
anz=floor(length(cal_micro)/NFFT)
for z=1:anz
    Y(z,:)=fft(cal_micro((z-1)*NFFT+1:z*NFFT),NFFT);

end

```

- Die Werte werden nun um einen Mittelwert aus allen Samples (“offset”) korrigiert und nachdem die Fensterlänge der FFT als eine Potenz von 2 festgelegt und mit “anz” die Anzahl der Berechnungsintervalle bestimmt wurde, werden mit der Anweisung “floor” die Werte aus “cal_micro” gerundet und eine FFT für jeweils ein Mikrofon durchgeführt
- Die Schleife über alle Mikrofone wird beendet

```

Y1=2*abs(Y) ./NFFT;
YY=mean(Y1,1);
f=fs/(NFFT) .*[0:NFFT/2-1];

```

- Für die korrekte Darstellung der Amplitude nach der FFT, wird deren Absolutwert zuerst mit 2 multipliziert und dann mit der Fensterlänge NFFT skaliert
- Anschließend berechnet man den Mittelwert über alle Berechnungsintervalle für jedes Mikrofon (YY)
- Die Berechnung von f dient unter anderem der grafischen Darstellung der FFT, die jedoch in diesem Programm nicht ausgegeben wird, und kommt auch bei der nachfolgenden Filterung zum Einsatz

```

Fil=find(f>1000);
YY(Fil)=0;
YY_max=max(YY);

```

- Obiger Programmabschnitt dient der Filterung aller Frequenzen, die größer als 1 kHz sind und setzt deren Amplituden Null; da bekannt ist, dass der Peak in etwa bei 250 Hz liegen wird, werden die restlichen Daten also nicht benötigt und deshalb gefiltert

```
cf(k)=p/YY_max
```

- Nun wird für jeden Channel ein Wert “cf”, die so genannte Kalibrierfunktion berechnet, der nicht nur die Differenz zwischen der vom Pistonphon ausgesendeten und der vom Mikrofon aufgezeichneten Amplitude ausgleicht, sondern auch dazu dient, das in Volt aufgenommene Signal später von Pascal in Dezibel umrechnen zu können. Aus diesem Grund wird der Druck durch die Amplitude in Volt dividiert

```
clear Y; clear Y1; clear YY; clear cal_micro;  
end
```

```
cal_name=[savepad '\calcf.dat'];  
save (cal_name, '-ascii','cf');
```

- Am Schluss werden alle nicht mehr benötigten Arrays gelöscht und für jedes Mikrofon der zugehörige Korrekturwert in einem File mit dem Namen “calcf.dat” abgespeichert.

Konvertierung in Dateiformate des DLR

Dateiname: Convert_to_DLR_without_filter_einfacherfilter.m

Dieses Programm konvertiert die Files mit den aufgezeichneten Messdaten in das notwendige Format und in die Ordnerstruktur des DLR.

Zu Beginn werden alle notwendigen Parameter eingegeben:

- TPNr = Bezeichnung des Testpunkts
- f_S = Abtastfrequenz
- flc = Frequenz des Tiefpassfilters (wird willkürlich gewählt)
- circumf_pos = Anzahl der Messpunkte in Umfangsrichtung
- max_time = Messzeit
- channel = Anzahl der Channels
- no_mics = Anzahl der Mikrofone
- ref_mic_1 und ref_mic_2 = Kanäle der beiden Referenzmikrofone

```
clear all
close all
clc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
#### Input %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

TPNr          = '1';
fs            = 60000;
flc          = 28000 / (fs/2);
circumf_pos  = 180;
max_time     = 20;
channel      = 28;
no_mics      = 24;
ref_mic_1    = 25;
ref_mic_2    = 26;
trig_1       = 27;
trig_2       = 28;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fpath='Z:\Amica\Messungen\C2\0\rawData' ;
fname=[fpath '\TP_' TPNr ];
data_path=[fpath];
load('calcf.dat');
[bb,aa] = butter(10, flc, 'low');
mkdir([data_path '\Data_of_traverse_steps']);
```

- trig_1 = Kanal des Hochdrucktriggers
- trig_2 = Kanal des Niederdrucktriggers

- “fpath” ruft den Pfad auf, in dem die Messergebnisse im .dat-Format vorzufinden sind
- “fname” bildet den Ordnernamen, in dem später die konvertierten Dateien abgelegt werden
- “data_path” übergibt einen Ordnernamen im Stringformat
- Mit dem Befehl “load” wird die Kalibrierfunktion, die zuvor mit der Datei “Mic_Calibration.m” ermittelt wurde, geladen
- Die Anweisung “butter” dient hier für die Verwendung eines Tiefpass-Butterworth-Filters. Es handelt sich dabei um einen Frequenzfilter, der so ausgelegt ist, dass der Frequenzgang unterhalb der Grenzfrequenz möglichst lange horizontal verläuft, Parameter ist die Frequenz “flc” die zuvor bei den Eingabeparametern erklärt wurde

- “mkdir” erstellt einen neuen Ordner mit dem angegebenen Pfad und dem Namen “Data_of_traverse_steps”

```
for circumf_counter=1:circumf_pos
    I=num2str(circumf_counter);
    if circumf_counter<10
        II=['0' I];
    else
        II=I;
    end

    mkdir([data_path '\Data_of_traverse_steps\RMA_' II '\RefMic\bin']);
    mkdir([data_path '\Data_of_traverse_steps\RMA_' II '\RmaMic\bin']);
    mkdir([data_path '\Data_of_traverse_steps\RMA_' II '\Trigger\bin']);

    name=[ fname '_' I '.dat']
    fid=fopen(name,'r','l'); fseek(fid,0,'bof');
    Ein = fread(fid,'float32');
    fclose(fid);
    clear fid;
```

- Eine Schleife über alle Umfangspositionen wird gestartet und innerhalb des Ordners “Data_of_traverse_steps” werden für jede einzelne Umfangsposition neue Ordner mit den Unterordnern RefMic \bin, RmaMic \bin und Trigger \bin erzeugt
- In oben stehendem Code werden außerdem die .dat-Files von jeder Umfangsposition, die die Messdaten enthalten, geöffnet, eingelesen, ihr Inhalt in die Matrix mit dem Namen “Ein” geschrieben und anschließend wieder geschlossen

In das Programm wurde ein Filter für fehlerhafte Rohdaten eingefügt; er überschreibt die Rohdaten, deren Wert ± 12 übersteigt, mit dem Mittelwert aus den beiden Nachbarwerten:

```

ind=find(Ein > 12.0);
if size(ind,1) > 1
    ind
    display('Error und Programm umschreiben')
elseif ind>0
    ind
    Ein(ind)
    Ein(ind)=(Ein(ind-1)+Ein(ind+1))/2;
    display('fehlerhafter Messwert => Wert wurde ersetzt c')

end

clear ind;

ind=find(Ein < -12.0);
if size(ind,1) > 1
    ind
    display('Error und Programm umschreiben')
elseif ind>0
    ind
    Ein(ind)
    Ein(ind)=(Ein(ind-1)+Ein(ind+1))/2;
    display('fehlerhafter Messwert => Wert wurde ersetzt f')
end

B=reshape(Ein,fs*max_time,channel);
A=B';

clear B; clear Ein;

for line_counter=1:no_mics
    J=num2str(line_counter);
    if line_counter<10
        JJ=['00' J];
    end
    if line_counter>=10 && line_counter<100
        JJ=['0' J];
    else
        JJ=J;
    end

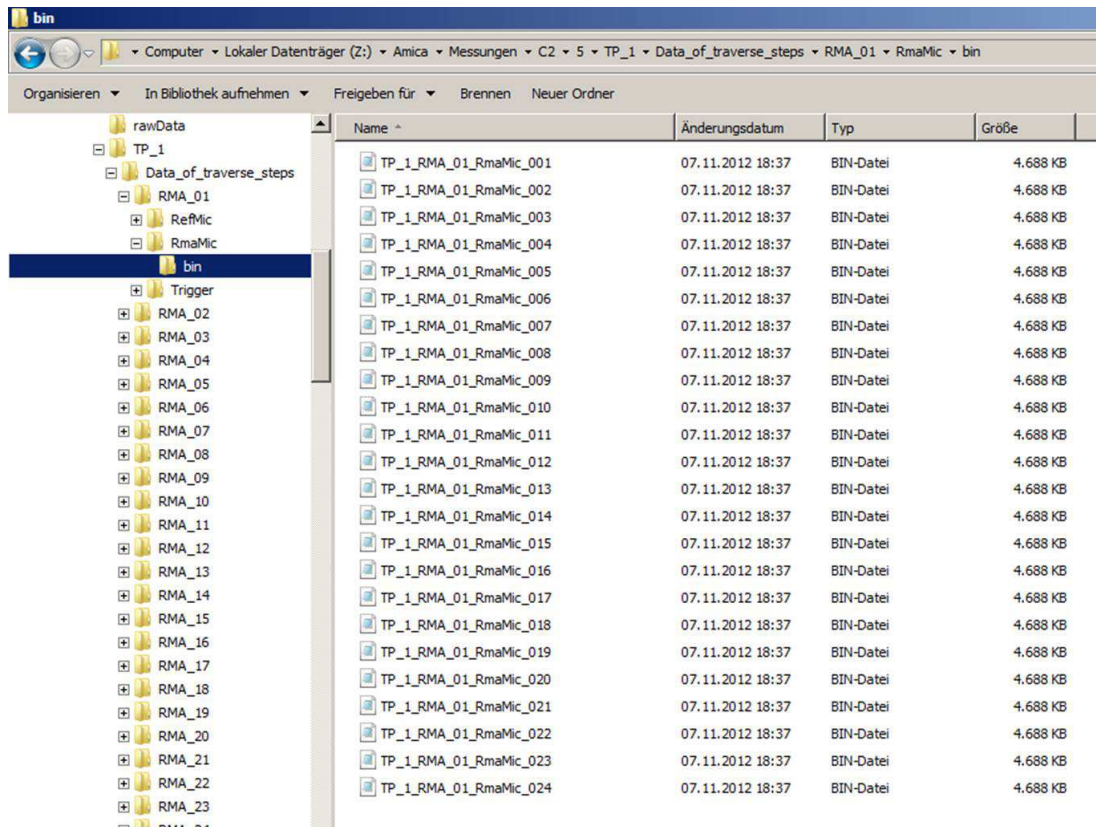
    signal=A(line_counter,:)*calcf(line_counter);
    %signal=filtfilt(bb, aa, signal1);
    new_file_name_RmaMic=[data_path '\Data_of_traverse_steps\RMA_ ' II '\RmaMic\bin\TP_ ' TPNr ' _RMA_ ' II ' _RmaMic_ ' JJ '.bin'];

    fid_RmaMic=fopen(new_file_name_RmaMic,'w','1');
    fwrite(fid_RmaMic,signal,'float32');
    fclose(fid_RmaMic);
    clear signal; clear signal1;

end

```


- Danach werden die Dimensionen der Matrix “Ein” verändert (siehe auch Abschnitt über Kalibration) und die neue Matrix “B” wird transponiert
- Alle nicht mehr benötigten Arrays werden gelöscht
- Anschließend startet eine neue Schleife über alle Mikrofone ohne Referenzmikrofone und Trigger, die mit einer if-Verzweigung für die korrekte Darstellung der Ordernamen beginnt
- “signal” ist ein Vektor, der alle Messdaten für jeweils ein Mikrofon enthält; entnommen wurden diese aus der Matrix “A”, die die Messdaten aller Mikrofone für alle Abtastpunkte enthält; “signal” ist also ein Vektor mit $f_S \cdot t$ Einträgen
- Die Daten eines Mikrofons werden dann mit dem Kalibrierwert, der zuvor berechnet wurde, multipliziert
- Die so erhaltenen Werte werden daraufhin in einem neu erstellten Ordner im angeführten Pfad abgespeichert; die Ordnerstruktur stellt sich nun folgendermaßen dar: rawData → TP_1 → Data_of_traverse_steps → RMA_II → RmaMic → bin → TP_1_RMA_II_RmaMic_JJ .bin (II steht für die Umfangsposition, JJ für die Mikrofonnummer)



```

trigger_si_1=A(trig_1,:);
trigger_si_2=A(trig_2,:);
ref_sig_1=A(ref_mic_1,:)*calcf(ref_mic_1);
ref_sig_2=A(ref_mic_2,:)*calcf(ref_mic_2);
%ref_sig_1=filtfilt(bb, aa, ref_si_1);
%ref_sig_2=filtfilt(bb, aa, ref_si_2);

new_file_name_RefMic_1=[data_path '\Data_of_traverse_steps\RMA_' II '\RefMic\bin\TP_' TPNr '_RMA_' II '_RefMic_0' num2str(ref_mic_1) '.bin'];
new_file_name_Trigger_1=[data_path '\Data_of_traverse_steps\RMA_' II '\Trigger\bin\TP_' TPNr '_RMA_' II '_Trigger_0' num2str(trig_1) '.bin'];

fid_RefMic_1=fopen(new_file_name_RefMic_1,'w','1');
fwrite(fid_RefMic_1,ref_sig_1,'float32');
fclose(fid_RefMic_1);

fid_trigger_1=fopen(new_file_name_Trigger_1,'w','1');
fwrite(fid_trigger_1,trigger_si_1,'float32');
fclose(fid_trigger_1);

new_file_name_RefMic_2=[data_path '\Data_of_traverse_steps\RMA_' II '\RefMic\bin\TP_' TPNr '_RMA_' II '_RefMic_0' num2str(ref_mic_2) '.bin'];
new_file_name_Trigger_2=[data_path '\Data_of_traverse_steps\RMA_' II '\Trigger\bin\TP_' TPNr '_RMA_' II '_Trigger_0' num2str(trig_2) '.bin'];

fid_RefMic_2=fopen(new_file_name_RefMic_2,'w','1');
fwrite(fid_RefMic_2,ref_sig_2,'float32');
fclose(fid_RefMic_2);

fid_trigger_2=fopen(new_file_name_Trigger_2,'w','1');
fwrite(fid_trigger_2,trigger_si_2,'float32');
fclose(fid_trigger_2);

clear A;

```

- Dasselbe was zuvor für die Mikrofone gemacht wurde, wird in oben stehenden Programm-Abschnitten auch für die Trigger bzw. die Referenzmikrofone durchgeführt, allerdings werden die Ergebnisse in andere Ordner gespeichert
- Ordnerstruktur für Referenzmikrofone: rawData → TP_1 → Data_of_traverse_steps → RMA_II → RefMic → bin → TP_1_RMA_II_RefMic_XX .bin (II steht für die Umfangsposition, XX für den Kanal des Referenzmikros), analog zu Abbildung 7
- Ordnerstruktur für die Trigger: rawData → TP_1 → Data_of_traverse_steps → RMA_II → Trigger → bin → TP_1_RMA_II_Trigger_YY .bin (II steht für die Umfangsposition, YY für den Kanal des Triggers), analog zu Abbildung 7

Adaptive Neuabtastung und Phase Averaging

Dateiname: pla_poli.f90

Input-File: pla_poli.dat (ist zu Zeit vorhanden, kann jedoch bei Bedarf auch selbst geschrieben werden)

Innerhalb der Routine “pla_poli.f90” findet sowohl die Adaptive Neuabtastung als auch das Phase Averaging statt. Dies kann durch Änderung des Input-Files sowohl mit dem Hoch- als auch den Niederdruck-Trigger durchgeführt werden. Bevor das Programm mittels des Fortran Command Windows gestartet wird, muss unbedingt in dem Ordner, in dem sich die beiden angeführten Files befinden, ein Unterordner mit dem Namen “save” erstellt werden, in dem dann am Ende die Ergebnisse abgespeichert werden.

```

program signal_smooth
USE MSFLIB
implicit none
real xret, triglvl, yy, xerr
integer:: nnrev, dim
parameter(dim=2000000)
parameter(nnrev=4000)
real, dimension(nnrev):: samples, deltax
real, dimension(dim):: a, trig
real, dimension(dim):: re, trig
real, dimension(20):: xip, yip
real, dimension(65536):: pla
real, dimension(nnrev, 1000):: rea
integer:: endpos, nmic, nn, numrev, re, redim
integer:: counta, countr, countt, hh, i, ii, ip, iret, k, kk, j, jj, numave, nperiod, step

character(3):: p1, p2, trignum, refnum
character(500):: path, pathname, pathref, pathtrig
character(500):: filename, fprefix, refname, trigname, savedir

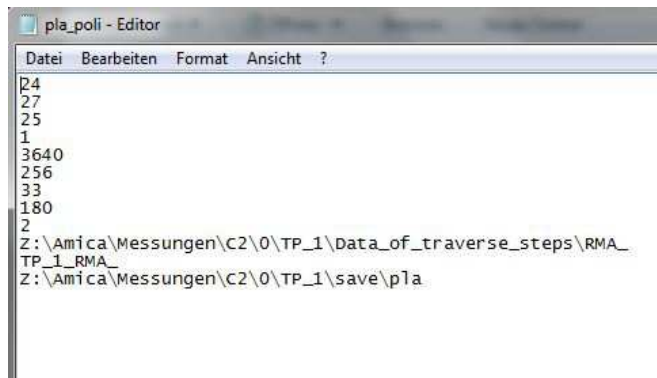
```

- Am Beginn des Fortran-Programms werden alle verwendeten Variablen deklariert, ihr Datentyp festgelegt und teils auch ihre maximale Größe angegeben.
- “dim” sollte dabei größer als $f_S \cdot t = 60000 \cdot 20$, also 1.200.000, sein und muss demnach bei Änderung der Messzeit oder der Abtastrate angepasst werden

```

open(unit=1, file='pla_poli.dat', status='old')
read(1,*) nmic
read(1,*) trignum
read(1,*) refnum
read(1,*) re
read(1,*) nperiod
read(1,*) nn
read(1,*) numrev
read(1,*) endpos
read(1,*) triglvl
read(1, '(A)') pathname
read(1, '(A)') fprefix
read(1, '(A)') savedir

```



```

pla_poli - Editor
Datei Bearbeiten Format Ansicht ?
24
27
25
1
3640
256
33
180
2
Z:\Amica\Messungen\C2\0\TP_1\Data_of_traverse_steps\RMA_
TP_1_RMA_
Z:\Amica\Messungen\C2\0\TP_1\save\pla

```

Im zweiten Schritt wird das Input-File mit dem Namen “pla_poli.dat”, welches unterhalb des Codes zu sehen ist, eingelesen und sein Inhalt einzelnen Variablen zugeordnet:

- nmic = Anzahl der Mikrofone
- trignum = Kanal des zu verwendenden Triggers; HP: Channel 27, LP: Channel 28 (HP oder LP, je nachdem, welche Effekte man separieren möchte)
- refnum = Kanal des zu verwendenden Referenzmikrofons
- re = 1 wenn kein Oversampling stattfinden soll
- nperiod = Anzahl der Umdrehungen während der Messzeit, siehe Kapitel 5.2
- nn = Resamples pro Umdrehung, siehe ebenfalls Kapitel 5.2
- numrev = Anzahl der Blöcke, über die gemittelt wird; numrev = 3 bedeutet bspw., dass von allen Umdrehungen während der Messzeit jede dritte Umdrehung ausgewählt und ein Mittelwert über diese drei Umdrehungen gebildet wird, den das Programm dann in einen Block schreibt, dem dann eine fixe Anzahl an Samples (nn) zugeordnet wird; ein Beispiel dazu ist im Abschnitt, wo das Phase Averaging beschrieben wird, angegeben
- endpos = Anzahl der Messpositionen in Umfangsrichtung

- `triglvl` = Triggerlevel; LP-Trigger: -8 bis -10 Volt, HP-Trigger: 2 Volt
- `pathname` = Ordner, in dem sich die Messdaten nach Anpassung an das DLR-Dateiformat befinden
- `fprefix` = Präfix des Files, das die Messdaten enthält
- `savedir` = Pfad, in dem die Daten nach Adaptive Resampling und Phase Averaging abgespeichert werden

Eine Schleife über alle Umfangspositionen wird begonnen:

```
close(1)
do k=1,endpos
  write(*,*) k
  !!!!!!! assignment of names !!!!!!!
  if (k<10) then
    write (pl, '(I1)') k
    path=pathname(:len(trim(pathname))//'0'//pl(:len(trim(pl))))//'\RasMic\bin\'
    pathtrig=pathname(:len(trim(pathname))//'0'//pl(:len(trim(pl))))//'\Trigger\bin\'
    pathref=pathname(:len(trim(pathname))//'0'//pl(:len(trim(pl))))//'\RefMic\bin\'
    trigname=fprefix(:len(trim(fprefix))//'0'//pl(:len(trim(pl))))//'_Trigger_0'//trignum(:len(trim(trignum)))/'.bin'
    refname=fprefix(:len(trim(fprefix))//'0'//pl(:len(trim(pl))))//'_RefMic_0'//refnum(:len(trim(refnum)))/'.bin'
  else if (k<100) then
    write (pl, '(I2)') k
    path=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RasMic\bin\'
    pathtrig=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\Trigger\bin\'
    pathref=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RefMic\bin\'
    trigname=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_Trigger_0'//trignum(:len(trim(trignum)))/'.bin'
    refname=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_RefMic_0'//refnum(:len(trim(refnum)))/'.bin'
  else
    write (pl, '(I3)') k
    path=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RasMic\bin\'
    pathtrig=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\Trigger\bin\'
    pathref=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RefMic\bin\'
    trigname=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_Trigger_0'//trignum(:len(trim(trignum)))/'.bin'
    refname=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_RefMic_0'//refnum(:len(trim(refnum)))/'.bin'
  end if
```

- Dateien (Messdaten nach Konvertierung in DLR-Formate) aus dem Ordner “Data_of_traverse_steps” werden an jeder Umfangsposition jeweils für die 24 Mikrofone, das gewählte Referenzmikrofon und den verwendeten Trigger eingelesen und einem Namen zugewiesen

```
open(unit=11, file=pathtrig(:len(trim(pathtrig))//trigname(:len(trim(trigname)))). status='old'.form='binary')
!!!! reading trigger !!!!!
countt=0
do kk=1,dim
  read(11,end=21) trig(kk)
  countt=countt+1
end do
continue
close(11)
write(*,*)countt
redim=countt*re
```

- Der Befehl “open” öffnet nun das .bin-File mit den Triggersignalen bei einer bestimmten Umfangsposition
- Danach wird die Datei gelesen und mittels des Zählers “countt” die Anzahl der Samples während der Messzeit von 20 s für eine Messposition gezählt

```

!!!!!! resample (oversampling) trigger !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
eventually use here an "if" for dimension of oversampling !!!!!!!!!!!
do kk=1,redim
  xret=real(kk-1)/real(re)+1
  iret=int(xret)
  do ip=1,3
    xip(ip)=iret+ip-1
    yip(ip)=trig(ip+iret-1)
  end do
  CALL POLINT(xip,yip,3,xret,YY,XERR)
  retrig(kk)=YY
end do

```

- Die Berechnung von “xret” in oben stehendem Code-Abschnitt ist in diesem Fall prinzipiell überflüssig, da hier das Oversampling stattfinden würde, was mit $re=1$ jedoch nicht der Fall ist
- “CALL POLINT” ruft eine Subroutine auf, die das Signal mittels einer Polynom-Interpolation mit drei Stützstellen (“xip” und “yip” sind die Koordinaten der Stützstellen) zwischen den diskreten Werten interpoliert, damit die darauf folgende Berechnung von Anfang und Ende der Umdrehungen überhaupt möglich ist und nicht an einer Stelle zwischen zwei diskreten Werten abgetastet wird, wo gar kein y-Wert vorliegt; mit “retrig(kk)=YY” liegt also an jeder Stelle x ein interpolierter y-Wert vor; “POLINT” ist dabei eine von Fortran vorgefertigte Subroutine für Polynom-Interpolationen

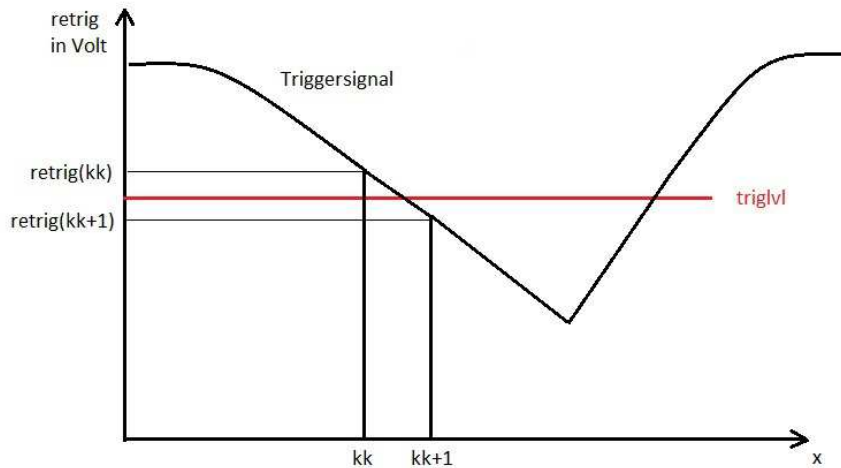
```

!!!!!! find start of each revolution !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
hh=1
do kk=1,redim-1
  if (((retrig(kk)>triglvl).and.(retrig(kk+1)<triglvl)).or.(retrig(kk)==triglvl)) then
    samples(hh)=kk
    !!!!!!!!! distance from trigger to exactly triglvl !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    deltax(hh)=(retrig(kk)-triglvl)/(retrig(kk)-retrig(kk+1))
    hh=hh+1
  end if
end do

```

- Nach dem Einlesen des Triggersignals wird nun mit Hilfe des Triggerlevels der Anfang bzw. das Ende einer Umdrehung der Welle festgestellt
- Liegt also der Triggerlevel zwischen dem y-Wert des Triggersignals beim Sampling mit der Nummer “kk” und dem y-Wert beim Sampling $kk+1$ oder fällt der Wert auf der y-Achse exakt mit dem Triggerlevel zusammen, so wird mit dem Zähler hh die Umdrehung als solche registriert und gezählt; “samples(hh)” steht für die Anzahl der Abtastungen während dieser Umdrehung
- Wenn nun der Interpolationswert für die y-Werte des Triggers mit dem Triggerlevel exakt übereinstimmt oder innerhalb bestimmter Grenzen um diesen liegt, so kennzeichnet das den Start bzw. das Ende einer Umdrehung

- Ist die Schleife durchlaufen, so steht hh für die exakte Anzahl der Umdrehungen während der Messzeit (in weiterer Folge wird von der Variable “hh” oft 1 abgezogen, da bei hh=1 mit dem Zählen begonnen wurde)
- Mit “deltax” wird durch lineares Interpolieren der Abstand in x-Richtung vom Trigger zum exakten Triggerlevel berechnet
- Die unterhalb stehende Skizze soll als zusätzliche Erklärung des Triggerlevels und der Schranken rund um diesen Wert dienen



```

do i=1,nmic
!!!! assignment of names !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  if (i<10) then
    write(p2,'(I1)') i
    if (k<10) then
      filename=fprefix(:len(trim(fprefix))//'0'//p1(:len(trim(p1))//'_RmaMic_00'//p2(:len(trim(p2))//'.bin'
    else
      filename=fprefix(:len(trim(fprefix))//p1(:len(trim(p1))//'_RmaMic_00'//p2(:len(trim(p2))//'.bin'
    end if
  else
    write(p2,'(I2)') i
    if (k<10) then
      filename=fprefix(:len(trim(fprefix))//'0'//p1(:len(trim(p1))//'_RmaMic_0'//p2(:len(trim(p2))//'.bin'
    else
      filename=fprefix(:len(trim(fprefix))//p1(:len(trim(p1))//'_RmaMic_0'//p2(:len(trim(p2))//'.bin'
    end if
  end if
  open(unit=10, file=path(:len(trim(path))//filename(:len(trim(filename))), status='old',form='binary')
  write(*,*) path(:len(trim(path))//filename(:len(trim(filename)))

```

- “do” öffnet eine Schleife über alle 24 Mikrofone
- Wie schon zuvor beim Trigger, werden hier die Dateien mit den Messdaten der 24 Mikrofone (Referenzmikrofone sind ausgenommen) einem Namen zugewiesen und anschließend geöffnet

```

! reading microphone data !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  counta=0
  do ii=1,countt
    read(10,end=22) a(ii)
    counta=counta+1
  end do
  continue

!! resample (oversampling) acq data !!!!!!!!!!!!!!!!!!!!!!!
  if (counta==countt) then
    do kk=1,redim
      xret=real(kk-1)/real(re)+1
      iret=int(xret)
      do ip=1,3
        xip(ip)=iret+ip-1
        yip(ip)=a(ip+iret-1)
      end do
      CALL POLINT(xip,yip,3,xret,YY,XERR)
      retrig(kk)=YY
    end do
  end if

```

- Die Inhalte der Files mit den Messdaten werden für jedes Mikrofon gelesen und die Anzahl der Samples gezählt
- Der Zähler für die Samples ist “counta”
- Für das Oversampling gilt wieder dasselbe wie schon für den Trigger: da im vorliegenden Fall $re=1$, ist die Berechnung von “xret” eigentlich nicht notwendig, jedoch wird auch hier wieder die Interpolation benötigt, um nachfolgend die Adaptive Neuabtastung durchführen zu können

```

! adaptive resampling !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  do j=1,hh-2
    do kk=1,nn
      xret=real(kk-1)/real(nn)*(real(samples(j+1)-samples(j))+deltax(j+1)-deltax(j))+deltax(j)+1
      iret=int(xret)
      do ip=1,5
        xip(ip)=iret+ip-1
        yip(ip)=retrig(ip+iret+samples(j)-1)
      end do
      CALL POLINT(xip,yip,5,xret,YY,XERR)
      rea(j,kk)=YY
    end do
  end do

```

- Am Anfang wird eine Schleife über alle Umdrehungen während der Messzeit geöffnet (hh)
- Nun wird für jede der Umdrehungen das Adaptive Resampling unter Verwendung der vom Benutzer vorgegebenen Anzahl der Resamples (nn), die für jede Umdrehung konstant ist, durchgeführt
- Mit “xret” werden die x-Positionen der neuen Abtastpunkte berechnet
- Anschließend werden wieder Stützstellen für eine Polynom-Interpolation ermittelt, dieses Mal jedoch fünf anstelle von drei, und mit “CALL POLINT”

wieder die Routine aufgerufen, die zwischen den diskreten Messwerten interpoliert, damit im Anschluss daran das Phase Averaging durchgeführt werden kann; da die Positionen der Abtastpunkte verschoben werden, wäre die Wahrscheinlichkeit ohne Interpolation sehr hoch, dass an dieser Stelle kein y-Wert vorliegt

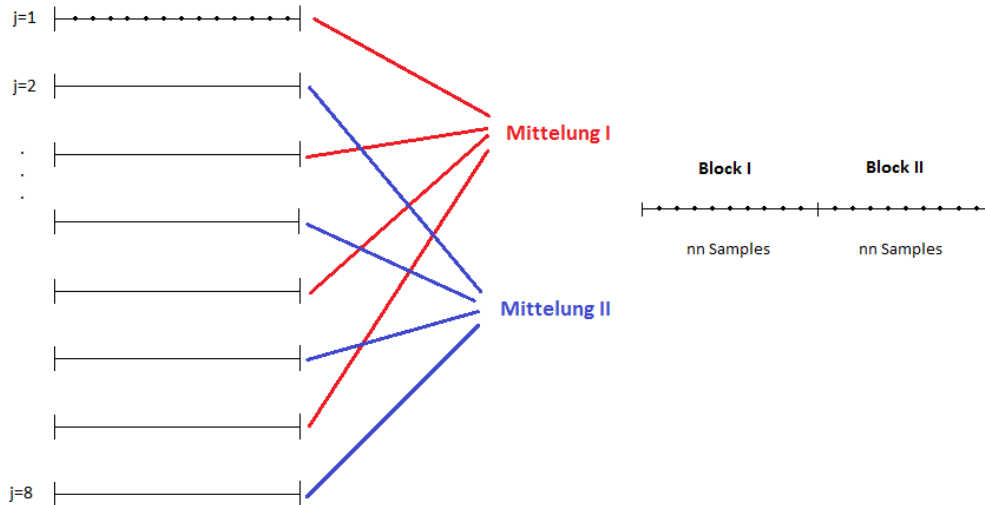
- Nach der Interpolation befinden sich im Array “rea(j,kk)” die Messdaten nach der Adaptiven Neuabtastung, wobei der Index “j” für die Umdrehungen und “kk” für für die neu definierte Anzahl an Samples steht

```
! phase averaging !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    pla=0
    step=1
    if (nperiod<(hh-1)) then
        numave=int(nperiod/numrev)
    else
        numave=int((hh-1)/numrev)
    end if
    write(*,*) hh,numave
    do j=1,numrev
        do kk=1,nn
            do jj=0,(numave*numrev-1),numrev
                pla(step)=pla(step)+rea(j+jj,kk)/real(numave)
            end do
            step=step+1
        end do
    end do
    write(*,*) step-1
!! end resampling !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

- Ergänzend zum Codeabschnitt, in dem das Phase Averaging durchgeführt wird, sollte unbedingt die auf der folgenden Seite zu findende Skizze betrachtet werden, die den Vorgang der Mittelung beim Phase Averaging beschreibt; anders als in Kapitel 5.3 beschrieben, wird hier jedoch kein exakter Winkel vorgegeben, bei dem jeweils die Werte gemittelt werden, sondern man bestimmt die exakte Anzahl an Umdrehungen während der Messzeit und unterteilt sie in die neue Anzahl an Samples (nn); danach nimmt man dann zum Beispiel jede zweite Umdrehung (numrev=2) und mittelt jedes einzelne Sample über die Menge “numrev”
- Mit “numave” wird also zuerst die Anzahl der Blöcke, wie sie auch in der Skizze beschrieben sind, festgelegt (“nperiod” ist hier die berechnete, theoretische Anzahl der Umdrehungen während der Messzeit ohne Drehzahlschwankungen und “hh” die exakte, die vorher im Programm ermittelt wurde)
- Die Mittelung wird nun innerhalb der folgenden drei “do”-Schleifen durchgeführt und soll der Einfachheit halber beispielhaft erklärt werden; nimmt man die erste aller Umdrehungen (j=1) und den ersten Abtastpunkt (kk=1), so wird in der dritten “do”-Schleife, wenn für numrev z.B. 17 gewählt wird, zuerst jede 17. Umdrehung gesucht, mit “rea(j+jj,kk)” die Messwerte an

allen Abtastpunkten $kk=1$ und von jeder 17. Umdrehung ermittelt und addiert und anschließend durch die Anzahl der Blöcke “numave” dividiert

- “pla(step)” setzt die Ergebnisse dann zu den bereits mehrmals erwähnten Blöcken zusammen (siehe unten), wobei “step” der Zähler für die Anzahl der Blöcke ist
- Dies geschieht für alle Abtastpunkte und bis alle Umdrehungen durchlaufen sind



In obigem Beispiel zur Erklärung der Variable “numrev” und des Phase Averaging wurde die Anzahl der Umdrehungen mit 8 angenommen, numrev ist 2. Das bedeutet jede zweite Umdrehung wird zusammengefasst und jeder einzelne Abtastpunkt gemittelt, was wiederum einen neuen Block ergibt. Diese Blöcke werden dann aneinander gefügt und jeder von ihnen wird nach Durchführung des Adaptive Resampling mit “nn” Samples abgetastet.

```
!!!! saving data !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      open(unit=12, file=savedir(:len(trim(savedir))))//fprefix(:len(trim(fprefix)))//p1(:len(trim(p1)))//'_RmaMic_0'//p2(:len(trim(p2)))//'.bin')
      write (12,100) (pla(kk), kk=1,step-1)
      close (12)
    end if
  close (10)
end do
!!!! end loop on all Mic !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

- Nach Beenden der Mittelung werden die Ergebnisse im Ordner “save”, der zu Beginn erstellt wurde, für jede Umfangsposition und für jedes einzelne Mikrofon abgespeichert

Die Ergebnisse für die Referenzmikrofone werden im Anschluss exakt auf dieselbe Art und Weise ermittelt:

```

!!!!!!! Start Same loop on reference microphone !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
open(unit=15, file=pathref(:len(trim(pathref))//refname(:len(trim(refname)))) status='old', form='binary')
write(*,*) pathref(:len(trim(pathref))//refname(:len(trim(refname))))
countr=0
do ii=1,countr
  read(15,end=25) a(ii)
  countr=countr+1
end do
continue
close (15)
!!!!!!! resample acq data Reference Mic !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
if (countr==countr) then
  do kk=1,redim
    xret=real(kk-1)/real(re)+1
    iret=int(xret)
    do ip=1,3
      xip(ip)=iret+ip-1
      yip(ip)=a(ip+iret-1)
    end do
    CALL POLINT(xip,yip,3,xret,YY,XERR)
    retrig(kk)=YY
  end do

!!!!!!! adaptive resampling Reference Mic !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
do j=1,hh-2
  do kk=1,nn
    xret=real(kk-1)/real(nn)*(real(samples(j+1)-samples(j))+deltax(j+1)-deltax(j))+deltax(j)+1
    iret=int(xret)
    do ip=1,5
      xip(ip)=iret+ip-1
      yip(ip)=retrig(ip+iret+samples(j)-1)
    end do
    CALL POLINT(xip,yip,5,xret,YY,XERR)
    rea(j,kk)=YY
  end do
end do
!!!!!!! phase averaging Reference Mic !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
pla=0
step=1
if (nperiod<(hh-1)) then
  numave=int(nperiod/numrev)
else
  numave=int((hh-1)/numrev)
end if
write(*,*) hh,numave
do j=1,numrev
  do kk=1,nn
    do jj=0,(numave*numrev-1),numrev
      pla(step)=pla(step)+rea(j+jj,kk)/real(numave)
    end do
    step=step+1
  end do
end do
write(*,*) step-1
!!!!!!! end resampling Reference Mic !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

!! saving data !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
open(unit=17, file=savedir(:len(trim(savedir))//fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//_RefMic_0//refnum(:len(trim(refnum))))//'.bin')
write(*,*) savedir(:len(trim(savedir))//fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//_RefMic_0//refnum(:len(trim(refnum))))//'.bin'
write (17,100) (pla(kk), kk=1,step-1)
close (17)

      end if
!!!!!!! End reference microphone !!!!!!!
    end do

100  format(f15.5)
103  format(2I7,2f15.5)

      END

```

Subroutine zur Polynom-Interpolation:

```

SUBROUTINE POLINT(XA, YA, N, X, Y, DY)
|*****|
|*      Polynomial Interpolation or Extrapolation      *|
|*      of a Discreet Function                        *|
|*      -----|
|* INPUTS:|
|*  XA:   Table of abscissas (N)                     *|
|*  YA:   Table of ordinates (N)                     *|
|*  N:    Number of points                           *|
|*  X:    Interpolating abscissa value                *|
|* OUTPUTS:|
|*  Y:    Returned estimation of function for X      *|
|*  DY:   Estimated error for Y                       *|
|*****|
PARAMETER(NMAX=25)
REAL XA(N), YA(N), X, Y, DY
REAL C(NMAX), D(NMAX)
REAL DEN, DIF, DIFT, HO, HP, W
NS=1
DIF=ABS(X-XA(1))
DO I=1, N
  DIFT=ABS(X-XA(I))
  IF(DIFT.LT.DIF) THEN
    NS=I
    DIF=DIFT
  ENDIF
  C(I)=YA(I)
  D(I)=YA(I)
END DO
Y=YA(NS)
NS=NS-1
DO M=1, N-1
  DO I=1, N-M
    HO=XA(I)-X
    HP=XA(I+M)-X
    W=C(I+1)-D(I)
    DEN=HO-HP
    IF(DEN.EQ.0.) Pause 'Error: two identical abscissas)'
    DEN=W/DEN
    D(I)=HP*DEN
    C(I)=HO*DEN
  END DO
  IF(2*NS.LT.N-M) THEN
    DY=C(NS+1)
  ELSE
    DY=D(NS)
    NS=NS-1
  ENDIF
  Y=Y+DY
END DO

RETURN
END
|After each column in the tableau XA
|is completed, we decide which correction,
|C or D, we want to add to our accumulating
|value of Y, i.e. which path to take through
|the tableau, forking up or down. We do this
|in such a way as to take the most "straight
|line" route through the tableau to its apex,
|updating NS accordingly to keep track of
|where we are. This route keeps the partial
|approximations centered (insofar as possible)
|on the target X. The last DY added is thus the
|error indication.

```

Rotor Synchronic Averaging (RSA)

Möchte man nicht die Einflüsse von HP und LP Rotor getrennt voneinander betrachten, sondern in erster Linie die Interaktionen zwischen den beiden Rotoren, dann verwendet man an Stelle von "pla_poli.f90" die Datei "pla_RSA.f90" für die Durchführung des Rotor Synchronic Averaging.

Dateiname: pla_RSA.f90 Input-File: pla_twotrigger.dat

```

program signal_smooth
USE MSFLIB
implicit none
real xret,yy,xerr
integer:: nnrev,dim
parameter(dim=2000000)
parameter(nnrev=7000)
real,dimension(nnrev):: samples,deltax
real,dimension(dim):: trig
real,dimension(dim):: retrig,retrig2
real,dimension(20):: xip,yip
real,dimension(65536):: pla
real,dimension(nnrev,4000):: rea
integer:: endpos,nmic,nn,numrev,re,redim
integer:: counta,countr,countt,hh,i,ii,ip,iret,k,kk,j,jj,numave,nperiod,step

character(3)::p1,p2,refnum
character(3),dimension(2)::trignum
character(500):: path,pathname,pathref,pathtrig
character(500):: filename,fprefix,refname,savedir
character(500),dimension(2)::trigname

!!!! Variable for FFT filter !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

integer:: MM,nft
integer:: nanoise,nanoise,check,no2
PARAMETER (MM=65392)
COMPLEX,dimension(MM):: fft1,fft2,ans,fft,tf,dumm
real:: fs,hhr,fres,nanoiser

!!!! Variable for RSA !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
real,dimension(dim,2):: tri,Npk
real,dimension(dim,2):: tri_int,samplesT,Trev
real,dimension(dim):: VPT,Npkpd,TrevLP
integer,dimension(2):: Nbl,redimT,tri_dim
integer,dimension(dim):: samples_s,samples_e
integer:: pp,xx,VPTNpk,VPTcontrol,Nfft,nLPF
real VPTtoll,TrevA,fLPF
real,dimension(2):: AvTrev,AvBPP,triglvl

!!!! Assignment for RSA !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Nbl(1)=36
Nbl(2)=72
VPTtoll=0.001
VPTcontrol=1

VPTNpk=216
fs=60000
Nfft=15

```


- Das Programm beginnt, wie auch schon bei “pla_poli.f90”, mit der Deklaration der Variablen und der Festlegung ihres Datentyps
- Hier treten im Vergleich zu “pla_poli.f90” noch zusätzliche Variablen für einen FFT Filter und die RSA auf
- Danach werden einigen Variablen Werte zugewiesen
- Nbl(1) = Anzahl der Schaufeln des HP Rotors
- Nbl(2) = Anzahl der Schaufeln des LP Rotors
- VPTtoll = Toleranzwert für den virtuellen Trigger (siehe Kapitel 5.4); maximaler zeitlicher Unterschied zwischen zwei Triggern, damit der virtuelle Trigger erstellt wird; hier 1/1000 der BPF des HP Rotors
- VPTcontrol wird später als Variable für eine Fallunterscheidung benötigt
- VPTNpk = konstante Anzahl an Blade Passing Periods (BPP); ist ein Vielfaches des Schaufelzahl des LP Rotors (hier: 72·3) und dient dazu, das Ende eines periodischen Signals festzulegen (siehe weiter unten)
- fs = Sampling Frequenz
- Nfft = Parameter für die Länge der FFT

```

open(unit=1, file='pla_twotrigger.dat', status='old')
read(1,*) nmic
read(1,*) trignum(1)
read(1,*) trignum(2)
read(1,*) refnum
read(1,*) nperiod
read(1,*) nn
read(1,*) numrev
read(1,*) endpos
read(1,*) triglvl(1)
read(1,*) triglvl(2)
read(1,'(A)') pathname
read(1,'(A)') fprefix
read(1,'(A)') savedir
close(1)

```

```

pla_twotrigger - Editor
Datei Bearbeiten Format Ansicht ?
24
27
28
25
6000
3100
1
180
2
-8
Z:\Amica\Messungen\C2\0\TP_1\Data_of_traverse_steps\RMA_
TP_1_RMA_
Z:\Amica\Messungen\C2\0\TP_1\save\pla

```

Eingelesen wird hier das Input-File mit dem Namen “pla_twotrigger.dat” und den Eingabeparametern werden danach Namen zugewiesen:

- nmic = Anzahl der Mikrofone
- trignum(1) = Kanal des HP Triggers
- trignum(2) = Kanal des LP Triggers
- refnum = Kanal des Referenzmikrofons
- nperiod = siehe Beschreibung zu “pla_poli.dat”; hier wird vorerst ein Wert angenommen (z.B. 6000) und dann durch den Wert, der bei der Auswertung mittels Fortran Command Window ermittelt wird, korrigiert
- nn = siehe Beschreibung zu “pla_poli.dat”; muss eine Potenz von 2 sein!
- numrev = siehe Beschreibung zu “pla_poli.dat”, ??
- endpos = Anzahl der Umfangspositionen
- triglvl(1) = Triggerlevel des HP Rotors
- triglvl(2) = Triggerlevel des LP Rotors
- pathname, fprefix, savedir = siehe Beschreibung zu “pla_poli.dat”

```

!!!!!! Parameter for fft filter !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
manoise=2**Nfft
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

OPEN (UNIT=177, file='ncoincidence.dat')
do k=1,endpos
  write(*,*) k
  !!!!!!!!! assignment of names !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  if (k<10) then
    write (pl, '(I1)') k
    path=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RnaMic\bin\'
    pathtrig=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\Trigger\bin\'
    pathref=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RefMic\bin\'
    do j=1,2
      trignum(j)=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_Trigger_0'//trignum(j)(:len(trim(trignum(j))))//'.bin'
    end do
    refname= fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_RefMic_0'//refnum(:len(trim(refnum)))/'.bin'
  else if (k<100) then
    write (pl, '(I2)') k
    path=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RnaMic\bin\'
    pathtrig=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\Trigger\bin\'
    pathref=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RefMic\bin\'
    do j=1,2
      trignum(j)=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_Trigger_0'//trignum(j)(:len(trim(trignum(j))))//'.bin'
    end do
    refname= fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_RefMic_0'//refnum(:len(trim(refnum)))/'.bin'
  else
    write (pl, '(I3)') k
    path=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RnaMic\bin\'
    pathtrig=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\Trigger\bin\'
    pathref=pathname(:len(trim(pathname))//pl(:len(trim(pl))))//'\RefMic\bin\'
    do j=1,2
      trignum(j)=fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_Trigger_0'//trignum(j)(:len(trim(trignum(j))))//'.bin'
    end do
    refname= fprefix(:len(trim(fprefix))//pl(:len(trim(pl))))//'_RefMic_0'//refnum(:len(trim(refnum)))/'.bin'
  end if
end do

```

- $manoise = 2^{Nfft}$; definiert einen Parameter für den FFT Filter
- Die Vergabe von Namen für die Dateien, die die Messdaten enthalten, erfolgt analog zu “pla_poli.f90”; zuerst werden wieder Namen für die Referenzmikrofone und die Trigger zugeordnet, die 24 restliche Mikrofone folgen weiter unten

```

!!!!!! reading trigger !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
do j=1,2
  open(unit=11, file=pathtrig(:len(trim(pathtrig))//trigname(j):len(trim(trigname(j))))), status='old',form='binary')
  countt=0
  do kk=1,dim
    read(11,end=21) tri(kk,j)
    countt=countt+1
  end do
  continue
  close(11)
  write(*,*)countt
end do
redim=countt

```

- Hier wird wieder analog zum gewöhnlichen Phase Averaging das Triggersignal eingelesen, allerdings für beide Trigger, also HP und LP (tri(kk,j), “kk” ist der Zähler für die Samples, “j” für die Trigger)
- Da redim=countt findet auch hier kein Oversampling statt, “countt” ist die Anzahl der Samples während der Messzeit

```

!!!RSA start
!!!!!!!!!!!! cutting the trigger !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
OPEN (UNIT=237,file='bla.dat')
do 22 pp=1,2
  hh=1
  do kk=1,redim-1
    if (((tri(kk,pp)<triglvl(pp)).and.(tri(kk+1,pp)>triglvl(pp))).or.(tri(kk,pp)==triglvl(pp))) then
      samplesT(hh,pp)=kk/fs
      hh=hh+1
    end if
    tri_dim(pp)=hh-1
  end do
end do

```

- Wie schon im File “pla_poli.f90” wird mit Hilfe des Triggerlevels der Anfang bzw. das Ende einer Umdrehung festgestellt, hier geschieht das sowohl für den Hochdruck- als auch für den Niederdruckrotor
- “samplesT(hh,pp)” ist die Zeit, die für “kk” Samples, also für den Ablauf einer Umdrehung benötigt wird
- “hh” ist der Zähler für die tatsächlich auftretenden Umdrehungen während der Messzeit pro Umfangsposition (20 s)
- Bei der Definition von “tri_dim” wird von “hh” 1 abgezogen, da bei “hh=1” mit dem Zählen begonnen wurde


```

do 25 pp=1,2
  !write(*,*) hh
  !! The trigger interval has to be divided in as many interval as the blades are!!!!!!!
  ii=1;
  do 23 kk=1,Nbl(pp)*(tri_dim(pp)-1).Nbl(pp)
    tri_int(kk,pp)=samplesT(ii,pp)
    xx=kk
    do 26 jj=1,Nbl(pp)-1
      xx=xx+1
      tri_int(xx,pp)=samplesT(ii,pp)+((samplesT(ii+1,pp)-samplesT(ii,pp))/Nbl(pp))*jj
    end do
    ii=ii+1
  end do
end do

do kk=1,(Nbl(1)*(tri_dim(1)-1))
  write(237,*) tri_int(kk,1), tri_int(kk,2)
end do

```

- Nun wird das Intervall zwischen zwei Triggern in die Anzahl der Schaufeln unterteilt
- Die erste Schleife läuft mit der Schrittweite “Nbl(pp)” (Schaufelzahl) bis zum Produkt aus Schaufelzahl und Anzahl der Umdrehungen
- Innerhalb der zweiten do-Schleife werden mit “tri_int” letztlich die Zeitpunkte berechnet, an denen ein Schaufeltrigger auftritt; “tri_int” ist also eine Matrix mit zwei Spalten, eine für den HP und eine für den LP Rotor, in deren Zeilen die Zeitpunkte der Original-Trigger sowie der Schaufeltrigger stehen
- “ii” ist der Zähler für die Umdrehungen
- Bis zum Produkt aus Schaufelzahl des Niederdruck-Rotors und dessen Umdrehungen während der Messzeit werden schließlich die Ergebnisse für die Zeitintervalle im zuvor erstellten File “bla.dat” abgespeichert

```

!!!! Calculating the Time revolution Period

      do 24 pp=1,2
        TrevA=0
        do kk=1, tri_dim(pp)-1

          !write(*,*) samplesT(kk+1,pp)-samplesT(kk,pp)
          TrevA=TrevA+samplesT(kk+1,pp)-samplesT(kk,pp)
        end do
        AvTrev(pp)=TrevA/real(tri_dim(pp)-1)
        AvBPP(pp)=AvTrev(pp)/real(Nbl(pp))
        write(*,*) AvTrev(pp), AvBPP(pp), TrevA, tri_dim(pp)
24      end do

!!!!!! Get the dimension on the vector with triggers + blade intervals

      do pp=1,2
        redimT(pp)=Nbl(pp)*(tri_dim(pp)-1)
      end do

close(237)

```

- Nach Durchlauf der zweiten do-Schleife mit der Bedingung “kk=1, tri_dim(pp)-1” ist “TrevA” die Zeit für den jeweiligen Trigger, die für die gesamte Anzahl an Umdrehungen “tri_dim(pp)” benötigt wird
- Damit ist “AvTrev” die Zeit, die für exakt eine Umdrehung benötigt wird
- “AvBPP” stellt daher die Zeitdauer dar, die eine Schaufel braucht, um einen ruhenden Beobachter zu passieren (Blade Passing Period)

```

select case (VPTcontrol)
case (1) !!
!!!!!!!!!!!!

open(unit=29,file='VPT.dat')
write(29,*) 'Sarcazzi'
xx=1
hh=1
do ii=1,redimT(2)
  do kk=xx,redimT(1)
    if (abs(tri_int(kk,1)-tri_int(ii,2))<=VPTtoll*(tri_int(kk+1,1)-tri_int(kk,1))*Nbl(1)) then
      ! the if condition impose to catch the trigger when the difference between a blade of the HP rotor and a blade
      ! of the LP rotor is smaller or equal to the HP blade passing time multiplied by a tolerance value.
      samples_s(hh)= int(tri_int(ii,2)*fs) ! this is the sample where the "periodic interval" begins
      samples_e(hh)= int(tri_int(ii+VPTNpk,2)*fs) ! this is the sample where the "periodic interval" finish

      write(29,*) samples_s(hh), samples_e(hh), samples_s(hh)-samples_e(hh)
      xx=kk
      hh=hh+1
    end if
  end do
end do
close (29)

```

- In diesem Abschnitt wird mittels des Parameter “VPTcontrol” eine Fallunterscheidung durchgeführt; ist der Wert 1, so tritt Fall 1 in Kraft
- Ist der Zeitunterschied zwischen dem Auftreten eines HP-Triggers (Schaufeltrigger oder Original-Trigger) und eines LP-Triggers kleiner als das Produkt aus der Toleranz “VPTtoll” und der Blade Passing Time des HP Ro-

tors, dann wird ein dritter Trigger, der diese Überlagerung kennzeichnet, definiert (siehe Kapitel 5.4)

- “samples_s” bestimmt hier für den LP Rotor den Abtastpunkt, bei dem das so genannte periodische Intervall beginnt
- “samples_e” legt mit dem anfangs festgelegten Wert “VPTNpk” willkürlich das Ende dieses Intervalls fest
- Beide Werte werden in dem erstellten File mit dem Namen “VPT.dat” gespeichert und beim Phase Averaging an Stelle der Samples während einer Umdrehung verwendet
- “hh” ist der Zähler für den dritten Trigger, also der Übereinstimmungen zwischen HP und LP Rotor

```
!!!! Calculation of the Low Pass Filter Frequency as half of the Resampling Frequency
fLPF=real(nn)/( VPTNpk * AvBPP(2) )/2.
!if (fLPF.gt.20000.) then
!   fLPF=20000.
!end if
write(*,*) 'low pass filter frequency', fLPF, 'check'
continue
```

- Mit “fLPF” wird die Frequenz des Tiefpass Filters basierend auf den Werten des LP Rotors und unter Einhaltung des Nyquist-Kriteriums berechnet

```
case(2) !!!!!!!!!!!!!!!!!!!!!!!!!!!!! "VPTcontrol=2" - Evaluating the Virtual Periodic Trigger (VPT)!!!!!!!!!!!!!!!!!!!!!!!!!!!!
pp=1   ! This is the VPT index
xx=1   ! This is the HP blade index
open(unit=39,file='VPT_2.dat')
do ii=1,redimT(2)-1
do kk=xx,redimT(1)
if (abs(tri_int(kk,1)-tri_int(ii,2))<=VPTtoll*(tri_int(kk+1,1)-tri_int(kk,1))*Nbl(1)) then
! the if condition impose to catch the trigger when the difference between a blade of the HP rotor and a blade
! of the LP rotor is smaller or equal to the HP blade passing time multiplied by a tolerance value.
TrevLP(pp)=(tri_int(ii+1,2)-tri_int(ii,2))*Nbl(2)
VPT(pp)= tri_int(ii,2)
write(39,*) VPT(pp)
xx=kk
pp=pp+1
go to 42
end if
end do
end do
42 continue
end do
close (39)
write(39,*)
do ii=2,pp
Npkpd(ii-1)=(VPT(ii)-VPT(ii-1))/TrevLP(ii)*Nbl(2)
write(39,*) ii-1, Npkpd(ii-1)
end do
close (39)
end select !!!!!!! STOP
write(*,*) hh,samples_s(1)
write(177,*) k,hh
```

- Auf den oben auftretenden Fall 2 soll hier nicht näher eingegangen werden, da er wegen “VPTcontrol = 1” nicht zum Einsatz kommt

- Anzumerken sei nur, dass in diesem Programm-Abschnitt die Berechnungen auf den Niederdruck-Rotor bezogen werden

```
nft=int(redim/(manoise))
fres=fs/real(manoise)
nancoiser=fLFF/fres
nanoise=int(nancoiser)
hhr=real(nft*manoise)/real(redim)*hh
write(*,*)hhr,hh
hh=int(hhr)
```

Die oben stehenden Größen werden in weiterer Folge für die Tiefpass-Filterung der Mikrofon-Signale benötigt:

- nft = Anzahl der Samples dividiert durch die Länge der FFT ("manoise" wurde weiter oben berechnet)
- fres = Auflösung der Frequenz
- nanoise = Stelle der Grenzfrequenz, bei der das Originalsignal abgeschnitten wird (dimensionslos)
- hhr = Anzahl der Übereinstimmung von HP und LP Trigger (3. Trigger)

```
do i=1,nmic
!!!!!! assignment of names !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  if (i<10) then
    write(p2,'(I1)') i
    if (k<10) then
      filename=fprefix(:len(trim(fprefix))//'0'//p1(:len(trim(p1))//'_RmaMic_00'//p2(:len(trim(p2))//'.bin')
    else
      filename=fprefix(:len(trim(fprefix))//p1(:len(trim(p1))//'_RmaMic_00'//p2(:len(trim(p2))//'.bin')
    end if
  else
    write(p2,'(I2)') i
    if (k<10) then
      filename=fprefix(:len(trim(fprefix))//'0'//p1(:len(trim(p1))//'_RmaMic_00'//p2(:len(trim(p2))//'.bin')
    else
      filename=fprefix(:len(trim(fprefix))//p1(:len(trim(p1))//'_RmaMic_00'//p2(:len(trim(p2))//'.bin')
    end if
  end if
  open(unit=10, file=path(:len(trim(path))//filename(:len(trim(filename))), status='old', form='binary')
  write(*,*) path(:len(trim(path))//filename(:len(trim(filename)))

!! reading microphone data !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  counta=0
  do ii=1,countt
    read(10,end=292) retrig(ii)
    counta=counta+1
  end do
  continue
```

- Wie schon zuvor für Trigger und Referenzmikrofon werden hier den Dateien mit den Messsignalen für die restlichen 24 Mikrofone Namen zugewiesen und anschließend die Mikrofon-Daten eingelesen

```
!!!!!!!!! FFT Classic Filtering!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

step=1
write(*,*) 'fft filter times', nft, ' up to', nft*manoise
do j=1,nft
  do kk=1,manoise
    retrig2(kk)=retrig(step)
    step=step+1
  end do
  CALL twofft(retrig2,retrig2,fft1,fft2,manoise)
  no2=manoise/2

!!!!!!! check fft !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  open(653,file='bah1.dat')
  do i=1,no2+1
    write(653,*) sqrt((real(fft(i)))**2+(imag(fft(i)))**2)
  end do
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!! Just LP filter !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  do 110 pp=1,no2+1
    if (pp.lt.nanoise) then
      ans(pp)=fft1(pp)/no2
    else
      ans(pp)=(0,0)
    end if
    continue

    ans(1)=cmplx(real(ans(1)),real(ans(no2+1)))
!!!!!!!!! check to compare fft !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  open(654,file='boh1.dat')
  do i=1,no2+1
    write(654,*) i*sqrt((real(ans(i)))**2+(imag(ans(i)))**2)
  end do
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!! Perform inverse transform on each kft block !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  call realft(ans,manoise,-1)
!!!!!!!!!!!! rewrite rms !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  do kk=1,manoise/2
    retrig(2*kk-1+manoise*(j-1))=real(ans(kk))
    retrig(2*kk+manoise*(j-1))=imag(ans(kk))
  end do
end do

!!!!!!! End anti noise filter !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

- Um das Rauschen zu vermindern, wird hier beim Signal der Mikrofone eine Tiefpass-Filterung durchgeführt
- Dazu wird mittels der von Fortran bereitgestellten Subroutine “twofft” und der darin enthaltenen Subroutine “four1” zuerst das Signal jedes Mikrofons in den Frequenzbereich übertragen
- “retrig” enthält für alle Mikrofone die Werte an jedem einzelnen Abtastpunkt
- “no2” ist die Hälfte von “manoise” wegen der Symmetrie der FFT
- Ist “pp” kleiner als die Anzahl der Samples bei denen sich die Grenzfrequenz befindet, dann liegt dort ein Wert “ans(pp)” für die Amplitude vor; wird die Grenze übertreten, so ist dieser Wert 0, also alles was über der Grenzfrequenz liegt wird herausgefiltert
- Die darauf folgende Zeile stellt eine Korrektur für die Subroutine “realft” dar, bei der das erste und das letzte Element aus “ans(pp)” miteinander

kombiniert werden

- Im Anschluss erfolgt mittels der Subroutine “realft”, die ebenfalls von Fortran zur Verfügung gestellt wird, und der Eingabe “-1” in der Klammer, die die Parameter enthält, die Rücktransformation in den Zeitbereich (wiederum unter Verwendung von “four1”)
- “retrig” enthält nun die Messwerte nach Tiefpass-Filterung

```

!!! adaptive resampling !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  do j=1,hh-2
    do kk=1,nn
      xret=real(kk-1)/real(nn)*(real(samples_e(j)-samples_s(j)))+1 !real(kk-1)/r
      iret=int(xret)
      do ip=1,5
        xip(ip)=iret+ip-1
        yip(ip)=retrig(ip+iret+samples_s(j)-1)
      end do
      CALL POLINT(xip,yip,5,xret,YY,XERR)
      rea(j,kk)=YY
    end do
  end do

!!! unlock to check signal !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  open(unit=19,file='m//p2//'_p'//pl//'.dat')
  do j=1,hh-1
    do kk=1,nn
      write(19,103) j,kk,rea(j,kk)
    end do
  end do
  close (19)
!!! phase averaging !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

  pla=0.
  step=1
  if (nperiod<(hh-1)) then
    numave=int(nperiod/numrev)
  else
    numave=int((hh-1)/numrev)
  end if
  write(*,*) hh,numave
  do j=1,numrev
    do kk=1,nn
      do jj=0,(numave*numrev-1),numrev
        pla(step)=pla(step)+rea(j+jj,kk)/real(numave)
      end do
      step=step+1
    end do
  end do
  write(*,*) step-1
!!!! end resampling !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

- Nun folgt wie gehabt das Adaptive Resampling und Phase Averaging, wobei bei der Bestimmung der x-Positionen der neuen Abtastpunkte nicht mehr die Samples während einer Umdrehung verwendet werden, sondern diejenigen zwischen Anfang und Ende des periodischen Intervalls

In gleicher Art und Weise wird für das Referenzmikrofon vorgegangen, deshalb wird der Code hier nicht weiter angeführt, nur die Subroutinen für die Tiefpass-Filterung wurden der Vollständigkeit halber eingefügt:


```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%1
SUBROUTINE reallft(DAT,n,isign)
  INTEGER isign,n
  REAL DAT(n)
  ! USES four1
  !Calculates the Fourier transform of a set of n real-valued data points. Replaces this data
  ! (which is stored in array data(1:n)) by the positive frequency half of its complex Fourier
  ! transform. The real-valued first and last components of the complex transform are returned
  ! as elements data(1) and data(2), respectively. n must be a power of 2. This routine
  ! also calculates the inverse transform of a complex data array if it is the transform of real
  ! data. (Result in this case must be multiplied by 2/n.)
  INTEGER i,i1,i2,i3,i4,n2p3
  REAL c1,c2,h1i,h1r,h2i,h2r,wis,wrs
  DOUBLE PRECISION theta,wl,wpi,wpr,wr,wtemp
  !Double precision for the trigonometric recurrences.
  theta=3.141592653589793d0/dble(n/2) !Initialize the recurrence.
  c1=0.5
  if (isign.eq.1) then
    c2=-0.5
    call four1(DAT,n/2,+1) !The forward transform is here.
  else
    c2=0.5
    theta=-theta
    !Otherwise set up for an inverse transform.
  endif
  wpr=-2.0d0*sin(0.5d0*theta)**2
  wpi=sin(theta)
  wr=1.0d0+wpr
  wi=wpi
  n2p3=n+3
  do 16 i=2,n/4
    !Case i=1 done separately below.
    i1=2*i-1
    i2=i1+1
    i3=n2p3-i2
    i4=i3+1
    wrs=sngl(wr)
    wis=sngl(wi)
    h1r=c1*(DAT(i1)+DAT(i3)) !The two separate transforms are separated out of
    h1i=c1*(DAT(i2)-DAT(i4)) !data.
    h2r=-c2*(DAT(i2)+DAT(i4))
    h2i=c2*(DAT(i1)-DAT(i3))
    DAT(i1)=h1r+wrs*h2r-wis*h2i !Here they are recombined to form the true trans
    DAT(i2)=h1i+wrs*h2i+wis*h2r !form of the original real data.
    DAT(i3)=h1r-wrs*h2r+wis*h2i
    DAT(i4)=-h1i+wrs*h2i-wis*h2r
    wtemp=wr
    wr=wr*wpr-wi*wpi+wr
    wi=wi*wpr+wtemp*wpi+wi
16  continue
  if (isign.eq.1) then
    h1r=DAT(1)
    DAT(1)=h1r+DAT(2)
    DAT(2)=h1r-DAT(2) !Squeeze the first and last data together to get
    !them all within the original array.
  else
    h1r=DAT(1)
    DAT(1)=c1*(h1r+DAT(2))
    DAT(2)=c1*(h1r-DAT(2))
    call four1(DAT,n/2,-1) !This is the inverse transform for the case isign=-1.
  endif
  return
END

```

```

SUBROUTINE twofft(data1,data2,fft1,fft2,n)
  INTEGER n
  REAL data1(n),data2(n)
  COMPLEX fft1(n),fft2(n)
  ! USES four1
  !Given two real input arrays data1(1:n) and data2(1:n), this routine calls four1 and
  ! returns two complex output arrays, fft1(1:n) and fft2(1:n), each of complex length n
  ! (i.e., real length 2*n), which contain the discrete Fourier transforms of the respective data
  ! arrays. n MUST be an integer power of 2.
  INTEGER j,n2
  COMPLEX h1,h2,c1,c2
  c1=cmplx(0.5,0.0)
  c2=cmplx(0.0,-0.5)
  do 14 j=1,n
    fft1(j)=cmplx(data1(j),data2(j)) !Pack the two real arrays into one complex
    !array.
14  continue
    call four1(fft1,n,1) !Transform the complex array.
    fft2(1)=cmplx(aimag(fft1(1)),0.0)
    fft1(1)=cmplx(real(fft1(1)),0.0)
    n2=n+2
    do 15 j=2,n/2+1
      h1=c1*(fft1(j)+conjg(fft1(n2-j))) !Use symmetries to separate the two trans
      h2=c2*(fft1(j)-conjg(fft1(n2-j))) !forms.
      fft1(j)=h1 !Ship them out in two complex arrays.
      fft1(n2-j)=conjg(h1)
      fft2(j)=h2
      fft2(n2-j)=conjg(h2)
15  continue
  return
END

```

```
SUBROUTINE FOUR1(DAT, NN, ISIGN)
Implicit none

REAL tempr, tempi
INTEGER i, istep, j, m, mmax, n, nn, isign
REAL*8 WR, WI, WPR, WPI, WTEMP, THETA
REAL, dimension(2*nn):: DAT
N=2*NN
J=1
DO 11 I=1, N, 2
IF(J.GT.I)THEN
TEMPR=DAT(J)
TEMPI=DAT(J+1)
DAT(J)=DAT(I)
DAT(J+1)=DAT(I+1)
DAT(I)=TEMPR
DAT(I+1)=TEMPI
ENDIF
M=N/2
IF((M.GE.2).AND.(J.GT.M))THEN
J=J-M
M=M/2
GOTO 1
ENDIF
J=J+M
CONTINUE
MMAX=2
IF(N.GT.MMAX)THEN
ISTEP=2*MMAX
THETA=6.28318530717959D0/(ISIGN*MMAX)
WPR=-2.D0*DSIN(.5D0*THETA)**2
WPI=DSIN(THETA)
WR=1.D0
WI=0.D0
DO 13 M=1, MMAX, 2
DO 12 I=M, N, ISTEP
J=I+MMAX
TEMPR=SNGL(WR)*DAT(J)-SNGL(WI)*DAT(J+1)
TEMPI=SNGL(WR)*DAT(J+1)+SNGL(WI)*DAT(J)
DAT(J)=DAT(I)-TEMPR

DAT(J+1)=DAT(I+1)-TEMPI
DAT(I)=DAT(I)+TEMPR
DAT(I+1)=DAT(I+1)+TEMPI
CONTINUE
WTEMP=WR
WR=WR*WPR-WI*WPI+WR
WI=WI*WPR+WTEMP*WPI+WI
CONTINUE
MMAX=ISTEP
GOTO 2
ENDIF
RETURN
END
```


FFT

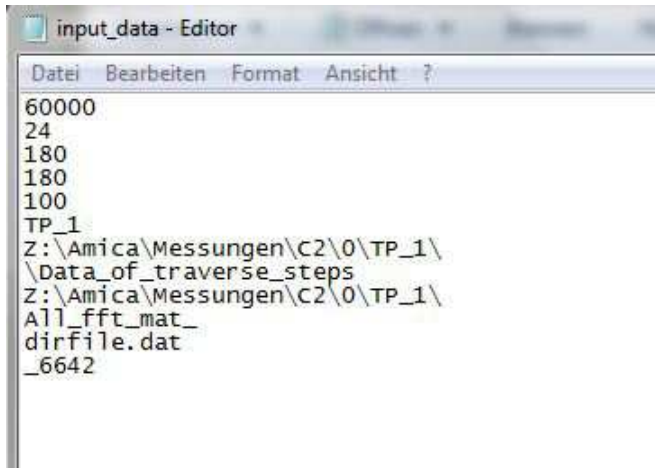
Dateiname: fft_all.m bzw. fft_ref.m (Referenzmikrofon)

Input-Files:

- input_data.dat
- pla_poli.dat
- ord.dat (beinhaltet die Position bzw. die Winkel der Mikrofone)
- sub_exchange.m (ersetzt in Dateinamen “.” durch “-”)
- sub_freq.m (beinhaltet die BPF und ihre Harmonischen für verschiedene Betriebspunkte, wobei hier nur einer davon betrachtet wird)

```
clear all;
close all;

fid = fopen('input_data.dat', 'r');
fs= fscanf(fid, '%d', 1)
number_mics=fscanf(fid, '%d', 1)
endposition=fscanf(fid, '%d', 1)
dummy=fscanf(fid, '%d', 1)
dummy=fscanf(fid, '%d', 1)
names=textscan(fid, '%s', 'delimiter', '\n', 'whitespace', '');
names=names{1}
TPNr=names{2,1}
name1=names{3,1};
name2=names{4,1};
savedir=names{5,1}
bpf=names{8,1};
bpf=str2num(bpf(2:length(bpf)))
fclose(fid);
```



```
input_data - Editor
Datei Bearbeiten Format Ansicht ?
60000
24
180
180
100
TP_1
Z:\Amica\Messungen\C2\0\TP_1\
\Data_of_traverse_steps
Z:\Amica\Messungen\C2\0\TP_1\
All_fft_mat_
dirfile.dat
_6642
```

Zu Beginn werden die Variablen aus dem Input-File mit dem Namen “input_data” eingelesen und einem Namen zugeordnet, alle mit “dummy” bezeichneten Parameter werden zwar eingelesen, jedoch in weiterer Folge nicht verwendet:

- f_s = Abtastfrequenz
- number_mics = Anzahl der Mikrofone
- endposition = Anzahl der Messpositionen in Umfangsrichtung
- “names = textscan()” liest den Text ab “TP_1” ein
- TPNr = Bezeichnung des Testpunkts
- name1 und name2 werden zwar eingelesen, aber danach nicht mehr verwendet
- savedir = Pfad, in dem die Messdaten liegen, bzw. die Ergebnisse der FFT abgespeichert werden
- bpf = Blade Passing Frequency: HP oder LP, je nachdem welcher Trigger verwendet wird (letzte Zeile des Input-Files); bei der FFT nach RSA ist Vorsicht geboten, da hier die 1. BPF des LP Rotors angegeben werden muss (bei RSA hat man sich auf den LP Rotor bezogen), für die AMA jedoch die Summe der ersten BPFs

Nach dem Einlesen von “input_data.dat” wird auch das File “pla_poli.dat”, das auch schon für die Adaptive Neuabtastung und das Phase Averaging verwendet wurde, eingelesen und den Variablen Namen zugewiesen:

```

fid = fopen('pla_poli.dat', 'r');
n_mic_check= fscanf(fid, '%d', 1);
trig_channel=fscanf(fid, '%d', 1);
%dummy=fscanf(fid, '%d', 1);
refmic_channel=fscanf(fid, '%d', 1);
dummy=fscanf(fid, '%d', 1);
dummy=fscanf(fid, '%d', 1);
nn=fscanf(fid, '%d', 1);
check=fscanf(fid, '%d', 1);
dummy=fscanf(fid, '%d', 1);
dummy=fscanf(fid, '%d', 1);
%dummy=fscanf(fid, '%d', 1);
names_pla=textscan(fid, '%s', 'delimiter', '\n', 'whitespace', '');
names_pla=names_pla{1}
dum=names_pla{2,1}
TP_check=names_pla{3,1}
pathname=names_pla{4,1}
fclose(fid);

```

- n_mic_check = Anzahl der Mikrofone
- trig_channel = Kanal des verwendeten Triggers
- refmic_channel = Kanal des Referenzmikrofons
- nn = Resamples pro Umdrehung (siehe Abschnitt über Adaptive Resampling und Phase Averaging)
- check = numrev aus dem Abschnitt über Phase Averaging
- “names_pla” liest wieder den Text ab Zeile 9 von “pla_poli.dat” ein
- TP_check = fprefix aus dem Abschnitt über Phase Averaging
- pathname = Pfad, in dem die Daten nach Adaptive Resampling und Phase Averaging abgespeichert wurden

```

if n_mic_check==number_mics

```

Das Programm wird nur dann fortgesetzt, wenn die Anzahl der Mikrofone in beiden Input-Files identisch ist.

```
b1 = 36;
max_dargestellte_frequenz = 15000;
max_f_plot_y = 160;
min_f_plot_y = 40;

xpos_vect = [364, 371.5, 379, 386.5, 394, 401.5, 409, 416.5, 424, 431.5, 439, 446.5,
xpos_ref = '164';

mat_speichern = 1;

startposition = 1;
d_alpha = 2;
start_angle = 0;

fs=bpf/(b1)*nn; %%%%%%%%%%%
%fs=bpf/216*nn; %%%%%%%%%%%
dd=1;
ddr=1;
ddR=1;
load 'ord.dat';
```

Die allgemeinen Eingaben beinhalten:

- `b1` = Anzahl der Schaufeln, je nach dem auf welchen der beiden Rotoren, HP oder LP, getriggert wird; wird die FFT nach der RSA für die Summe aus beiden Rotoren durchgeführt, so nimmt man die Schaufelzahl des LP Rotors, da auch beim Wert “VPTNpk” (siehe RSA) derjenige für den LP Rotor gewählt wurde
- `max_dargestellte_frequenz` = Grenze für die x-Achse, bis zu der die FFT grafisch dargestellt wird
- `max_f_plot_y` und `min_f_plot_y` = Grenzwerte für die y-Achse
- Der Vektor “`xpos_vect`” enthält die x-Positionen der 24 Mikrofone, “`xpos_ref`” die axiale Position des Referenzmikrofons
- wenn “`mat_speichern = 1`”, dann werden die Ergebnisse der FFT bei bestimmten Frequenzen gespeichert (siehe weiter unten)
- `d_alpha` = Schrittweite der Messpunkte in Umfangsrichtung
- `start_angle` = Winkel, bei dem die Messung beginnt

- f_S stellt die berechnete Sampling-Frequenz dar, die sich aufgrund der Adaptiven Neuabtastung ergibt
- “dd”, “ddr” und “ddR” werden in weiterer Folge als Zähler verwendet und an dieser Stelle deklariert
- Die im Anschluss geladene Datei “ord.dat” wird in erster Linie für die RMA benötigt und berücksichtigt den Winkelversatz zwischen den beiden Mikrofonreihen (siehe Kapitel 4.2)

```
for k=1:number_mics
    mic_nr = num2str(k);
    start_angle=ord(k);

    xpos=num2str(xpos_vect(k));
    run sub_exchange;
    r_Pos_Nr = '1';

    [savepath] = [savedir TPNr '\kreis_mic_x' xpos '\'];
    mkdir(savepath);

for i = startposition:endposition
    if (i-1)*d_alpha+start_angle>359
        winkelpos = (i-1)*d_alpha+start_angle-360;
    else
        winkelpos = (i-1)*d_alpha+start_angle;
    end
    winkelpos_name=int2str(winkelpos);
```

- Mit dem ersten “for” startet eine Schleife über alle 24 Mikrofone
- Für jedes Mikrofon wird der Startwinkel festgelegt, der entweder 0° oder 6° ist, je nachdem in welcher Mikrofonreihe sich der betrachtete Sensor befindet
- Anschließend werden die axialen Positionen in einen String umgewandelt und mit dem Unterprogramm “sub_exchange” das Komma darin in einen Bindestrich umgewandelt
- Ist “r_Pos_Nr” gleich 1, so bedeutet das, dass die innere Kanalwand betrachtet wird, ist der Wert 2, so betrachtet man die äußere Kanalwand; nachfolgend wird diese Zahl jedoch nur für die Namensgebung der Ergebnisdateien verwendet

- Im momentanen Arbeitsverzeichnis werden mit dem Befehl “mkdir” Ordner mit dem Namen “kreis_mic_x” und der axialen Position erstellt, in denen die Ergebnisse der FFT abgespeichert werden
- Die zweite for-Schleife legt fest, dass, wenn der Traversierungswinkel 360° überschreitet, wieder bei 0° zu zählen begonnen wird

```
name = [ TP_check int2str(i) '_RmaMic_0' mic_nr '.bin'];  
name_tot = [pathname name];
```

```
A= load (name_tot);
```

- Mit “name” bzw. “name_tot” wird dem Pfad, in dem sich für jedes Mikrofon die Dateien nach Durchführung der Neuabtastung und des Phase Averaging befinden (Ordner “save”), ein Name zugewiesen und anschließend die Inhalte dieser .bin-Datei in die Matrix A geschrieben
- A enthält nach Durchlauf beider for-Schleifen sowohl die Messdaten für alle 24 Mikrofone als auch für alle 180 Messpunkte

```
l=length(name);  
j=1;  
for j=1:l  
    if name(j) ~= '_'  
        namenew(j)=name(j);  
    else  
        namenew(j)='-';  
    end;  
end;
```

- Oben stehende for-Schleife ersetzt in den File-Namen die Unter- durch Bindestriche

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    Nfft=2^13;%Nfft=3100; %scare!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    if Nfft/nn>check
        disp (['waaaaaaaaaaaa wrong!!!! reduce fft length!!!!!!!!'])
        break
    end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% FFT %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

YA = fft(A,Nfft);
YA = (YA(1:length(YA)/2))/(Nfft/2^0.5); % for further process
f = (0:Nfft-1)*(fs/Nfft);
Pyy = abs(YA);
max_f_plot=round((length(f)/2)/(fs/2)*max_dargestellte_frequenz);
Pyy=Pyy(1:max_f_plot);

```

- Die Variable “Nfft” legt die Länge der FFT fest, die eine Potenz von 2 sein sollte (siehe Kapitel 3.6)
- Danach muss sicher gestellt werden, dass die Länge der FFT nicht das Produkt aus Resamples (nn) und Anzahl der gemittelten Blöcke (check, siehe dazu Abschnitt über Phase Averaging) und damit die Gesamtzahl der Samples während der Messzeit übersteigt
- Mit “fft(A,Nfft)” wird die FFT der Matrix A mit der Länge Nfft durchgeführt und deren Ergebnisse in “YA” gespeichert
- Aufgrund der Symmetrie wird für die weiteren Betrachtungen nur die erste Hälfte der FFT betrachtet und außerdem die Werte mittels einer Division durch $Nfft/\sqrt{2}$ (Effektivwert der FFT-Länge) normiert
- Weiters werden im Folgenden nur die Absolutwerte der Amplituden betrachtet (abs(YA))
- Außerdem muss der Frequenzwert “f” richtig skaliert und mit “max_f_plot” sein maximaler Wert, der auf der x-Achse dargestellt wird, ermittelt werden

```

Pyyyr(:,ddr) = Pyy;
ddr=ddr+1;
if rem(i,10)==0
    disp(['files to go:' int2str(endposition*number_mics-((ddr-1)))]);
end

run sub_freq

REA1(i,1)=real(max(YA((snf(1)-5):(snf(1)+5))));
REA1(i,2)=imag(max(YA((snf(1)-5):(snf(1)+5))));
REA2(i,1)=real(max(YA((snf(2)-5):(snf(2)+5))));
REA2(i,2)=imag(max(YA((snf(2)-5):(snf(2)+5))));
REA3(i,1)=real(max(YA((snf(3)-5):(snf(3)+5))));
REA3(i,2)=imag(max(YA((snf(3)-5):(snf(3)+5))));

clear Y;
clear B_cut; clear B; clear B_cut2;
clear B_cut_m2;
clear A;
clear w; clear Ein; clear trig;
clear refmic; clear Pyy;

```

- In “Pyyyr” sind nun die Ergebnisse nach Durchführung der FFT für alle 180 Messpositionen und die 24 Mikrofone enthalten; “ddr” stellt hier den Zähler für die Mikrofone dar
- Mit “run sub_freq” wird nachfolgendes Matlab-File ausgeführt, das die BPF und ihre erste Harmonische bzw. die Summe der BPFs von HP und LP Rotor beinhaltet; betrachtet wird nur der erste Fall (Cutback), da das vierte Zeichen von “TPNr” (siehe input_data.dat) ein Einser ist und daher der ersten if-Verzweigung entspricht
- Außerdem wird in genanntem Matlab-File der Wert “snf” berechnet, der später neben “sf(1)”, “sf(2)” und “sf(3)” für das Abspeichern der Dateien zum Einsatz kommt
- Die folgenden sechs Zeilen ermitteln sowohl Real- als auch Imaginärteil der maximalen Amplituden nach Durchführung der FFT und zwar bei den drei Frequenzen aus der Subroutine “sub_freq” (BPF + 1.Harmonische + Summe aus LP und HP); die Informationen werden dann ausschließlich für diese drei Frequenzen abgespeichert, da man ansonsten $N_\varphi \cdot N_{ax} \cdot 2^{13}$ zu speichernde Werte erhalten würde

sub_freq.m Anfang

```

if TPNr(4) == '1' % TP-001 = Cutback
    sf(1)=6642; %DIFF
    sf(2)=13284; %wrong
    sf(3)=10910; %wrong
    %sf(1)=4259; %BPF LP
    %sf(2)=6640; %BPF HP
    %sf(3)=10900; %SUM
end
if TPNr(4) == '2' % TP-002 = Sideline
    sf(1)=7090; %wrong
    sf(2)=7560; %wrong
    sf(3)=15120; %wrong
end

if TPNr(4) == '3' % TP-003 = Approach
    sf(1)=4130; %4090 TP 033
    sf(2)=4967;
    sf(3)=9935;
end

for ifreq=1:3
    snf(ifreq)=round(sf(ifreq)*Nfft/fs);
end

```

sub_freq.m Ende

Nun werden die berechneten Informationen für die drei oben angeführten Frequenzen im angegebenen Pfad und in den zu Beginn erstellten Ordnern für jede axiale Position aus dem Vektor "xpos_vect" gespeichert:

```

#####
#### read and create the fft's #####
##### END #####
end

if mat_speichern == 1
    %##### file schreiben als *.dat #####
    savefile = ([savepath 'All_fft_mat_' xpos '_' r_Pos_Nr '_' winkelpos_name '_' int2str(sf(1)) '.dat']);
    save (savefile, '-ascii','REA1');
    savefile = ([savepath 'All_fft_mat_' xpos '_' r_Pos_Nr '_' winkelpos_name '_' int2str(sf(2)) '.dat']);
    save (savefile, '-ascii','REA2');
    savefile = ([savepath 'All_fft_mat_' xpos '_' r_Pos_Nr '_' winkelpos_name '_' int2str(sf(3)) '.dat']);
    save (savefile, '-ascii','REA3');
end
end

```

```
%if ddr>10
PMeanR=mean(Pytyr,2);
PMeanR=20*log10(PMeanR/(2e-5));
hold off;
figure('Name',['FFT ave tip'],'NumberTitle','off'); plot(f(1:(max_f_plot)), PMeanR(1:(max_f_plot)));axis ([0 max_dargestellte_
hold on;
title(['FFT ave tip']);grid;
xlabel('Frequency [kHz]');ylabel('SPL [dB]')
saveas(gcf,([savepath '\fig_fft_mean_tip' ]), 'fig')
saveas(gcf,([savepath '\fig_fft_mean_tip' ]), 'tiff')
% end
% else
% disp(['Program ended!!!!!!!!!! Pls check IP names!!!!!!!!!!'])
% end
else
disp(['Program ended!!!!!!!!!! Pls check mic number!!!!!!!!!!'])
end
```

- Bevor die FFT grafisch dargestellt wird, wird mit “mean” an jeder der 180 Umfangspositionen über die 24 Mikrofone gemittelt und dieser Wert anschließend in Dezibel umgerechnet
- Danach wird der FFT-Graph erstellt und die Bilder im Ordner der letzten axialen Position abgespeichert
- Würde die Anzahl der Mikrofone in den beiden Input-Files nicht übereinstimmen, so würde das Programm abgebrochen und die Fehlermeldung “Program ended!!!!!!!!!! Pls check mic number!!!!!!!!!!” angezeigt werden

Die Durchführung der FFT für das Referenzmikrofon erfolgt analog, allerdings unter Verwendung der Datei `fft_ref.m`.

Azimutalmodenanalyse (AMA)

Dateiname: AMA_aus_fft.f90 und AMA_for_RMA_mat.m

Input-Files:

- input_data.dat
- list.dat (Liste mit Umfangspositionen)
- list2.dat
- list3.dat
- dirfile.dat (beinhaltet die Namen der Ordner, in denen die Ergebnisse der FFT abgespeichert wurden)

AMA_aus_fft.f90:

```
program signal_smooth
USE MSFLIB
implicit none
real,dimension(2100,24,3)::aa
real,dimension(65536,4)::a,rm
integer,dimension(300,3)::lis,lis2
integer,dimension(300)::ord,ord3
complex,dimension(500)::ca,crm
complex,dimension(500,24,2)::am
complex::amn
integer::endpos,smode
integer::countf,countr,f,hh,i,ii,k,kk,m,mn,nk,nn,rf
integer::dim
character(3)::prad
character(10)::freq
character(500)::pathname,suffixname,TPname
character(500)::dirfile,filename,fprefix,refname,savedir
character,dimension(30)::dir*500,pdir*5,ord2*3
character,dimension(30,3)::list*3
dim=70000
```

- Zuerst werden wieder alle verwendeten Variablen deklariert, ihr Datentyp festgelegt und teils auch ihre maximale Größe angegeben
- Siehe dazu auch die Beschreibung des Fortran-Programms für Phase Averaging und Adaptive Resampling
- Danach folgt, wie bereits mehrmals erläutert, das Einlesen der Parameter aus dem Input-File “input_data.dat”

```
open(unit=1,file='input_data.dat',status='old')
read(1,*) endpos
read(1,*) endpos
read(1,*) endpos
read(1,*) endpos
read(1,*) smode
read(1,'(A)') TPname
read(1,'(A)') pathname
read(1,'(A)') suffixname
read(1,'(A)') savedir
read(1,'(A)') fprefix
read(1,'(A)') dirfile
read(1,'(A)') freq
close(1)
```

- Die ersten drei mit “endpos” bezeichneten Variablen werden im weiteren Verlauf des Programms nicht benötigt
- Die vierte mit “endpos” benannte Variable ist die Anzahl der Umfangspositionen
- smode = Anzahl der Moden in positiver und negativer Richtung für die AMA
- TPname = Bezeichnung des Projektes (also TP_1)
- pathname = Pfad des Hauptordners, in dem die Auswertung durchgeführt wird
- suffixname = Ordner, der die Messdaten nach Konvertierung ins DLR-Format enthält
- savedir = Pfad, in dem die Ergebnisse gespeichert werden
- fprefix = Präfix der Files, nach Durchführung der FFT (All_fft_mat)
- dirfile = File, in dem die Namen der Ordner enthalten sind, die bei der FFT erstellt werden
- freq = Frequenz, bei der die AMA durchgeführt wird; hier ist entweder die erste BPF oder eine Harmonische anzugeben (für HP und LP Trigger), bei der RSA muss hier die Summe aus der ersten BPF des HP Rotors und des LP Rotors eingetragen werden

```

open(unit=2,file='list.dat',status='old')
do kk=1, endpos
  read(2,*,end=44) (lis(kk,i), i=1,2)
end do
continue
close(2)
open(unit=3,file='list2.dat',status='old')

nk=0
do kk=1,dim
  read(3,*,end=45) (list(kk,i), i=1,2)
  nk=nk+1
end do

continue
close(3)
open(unit=4,file='list3.dat',status='old')
do kk=1,180
  read(4,*,end=46) (lis2(kk,i), i=1,2)
end do
continue
close(3)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

open(unit=31,file=dirfile,status='old')
countf=0
do kk=1,dim
  read(31,*,end=41) dir(kk),pdir(kk)
  countf=countf+1
end do
continue
countf=countf-1

```

- Oben werden die Files “list.dat”, “list2.dat”, “list3.dat” und “dirfile.dat” geöffnet und eingelesen; ihre Funktion wurde bereits weiter oben erklärt
- Beim Einlesen der Datei “dirfile.dat” steht “dir(kk)” für die erste Spalte des dat-Files und “pdir(kk)” für die zweite Spalte
- “countf” ist der Zähler für die Anzahl an x-Positionen und gleichzeitig auch die Anzahl der Mikrofone

```

!!!!!!!!!!!! reading ref_microphone file !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

refname= fprefix(:len(trim(fprefix))//pdir(countf+1)(:len(trim(pdir(countf+1))))//'_2'//freq(:len(trim(freq)))//'.dat')
countf=0
open (unit=32, file=savedir(:len(trim(savedir))//TPName(:len(trim(TPName))//dir(countf+1)(:len(trim(dir(countf+1))))//refname, status='old')
write(*,*) savedir(:len(trim(savedir))//TPName(:len(trim(TPName))//dir(countf+1)(:len(trim(dir(countf+1))))//refname
do kk=1, endpos
  read(32,*,end=42) (ra(kk,ii), ii=1,2)
  countf=countf+1
end do
continue
close(32)

do rf=1,countf
  cra(rf)=cplx(ra(rf,1),ra(rf,2))
end do

open(unit=57,file=savedir(:len(trim(savedir))//TPName(:len(trim(TPName))//'\AMA_complex'//freq(:len(trim(freq)))//'.dat', status='replace')
open(unit=58,file=savedir(:len(trim(savedir))//TPName(:len(trim(TPName))//'\AMA_abs'//freq(:len(trim(freq)))//'.dat', status='replace')

```

- Begonnen wird mit dem Öffnen und Lesen der Datei für das Referenzmikrofon, die die Ergebnisse nach Durchführung der FFT beinhaltet (Ordner mit dem Präfix “ref_mic_” und Datei mit dem Präfix “All_fft_mat_”)

- Dieses .dat-File beinhaltet zwei Spalten, die den Real- und Imaginärteil der Ergebnisse darstellen, und 180 Zeilen, eine Zeile für jede Umfangsposition
- “countr” ist der Zähler für die Anzahl der Messpunkte in Umfangsrichtung
- In “rm(kk,ii)” werden dann die Werte gespeichert, wobei “kk” der Index für die Umfangsposition und “ii” der Index für die Spalte ist
- Der Befehl “cplx” gibt nun für jede Umfangsposition eine komplexe Zahl zurück, deren Realteil “rm(rf,1)” und deren Imaginärteil “rm(rf,2)” ist
- Anschließend werden die Dateien mit dem Präfix “AMA_complex” bzw. “AMA_abs” erstellt, in die später die Ergebnisse der AMA gespeichert werden

```

!!!!!!!!!!!! loop on hub/tip !!!!!!!!!!!!!
do 1 hh=1,1
  write (prad,'(I1)') hh

!!!!!!!!!!!! loop on xpositions !!!!!!!!!!!!!
do 2 f=1,countf

  if (mod(f,2).eq.0) then
    do i=1,endpos
      ord(i)=lis(i,2)
      ord3(i)=lis2(i,2)
    end do
    do i=1,nk
      ord2(i)=list(i,2)
    end do
  else
    do i=1,endpos
      ord(i)=lis(i,1)
      ord3(i)=lis2(i,1)
    end do
    do i=1,nk
      ord2(i)=list(i,1)
    end do
  end if

  nn=0

```

- Zunächst wird eine Schleife über alle axialen Positionen (countf) gestartet
- Die darauf folgende if-Verzweigung legt fest, wie die Mikrofone untereinander bzw. mit welchen Referenzmikrofonen sie kombiniert werden und ist nur dann wirklich von Bedeutung, wenn mehrere mit Sensoren bestückte Mikrofonplatten und damit auch mehrere Referenzmikrofone verwendet werden, deren Messergebnisse zusammengeführt werden müssen (Kriterium für die Verzweigung ist, ob es sich um eine axiale Position mit gerader oder ungerader Nummer handelt)

```
!!!!!!! Reading all data !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
do k=1,nk
  filename=fprefix(:len(trim(fprefix)))//pdir(f(:len(trim(pdir(f))))//'_/'//prad(:len(trim(prad)))//'_/'//ord2(k(:len(trim(ord2(k))))//freq(:len(trim(tr
write(*,*) pathname(:len(trim(pathname)))//dir(f(:len(trim(dir(f))))//filename
open (unit=33, file=savedir(:len(trim(savedir)))//TPhame(:len(trim(TPhame)))//dir(f(:len(trim(dir(f))))//filename, status='old')
do kk=1,dim
  read(33,*,end=43) (a(kk,ii), ii=1,2)
end do
continue
close(33)
do i=1,endpos/nk
  nn=nn+1
  ca(nn)=cplx(a(i,1),a(i,2))
end do
end do
```

- Das Einlesen der Daten für die restlichen 24 Mikrofone erfolgt analog zum Referenzmikrofon
- Es werden wieder Real- und Imaginärteil der Werte gespeichert, diesmal in “a(kk,ii)” und danach mit dem Befehl “cplx” eine komplexe Zahl mit dem Namen “ca(nn)” zurückgegeben; “nn” ist in diesem Fall der Zähler für die Umfangspositionen, da bei einer einzigen Mikrofonplatte nk=1 ist

```
!!!!!!! AMN decomposition !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
mn=1
do m=-smode,smode
  am(m,f,hh)=(0.,0.)
  aam=(0.,0.)
  do k=1,endpos
    am(mn,f,hh)=sqrt(2.)*ca(ord3(k))*CONJG(crm(ord(k)))*exp(real(am)*real(k-1)*2.*(0.1)*0.0174532925199)/(crm(ord(k))*CONJG(crm(ord(k))))**0.5
    aam=am(mn,f,hh)+aam
  end do
  am(m,f,hh)=aam/real(endpos)
  mn=mn+1
end do
write(*,*) hh,f
!!!!!!! end decomposition !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

- Nun wird für den angegebenen Moden-Bereich (smode) “am” nach Gleichung 5.20 berechnet, wobei der Winkel ins Bogenmaß umgerechnet wird (Multiplikation des momentanen Winkels mit 0.017...)
- Mit $\sqrt{2}$ wird multipliziert, weil zuvor bei der FFT mit genau diesem Wert normiert wurde
- In diesem Code-Abschnitt wird außerdem eine Mittelung der Werte über alle 180 Umfangspositionen durchgeführt
- “mn” ist der Zähler für die Anzahl der Moden

```

!!!!!!! Writing complex (for RMA) and modulus (for plot)!!!!!!!!!!!!!!!!!!!!!!
      do mn=1, mn-1
        aa(mn, f, hh)=0.5*(abs(am(mn, f, hh)))**2
        write(58,100) aa(mn, f, hh)
        write(57,101) real(am(mn, f, hh)), aimag(am(mn, f, hh))
      end do

end do
end do

close(57)
close(58)
format(f18.8)
format(2f18.8)

END

```

- Zum Schluss wird für jede Mode die komplexe Amplitude nach Gleichung 5.21 berechnet und diese dann in der zuvor erstellten Datei mit dem Präfix “AMA_abs” abgespeichert
- In “AMA_complex” werden Real- und Imaginärteil von “am” für jede Mode gespeichert

Nach dem Erstellen der beiden oben genannten Dateien mit Hilfe des Fortran-Programms, erfolgt die endgültige Auswertung der AMA mit folgender Matlab-Routine:

AMA_for_RMA_mat.m:

```

clear all;
close all;

schriftgroesse=30;
zahlengroesse=20;
schrift='Helvetica';
schriftart1='bold';
schriftart2='italic';

```



```

fid = fopen('input_data.dat', 'r');
dummy= fscanf(fid,'%d',1);
nmic=fscanf(fid,'%d',1)
dummy=fscanf(fid,'%d',1);
dummy=fscanf(fid,'%d',1);
nmode=fscanf(fid,'%d',1)*2+1;
names=textscan(fid, '%s', 'delimiter', '\n', 'whitespace', '');
names=names{1}
TPNr=names{2,1}
name1=names{3,1};
name2=names{4,1};
savedir=names{5,1}
fama=names{8,1};
fclose(fid);

```

- Am Anfang des Programms werden zuerst Variablen für die grafische Darstellung der AMA deklariert (schriftgrosse, zahlengrosse, etc.)
- Danach werden, wie schon im Abschnitt über das Programm “fft_all.m” erklärt, den Parametern aus dem File “input_data.dat” Namen zugeordnet
- Im Gegensatz zur FFT werden hier jedoch “name1” und “name2” verwendet
- Vorsicht ist hier beim Einlesen von “nmode” geboten, da an dieser Stelle mit 2 multipliziert wird und daher weiter unten wiederum durch 2 dividiert werden muss

```

[pathname] = [name1 TPNr name2]
[savepath] = [savedir TPNr ]

AMA_complex=load(['AMA_complex' fama '.dat']);
AMA_abs=load(['AMA_abs' fama '.dat']);

for k=1:length(AMA_complex)
    Y(k)=AMA_complex(k,1)+j*AMA_complex(k,2);
end

BB=reshape(Y,nmode,nmic);
BA=reshape(AMA_abs,nmode,nmic);
x=-((nmode-1)/2):(nmode-1)/2;

```

- Nach dem Festlegen des Pfads, wo einerseits die Messergebnisse vorzufinden sind, andererseits die Ergebnisse nach Durchführung der AMA abgespeichert werden, werden die beiden Dateien, die mit der zuvor beschriebenen Fortran-Routine erstellt wurden, geladen

- In der for-Schleife werden die beiden Spalten mit Real- und Imaginärteil aus “AMA_complex” wieder zu einer komplexen Zahl “Y(k)” zusammengesetzt
- Mit “reshape” wird die Dimension der Matrix, die die komplexen Werte enthält, und der Matrix, die die komplexen Amplituden beinhaltet, verändert (Matrizen BB und BA); die Anzahl der Spalten entspricht bei beiden “nmode”, die der Zeilen “nmic”
- “x” legt den Bereich der Moden auf der x-Achse fest; hier wird, wie schon weiter oben erwähnt, durch 2 dividiert

```
if str2double(fama(2:end)) == 6642
    mvect=(-44:8:36);
elseif str2double(fama(2:end)) == 13284
    mvect=(-96:8:80);
elseif str2double(fama(2:end)) == 10910
    mvect=(-84:8:68);
elseif str2double(fama(2:end)) == 4260
    mvect=(-32:8:24);
elseif str2double(fama(2:end)) == 8520
    mvect=(-64:8:48);
end
```

- In der obigen if-Verzweigung werden, je nachdem ob es sich um die Auswertung mit dem HP Trigger, dem LP Trigger oder der RSA handelt, für jede BPF und ihre erste Harmonische die Moden angegeben, die sich durch den Kanal ausbreiten
- “fama” ist hier die Frequenz, die im Input-File angegeben werden muss und somit festlegt, um welchen Trigger bzw. um welche Harmonische es sich handelt

```

%considering all the 24 mics =24
summ(length(mvect))=0;
for k=1:nmic
    for i=1:length(mvect)
        summ(i)=BA(mvect(i)+(nmode-1)/2+1,k)/(nmic)+summ(i);
    end
end

total=20*log10(mean(summ)/2e-5);

fig1=figure; plot(x,20*log10(BA(:,1:k)/2e-5)); axis ([-((nmode-1)/2) (nmode-1)/2 60 200]);
str={['Modes mean value: ' num2str(total) ]};
xlabel(str)
saveas(fig1,['fig_Mode_24' fama ], 'tiff')
saveas(fig1,['fig_Mode_24' fama ], 'fig')

```

- Innerhalb der beiden for-Schleifen wird zuerst über die Anzahl der Mikrofone gemittelt (Division durch “nmic”)
- Mit der Variable “total” erfolgt nicht nur die Umrechnung in dB, sondern auch die Mittelwertbildung über alle Moden; dieser Wert wird in der Grafik unterhalb der x-Achse angezeigt
- Der Plot wird anschließend jedoch nicht für einen Mittelwert, sondern für jedes der Mikrofone dargestellt und enthält somit 24 verschiedenen Graphen für alle Moden (Werte in BA(:,1:k) werden in dB umgerechnet)

```

%%mean of each mode over all 24 mics%%
for m=1:nmode
    meanModes(m,1)=mean(BA(m,1:nmic));
end

fig2=figure; bar(x,20*log10(meanModes/2e-5)); axis ([-((nmode-1)/2) (nmode-1)/2 60 200]);
$secondline={['Modes mean value: ' num2str(total) ]};
xlabel('Mode m', 'FontSize',schriftgroesse,'FontName',schrift, 'FontAngle',schriftart2,'fontWeight', schriftart1);
ylabel('SPL [dB]', 'FontSize',schriftgroesse,'FontName',schrift, 'FontAngle',schriftart2,'fontWeight', schriftart1);

saveas(fig2,['mean_fig_Mode_24' fama ], 'tiff')
saveas(fig2,['mean_fig_Mode_24' fama ], 'fig')

hold on
for i=1:length(mvect)
    ind=mvect(i);
    meanmodes(i)=meanModes(ind+101);
end
bar(mvect,20*log10(meanmodes/2e-5),'FaceColor',[0 0 0],'BarWidth',0.1)
clear meanModes

```

- Bei jeder Mode wird nun der Mittelwert über alle 24 Mikrofone gebildet und anschließend grafisch dargestellt
- Auch diese Grafik wird sowohl als “tiff”, als auch als “fig” abgespeichert
- Zusätzlich werden die Grafiken, die die über alle 24 Mikrofone gemittelten Werte enthalten, als Balkendiagramme dargestellt

Dasselbe wie zuvor für alle 24 Mikrofone, wird im Anschluss zuerst für die vorderen 12 und danach für die hinteren 12 Mikrofone durchgeführt:

```

%considering the first 12 mics =12A
summ(length(mvect))=0;
for k=1:nmic/2
    for i=1:length(mvect)
        summ(i)=BA(mvect(i)+(nmode-1)/2+1,k)/(nmic/2)+summ(i);
    end
end

total=20*log10(mean(summ)/2e-5);

fig3=figure; plot(x,20*log10(BA(:,1:k)/2e-5)); axis ([-((nmode-1)/2) (nmode-1)/2 60 200]);
str={'Modes mean value: ' num2str(total) };
xlabel(str)
saveas(fig3,(['fig_Mode_12A' fama ]), 'tiff')
saveas(fig3,(['fig_Mode_12A' fama ]), 'fig')

***mean of each mode of the first 12 mics***
for m=1:nmode
    meanModes(m,1)=mean(BA(m,1:nmic/2));
end

fig4=figure; bar(x,20*log10(meanModes/2e-5)); axis ([-((nmode-1)/2) (nmode-1)/2 60 200]);
xlabel('Mode m', 'FontSize',schriftgroesse,'FontName',schrift, 'FontAngle',schriftart2,'fontWeight', schriftart1);
ylabel('SPL [dB]', 'FontSize',schriftgroesse,'FontName',schrift, 'FontAngle',schriftart2,'fontWeight', schriftart1);
saveas(fig4,(['mean_fig_Mode_12A' fama ]), 'tiff')
saveas(fig4,(['mean_fig_Mode_12A' fama ]), 'fig')

hold on
for i=1:length(mvect)
    ind=mvect(i);
    meanmodes(i)=meanModes(ind+101);
end
bar(mvect,20*log10(meanmodes/2e-5), 'FaceColor',[0 0 0], 'BarWidth',0.1)
clear meanModes

```

```

%considering the last 12 mics =12B
summ(length(mvect))=0;
for k=(nmic/2+1):nmic
    for i=1:length(mvect)
        summ(i)=BA(mvect(i)+(nmode-1)/2+1,k)/(nmic/2)+summ(i);
    end
end

total=20*log10(mean(summ)/2e-5);

fig5=figure; plot(x,20*log10(BA(:,(nmic/2)+1):k)/2e-5); axis ([-((nmode-1)/2) (nmode-1)/2 60 200]);
str={'Modes mean value: ' num2str(total) '};
xlabel(str)
saveas(fig5,(['fig_Mode_12B' fama ]), 'tiff')
saveas(fig5,(['fig_Mode_12B' fama ]), 'fig')

%%mean of each mode of the last 12 mics%%
for m=1:nmode
    meanModes(m,1)=mean(BA(m,nmic/2+1:nmic));
end

fig6=figure; bar(x,20*log10(meanModes/2e-5),'FaceColor',[1 1 1],'BarWidth',0.1); axis ([-((nmode-1)/2) (nmode-1)/2 60 200]);
xlabel('Mode m', 'FontSize',schriftgroesse,'FontName',schrift, 'FontAngle',schriftart2,'fontWeight', schriftart1);
ylabel('SPL [dB]', 'FontSize',schriftgroesse,'FontName',schrift, 'FontAngle',schriftart2,'fontWeight', schriftart1);
saveas(fig6,(['mean_fig_Mode_12B' fama ]), 'tiff')
saveas(fig6,(['mean_fig_Mode_12B' fama ]), 'fig')

hold on
for i=1:length(mvect)
    ind=mvect(i);
    meanmodes(i)=meanModes(ind+101);
end
bar(mvect,20*log10(meanmodes/2e-5),'FaceColor',[0 0 0],'BarWidth',0.1)

for k=1:nmode
    for i=1:nmic
        B(i,k)=BB(k,nmic-i+1); %Solution 1
        % B(i,k)=BB(k,i); %Solution 2
    end
end

B1=B((nmic/2+1):nmic,:); %Solution 1
%B1=B(1:nmic/2,:); %Solution 2
am_komplex=B1;
savefile = ([savepath '\for_am_komplex_' TPNr '_r_01_f' fama '.mat']);
save (savefile, 'am_komplex');

B2=B(1:nmic/2,:); %Solution 1
%B2=B((nmic/2+1):nmic,:); %Solution 2
am_komplex=B2;
savefile = ([savepath '\for_am_komplex_' TPNr '_r_02_f' fama '.mat']);
save (savefile, 'am_komplex');

```

Mit Hilfe der beiden oben angeführten Schleifen werden die komplexen Werte aus der Matrix BB zuerst für die 12 vorderen Mikrofone und danach für die 12 hinteren Mikrofone in zwei Dateien abgespeichert. Diese Files werden nachfolgend für die Durchführung der Radialmodenanalyse benötigt.

Radialmodenanalyse (RMA)

Dateiname: RMA_aus_AMA_DREAM_swirl.m

Input-Files:

- input_data.dat
- sub_stroemungsparameter.m (enthält die Strömungsparameter)
- eigenw.dat (enthält die Eigenwerte des Kanals)

```
clc;
clear all;

fid = fopen('input_data.dat', 'r');
dummy= fscanf(fid, '%d', 1);
dummy=fscanf(fid, '%d', 1)
dummy=fscanf(fid, '%d', 1);
dummy=fscanf(fid, '%d', 1);
nmode=fscanf(fid, '%d', 1)*2+1
TPNr=fscanf(fid, '%s', 1)
TPNr=TPNr(4:4)
name1=fscanf(fid, '%s', 1)
name2=fscanf(fid, '%s', 1)
savedir=fscanf(fid, '%s', 1)
dum=fscanf(fid, '%s', 1)
dum=fscanf(fid, '%s', 1)
gesuchte_f=fscanf(fid, '%s', 1)
gesuchte_f=str2num(gesuchte_f(2:length(gesuchte_f)))
fclose(fid);
```

- Wie bereits mehrmals beschrieben, werden zuerst die Parameter aus dem File “input_data.dat” eingelesen und einem Namen zugeordnet


```

run sub_stroemungsparameter

##### geometry parameters #####
gaskonst_R = 287.05;
radius_tip = 0.400; % outside radius [m]
radius_hub = 0.339; % inside radius [m]
omega=30./(radius_tip+radius_hub)*2. % ct/r
dx =0.006 ; % distance between the microphones [m]

erg_tiff_speichern = 1;
ampl_RMA_ploten = 1;
inv=0; %evaluated the invers AMA's; 1=yes, 0=no
P_min_disp = 70; %shown maximum sound power
P_max_disp = 120; %shown minimal sound power
minplo      = 70; %shown minimum mode amplitude
maxplo      = 120; %shown maximum mode amplitude

[homepfad]=[savedir 'TP_' TPNr '\\'];
[loadpath]=[savedir 'TP_' TPNr '\\'];

```

- Das Programm “sub_stroemungsparameter.m” wird aufgerufen
- Es folgt die Eingabe bestimmter Geometrieparameter, sowie der Gaskonstanten R und einiger Parameter für die grafische Darstellung der RMA
- Mit der Berechnung von “omega” mittels eines Festkörper-Drallmodells wird hier auch der Drall der Strömung berücksichtigt
- Des Weiteren wird mit “homepfad” der Pfad festgelegt, in dem sich die Datei mit den Eigenwerten des Strömungskanals befindet und mit “loadpath” der, wo alle zur Auswertung verwendeten Programme gespeichert sind

```

vorname = '';
if inv == 1
    vorname = 'inv_';
end

n_max = 10;

```

- Die Variable “vorname” wird weiter unten im Programm für die Namensgebung der abgespeicherten Dateien verwendet
- “n_max” gibt einen hypothetischen Wert für die maximale sich radial ausbreitende Mode und wird im Laufe des Programms neu berechnet

```

load('eigenw.dat','-mat');

% computation of values everywhere in the field are assumed to be constant
schallgeschw=sqrt(1.4*gaskonst_R*stat_temp) %im m/s
k=2*pi*gesuchte_f/schallgeschw;
roh=stat_druck*10^5/(gaskonst_R*stat_temp);

%% read in of the complex azimuthalmoden-files for the inner duct wall
loadfile = ([loadpath vorname 'for_am_komplex_IP_' TPNr '_r_01_f_' int2str(gesuchte_f) '.mat'])
load(loadfile, 'am_komplex');
Am1=am_komplex;%(1:6,:);

%% read in of the complex azimuthalmoden-files for the outer duct wall
loadfile = ([loadpath vorname 'for_am_komplex_IP_' TPNr '_r_02_f_' int2str(gesuchte_f) '.mat'])
load(loadfile, 'am_komplex');
Am2=am_komplex;%(1:6,:);

```

- Die Eigenwerte des Strömungskanals können entweder mit Hilfe eines Matlab-Files, das am Ende dieser Beschreibung angeführt ist, berechnet werden (erstellt wird dabei das File “eigenw.dat”) oder vom DLR übernommen werden
- Nach dem Laden der Datei, die die Eigenwerte des Kanals enthält, werden mit Hilfe der Größen aus “sub_stroemungs-parameter” Schallgeschwindigkeit, Wellenzahl k und Dichte berechnet
- In “eigenw.dat” befinden sich die Parameter σ_{mn} und Q_{mn} (siehe dazu Kap. 5.5), die von dem Verhältnis von innerem zu äußerem Kanalradius und der maximalen radialen Mode abhängen; diese Parameter müssen die Bedingungen der schallharten Wand im Kanal erfüllen
- Danach werden die beiden Files, die am Ende des Programms “AMA_for_RMA_mat.m” erstellt wurden und die komplexen Werte der AMA enthalten (eines der Files enthält die Werte für die vorderen, eines die für die hinteren 12 Mikrofone), geladen und ihre Werte in eine Matrix geschrieben (Zeilen stehen für die x-Positionen, Spalten für m)


```

[nr_x,m_summe] = size(Am2);
m_max=(m_summe-1)/2;

%% counter variable -m_max = i=1
m=-1*m_max;

for i=1:m_summe
m=i-(m_max+1);
for n=1:n_max+1 % if n = 1 then the mode 0 meant!

sigma(i,n)=eigenw(abs(m)+1,n,1); %sigma and Q be read from the file
Q=eigenw(abs(m)+1,n,2);
% QQ(i,n)=Q;

%computation of alfa and the positive and negative axial wave numbers
alfa(i,n) = sqrt(1-(1-machzahl^2) * sigma(i,n)^2/((k-m*omega/schallgeschw)*radius_tip)^2));
k_mn_plus(i,n)=(k-m*omega/schallgeschw)/(1-machzahl^2)*(-1*machzahl+alfa(i,n));
k_mn_minus(i,n)=(k-m*omega/schallgeschw)/(1-machzahl^2)*(-1*machzahl-alfa(i,n));

Jm1=besselj(m,radius_hub/radius_tip*sigma(i,n));
Jm2=besselj(m,sigma(i,n));

Ym1=bessely(m,radius_hub/radius_tip*sigma(i,n));
Ym2=bessely(m,sigma(i,n));

f1_dach(i,n)=Jm1+Q*Ym1;
f2_dach(i,n)=Jm2+Q*Ym2;

if m==0 && n==1
f1_dach(i,n)=Jm1;
f2_dach(i,n)=Jm2;
end

```

- Nach Start der beiden Schleifen, eine für die azimutale Ordnung m und eine für die radiale Ordnung n , werden aus “eigenw.dat” die Werte σ_{mn} und Q_{mn} gelesen und ein Faktor “alfa” berechnet
- Mit “alfa” werden schließlich die positive und die negative axiale Wellenzahl berechnet (siehe Kap. 5.5)
- Der Befehl “besselj” bzw. “bessely” berechnet die Bessel- bzw. die Neumannfunktion zuerst für die Nabe ($Jm1$ und $Ym1$) und dann für die Messposition am Gehäuse ($Jm2$ und $Ym2$)
- In den Klammern werden die benötigten Parameter für die Berechnung angegeben

```

if n==1 && m==0
    F1mn=1/2*(1-(radius_hub/radius_tip)^2);
    % F2mn=1/2*(1-(1)^2)+eps % ist 0
else
    F1mn=1/2*((1-m^2/sigma(i,n)^2)*(f2_dach(i,n)*conj(f2_dach(i,n)))-((radius_hub/radius_tip)^2-m^2/sigma(i,n)^2)*(f1_dach(i,n)*conj(f1_dach(i,n))));
    % F2mn=1/2*((1-m^2/sigma(i,n)^2)*(f2_dach(i,n)*conj(f2_dach(i,n)))-(1^2-m^2/sigma(i,n)^2)*(f1_dach(i,n)*conj(f1_dach(i,n))));
end
F1(i,n)=F1mn;
F2(i,n)=F2mn;

f1(i,n)=(1/sqrt(F1mn))*(Jm1+Q*Ym1);
f2(i,n)=(1/sqrt(F1mn))*(Jm2+Q*Ym2);

```

- Mittels der Bessel- und der Neumannfunktion werden dann die Funktionen “f1_dach” (Nabe) und “f2_dach” (Gehäuse) berechnet, die für die Ermittlung der Normalisierungsfunktion “F1mn” benötigt werden
- Eine Fallunterscheidung für $m=0$ und $n=1$ wird vorgenommen
- Mit der Normalisierungsfunktion werden schließlich die Zylinderfunktionen “f1”(Nabe) und “f2” (Gehäuse) ermittelt
- Diese Vorgehensweise ist in Kapitel 5.5 genau beschrieben

```

kln(1:n_max)=50;
for n=1:n_max
    i=m_max-1;
    k_mn_plus(i,n)=k/(1-machzahl^2)*(-1*machzahl+alfa(i,n));
    if imag(k_mn_plus(i,n)) ~= 0
        kln(n)=n;
    end
end
n_max=min(kln)-2;

m=-1*m_max;
h=0;
count=1;

```

- In obigem Abschnitt wird die tatsächliche, höchste Mode berechnet, die sich in radialer Richtung ausbreiten kann
- Zu diesem Zweck werden alle imaginären axialen Wellenzahlen und die zugehörigen nicht ausbreitungsfähigen radialen Moden ermittelt; die kleinste dieser Moden stellt genau die Grenze zu denjenigen dar, die ausbreitungsfähig sind
- “n_max” ist also die höchste ausbreitungsfähige Mode in radialer Richtung

```
for i=1:m_summe
    m=i-(m_max+1);
    for n=1:n_max+1 %wenn n=1 dann ist die mode 0 gemeint!
        for x=1:nr_x
            %
            %
            %
            A1(x)=Am1(x,i);
            A2(x)=Am2(x,i);
            %
            W11(x)=f1(i,n)*exp(j*k_mn_plus(i,n)*x*dx);
            W12(x)=f2(i,n)*exp(j*k_mn_plus(i,n)*x*dx);
            %
            W21(x)=f1(i,n)*exp(j*k_mn_minus(i,n)*x*dx);
            W22(x)=f2(i,n)*exp(j*k_mn_minus(i,n)*x*dx);
            end
            count=count+1;
            % transfer of the matrices
            A(1:length(A1),1)=A1(:);
            A(1:length(A2),1)=A2(:);
            h=h+1;
            %% r1 obere hälfte von W bei n; immer 2 spalten pro n da in der
            %% formel zwei einträge sind (und dann auch A+ und A- rauskommt)
            %% r1 is the upper half of W
            %
            W(1:length(W11),h)=W11;
            W(1:length(W12),h)=W12;
            h=h+1;
            %% r2 is the lower half of W
            %
            W(1:length(W21),h)=W21;
            W(1:length(W22),h)=W22;
        end
    end
end
```

- Für die Berechnung der modalen Amplituden in radialer Richtung werden sowohl die Matrix W_m mit den akustischen Modellgleichungen als auch der Vektor, der die azimutalen modalen Amplituden enthält, definiert
- Vergleiche dazu Abschnitt 5.6: “W12” im Programm entspricht W_m^+ und “W22” entspricht W_m^-

- “A2” enthält die modalen Amplituden in azimuthaler Richtung

```

h=0;

if m~=0

disp(['current mode analyzed: ' int2str(i) ' ' int2str(n) ]);
L = lsqr(W,A,3e-16,3000);
L=pinv(W)*A;
erg=20*log10(abs(L)/(2e-5)); %result in dB
erg=20*log10(abs(L)); %result in dB, sijetsma



---



%% sonderfall ebene welle; hier ist m=n=0 um die normale berechnung
%% weiter zu verwenden zu können wird matrix umgeschrieben
else
disp(['current mode analyzed: ' int2str(i) ' ' int2str(n) ]);
[W_zeilen,W_spalten]=size(W);
[A_zeilen,A_spalten]=size(A);
W2(:,:)=W(:,3:W_spalten);
L=inv((W2'*W2))*W2'*A;
L = lsqr(W2,A,3e-16,3000);

erg=20*log10(abs(L)/(2e-5)); %result in dB
erg=20*log10(abs(L)); %result in dB, nach sijetsma

L2(1:length(L)+2)=0;
L2(1,3:length(L)+2)=L;
L=L2;

erg2(1:length(erg)+2)=0;
erg2(1,3:length(erg)+2)=erg;
erg=erg2;

clear erg2;
clear L2;
end

n=1;
lauf=1;

while n<=n_max+1
    A_plus_komplex(n,i)=L(lauf);
    A_minus_komplex(n,i)=L(lauf+1);

    A_plus(n,i)=erg(lauf); %%result in dB
    A_minus(n,i)=erg(lauf+1); %%result in dB

    n=n+1;
    lauf=lauf+2;
end
end

```

- Bevor nun mit der in Matlab implementierten Methode der kleinsten Fehlerquadrate (Befehl “lsqr”) der Lösungsvektor L berechnet wird, muss noch eine Fallunterscheidung für $m=0$ vorgenommen werden; ist $m=0$, muss die Matrix W umgeschrieben werden, erst dann kann L ermittelt werden
- In “erg” werden die Ergebnisse in dB abgespeichert
- Anschließend werden zuerst die komplexen radialen Amplituden sowohl in (A_plus_komplex) als auch entgegen (A_minus_komplex) der Strömungsrichtung für alle radialen Moden definiert, danach folgen die Amplituden in dB
- Die Laufvariable “lauf” wird deshalb um 2 erhöht, weil jeweils jeder zweite Eintrag des Vektors sich in bzw. entgegen der Strömungsrichtung ausbreitet (siehe Gleichung 5.24)

```

for i=1:m_summe
    for n=1:n_max+1 %wenn n=1 dann ist die mode 0 gemeint!
        if imag(k_mn_plus(i,n))~= 0 || imag(k_mn_minus(i,n))~= 0
            A_plus(n,i)=NaN;
            A_minus(n,i)=NaN;
        end
    end
end
end

```

- Für den Fall, dass die axiale Wellenzahl imaginär ist, wird die Amplitude in dB als “not a number” definiert, hier kann keine Schallausbreitung stattfinden

```

H=10;
xachse=-1*m_max:1:m_max;
xachse2=linspace(-1*m_max, m_max, (2*m_max)/H+1);

if ampl_RMA_ploten == 1

    %%amplitude of the radial modes of A plus
    figure('Name', ['Radialmodenanalyse '], 'NumberTitle', 'off');
    subplot(2,1,1)
    image(flipud(A_plus), 'CDataMapping', 'scaled')
    title(['Amplituden der Radialmoden A'+ von ' int2str(gesuchte_f) ' Hz]);
    xlabel('azimutale Ordnung m')
    ylabel('radiale Ordnung n')
    hot_ud=flipud(hot);
    colormap(hot_ud)
    caxis([minplo maxplo])
    colorbar

    yax=linspace(n_max,0,n_max+1);

```

```

ylim = get(gca, 'YLim'); % ylimit
set(gca, 'YTick', linspace(ylim(1), ylim(2), n_max+1));
set(gca, 'YTickLabel', num2cell(yax))

xlim = get(gca, 'XLim'); % xlimit
set(gca, 'XTick', linspace(xlim(1), xlim(2), (2*m_max+1)/H+1));
set(gca, 'XTickLabel', num2cell(xachse2))

%amplitude of the radial modes of A minus
figure('Name', ['Radialmodenanalyse '], 'NumberTitle', 'off');
subplot(2,1,2)
image(flipud(A_minus), 'CDataMapping', 'scaled')
title(['Amplituden der Radialmoden A^- von ' int2str(gesuchte_f) ' Hz']);
xlabel('azimutale Ordnung m')
ylabel('radiale Ordnung n')
colormap(hot_ud);
caxis([minplo maxplo]);
colorbar

ylim = get(gca, 'YLim'); % ylimit
set(gca, 'YTick', linspace(ylim(1), ylim(2), n_max+1));
set(gca, 'YTickLabel', num2cell(yax))

xlim = get(gca, 'XLim'); % xlimit
set(gca, 'XTick', linspace(xlim(1), xlim(2), (2*m_max+1)/H+1));
set(gca, 'XTickLabel', num2cell(xachse2))

```

- Ein Teil der Ergebnisse wird dann grafisch dargestellt, und zwar die Amplituden der Radialmoden sowohl in als auch entgegen der Strömungsrichtung; dazu werden auf der y-Achse die radiale Modenordnung n und auf der x-Achse die azimutale Modenordnung m aufgetragen
- Die beiden Grafiken (eine für A^+ und eine für A^-) werden in einem Fenster in Form von Barplots angezeigt

```

for i=1:m_summe
    m=i-(m_max+1);

    for n=1:n_max+1 %wenn n=1 dann ist die mode 0 gemeint!
        P_plus(i,n)=0;
        P_minus(i,n)=0;
        if imag(k_mn_plus(i,n))== 0 || imag(k_mn_minus(i,n))== 0

                P_plus(i,n)=(pi*radius_tip^2)/(roh*schallgeschw)*alfa(i,n)*(1-machzahl^2)^2/(1-alfa(i,n)*machzahl)^2*(abs(A_plus_komplex(n,i)))^2*k^C(i,n);
                P_minus(i,n)=(pi*radius_tip^2)/(roh*schallgeschw)*alfa(i,n)*(1-machzahl^2)^2/(1+alfa(i,n)*machzahl)^2*(abs(A_minus_komplex(n,i)))^2*k^C(i,n);
        end
    end
end

```

- Für den Fall, dass eine Mode ausbreitungsfähig ist, also “ k_{mn_plus} ” bzw. “ k_{mn_minus} ” nicht imaginär ist, wird ihre Schalleistung in und entgegen

```

counter=1;
for i=1:m_summe
    for n=1:n_max+1
        res(1,counter)=i-(m_max+1);
        res(2,counter)=n-1;
        res(3,counter)=10*log10(P_plus(i,n)/10^(-12));
        res(4,counter)=10*log10(P_minus(i,n)/10^(-12));
        counter=counter+1;
        % PP_plus_neu(h+1)=PP_plus(i);
        end
    end

savefile_dat = ([loadpath vorname 'Amplitude_RMA_TP_' TPNr '_f_' num2str(gesuchte_f), '.dat']);
res=res';
save (savefile_dat, '-ascii','res');

```

der Strömungsrichtung für alle azimuthalen Moden berechnet; “P_plus” bzw. “P_minus” sind also Matrizen, die aus “n_max” Spalten und “m_summe” Zeilen bestehen

- Die Ergebnisse werden anschließend logarithmiert und gemeinsam mit der radialen und azimuthalen Modenordnung in der Matrix “res” gespeichert, diese wiederum speichert das Programm in ein zuvor erstelltes .dat-File

```

PP_plus=10*log10(sum(P_plus')/(10^(-12)));
% PP_plus=20*log10(mean(P_plus));

figure('Name',['Radialmodenanalyse '],'NumberTitle','off');
bar(xachse,PP_plus)

hold on
PP_minus=10*log10(sum(P_minus')/(10^(-12)));

bar(xachse, PP_minus, 'g')

axis([-m_max m_max P_min_disp P_max_disp])

if inv == 1
    axis([-m_max m_max 100 180])
end

```



```

PP_plus_tot=10*log10(sum(sum(P_plus'))/((10^(-12))));
PP_minus_tot=10*log10(sum(sum(P_minus'))/((10^(-12))));

grid
title(['Schalleistung von P_m^+ und P_m^- von TP ' TPNr ' bei f = ' int2str(gesuchte_f) ' Hz']);
str(1)={'azimutale Ordnung m'};
str(2)={' '};
str(3)={'\Sigma P_m^+: ' num2str(PP_plus_tot,4) ' dB; \Sigma P_m^-: ' num2str(PP_minus_tot,4) ' dB'};
xlabel(str)

ylabel('Schalleistung P_m [dB]')
legend(['P_m^+' ],['P_m^-'])
hold off

```

- Um die Schalleistung darstellen zu können, wird zuerst die transponierte Matrix “P_plus” bzw. “P_minus” summiert, das heißt für jede azimutale Ordnung werden die Werte aller radialer Moden zusammengezählt und dann zu einem Zeilenvektor zusammengesetzt, und diese Summenwerte dann logarithmiert
- Für jede azimutale Modenordnung wird also die gesamte Schalleistung in und entgegen der Strömungsrichtung grafisch dargestellt
- Des Weiteren wird ein Summenwert über alle radialen und alle azimutalen Modenordnungen ermittelt (“PP_plus_tot” und “PP_minus_tot”) und im Fenster ebenfalls angezeigt (ΣP_m^+ und ΣP_m^-)

```

for i=1:m_summe
    m=i-31;
    if imag(k_mn_plus(i,1))~= 0 || imag(k_mn_minus(i,1))~= 0 %|| m==0
        PP_plus(i)=0;
        PP_minus(i)=0;
    end
end

disp(['maximale radial Ordnung = ' int2str(n_max)]);

if erg_tiff_speichern == 1
    saveas(gcf, ([loadpath vorname 'RMA_TP_' TPNr '_f_' num2str(gesuchte_f)]), 'tiff')
    % saveas(gcf, ([loadpath '\' vorname 'RMA_TP_' TPNr '_f_' num2str(gesuchte_f)]), 'png')
    saveas(gcf, ([loadpath vorname 'RMA_TP_' TPNr '_f_' num2str(gesuchte_f)]), 'fig')
end

h=1;
for i=1:length(PP_plus);
    result(1,i)=i-(m_max+1);
    result(2,i)=PP_plus(i);
    result(3,i)=PP_minus(i);
    % PP_plus_neu(h+1)=PP_plus(i);
end

```



```
savefile_txt = ([loadpath vorname 'PP_plus_RMA_TP_' TPNr '_f_' num2str(gesuchte_f), '.txt']);
result=result';
save (savefile_txt, '-ascii','result');
clear W2;
```

- Ist eine Mode nicht ausbreitungsfähig, so wird ihre Schalleistung Null gesetzt
- Das erstellte Bild, das die Schalleistungen enthält, wird sowohl als tiff-Datei als auch als Matlab-Figure im Arbeitsverzeichnis gespeichert, bevor die Schalleistungen gemeinsam mit der azimuthalen Modenordnung in die Matrix “result” geschrieben und ebenfalls abgespeichert werden

Matlab-File zur Berechnung der Kanal-Eigenwerte:

```
%Berechnen der Eigenwerte zur Auswertung der Akustikmessungen
z=0:0.001:100;
eta=0.659574; %Nabenverhältnis r_hub/r_tip; r_tip=R;

Mode_max=2;
%Erstellen der Besselfunktion erster Art Jm und der Besselfunktion zweiter
%Art Ym (oder auch Neumannfunktion genannt)

for m=0:Mode_max; % m ist die Umfangsmodenordnung und auch die Ordnung der Besselfunktionen;

    Jm(m+1,:)=besselj(m,z); Jm_eta(m+1,:)=besselj(m,eta.*z);
    Ym(m+1,:)=bessely(m,z); Ym_eta(m+1,:)=bessely(m,eta.*z);

end

%Bestimmung der Ableitungen
d_Jm=gradient(Jm); d_Jm_eta=gradient(Jm_eta);
d_Ym=gradient(Ym); d_Ym_eta=gradient(Ym_eta);

for m=0:Mode_max
    %Aus der analytischen Berechnung folgt, daß die
    %Koeffizientendeterminante verschwinden muß->charakteristische
    %Bestimmungsgleichung
    delta=d_Jm_eta(m+1,:).*d_Ym(m+1,:)-d_Jm(m+1,:).*d_Ym_eta(m+1,:);

    %Nullstellensuche; in einer Zeile stehen die Werte für eine
    %Modenordnung; z.B. erste Zeile sind die Werte für die 0te Mode (m=0);
    k=0;
    for zaehler=1:length(z)-1
        if m==0 & k==0
            k=k+1;
            Sigma(m+1,k)=0;
            Qmn(m+1,k)=0;
            eigenw(k,1,m+1)=m;
            eigenw(k,2,m+1)=k-1;
            eigenw(k,3,m+1)=Sigma(m+1,k);
            eigenw(k,4,m+1)=Qmn(m+1,k);
        end
    end
end
```

```
if delta(zaeehler+1)<0 & delta(zaeehler)>0
    k=k+1;
    Sigma(m+1,k)=z(zaeehler);
    Qmn(m+1,k)=-d_Üm_eta(m+1,zaehler)./d_Ym_eta(m+1,zaehler);
    eigenw(k,1,m+1)=m;
    eigenw(k,2,m+1)=k-1;
    eigenw(k,3,m+1)=Sigma(m+1,k);
    eigenw(k,4,m+1)=Qmn(m+1,k);
end
if delta(zaeehler+1)>0 & delta(zaeehler)<0
    k=k+1;
    Sigma(m+1,k)=z(zaeehler);
    Qmn(m+1,k)=-d_Üm_eta(m+1,zaehler)./d_Ym_eta(m+1,zaehler);
    eigenw(k,1,m+1)=m;
    eigenw(k,2,m+1)=k-1;
    eigenw(k,3,m+1)=Sigma(m+1,k);
    eigenw(k,4,m+1)=Qmn(m+1,k);
end
```

```
    %In die richtige Form für die Auswertung bringen
    %3d Matrix mit k-Zeilen und 4 Spalten; die Spalten sind die Werte für m,k,
    %Sigma und Qmn-->das sind die eigenw Matrizen
end
clear delta
end

save 'eigenw.dat' eigenw %speichert die Variable eigenw als eigenw.dat ab wie sie für die Akustikauswertung gebraucht wird
```