



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMOBILNÍHO A DOPRAVNÍHO INŽENÝRSTVÍ

INSTITUTE OF AUTOMOTIVE ENGINEERING

DIAGNOSTICKÝ NÁSTROJ PŘEVODOVÝCH ÚSTROJÍ

TRANSMISSION DIAGNOSTIC TOOL

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Jakub Škvarka

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Aleš Prokop, Ph.D.

BRNO 2024

Zadání diplomové práce

Ústav:	Ústav automobilního a dopravního inženýrství
Student:	Bc. Jakub Škvarka
Studijní program:	Automobilní a dopravní inženýrství
Studijní obor:	bez specializace
Vedoucí práce:	Ing. Aleš Prokop, Ph.D.
Akademický rok:	2023/24

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Diagnostický nástroj převodových ústrojí

Stručná charakteristika problematiky úkolu:

Práce je zaměřena na vytvoření diagnostického nástroje pro analýzu vibro–akustického projevu převodových ústrojí v prostředí Matlab/Python a následnou aplikaci klíčových funkcionalit na poskytnutá data z technického experimentu.

Cíle diplomové práce:

Vyhotovení rešerše v oblasti vibro–akustického projevu převodových ústrojí a nástrojů diagnostiky.
Vyhotovení diagnostického nástroje v prostředí Matlab/Python.
Provedení analýzy vybraných naměřených dat.
Zhodnocení dosažených výsledků a vyvození závěrů.

Seznam doporučené literatury:

NAUNHEIMER, Harald, Bernd BERTSCHE, Joachim RYBORZ, Wolfgang NOVAK. Automotive Transmissions. Springer Verlag, 1999, ISBN 978-3-642-16213-8.

SMITH, Derek J. Gear noise and vibration. 2nd edition. New York: Marcel Dekker, 2003, 297 s. ISBN 824741293.

SMETANA, Ctirad. Hluk a vibrace. Praha: Sdělovací technika, 1998, 250 s. ISBN 8090193625.

TŮMA, Jiří. Vehicle gearbox noise and vibration: measurement, signal analysis, signal processing and noise reduction measures. Chichester: John Wiley, 2014, 243 s. ISBN 9781118359419.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2023/24

V Brně, dne

L. S.

prof. Ing. Josef Štětina, Ph.D.
ředitel ústavu

doc. Ing. Jiří Hlinka, Ph.D.
děkan fakulty

ABSTRAKT

Táto diplomová práca sa zameriava na vývoj nástroja pre diagnostiku prevodových ústrojenstiev v programovacom jazyku Python. Cieľom práce je vytvoriť flexibilné softvérové riešenie umožňujúce získavanie a analýzu dát hlavne z prevodoviek pomocou analyzátora LAN-XI, ktoré je prispôsobiteľné individuálnym potrebám užívateľa a umožňuje automatizáciu procesov. Vyvinutý nástroj plnohodnotne nahrádza konvenčný softvér, pričom poskytuje možnosti automatizácie a prispôsobenia procesu získavania dát. V rámci práce je tiež prezentovaný konkrétny príklad využitia nástroja na reálnych dátach z technického experimentu, ktorý demonštruje jeho funkčnosť a praktické využitie.

KLÍČOVĚ SLOVÁ

signál, Fourierova transformácia, analyzátor, Python, vývoj softvéru, zber dát

ABSTRACT

This master's thesis focuses on the development of a tool for diagnosing transmission systems in programming language Python. The aim of this work is to create a flexible software solution that enables data acquisition and analysis primarily from gearboxes using the LAN-XI analyser, which is adaptable to the individual needs of the user and allows for process automation. The developed tool fully replaces conventional software, providing opportunities for automation and customization of the data acquisition process. The thesis also presents a specific example of the tool's application on real data from a technical experiment, demonstrating its functionality and practical use.

KEYWORDS

signal, Fourier transformation, analyser, Python, software development, data acquisition

BIBLIOGRAFICKÁ CITÁCIA

ŠKVARKA, Jakub. Diagnostický nástroj převodových ústrojí. Online, Diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automobilního a dopravního inženýrství, 2024. Vedoucí práce Aleš Prokop. Dostupné z <https://www.vut.cz/studenti/zav-prace/detail/156347>.



ČESTNÉ PREHLÁSENIE

Prehlasujem, že táto práca je mojím pôvodným dielom, spracoval som ju samostatne pod vedením Ing. Aleša Prokopa, Ph.D. a s použitím informačných zdrojov uvedených v zozname.

V Brne dňa 24. mája 2024

.....

Jakub Škvarka

POĎAKOVANIE

Ďakujem vedúcemu diplomovej práce Ing. Alešovi Prokopovi, Ph.D. za ústretovosť a cenné pripomienky pri vedení práce a taktiež za poskytnuté vybavenie potrebného k jej vypracovaniu. V neposlednom rade ďakujem rodičom a partnerke Sofii za podporu počas celého štúdia.

OBSAH

Úvod.....	11
1 Vibrácie a hluk v prevodových ústrojenstvách.....	12
1.1 Vibrácie budené ozubeným súkolím.....	13
1.2 Vibrácie budené ložiskami.....	13
1.2.1 Rozmery ložiska	13
1.3 Veličiny popisujúce meraný signál.....	15
1.3.1 Popis signálu vibrácií	15
2 Monitoring vibroakustického prejavu.....	17
2.1 Veličiny pri meraní vibrácií.....	17
2.1.1 Kontaktné meranie.....	18
2.1.2 Bezkontaktné meranie	20
2.2 Veličiny pri meraní hluku	21
2.2.1 Snímače zvuku.....	22
2.3 Meranie otáčok	23
3 Klasifikácia signálov a ich analýza.....	24
3.1 Elementárne deterministické signály	25
3.2 Popis signálu	25
3.3 Merací reťazec	25
3.3.1 Analóg- Digitálny prevodník.....	26
3.3.2 Popis vibroakustického signálu	27
3.4 Fourierova transformácia	28
3.4.1 Fourierove rady	28
3.4.2 Fourierova transformácia všeobecného signálu	28
3.4.3 Diskrétna Fourierova transformácia	29
3.4.4 Rýchla Fourierova transformácia (FFT).....	31
3.4.5 Krátkodobá Fourierova transformácia (STFT).....	32
3.4.6 Voľba časového okna	35
4 Tvorba diagnostického nástroja.....	38
4.1 Programovací jazyk a knižnice	39
4.2 Stanovenie požiadaviek	40
4.3 Logika kódu	40
4.4 Spracovanie dát.....	41
4.4.1 Načítanie dát.....	41
4.4.2 Časová doména.....	43
4.4.3 Frekvenčná doména	44
4.4.4 Spektrogram	45
4.5 Diagnostika	48
4.5.1 Otáčky.....	48
4.5.2 Výpočet teoretických frekvencií.....	50
4.5.3 Načítanie dát z experimentu	50
4.5.4 Filtrovanie frekvencií	50
4.5.5 Grafické zobrazenie	51
4.6 Online zber dát.....	51

4.6.1	Komunikácia s analyzátorom LAN-XI.....	52
4.6.2	Spustenie nahrávania.....	54
4.6.3	Ukladanie dát	56
4.6.4	Online zobrazenie dát.....	59
4.7	Užívateľské rozhranie	59
4.7.1	Návrh.....	60
4.7.2	Finálna verzia.....	62
4.8	Validácia meraných dát	63
4.9	Možnosti vylepšenia.....	64
5	Analýza dát z prevodovky	65
5.1	Analyzované zdroje vibrácií.....	65
5.2	Definícia dát z merania.....	66
5.3	Postup pri diagnostike	67
	Záver	76
	Použité informačné zdroje.....	77
	Zoznam použitých skratiek a symbolov	80

ÚVOD

Prevodové ústrojenstvá zohrávajú neodmysliteľnú súčasť v mnohých technických aplikáciách zabezpečujúce prenos a zmenu veľkosti točivého momentu. Pre zaručenie ich bezchybného chodu sa vyžaduje spoľahlivá diagnostika a analýza vibroakustického prejavu. Potreba diagnostiky je ešte dôležitejšia vzhľadom na narastajúce požiadavky na výkon, spoľahlivosť a efektívnosť prevodoviek, ako aj kvôli prísnejším legislatívnym a hygienickým normám.

Pomocou systémov na zber dát označované aj ako DAQ, sú získavané dáta pre následnú analýzu a diagnostiku. Slúžia ako most medzi fyzickým svetom a digitálnou doménou, umožňujúce užívateľovi zachytiť, spracovať a následne interpretovať skutočné javy s veľkou presnosťou. Na tento účel je potrebné mať k dispozícii snímače, analyzátor a vhodný softvér na ovládanie a spracovanie týchto dát. Avšak nie každý softvér umožňuje úplné prispôbenie požiadavkám merania. Práve preto je cieľom tejto diplomovej práce, vytvoriť softvér v prostredí Python, ktorý umožní diagnostiku prevodových ústrojenstiev a taktiež prostredníctvom analyzátora LAN-XI, umožní zber dát podľa konkrétnych požiadaviek na nastavenie. Bude náhradou za komerčne dostupný softvér a umožní nastaviť meranie podľa špecifických potrieb užívateľa. Kvôli tomu je potrebné nadviazať spojenie s analyzátorom LAN-XI a pochopiť podstatu prenosu dát tak, aby bolo možné dáta ukladať v potrebnom formáte a následne ich aj analyzovať a graficky zobrazit'.

Pre tento účel je najprv vytvorená rešerš v oblasti vzniku a meraní vibrácií s následným detailným popisom princípu analýzy signálu na základe rýchlej Fourierovej transformácie. Druhá časť práce pojednáva o možnostiach vývoja samotného softvéru pomocou rôznych knižníc v prostredí Python. Je tu detailne popísaný postup práce a taktiež dôvody, výhody a nevýhody jednotlivých knižníc. Táto práca slúži aj ako názorná ukážka postupu práce s vytvoreným softvérom, pretože je v nej analyzovaný signál z technického experimentu. Sú tu demonštrované jednotlivé funkcie softvéru, pre preukázanie jeho využiteľnosti v praxi.

1 VIBRÁCIE A HLUK V PREVODOVÝCH ÚSTROJENSTVÁCH

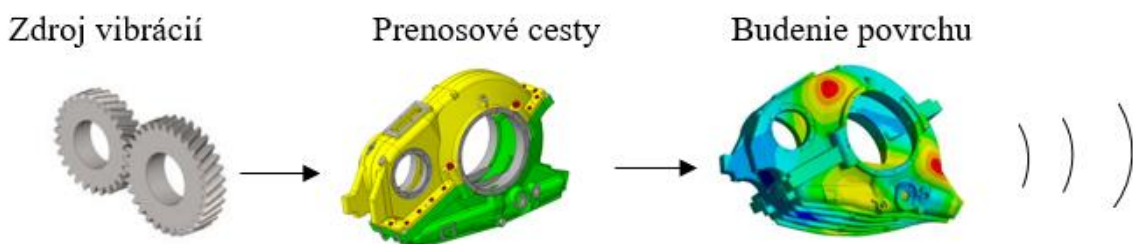
Prevodovky sú dôležitým prvkom v mnohých mechanických systémoch, ktoré slúžia na zmenu točivého momentu. Ich úloha je kľúčová pre riadenie rýchlosti, sily a smeru pohybu, a to nielen v ťažkých priemyselných strojoch a osobných vozidlách ale aj v spotrebných elektronických zariadeniach.

Hlavnými požiadavkami na prevodovky je ich spoľahlivosť a schopnosť nepretržitého chodu. Podstatou prevodoviek je presný a spoľahlivý prenos točivého momentu, čo je základom pre optimálne fungovanie celého systému. Okrem toho musia mať prevodovky vysokú účinnosť, aby minimalizovali straty. S narastajúcimi technologickými nárokmi sa požaduje aj zvyšujúci sa rozsah otáčok a preneseného momentu, aby prevodovky dokázali zvládať širokú škálu pracovných podmienok. Zároveň je požiadavka na čo najnižšie vibrácie počas chodu. Keďže vplyvom dynamických procesov sa časti prevodoviek opotrebovávajú, je potrebné zabezpečiť pravidelný servis a údržbu. Diagnostika je kľúčovým nástrojom na určenie stavu prevodoviek a odhalenie potenciálnych problémov. Umožňuje pomocou analýzy odozvy stanoviť zdroj budiacich silových účinkov iniciovaných v dôsledku zhoršujúceho sa stavu zariadenia.

Vibroakustika je neoddeliteľnou súčasťou diagnostiky prevodových ústrojenstiev. Spája štúdium vibrácií a zvuku a poskytuje dôležité informácie o funkčnosti prevodoviek. Zdrojmi budenia v prevodových ústrojenstvách sú najmä ozubené prevody - ozubená. Aby vznikali vibrácie, je potrebná variácia síl, ktorá generuje vibrácie v komponentoch a tie sú ďalej prenesené do štruktúry a do okolia cez elastické médiá vid' Obr.1.1. Vtedy sa hovorí o odozve na silové účinky. Odozva sa môže šíriť cez štruktúru alebo cez vzduch a touto cestou sa dostane až k prijímateľovi, ktorým je väčšinou samotný užívateľ.

Variácia síl môže byť zapríčinená kombináciou pružných deformácií pri odvalovaní súkolí, malých odchýlok tvaru zubu a kolísaním otáčok pohonu. Tým vzniká relatívna zmena posunutia medzi súkolím a tak aj dynamická odozva v podobe variácie sily. Táto sila môže meniť svoju amplitúdu, smer alebo pôsobisko. [22],[23]

Medzi hlavné príčiny zmeny amplitúdy patria premenlivá tuhosť záberu a rázy zubov pri vstupe do záberu. Sú spôsobené deformáciou zubov a nepresnými rozstupmi, alebo aj výrobnými nedokonalosťami. Zmena smeru sily v zábere vzniká kvôli trecím silám. Tieto sily nie sú hlavným dôvodom vibrácií, avšak môžu mať negatívny vplyv na ozubenie, kde môžu spôsobiť opotrebenie boku zuba, tým odchýlku od teoretického tvaru a následné vibrácie. Tieto aspekty nedokonalosti ozubenia vyjadruje chyba prevodu (Transmission Error, TE). Je vyjadrená ako rozdiel medzi skutočnou polohou hnaného kola a jeho teoretickou polohou. [22],[23]



Obr. 1.1 Schéma vibroakustického systému

Ďalším zdrojom vibrácií sú valivé ložiská, ktoré sú súčasne aj dôležitým aspektom v prenosovej ceste hluku. Vibrácie vznikajú odvalovaním valivých členov ložiska po vnútornej a vonkajšej dráhe. Ich frekvencie vibrácií sú ovplyvnené nerovnosťami povrchu alebo nepravidelnosťami funkčných plôch dané výrobou. Výsledné vibrácie sa prejavujú v nízkofrekvenčnom pásme len pri silne opotrebovaných ložiskách, inak v oblasti vysokých frekvencií, kde je energia vibrácií malá. [22],[23]

1.1 VIBRÁCIE BUDENÉ OZUBENÝM SÚKOLÍM

Ako už bolo spomenuté, v prevodových ústrojenstvách je hlavným zdrojom budenia hlavne ozubenie. Pri diagnostike signálu odozvy na silové budenie sa využívajú frekvencie, ktoré reprezentujú jednotlivé fyzikálne javy. Základom spracovania je frekvenčná doména. V prevodových ústrojenstvách sa vždy nachádzajú tieto frekvenčné zložky a ich násobky:

Otáčková frekvencia

$$f_{ot} = k \cdot \frac{n}{60}, \quad k=1,2,3... \quad (1)$$

kde n sú otáčky pre príslušný hriadeľ.

Zubová frekvencia

$$f_z = z \cdot f_{ot} \quad (2)$$

kde z je počet zubov kolesa.

Postranné pásma zubovej frekvencie

$$f_{pp} = f_z \pm l \cdot f_{ot}, \quad l=1,2,3... \quad (3)$$

1.2 VIBRÁCIE BUDENÉ LOŽISKAMI

Idealizované valivé ložiská s pracovnými povrchmi bez nerovností a s ideálne presnými geometrickými rozmermi nie sú zdrojom vibrácií a hluku. Až nedokonalosti ich geometrického tvaru a poruchy dráh alebo valivých členov budia vibrácie. Vplyvom opotrebenia sa časom na pracovných plochách môžu vyskytovať rôzne druhy porúch ako napríklad pitting, alebo praskliny a lomy. Nárazom valivého člena na tieto lokálne nerovnosti spôsobia silové impulzy, ktoré vybudia rýchle tlmené vysokofrekvenčné kmity mechanickej štruktúry ložiska. Pomocou rozmeru ložiska, počtu valivých členov a otáčok je možné vypočítať frekvenciu opakovania týchto kmitov. [24]

1.2.1 ROZMERY LOŽISKA

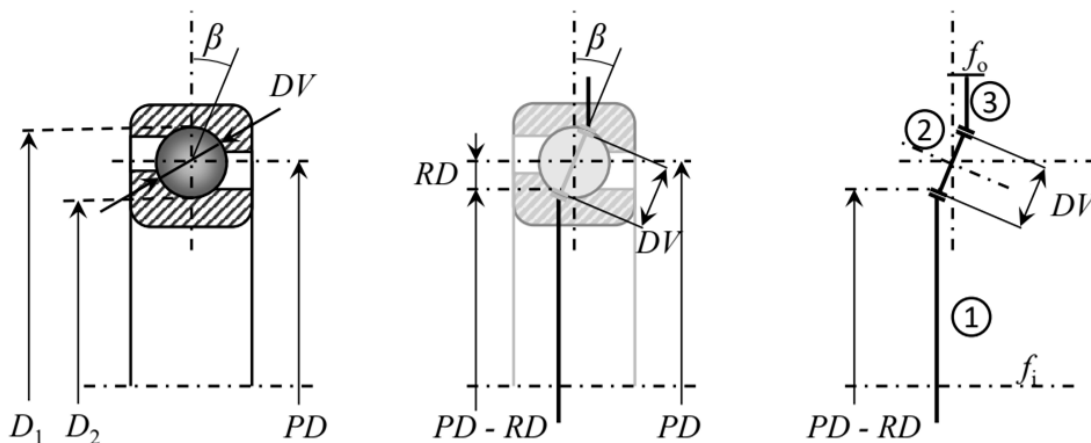
Do výpočtu frekvencií prejavu väd vstupujú nasledujúce rozmery ako je vidieť na Obr.1.2.

- Priemer valivého člena – DV
- Priemer valčeka - RD
- Roztečený priemer – PD
- Priemer vonkajšej dráhy - DI

- Priemer vnútornej dráhy – D_2
- Stykový uhol - β
- Počet valivých členov – n

Vo výpočtoch vystupujú nasledujúce frekvencie otáčania

- Frekvencia otáčania vnútorného krúžku vzhľadom na okolie - f_i
- Frekvencia otáčania vonkajšieho krúžku vzhľadom na okolie - f_o
- Frekvencia otáčania klietky vzhľadom na okolie – f_c
- Frekvencia otáčania valivého člena vzhľadom na klietku – f_r



Obr. 1.2 Rozmery ložiska pre výpočet frekvencií závad [24]

Pomocou týchto frekvencií je možné ďalej dopočítať frekvencie prejavu väd. Do výpočtu ešte vstupuje pomer priemeru valivého člena k roztečnému priemeru ako x .

$$x = \frac{RD}{PD} = \frac{DV}{PD} \cos \beta \quad (4)$$

$$f_{BPFO} = \frac{n}{2} (f_i - f_o) (1 - x) \quad (5)$$

kde f_{BPFO} je frekvencia prejavu vady na vonkajšej dráhe a n je počet valivých členov.

$$f_{BPFI} = \frac{n}{2} (f_i - f_o) (+x) \quad (6)$$

kde f_{BPFI} je frekvencia prejavu vady na vnútornej dráhe.

$$f_{BSF} = \frac{1}{2} (f_i - f_o) \left(\frac{1 - x^2}{x} \cos \beta \right) \quad (7)$$

kde f_{BSF} je frekvencia prejavu vady valivého člena.

$$f_{FTF} = \frac{1}{2} [f_o(1 + x) + f_i(1 - x)] \quad (8)$$

kde f_{FTF} je frekvencia otáčania klietky.

Na to aby bolo možné spracovať tieto frekvencie a diagnostikovať konkrétnu prevodovku, je nutné mať k dispozícii signál popisujúci toto chovanie. Ten je získaný pomocou technického experimentu a zberu dát. Budenie aj odozva môže mať charakter rôznych typov signálov. Je potrebné ich vhodne popísať pomocou príslušných veličín ako je vidieť v nasledujúcich kapitolách.

1.3 VELIČINY POPISUJÚCE MERANÝ SIGNÁL

Vibrácie a hluk sú momentálne v automobilovom priemysle vnímané ako stále väčší problém, či už z hľadiska komfortu alebo zvukového znečistenia prostredia. Hluk vo vozidle dokáže zmeniť jeho celkový dojem a zážitok z jazdy. Preto je potrebné vedieť tieto javy merať a vyhodnocovať pomocou vhodných prostriedkov. [14]

Základným problémom diagnostiky je, že sa analyzuje až odozva na pôsobiace príčiny. V prípade vibračnej diagnostiky sú touto odozvou vibrácie, ktorých charakter sa líši od pôsobiacej sily.

1.3.1 POPIS SIGNÁLU VIBRÁCIÍ

Najjednoduchším prípadom budenia vibrácií je periodická harmonická sila. V realite sa väčšinou vyskytujú sily, ktoré sa dajú vyjadriť súčtom harmonických síl. Preto sa môžu popísať vlastnosti periodickej sily a jej vplyv na vibračnú odozvu pomocou harmonickej sily a odozvy.

$$f(t) = F \cdot \sin(\omega t + \varphi_F) \quad (9)$$

kde F je amplitúda budiacej sily, ω je kruhová frekvencia budiacej sily, t je čas a φ je počiatočný fázový posun budiacej sily. Výchylku kmitania je možné vyjadriť ako

$$x(t) = X \cdot \sin(\omega t + \varphi_F - \varphi) \quad (10)$$

kde X je amplitúda vynúteného kmitania a φ je fázový posun výchylky za budiacou silou.

V diagnostike sa dá často stretnúť aj s impulznou silou, ktorá spôsobí vychýlenie telesa z rovnovážnej polohy a následné kmitanie v jednej alebo viacerých vlastných frekvenciách, napríklad pri poruchách v ložisku. Ďalej môže pôsobiť na teleso sila s náhodným charakterom, pri ktorej môže byť vybudená tak isto vplyvom skokovej zmeny sily voľné kmitanie na vlastnej frekvencii. Špeciálnym typom kmitania je samobudené kmitanie, kedy dochádza ku kladnej spätnej väzbe medzi kmitaním objektu na vlastnej frekvencii a aerodynamickými silami.

Pre popis vlastností signálu je potrebné definovať niekoľko veličín. V technickej praxi sa počet cyklov za sekundu používa frekvencia vyjadrená v hertzoch ako

$$f = \frac{\omega}{2\pi} \text{ [Hz]} \quad (11)$$

Jej prevrátenou hodnotou je perióda

$$T = \frac{1}{f} = \frac{2\pi}{\omega} \quad [\text{s}] \quad (12)$$

Pri technickej diagnostike dosahujú okamžité hodnoty meraných veličín rôzne hodnoty rýchlo sa meniace sa v čase, preto boli zavedené veličiny, ktoré ich lepšie charakterizujú. Medzi také patrí efektívna hodnota (RMS) definovaná ako:

$$x_{ef} = \sqrt{\frac{1}{T} \int_0^T x^2(t) dt}, \quad (13)$$

kde x je okamžitá hodnota v čase t a T je perióda, pre ktorú sa určuje efektívna hodnota. Ďalej sa používa stredná hodnota definovaná nasledovne

$$x_{str} = \frac{1}{T} \int_0^T |x(t)| dt \quad (14)$$

Prevládajúci tvar časového signálu sa dá usúdiť pomocou činiteľa výkmitu (crest factor) ako:

$$CF = \frac{x_a}{x_{ef}}, \quad (15)$$

kde x_a je vrcholová hodnota – maximálna amplitúda signálu. Pre nižšie hodnoty CF (~3) prevláda sínusový charakter signálu, pri väčšej hodnote prevláda impulzný charakter. Je to taktiež jedna z metód pre kontrolu stavu ložísk.

Ďalším parametrom charakterizujúci signál je špicatosť (kurtosis). Je to štatistický parameter, ktorý hovorí ako veľmi je signál koncentrovaný okolo priemeru v porovnaní s jeho šírkou. Vyššia hodnota naznačuje, že signál má viac extrémnych hodnôt. Je definovaný nasledovne. [27]

$$K = \frac{\frac{1}{N} \sum x_i^4}{\left(\frac{1}{N} \sum x_i^2\right)^2} \quad (16)$$

2 MONITORING VIBROAKUSTICKÉHO PREJAVU

Mechanické vibrácie štruktúr budené rôznym charakterom, majú podiel na stratách energie. Zvolením vhodného postupu merania a následného spracovania signálu je možné získať potrebné dáta a informácie o tom, ako sa vibrácie prejavujú. Tým sa zvolí správne riešenie pri ich eliminácii. Meranie časového priebehu signálu môže prebiehať dvomi spôsobmi: záznamom alebo priamym vyhodnotením v reálnom čase. Záznam slúži na uchovanie týchto charakteristík signálu na neskoršie vyhodnotenie či už ako jednoduchý časový záznam, FFT spektrum alebo multispektrum označované aj ako spektrogram. Problematikou šírenia vibrácií a následnou čiastočnou premenou na akustickú energiu sa zaoberá oblasť označovaná skratkou NVH (Noise Vibration and Harshness).

2.1 VELIČINY PRI MERANÍ VIBRÁCIÍ

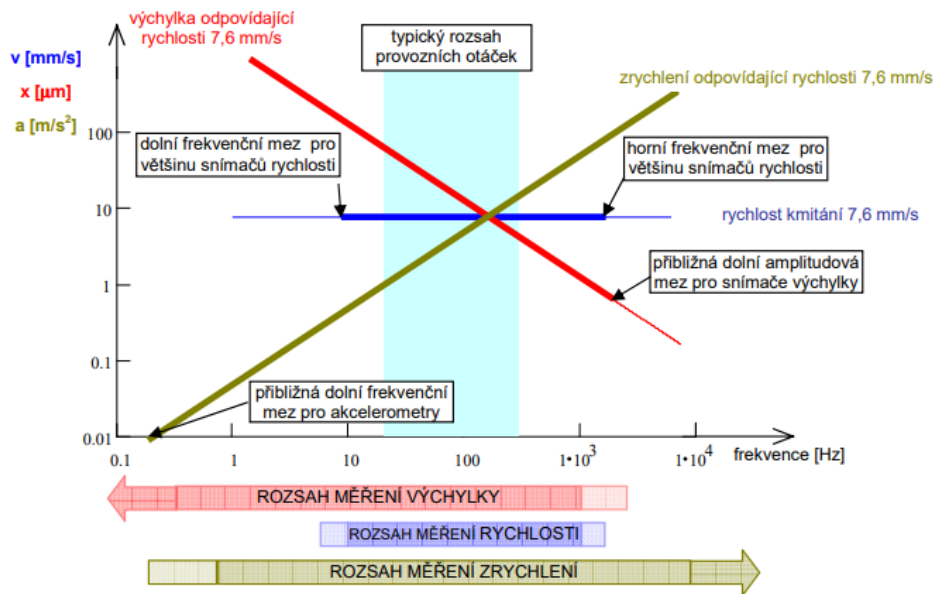
V spojitosti s kmitaním mechanických systémov sa používajú rôzne snímače na meranie pohybu a sily. Meranie sily je realizované pomocou tenzometrov. Fungujú na princípe elektrických odporov citlivých na deformáciu mechanickým napätím. Taktiež sa používajú piezoelektrické snímače. Ich princíp fungovania pozostáva z piezoelektrického javu, ktorý je vyvolaný mechanickým napätím. Takýto snímač je v podstate zdrojom elektrického náboja, ten však treba transformovať na elektrické napätie a zosilniť. Vo všeobecnosti snímače pohybu alebo posunutia fungujú na princípe, kde pohyb súvisí so zmenou elektrického odporu, indukčnosti alebo kapacity. [1]

Pri meraní pohybu existuje viacero veličín, v ktorých sa bude zaznamenávať. Konkrétne ide o meranie dráhy, rýchlosti alebo zrýchlenia. Pomocou derivácii alebo integrácii stačí merať jednu z veličín a ostatné je možné vypočítať vid' *Tabuľka 1*. Rýchlosť je vždy oneskorená s fázovým posunom 90° za výchylkou a zrýchlenie o 90° za rýchlosťou. [1]

Voľba meranej veličiny ma veľký vplyv na presnosť merania. Pri meraní je nutné uvažovať aj nepriaznivé vplyvy pôsobiace na presnosť merania. Je vhodné zvoliť takú meraciu veličinu, aby dávala dostatočný odstup signálu od šumu. Na Obr.2.1 je vidieť, že pri meraní vyšších frekvencií ako napríklad zubovej frekvencie je vhodné použiť snímač zrýchlenia. Naopak pri nižších frekvenciách ako napríklad otáčková frekvencia, je vhodné vyhodnocovať výchylku keďže amplitúda je vtedy vyššia. Medzi tým leží rozsah vyhodnocovania rýchlosti. [1]

Tabuľka 1

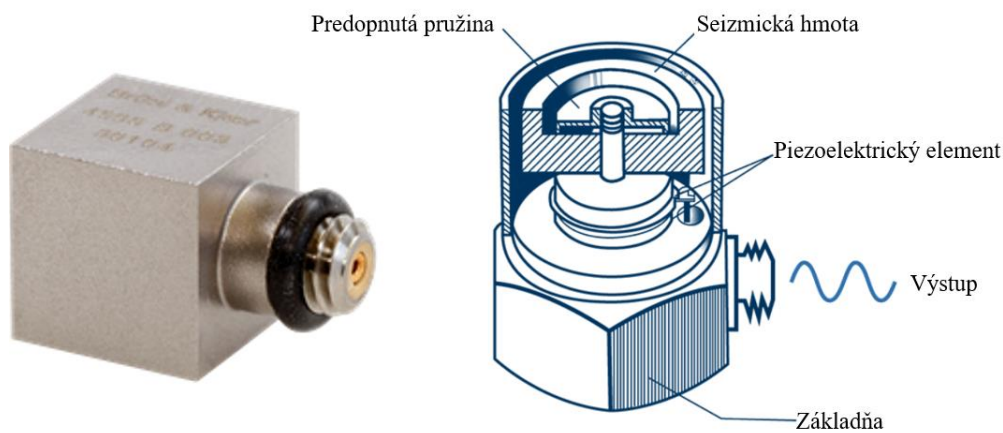
Dráha	$s(t)$	$s(t) = \int v(t) dt$
Rýchlosť	$v(t) = \frac{ds(t)}{dt}$	$v(t) = \int a(t) dt$
Zrýchlenie	$a(t) = \frac{dv(t)}{dt}$	$a(t)$



Obr. 2.1 Obmedzenia pri meraní [1]

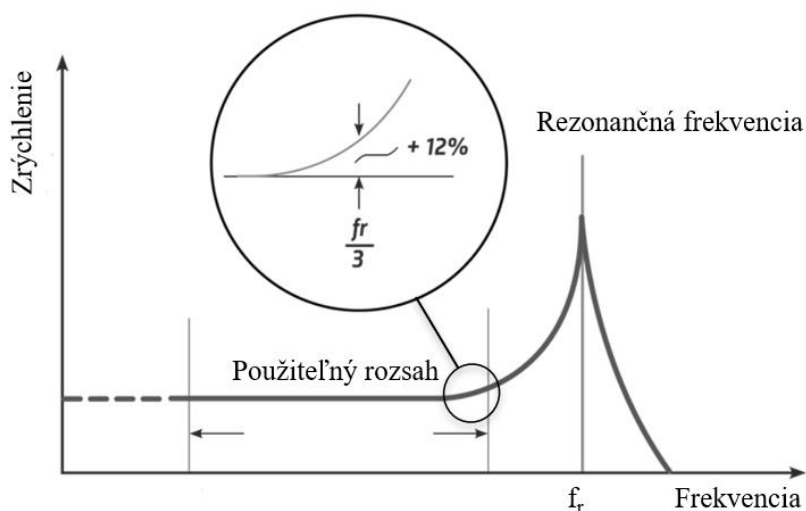
2.1.1 KONTAKTNÉ MERANIE

Pre pohyb vo forme vibrácii sa najviac používa snímač zrýchlenia – akcelerometer vid' Obr. 2.2. Je najuniverzálnejší a dokáže pracovať v najširšom frekvenčnom pásme merania. Funguje taktiež na princípe piezoelektrického javu. Preto nie je potrebné dodatočné napájanie. Pri pohybe plášťa akcelerometra oproti seizmickej hmote vzniká v piezoelektrickom člene mechanické napätie a následne indukuje elektrický náboj úmerný zotrvačnej sile. Rozsah merania frekvencií piezoelektrických akcelerometrov sa pohybuje v rozsahu medzi 0,1 Hz až 20 kHz v závislosti od spôsobu uchytenia, veľkosti a hmotnosti samotného akcelerometru. Existujú však aj snímače rýchlosti a snímače polohy. Pre tieto účely sa však najviac hodia akcelerometre, preto bude ďalej pojednávané len o týchto snímačoch.[1],[3]



Obr. 2.2 Snímač zrýchlenia [2], [3]

Po správnom zvolení meranej veličiny, v tomto prípade zrýchlenie, je taktiež potrebné zohľadniť niekoľko ďalších aspektov. Citlivosť akcelerometra sa dá vyčítať z tzv. kalibračnej krivky, kde je vidieť tak isto aj jeho frekvenčný rozsah. Ak je frekvencia meraného kmitania pod dolnou frekvenciou medze snímača, tak sa zotrvačná hmota pohybuje spoločne s meranou štruktúrou a nevytvára elektrický náboj. Ak je nad hornou medzou, tak je potrebné počítať so skreslenou amplitúdou v dôsledku rezonancie. Keďže aj snímač je mechanický systém, má rezonančnú frekvenciu a je potrebné aby bolo meranie vyhotovené rádovo do 30% hodnoty vlastnej frekvencie snímača, kedy merané hodnoty budú s maximálnou chybou 12%, ako je vidieť na Obr. 2.3.[1] [3] [27]



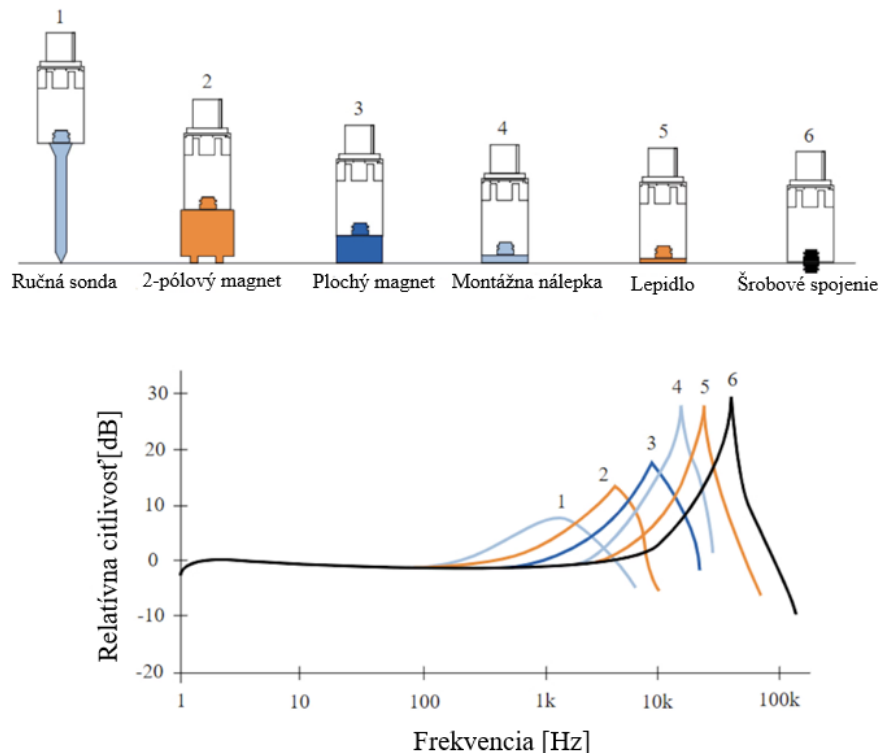
Obr. 2.3 Frekvenčný rozsah snímača [3]

Všeobecne sa dá povedať, že väčšie snímače majú lepšiu citlivosť a dokážu merať nižšie frekvencie. Majú však vyššiu hmotnosť a nižšiu rezonančnú frekvenciu.

Obmedzenie použiteľnosti snímača ovplyvňuje aj montáž akcelerometru. Každé pripevnenie snímača pridáva ďalšiu hmotu a pružinu do systému s jedným stupňom voľnosti. Nevhodným pripevnením sa môže úplne znehodnotiť výsledné meranie. Pridržanie rukou je najhorší možný variant, kedy sa použiteľné frekvencie merania pohybujú rádovo len v stovkách a slúži len pre orientačné meranie. Prednostne sa používa montáž pomocou včelieho vosku, lepiacej pásky, lepidla alebo šróbového spojenia – to vykazuje najvyššiu vlastnú frekvenciu, takže umožňuje najvyšší rozsah merania vid' Obr. 2.4. [29]

Plocha pod základňou snímača musí byť čistá a rovná, snímač má dosadať celou plochou základni a diera pre šrob musí byť kolmá k meranému povrchu. Ak je použité lepidlo je potrebné zvoliť také, ktoré nemá mäkkú doplnkovú zložku, pretože by znížilo frekvenčný rozsah. Dosadacia plocha musí byť taktiež rovná čistá a navyše aj zdrsnená. Takýmto spôsobom sa dá priblížiť frekvenčnému rozsahu šróbového spojenia. Pripevnenie včelím voskom sa používa hlavne pri malých snímačoch v laboratórnych meraniach. Mierne znižuje použiteľný frekvenčný rozsah a teplota pri meraní nesmie presiahnuť 40°C. Pri čistom povrchu a správnom osadení je možné aj s včelím voskom dosiahnuť vysoké hodnoty rezonančnej frekvencie. Rozsah zrýchlenia môže byť až do 100 m/s². Avšak pri úspešnom meraní, je dôležité taktiež vybrať správne miesto merania, nie len typ upevnenia. Nesprávne zvolené miesto môže pri diagnostike skresliť výsledný signál a tým zhoršiť alebo úplne

zamedziť následnému spracovaniu signálu. Keďže sa jedná o kontaktné meranie, je potrebné dodržať maximálnu zmenu hmotnosti štruktúry tým, že osadíme snímač.



Obr. 2.4 Rezonančná frekvencia závislá od druhu montáže snímača [29]

Všeobecne platí, že hmotnosť snímača by nemala presahovať 10% hmotnosti meraného systému, pretože podľa rovnice pre vlastnú frekvenciu sústavy s jedným stupňom voľnosti v tvare

$$\Omega = \sqrt{\frac{k_s}{m}}, \quad (17)$$

kde Ω je vlastná frekvencia, k_s je tuhosť sústavy a m je jej hmotnosť, vstupuje práve hmotnosť a tá mení výslednú vlastnú frekvenciu sústavy. Tak isto by nemal byť snímač umiestnený v mieste s rezonanciou. [3], [27]

2.1.2 BEZKONTAKTNÉ MERANIE

Snímanie zrýchlenia sa dá merať aj bezkontaktné na základe merania výchylky alebo rýchlosti a následného derivovania. Existuje niekoľko typov snímačov napríklad indukčnosť, indukčný alebo kapacitné snímače vzdialenosti. Je s nimi možné merať napríklad výchylky hriadele alebo natočenie kolies pri systéme ABS. Ich výhodou je, že neovplyvňujú celkovú hmotnosť systému.

Pri bezkontaktnom meraní vibrácií je možné použiť aj laserové meranie. Laserový skenovací vibrometer vid' Obr. 2.5, je nákladné zariadenie ale výhodou je, že sa dajú stanoviť vibrácie aj na mieste, ku ktorému sa nedá fyzicky bližšie dostať. Funguje na princípe merania rýchlosti pomocou Dopplerového javu, kedy sa sleduje zmena frekvencie vysielaného signálu oproti prijímanému v dôsledku pohybu sledovaného objektu. Výhodou je, že objekt nie je

ovplyvnený hmotnosťou snímača, je možné merať z rôznych vzdialeností. Taktiež majú veľký frekvenčný rozsah od 0 do 30 MHz a rozsah meranej rýchlosti od 10^{-7} do $30 \text{ m}\cdot\text{s}^{-1}$. [10]



Obr. 2.5 Laserový vibrometer [30]

2.2 VELIČINY PRI MERANÍ HLUKU

Ako už bolo spomenuté, mechanický hluk sa šíri naprieč elastickým médiom. Môže byť spôsobený napríklad vybudením povrchu prevodovej skrine. Mechanická energia je odovzdaná plynnému prostrediu, ktorým sa šíri do okolia pomocou vlnoplôch. Jedná sa o periodicky opakujúce sa miesto v priestore v ktorom sú zhustené alebo zriedené molekuly, teda dochádza k zmene tlaku oproti barometrickému tlaku. Na barometrickom tlaku s hodnotou približne $100\,000 \text{ Pa}$ je namodulovaný tzv. akustický tlak, ktorý má oveľa nižšiu hodnotu vid' Obr.2.6. Zdravé ľudské ucho začína vnímať hodnoty od $2 \cdot 10^{-5} \text{ Pa}$ pri frekvencii 1000 Hz.

Na Obr.2.6. tiež vidno schematicky znázornené šírenie zvuku v prostredí. V závislosti od prostredia sa mení rýchlosť jeho šírenia a tým aj **vlnová dĺžka** λ . Vlnová dĺžka je definovaná :

$$\lambda = \frac{c}{f} \quad (18)$$

kde c je rýchlosť šírenia sa zvuku v danom prostredí, pre vzduch je to približne $330 \text{ m}\cdot\text{s}^{-1}$. [15]

Keďže **akustický tlak** nadobúda hodnoty v širokom rozsahu, je efektívne vyjadrovať ho v logaritmickej mierke pomocou **hladiny akustického tlaku** L_p vyjadrenej v decibeloch (dB). Dá sa vyjadriť ako :

$$L_p = 10 \cdot \log \frac{p^2}{p_0^2} = 20 \cdot \log \frac{p}{p_0}, \quad (19)$$

kde p_0 je referenčný tlak a jeho hodnota je $2 \cdot 10^{-5} \text{ Pa}$. Hodnota prahu bolesti je oproti tomu až 100 Pa čo zodpovedá približne 130 dB. Charakterizuje akustickú doménu v danom mieste od zdroja.

V spojení s hladinou akustického tlaku súvisí aj skalárna veličina **akustická intenzita**, ktorá definuje množstvo a smer toku akustickej energie v danom prostredí. **Hladina akustickej intenzity** charakterizuje akustickú doménu v danom mieste a smere. Je daná vzťahom

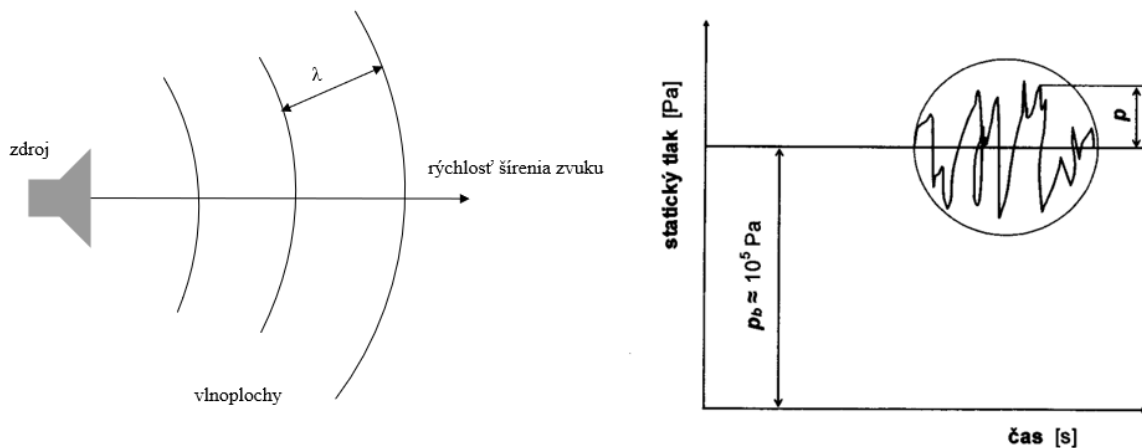
$$L_I = 10 \cdot \log \frac{I}{I_0}, \quad (20)$$

kde I je akustická intenzita v smere šírenia akustických vln a I_0 je referenčná hodnota $10^{-12} \text{ W} \cdot \text{m}^{-2}$. Vyjadruje sa taktiež v jednotkách dB.

Posledná charakteristika je **akustický výkon** zdroja zvuku. Ten vyjadruje aké množstvo energie prejde alebo dopadne na danú priestorovú plochu za jednotku času. **Hladina akustického výkonu** je daná vzťahom

$$L_W = 10 \cdot \log \frac{P}{P_0}, \quad (21)$$

kde P_0 je referenčná hodnota 10^{-12} W . [16]



Obr. 2.6 Šírenie zvuku a priebeh statického tlaku vo vzduchu [15]

2.2.1 SNÍMAČE ZVUKU

Charakteristickou veličinou pri meraní zvuku je akustický tlak. Na jeho stanovenie sa najčastejšie používa kondenzátorový mikrofón vid' Obr.2.7. Zmeny tlaku vyvolajú kmity membrány a tým zmenu kapacity, ktorá sa elektronicky vyhodnocuje. Výhodou tohto mikrofónu je stálosť frekvenčnej charakteristiky v celom počuteľnom rozsahu, pričom citlivosť je nezávislá na veku mikrofónu. Kondenzátorové mikrofóny vyžadujú polarizačné napätie a ich signál je potrebné zosilniť predzosilňovačom, preto sa doň privádza napájacie a polarizačné napätie.[31], [4]

Ľudské ucho vníma zvuk v rozmedzí od 20 Hz do 20 kHz. Avšak fyziologické vnímanie zvuku nie je rovnako citlivé na všetkých frekvenciách a amplitúdach. To znamená, že efektívna hodnota akustického tlaku, ktorá je vnímaná s rovnakou hlasitosťou, nie je pre rôzne frekvencie konštantná.



Obr. 2.7 Kondenzátorový mikrofón Brüel & Kjær [32]

Ľudské ucho je najcitlivejšie na frekvenciu okolo 1 kHz. Bola zavedená jednotka fón, ktorá hodnotí hlasitosť vnímanú ľudským uchom. Aby nameraná efektívna hodnota zodpovedala citlivosti ucha, v praxi sa používajú váhové filtre, ktoré stanovujú korekciu pre každú frekvenciu. Frekvenčná váha typu A zodpovedá hlasitosti na úrovni 40 fónov, teda tak ako by človek vnímal referenčný tón s frekvenciou 1 kHz a hlasitosťou 40 dB. Takto korigovaná hladina akustického tlaku sa označuje $L_{p(A)}$ [dB] alebo označenie typu použitého filtra v jednotke [dBA], [dB(A)]. Filtre B, C a D sa používajú na prepočet vyšších hladín rovnakej hlasitosti a sú určené pre špeciálne účely, ako je letectvo (filter D) alebo stavebná akustika (filter C). V diagnostike sa však váhové filtre nepoužívajú, jedine v prípade potreby zohľadniť vnímanie ľudského ucha daných frekvencií. [4], [31]

2.3 MERANIE OTÁČOK

Pri technickej diagnostike vibroakustického signálu je nutné vedieť presné otáčky meraného stroja. Budiace frekvencie významných zdrojov frekvencií súvisia práve s otáčkovou frekvenciou. Otáčky je možné získať aj priamo z meraného signálu pomocou rýchlej Fourierovej transformácie. Avšak v praxi sa tento spôsob veľmi nevyužíva z dôvodu zlej rozlíšiteľnosti, hlavne pri nízkych otáčkach. Existujú oveľa účinnejšie a presnejšie metódy ako napríklad tacho sonda alebo Hallova sonda.

Tacho sonda vid' Obr.2.8. pracuje na optickom princípe, kedy sa odraz sondou vysielaného signálu odrazí od reflexnej značky na hriadeli a spôsobí skokovú zmenu výstupného napätia. Tieto napät'ové špičky je možné ďalej spracovať. Optická tacho sonda je pomerne jednoduchá na inštaláciu a má ďalšie výhody ako napríklad možnosť merania na väčšiu vzdialenosť, aj cez sklo, alebo cez malý otvor stroja, kedy stačí zameranie lúča cez tento otvor.[11]



Obr. 2.8 Otáčková sonda Brüel & Kjær [11]

3 KLASIFIKÁCIA SIGNÁLOV A ICH ANALÝZA

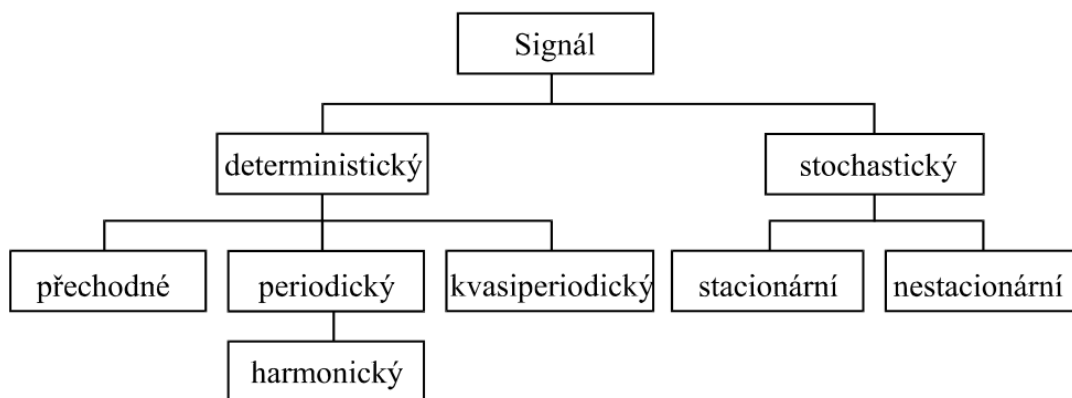
Pojem “signál” sa vo všeobecnosti vzťahuje iba na časové funkcie zobrazujúce priebeh fyzikálnych veličín v čase. Obsahuje iba zmysluplné informácie na rozdiel od šumu. V tejto práci sa však pojem signál bude používať na každú časovú funkciu vrátane náhodného šumu. Tieto informácie majú taktiež výpovednú hodnotu o objekte, ktorý tento signál generuje. Aby bolo možné takéto informácie získať, je potrebné mať k tomu vhodný nástroj na spracovanie signálu. Tvorba diagnostického nástroja je hlavnou náplňou a cieľom tejto práce. [4]

Na Obr. 3.1 je vidno klasifikáciu signálu. Prvé rozdelenie je na deterministické a náhodné. Deterministický signál sa dá predvídať, pretože jeho priebeh je determinovaný napr. funkčným predpisom. Naopak stochastický (náhodný) signál sa nedá predvídať. [4]

Deterministické signály sa ďalej delia na prechodné, ktoré sú časovo obmedzené a časovo neobmedzené, teda trvajú nekonečne dlhú dobu. Sú to signály periodické a kváziperiodické. Periodický signál je zložený z harmonických signálov, ktoré sú sínusovky s násobkom základnej frekvencie. [4]

Stochastické signály sa delia na stacionárne a nestacionárne. Charakter týchto signálov sa môže vplyvom dĺžky časového úseku javiť ako ustálený alebo neustálený. Tým pádom je charakteristika v niektorých prípadoch závislá od doby trvania signálu. [4]

Ďalej sa tieto signály môžu deliť na dve špeciálne skupiny a to úzkopásmové a širokopásmové. Úzkopásmový náhodný signál je zložený hlavne z harmonických zložiek s frekvenciami, ktoré sú blízko k sebe v úzkom pásme okolo nulovej frekvencie. To znamená, že jeho spektrum má iba jeden výrazný vrchol. V časovom priebehu môže tento signál obsahovať oscilácie s takmer konštantnou frekvenciou, ale s náhodnými zmenami amplitúdy. [4]



Obr. 3.1 Klasifikácia signálov podľa náhodnosti priebehu[4]

Širokopásmový signál je úplne náhodný a nemá žiadnu stálu frekvenciu. Je charakterizovaný tým, že je zložený z harmonických zložiek s frekvenciami zo širokého pásma. Spektrum takého signálu obsahuje mnoho rôznych frekvencií, ktoré sa môžu náhodne meniť v čase. Širokopásmové signály sú obvykle spojované so šumom. Môže to byť napríklad záznam hluku, teda akustického tlaku v mieste merania. [4]

3.1 ELEMENTÁRNE DETERMINISTICKÉ SIGNÁLY

Harmonický signál je všeobecne definovaný ako časová funkcia $x(t) = a \cos \omega t + b \sin \omega t$ kde a, b sú reálne čísla, ω je uhlová rýchlosť a t je čas. Súvislosť uhlovej frekvencie ω a frekvencie f je definovaná ako $\omega = 2\pi f$. Pomocou amplitúdy A a počiatočnou fázou φ je výhodné znázorniť harmonický signál Eulerovým vzorcom nasledovne: [4]

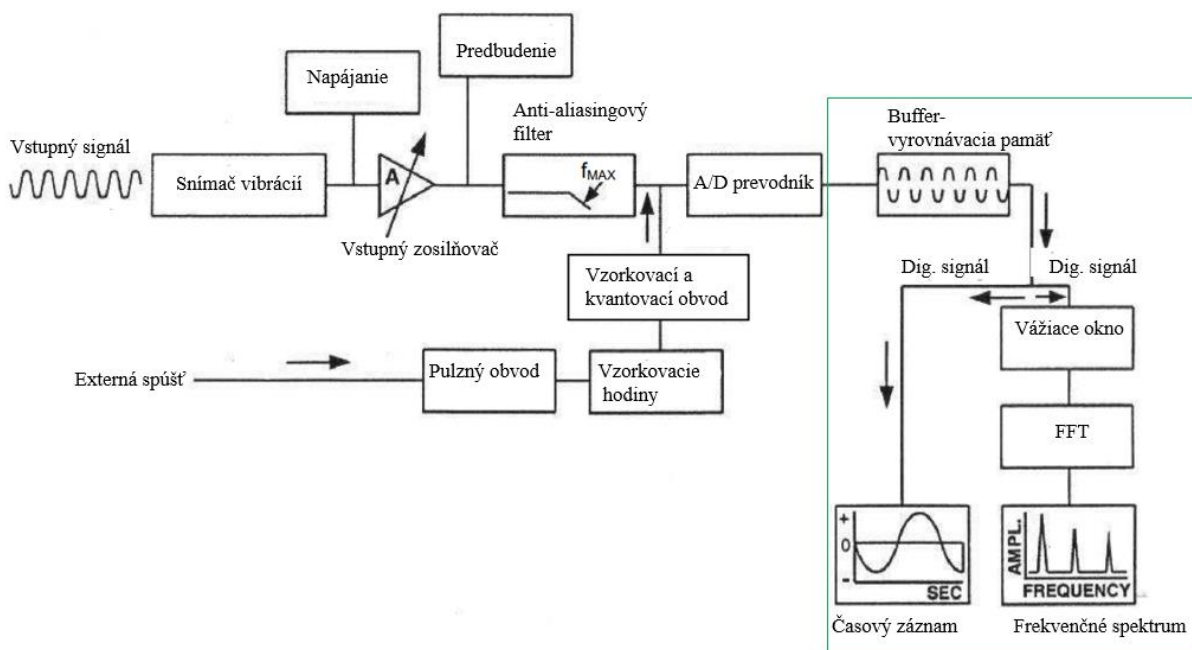
$$x(t) = A \cos(\omega t + \varphi) = \frac{A}{2} e^{j\omega t + \varphi} + \frac{A}{2} e^{-j\omega t + \varphi} \quad (22)$$

3.2 POPIS SIGNÁLU

Hlavnou úlohou signálu je preniesť informáciu. Či už sa jedná o textovú správu, zvukovú nahrávku alebo informáciu o výchyľke, všetky tieto dáta sa dajú reprezentovať číslami. Existujú dve základné skupiny signálov a to analógový a digitálny. Analógový signál sa časom mení a je spojitý. Oproti tomu digitálny signál je nespojitý a diskretný. Dá sa povedať, že ide a postupnosť údajov vyjadrená najčastejšie binárnym kódom- pomocou jednotiek a núl.

3.3 MERACÍ REŤAZEC

Bez analyzátoru by nebolo možné diagnostikovať fyzikálne javy. Služi ako prekladač fyzikálnych javov na digitálne dáta, s ktorými je možné ďalej manipulovať. Analógový signál zo snímača vibrácií prechádza cez zosilňovač a anti-aliasingový filter do A/D prevodníka vid' Obr. 3.2. Tu sa pomocou kvantizácie a vzorkovania získa digitálna hodnota, ktorá ďalej putuje do vyrovnávacej pamäte – zásobníka. Odtiaľto je možné signál zobrazit' v časovej doméne alebo ďalej spracovať a zobrazit' frekvenčné spektrum daného signálu. Táto diplomová práca sa zaoberá hlavne vyznačenou časťou na Obr. 3.2

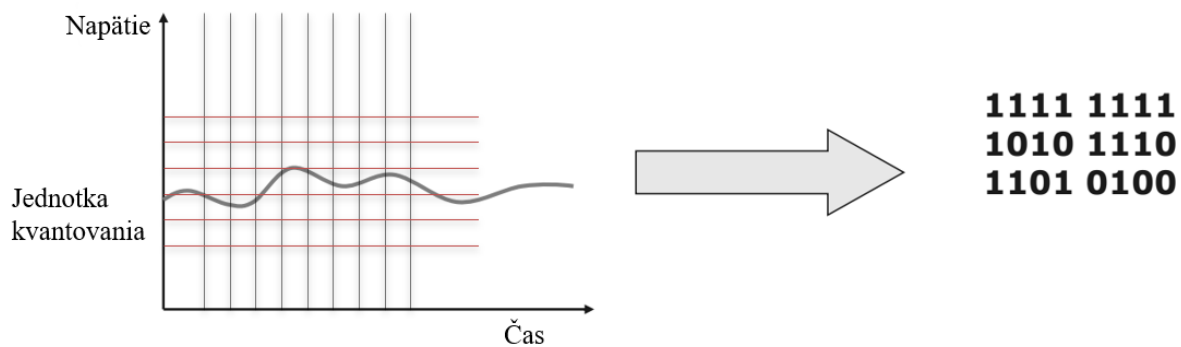


Obr. 3.2 Schéma meracieho reťazca a vyznačená oblasť záujmu [1]

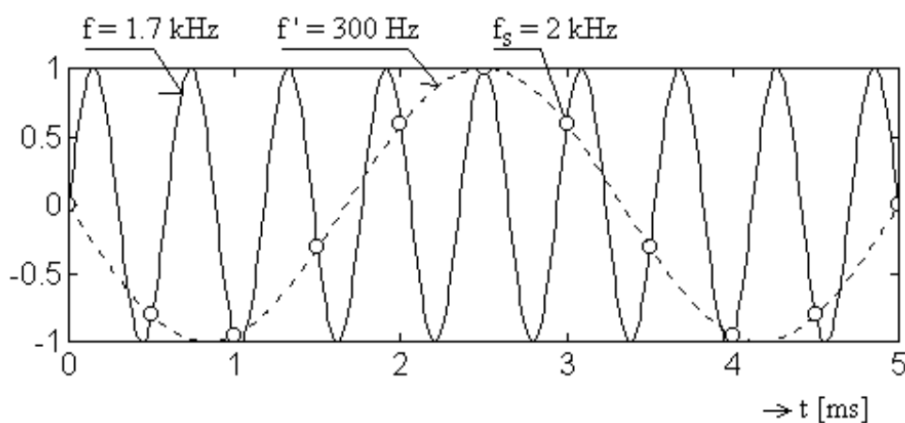
3.3.1 ANALÓG- DIGITÁLNY PREVODNÍK

Je to zariadenie, ktoré prevádza analógový signál na digitálne dáta vid' Obr. 3.3. Pomocou dvoch procesov – kvantizácie (vodorovne) a vzorkovania (zvisle) sa nadelí analógový signál podľa toho, ako sa zvolí vzorkovacia frekvencia a aké rozlíšenie má prevodník. V minulosti to bolo 8-bit, 16-bit a dnes aj viac ako 24-bit čo znamená 2^{24} alebo približne 16,8 milióna úrovní rozlíšenia meranej hodnoty. Vzorkovacia frekvencia delí signál z hľadiska času, teda určuje ako často sa zapisuje nová hodnota.

S tým súvisí chyba typu aliasing a Nyquist-Shanonov teorém. Popisuje jav pri ktorom dochádza k skresleniu výsledného signálu, kvôli zle zvolenej vzorkovacej frekvencii. Hovorí o tom, že vzorkovacia frekvencia musí byť minimálne 2,56-násobne vyššia ako zaznamenaná najvyššia frekvencia. Grafické znázornenie je na Obr. 3.4. Vtedy dochádza k javu zvanému aliasing - prekladanie spektier. Otvplyvňuje zloženie spektier tak, že sa v spektre objavajú falošné zložky, ktoré v skutočnosti prislúchajú zložkám s vyššou frekvenciou. Ak by skutočný



Obr. 3.3 Jednotka kvantovania a vzorkovacia perióda deliaca analógový signál [26]

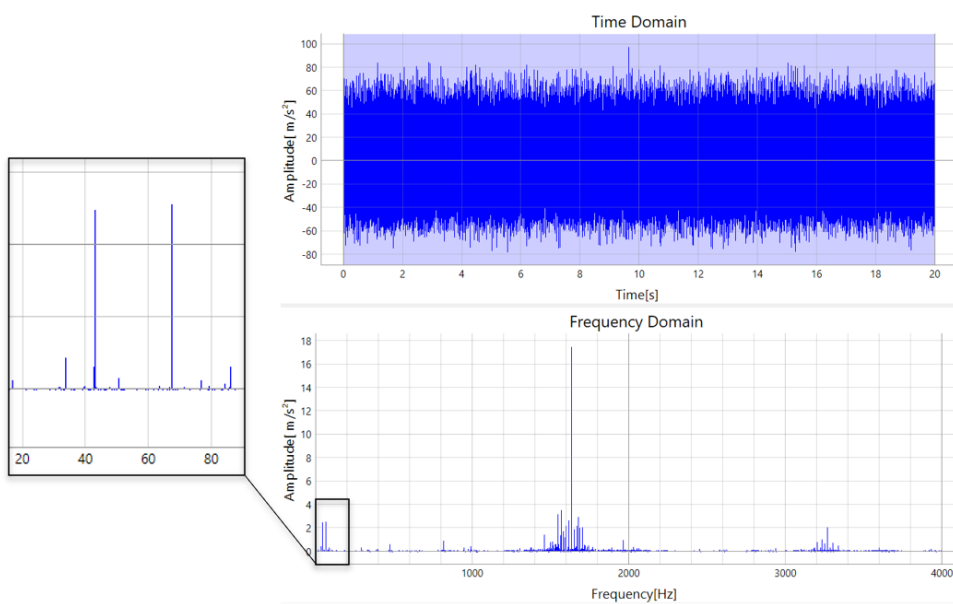


Obr. 3.4 Chyba typu aliasing [4]

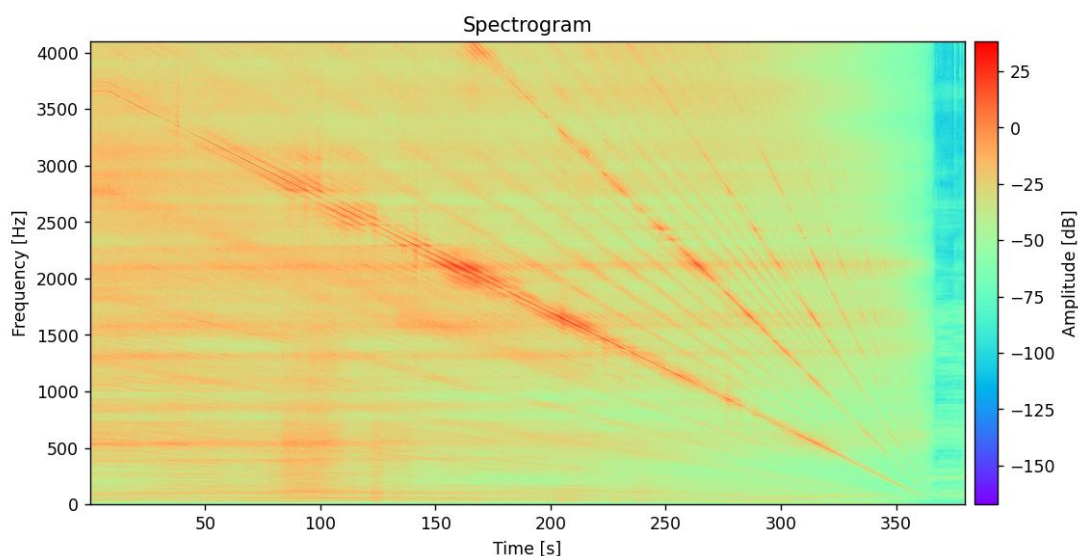
harmonický signál mal rovnakú frekvenciu ako frekvencia vzorkovania, tak by frekvencia zdanlivého signálu bola rovná nule. V praxi sa však používa aj 6 až 10 násobok maximálnej frekvencie. [1], [26]

3.3.2 POPIS VIBROAKUSTICKÉHO SIGNÁLU

Tieto cenné informácie o hodnotách frekvencií sa dajú získať aj zo signálu pomocou Fourierovej transformácie. Na Obr.3.5. je vidieť priebeh signálu v čase, ktorý nemá veľkú výpovednú hodnotu a frekvenčnú doménu. V nej je jasne vidieť dominantnú frekvenciu v okolí 1600 Hz s postrannými pásmami a otáčkovú frekvenciu na začiatku s hodnotou približne 40 Hz. Vo vyšších frekvenciách v okolí hodnoty 3250 Hz je vidno zvýšenie amplitúdy, ktoré môže byť spôsobené harmonickým násobkom zubovej frekvencie. Avšak tento prístup je možný len pri ustálených dejoch, kedy sa otáčky nemenia a daný jav je v čase nemenný. V prípade realizácie rozbehových alebo dobehových skúšok v celom rozsahu pracovných otáčok je potrebné vykresliť multispektrum, alebo aj inak nazývaný spektrogram. V priebehu zmeny otáčok sa môžu fyzikálne javy v štruktúre prejavovať s rôznou intenzitou, preto je potrebné mať prehľad o čase, frekvencii a aj o amplitúde v jednom grafe.



Obr. 3.5 Ukážka z pripravovaného softvéru – časová a frekvenčná doména



Obr. 3.6 Ukážka z pripravovaného softvéru – spektrogram v knižnici Matplotlib s farebnou mapou rainbow (jet)

Multispektrum je spojenie viacerých frekvenčných spektier do jedného 3D grafu, kedy sa vyberie určitá časť signálu a uskutoční sa FFT len na tomto úseku. Takto sa prejde celý signál a výsledkom je spomenutý 3D graf. Spektrogram môže byť vykreslený aj ako 2D graf kedy amplitúdu predstavuje farebná škála vid'. Obr.3.6. Na x-ovej osy je čas a na osy y zase frekvencia. V nasledujúcej kapitole je rozobraný princíp FFT a tvorby multispektra.

3.4 FOURIEROVA TRANSFORMÁCIA

Ako už bolo spomenuté, priebeh signálu sa dá znázorniť v čase - časovej doméne. Jeho vlastnosti avšak vhodne dopĺňa aj rozklad na súbor elementárnych funkcií. V technických aplikáciách je najbežnejší rozklad na súbor harmonických funkcií, ktoré sa líšia amplitúdou, uhlovou frekvenciou a počiatočnou fázou. Signál sa dá zobrazit' aj ako závislosť amplitúdy na frekvencii. Vtedy sa hovorí o frekvenčnej oblasti - doméne. Rozklad periodickej funkcie na harmonické zložky sa nazýva Fourierova rada. Neperiodické funkcie sa rozkladajú pomocou Fourierovej transformácie. [4]

3.4.1 FOURIEROVE RADY

Periodický signál sa dá rozložit' na všeobecne nekonečný rozklad so zložkami frekvencií. Je charakterizovaný tým, že jeho hodnoty sa periodicky opakujú s periódou T , čo znamená

$$x(t) = x(t + iT) \quad i = \pm 1, \pm 2, \pm 3, \dots \quad (23)$$

kde T je perióda a i je násobok danej funkcie.

Tento proces sa nazýva rozklad na Fourierovu radu. Umožňuje reprezentáciu periodického signálu ako súčet sínusových a kosínusových funkcií s rôznymi frekvenciami a amplitúdami. Ďalej umožňuje jednoduché zobrazenie zložitých periodických signálov pri analýze a spracovaní. K vyjadreniu tohoto rozkladu sa používajú Fourierove koeficienty, ktoré sa vypočítajú integráciou. Hovorí nám koľko z funkcie $x(t)$ je v kosínusovej alebo sínusovej projekcii. Táto metóda nie je vhodná pre náhodné alebo neperiodické signály.

$$F_k = \frac{1}{T} \int_0^T x(t) e^{2\pi f_k t} dt, \quad k = 0, \pm 1, 2, \dots \quad (24)$$

kde F_k sú koeficienty Fourierovej rady, T je perióda priebehu funkcie, f je jej frekvencia a t je čas.

Inak povedané, Fourierova rada predstavuje rozklad signálu na nekonečný počet dvojíc vektorov, rotujúcich proti sebe. Je to spôsob ako môže byť vyjadrená periodická funkcia pomocou harmonických zložiek.[4]

3.4.2 FOURIEROVA TRANSFORMÁCIA VŠEOBECNÉHO SIGNÁLU

Všeobecný alebo neperiodický signál sa dá rozložit' na harmonické zložky pomocou Fourierovej transformácie. Takýto rozklad obsahuje zložky so všetkými frekvenciami s nekonečne malou amplitúdou. Spektrum takéhoto signálu je spojitá funkcia frekvencie. Definičné vzorce priamej a spätnej Fourierovej transformácie pre signál $x(t)$ sú nasledujúce:

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt, \quad (25)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega, \quad (26)$$

kde $X(\omega)$ má význam signálu transformovaného do frekvenčnej oblasti a $x(t)$ je originál. [4]

Definičný vzťah priamej Fourierovej transformácie sa dá vysvetliť nasledovne. Eulerov vzorec hovorí, že ak:

$$e^{jn} \quad (27)$$

kde j je komplexná jednotka a n je ľubovoľné číslo, dostali by sme sa po jednotkovej kružnici v komplexnej rovine o jej n -násobok dĺžky.

Pridaním frekvencii f do vzťahu, získame rotáciu s frekvenciou f cyklov za sekundu po kružnici v komplexnej rovine. Hodnota 2π reprezentuje celú dĺžku kruhu a t je premenná času. Tým získame nasledovný výraz:

$$e^{-j2\pi ft} \quad (28)$$

Výraz v rovnici 28 je vynásobený funkciou $x(t)$ a sú získané hodnoty tejto funkcie okolo pomyselného kruhu v komplexnej rovine.

$$x(t) \cdot e^{-j2\pi ft} \quad (29)$$

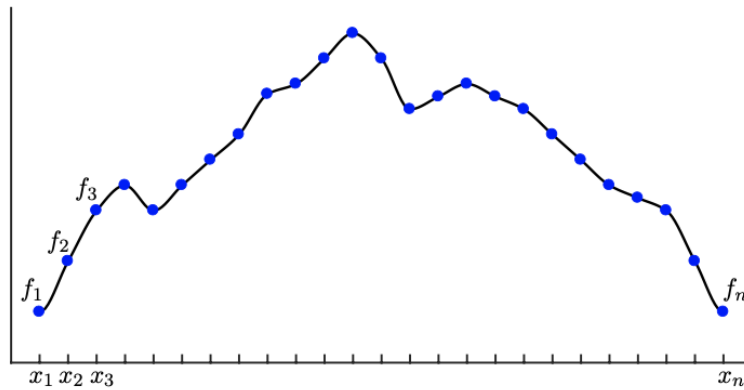
Následná integrácia môže byť interpretovaná ako nájdenie pomyselného ťažiska tohto signálu okolo kruhu. Nahradením $2\pi f = \omega$ je získaný definičný vzťah pre Fourierovu transformáciu.

$$\int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (30)$$

V určitých prípadoch nadobúda hodnota tohto integrálu hodnoty výrazne odlišné od 0 a to práve vtedy ak sa s frekvencia dostane na hodnotu, s ktorou daný signál osciluje. Ak by signál pozostával z viacerých takýchto frekvencií, bolo by získané pre každú z týchto frekvencií vychýlenie. To znamená že signál zložený s ľubovoľného počtu frekvencií sa dá týmto spôsobom rozdeliť a dajú sa identifikovať jednotlivé harmonické zložky signálu, ktoré sú dominantné.

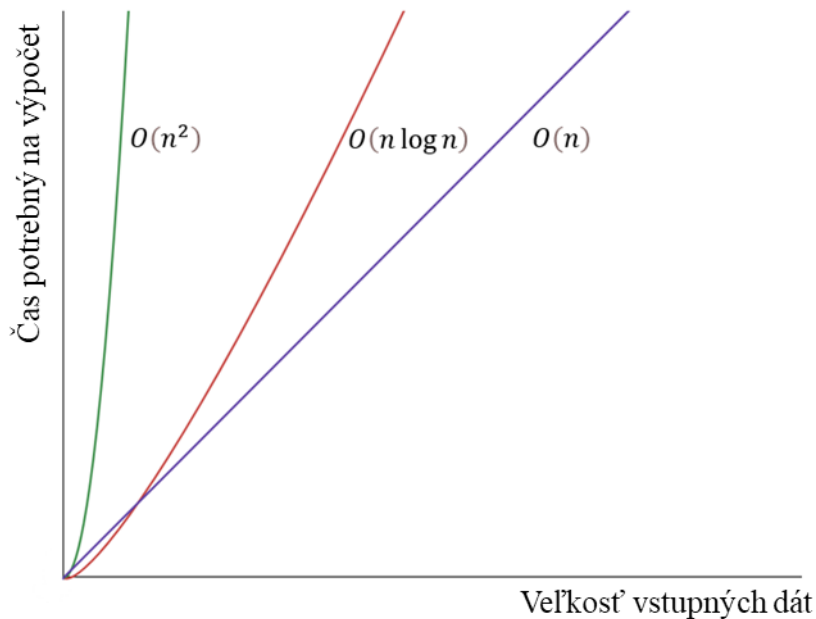
3.4.3 DISKRÉTNÁ FOURIEROVA TRANSFORMÁCIA

Doteraz boli vysvetlené pojmy Fourierova transformácia a Fourierove rady pre spojité signály $x(t)$. Pri práci s reálnymi dátami je potrebné aproximovať Fourierovu transformáciu na diskretných vektoroch z dát. Výsledná diskretná Fourierova transformácia (DFT) je v podstate diskretizovaná Fourierova rada pre vektory dát $[x_1 x_2 x_3 \dots x_n]^T$ získaná diskretizáciou funkcie $x(t)$ so vzorkovaním Δx vid'. Obr. 3.7. [5]



Obr. 3.7 Diskretizácia funkcie f pre potreby DFT [5]

DFT je užitočný nástroj pri numerických výpočtoch a aproximácii, ale má nevýhodu v náročnosti výpočtu pre veľmi veľký počet n . Výpočet obsahuje maticu $n \times n$, ktorá potrebuje na výpočet $O(n^2)$ operácií. To znamená, že časová náročnosť na výpočet rastie s druhou mocninou. Až pri vynájdení revolučného algoritmu rýchlej Fourierovej transformácie (FFT) sa podarilo znížiť časovú náročnosť na výpočet na $O(n \log n)$ vid' Obr. 3.8.. Práve tento vynález umožňuje spracovanie signálu v reálnom čase napríklad pri komunikácii založenej na kompresii obrazu a zvuku. Využíva sa však aj pri odstraňovaní šumu v signáli. [5]



Obr. 3.8 O -notácia časovej zložitosti algoritmov

Pri vzorkovaní vznikajú určité požiadavky na vzťah medzi frekvenciou vzorkovania a frekvenčným rozsahom signálu. Bude sa uvažovať, že za jednu periódu je nameraných N diskretných hodnôt spojitého signálu. Postupnosť N vzoriek sa nazýva záznam a celkový počet vzoriek je dĺžka záznamu. Perióda signálu T teda jeho dĺžka, sa vypočíta nasledujúcim vzťahom

$$T = N \cdot \Delta t \quad (31)$$

kde N je počet vzoriek záznamu a Δt je čas medzi vzorkami.

Vzorkovacia frekvencia je daná prevrátenou hodnotou periódy vzorkovania

$$f_s = \frac{1}{\Delta t} \quad (32)$$

Prevrátená hodnota dĺžky snímania sa označuje ako základná frekvencia snímania záznamu

$$f_0 = \frac{1}{T} \quad (33)$$

Pre frekvenciu vzorkovania potom platí

$$f_s = N \cdot f_0 \quad (34)$$

Diskrétna Fourierova transformácia je daná :

$$X_k = \sum_{i=0}^{n-1} x_i e^{-j2\pi i \frac{k}{n}} \quad (35)$$

Maticovým násobením je získaný výsledný vektor \hat{x} :

$$\begin{bmatrix} \widehat{x}_1 \\ \widehat{x}_2 \\ \widehat{x}_3 \\ \dots \\ \widehat{x}_n \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 0 & \omega & \omega_n^2 & \dots & \omega_n^{n-1} \\ 0 & \omega_n^2 & \omega_n^4 & \dots & \omega_n^{2(n-1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \omega_n^{(n-1)} & \omega_n^{2(n-1)} & \dots & \omega_n^{(n-1)^2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \\ x_n \end{bmatrix} \quad (36)$$

Výsledný vektor \hat{x} obsahuje Fourierove koeficienty pre vstupný vektor x . DFT matica F je Vandermondova matica, keďže v každom svojom riadku obsahuje po sebe idúce členy geometrickej postupnosti začínajúcej číslom 1. Taktiež obsahuje komplexné čísla, čiže výsledný vektor \hat{x} obsahuje amplitúdu aj fázu. Táto vlastnosť má vhodné fyzikálne využitie. Taktiež môžeme konštatovať, že táto matica má hierarchickú a symetrickú štruktúru, čo sa ďalej využije pri rýchlej Fourierovej transformácii. [3]

3.4.4 RÝCHLA FOURIEROVA TRANSFORMÁCIA (FFT)

FFT je spôsob akým efektívnejšie spočítať DFT. Ako už bolo spomenuté, násobenie DFT maticou F je potrebné $O(n^2)$ operácií, zatiaľ čo pri rýchlej Fourierovej transformácii $O(n \log n)$. Napríklad bežný zvukový signál je vzorkovaný frekvenciou 44,1 kHz - 44100 vzoriek za sekundu. Pri dĺžke snímania 10 sekúnd by mal vektor f rozmer $n = 4,41 \cdot 10^5$. Výpočtom cez DFT by bolo potrebné 200 miliard násobení. Pre porovnanie s algoritmom FFT je potrebné iba 6 miliónov násobení. To je zrýchlenie procesu o vyše 30000 krát. Hlavným princípom ktorý stojí za rýchlou Fourierovou transformáciou je efektívnejšie

rozloženie DFT matice F . Ak je počet dát n mocninou 2, napríklad $n = 1024 = 2^{10}$, tak DFT matica F_{1024} môže byť zapísaná ako:

$$\hat{x} = F_{1024} f x = \begin{bmatrix} I_{512} & -D_{512} \\ I_{512} & -D_{512} \end{bmatrix} \begin{bmatrix} F_{512} & 0 \\ 0 & F_{512} \end{bmatrix} \begin{bmatrix} x_p \\ x_n \end{bmatrix} \quad (37)$$

kde x_p sú párne indexy vektora x a x_n sú nepárne. I_{512} je jednotková matica s rozmerom 512×512 . Matica D_{512} je daná:

$$D_{512} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & \omega & 0 & \dots & 0 \\ 0 & 0 & \omega^2 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \omega^{511} \end{bmatrix} \quad (38)$$

Ak $n = 2^p$, tento proces môže byť opakovaný teda, je použitý rekurzívne a F_{512} môže byť reprezentované F_{256} , ďalej F_{128} atď.. Ak $n \neq 2^p$ tak k vektoru môžu byť pridané nuly, až dokým nebude mocninou čísla 2. FFT potom zahŕňa efektívne prelínanie párných a nepárnych indexov menších vektorov x a výpočet niekoľkých menších 2×2 výpočtov DFT.[3]

3.4.5 KRÁTKODOBÁ FOURIEROVA TRANSFORMÁCIA (STFT)

Aj keď Fourierova transformácia ponúka detailné informácie o obsahu frekvencií daného signálu, nehovorí nič o tom, kedy sa tieto frekvencie vyskytujú. Ako už bolo spomenuté, v oblasti vibroakustického merania, kedy sa robí meranie celého rozsahu prevádzkových otáčok, je dôležité poznať aj konkrétny čas výskytu. Fourierova transformácia je schopná zmysluplne spracovať iba periodické a stacionárne signály. Pre signály s nestacionárnymi frekvenciami je potrebné charakterizovať frekvenčné spektrum zároveň s časom. K tomuto účelu slúži krátkodobá Fourierova transformácia, v anglickej literatúre short-time Fourier transform (STFT). Pri tomto postupe sa Fourierova transformácia pozerá na časť signálu cez vážiace okno, cez ktoré je približne stacionárny. Časť signálu sa vynásobí vážiacim oknom, následným preložením s druhou časťou signálu sa proces opakuje až do konca celkového signálu. Takto sa pre každý segment vykoná FFT vid' Obr.3.9. STFT je definovaná nasledujúcim vzťahom a je závislá na čase a frekvencii :

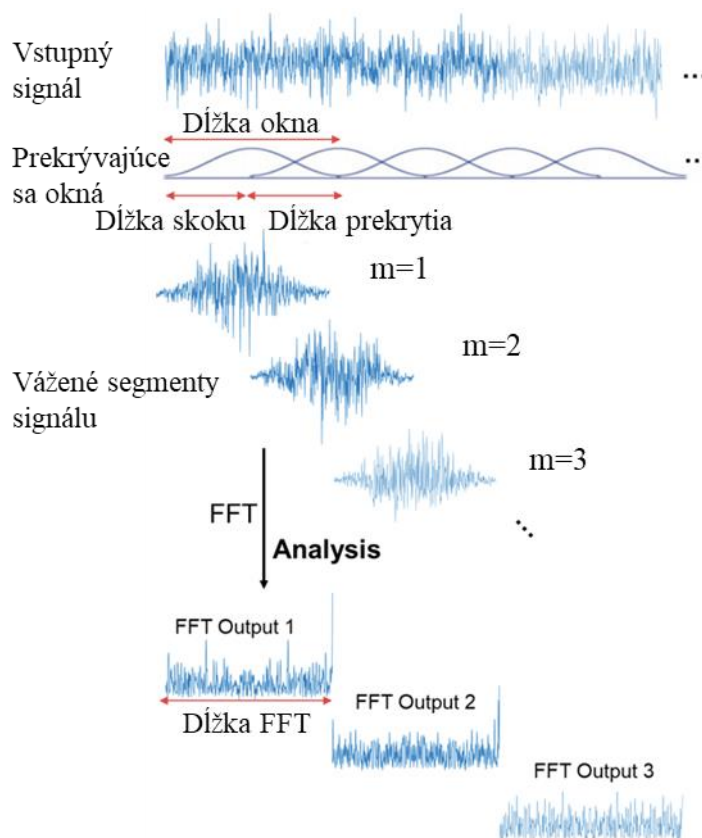
$$X(\omega, \tau) = \int_{-\infty}^{\infty} x(\tau) e^{-j\omega\tau} g(t - \tau) d\tau, \quad (39)$$

kde $g(t - \tau)$ je funkcia vážiaceho okna, ktorá sa časom posúva.

Podobne ako pri DFT aj tu musí byť vzťah definovaný pre diskretizované dáta nasledovne :

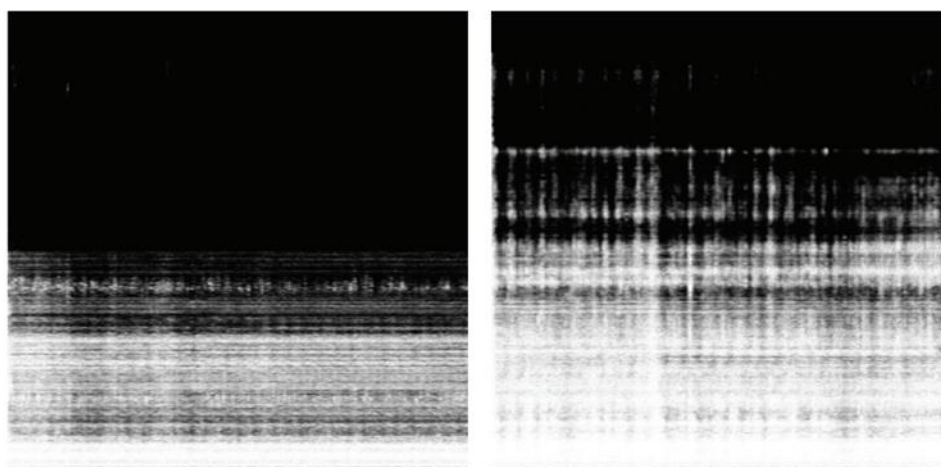
$$X_{(k,m)} = \sum_{n=0}^{N-1} g(n)x(n + mH) e^{-j2\pi n \frac{k}{N}} \quad (40)$$

Kde k je index frekvencie, m je index počítaného segmentu, N je veľkosť segmentu, n je poradie vzorky signálu v segmente a H je veľkosť skoku medzi segmentmi.[5]



Obr. 3.9 Postup pri algoritme STFT [8]

STFT je možné využiť aj v iných oblastiach ako je napríklad hudobný priemysel, spracovanie reči pomocou strojového učenia ale aj priamo v diagnostike prevodových ústrojenstiev, kedy je možné nájsť vady v ložiskách alebo konkrétne vzory v charaktere signálu. Na to je potrebné zo signálu v čase vytvoriť pomocou STFT takzvaný časovo frekvenčný obraz - TFI (Time-frequency images).[6] Je to v podstate 2 dimenzionálna reprezentácia obrazu, ktorý



(a) Bez vady

(b) Vada na vnútornej dráhe

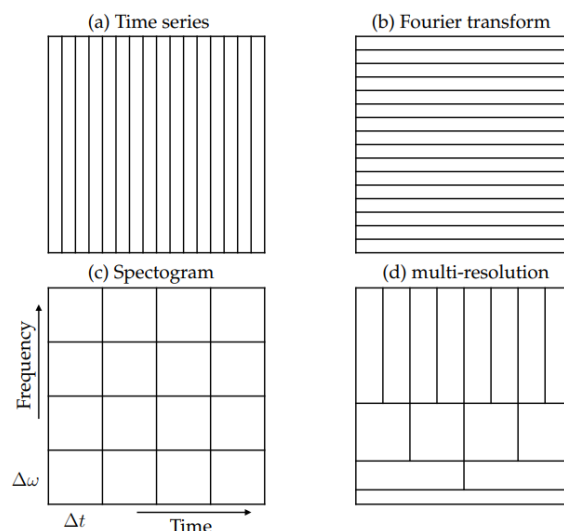
Obr. 3.10 Čiernobiele obrázky spektrogramov z ložísk [7]

môže byť pre človeka čitateľný vid' Obr.3.10. , ale v tomto prípade je potrebný jeho číselný tvar. Táto číselná reprezentácia je vhodným vstupom pre neurónové siete, ktoré rozhodnú o jeho klasifikácii. Pomocou zvolených natrénovaných modelov je možné klasifikovať takéto dáta z merania. Na Obr. 3.9.(a) je vidieť príklad TFI z ložiska ktoré nemá žiadnu vadu a Obr.3.9 (b) z ložiska, ktoré má vadu na vnútornom krúžku. Voľným okom je vidieť, že na obrázku vpravo je zastúpené väčšie množstvo frekvencií. Vyššie spomenutým postupom bolo v štúdiu získaná presnosť klasifikácie rôznych väd v ložisku až 97%. Autori štúdie hovoria o tom, že tento postup založený na STFT a neurónových sieťach má vysokú účinnosť oproti tradičným metódam a umožňuje automatizáciu rozoznávania väd naprieč rôznym spektrom aplikácií. Je však nutné pracovať na výpočtovej náročnosti a rôznych pracovných podmienkach hlavne v reálnom čase. [7]

Kompromis medzi rozlíšením

V spektrograme nie je možné získať ľubovoľne presné rozlíšenie frekvencie a času zároveň. Existuje kompromis medzi rozlíšením frekvencie a času. Tento jav popisuje Heisenbergov princíp neurčitosti, ktorý hovorí, že čím presnejšie určíme jednu veličinu, tým menej presne tú druhú. Pri pohľade na časovú doménu v Obr. 3.11 (a) je vidieť, že je perfektne rozlíšiteľná ale neobsahuje žiadne informácie o frekvencii, tak isto pri frekvenčnom spektre, ktoré naopak neposkytuje informácie o čase vid' Obr.3.11(b). Spektrogram vid' Obr. 3.11(c) rozlišuje obe veličiny, avšak ak je frekvenčné spektrum jemne rozlíšené, strácajú sa informácie o tom, kedy daný jav nastal. Naopak pri rozlíšení času, zase nie sú dostupné informácie o presnej frekvencii.[5]

V prípade potreby naozaj presného a jemného rozlíšenia v oboch oblastiach, existuje Vlnková transformácia (anglicky Wavelet transform), ktorá dosahuje optimálnu rovnováhu medzi



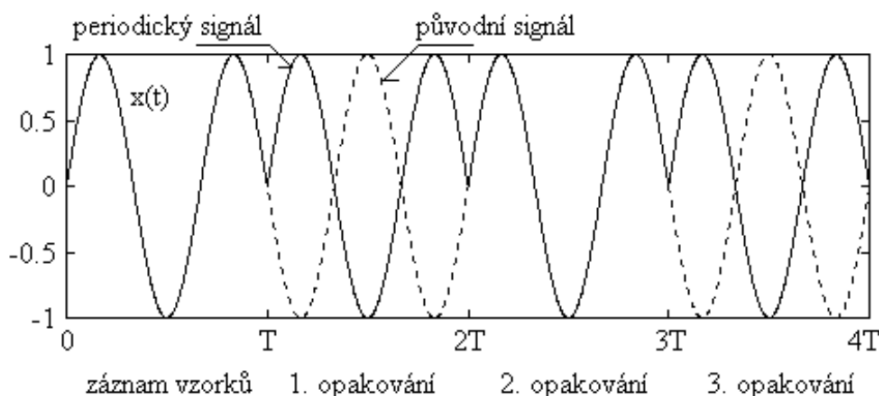
Obr. 3.11 Porovnanie rozlíšenia rôznych typov zobrazenie frekvencií a času [5]

rozlíšeniami času a frekvencie a tým prevyšuje STFT. Vo vyšších frekvenciách je rozlíšenie času jemnejšie a v nižších zase hrubšie. Vychádza z toho, že prechodové javy sa vyskytujú vo vysokých frekvenciách, teda je potrebné mať viac informácií o čase a naopak pri nízkych frekvenciách sa predpokladá väčšie frekvenčné rozlíšenie vid' Obr.3.11(d). Je vhodné ju využiť pri rapidne sa meniacom signáli, alebo v oblasti spracovania obrazu, fotiek a kompresii

veľkých dát. Ďalej však nebude táto transformácia rozvíjaná, pretože pre potreby spracovania signálu je dostačujúci algoritmus FFT a STFT.[5],[9]

3.4.6 VOĽBA ČASOVÉHO OKNA

Medzi veličinami pri meraní spektra signálu, ako sú frekvenčný rozsah, dĺžka záznamu a doba merania, je priama súvislosť. Na obrázku 3.12 je vidieť takzvané časové okno, čo predstavuje výsek pre výpočet DFT a je považovaný za jednu jeho periódu. Je však vidieť, že pôvodný signál vyzerá inak a to sa prejaví aj v zložení spektra tak, že obsahuje falošné zložky, ktoré v spektre naozaj nie sú. [4].



Obr. 3.12 Pôvodný harmonický signál a jeho periodické pokračovanie [4]

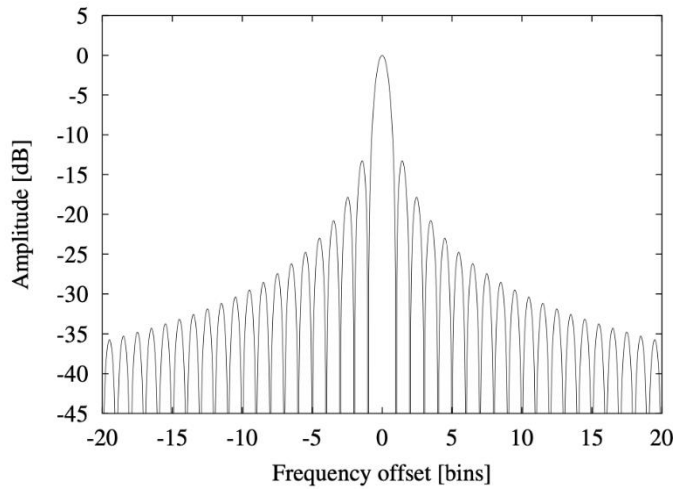
Ak by bol zvolený úsek s dĺžkou N zo signálu obsahujúci len čistý sínusový signál a je vypočítaná DFT, nezískal by sme iba jeden ostrý vrchol vo frekvenčnej doméne s hodnotou frekvencie daného signálu. S veľkou pravdepodobnosťou by bol získaný výsledok podobný tomu na Obr. 3.13. Dôvodom je fakt, že DFT považuje signál za periodický, teda že časový signál s dĺžkou N sa opakuje donekonečna. Ak však frekvencia sínusového signálu nie je presný násobok periódy signálu, tento predpoklad neplatí. Týmto spôsobom vzniká diskontinuita medzi poslednou a prvou vzorkou, kvôli cyklickému pokračovaniu.

Možným riešením je násobenie signálu s funkciou časového okna ešte pred použitím DFT. Vlastnosťou časového okna je, že začína v blízkosti nuly alebo priamo v nule, následne sa zvýši na maximum v strede časového radu a opäť sa zníži na konci vid' Obr. 3.14. Tým sa docielu odstránenie nespojitosti. Existuje mnoho funkcií časových okien a zvyčajne zahŕňajú kompromis medzi šírkou výsledného maxima vo frekvenčnej oblasti, presnosťou amplitúdy a rýchlosťou poklesu spektrálneho presakovania do iných frekvenčných intervalov. [12]

Pojem spektrálne presakovanie priamo súvisí s tým, že signál v okne pozorovania nevykazuje periodický vzor. Ak je takýto signál periodicky predĺžený budú na hraniciach pozorovaného okna vznikáť diskontinuity. Tieto diskontinuity vytvárajú spektrálne príspevky a šíria sa po celej frekvenčnej sade vid' Obr.3.14. Práve aplikovaním časových okien sa upravuje a presňuje výsledok DFT (pomocou FFT) tým, že sa zredukuje spektrálne presakovanie.[13]

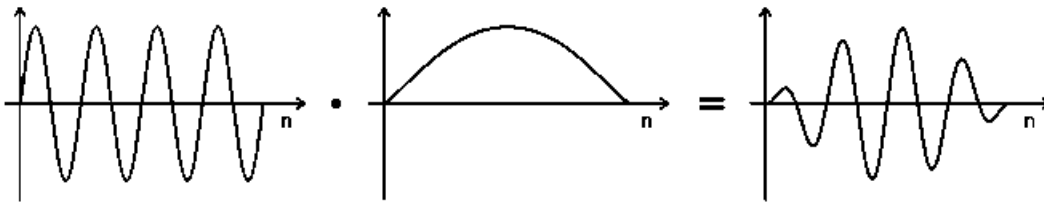
Ak je uvažovaný periodický signál a je možné vybrať celú jeho periódu alebo celočíselné násobky na predĺženie, je to ideálny prípad. Ak úsek nie je celočíselným násobkom, vtedy

dochádza k diskontinuite a spomínanému spektrálnemu presakovaniu. Kvôli tomu je vhodné použiť okno, s hodnotou signálu na oboch koncoch blížiacou sa k 0.

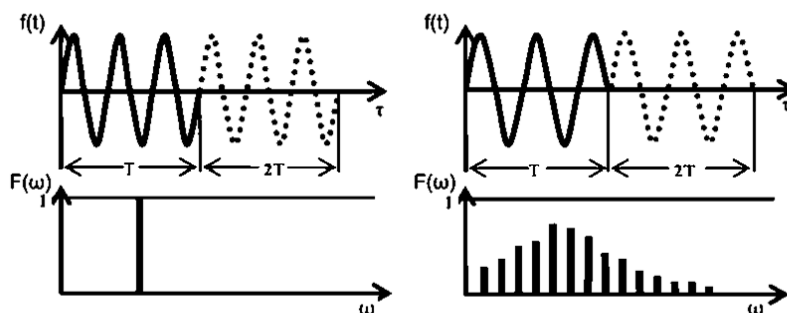


Obr. 3.13 Spektrálna odozva obdĺžnikového okna [4]

DFT sa dá vylepšiť zmenami funkcie okna a jej optimalizáciou. Ak výrez obsahuje jeden a pól periódy, jedná sa o najhorší prípad. Dochádza k rozmazaniu hlavnej spektrálnej hodnoty a poklesu jej amplitúdy. Energia harmonického signálu sa rozloží do susediacich frekvencií.



Obr. 3.14 Násobenie signálu s funkciou časového okna [37]



Obr. 3.15 Presakovanie spektra pri celom násobku periódy a pri polovičnom násobku [37]

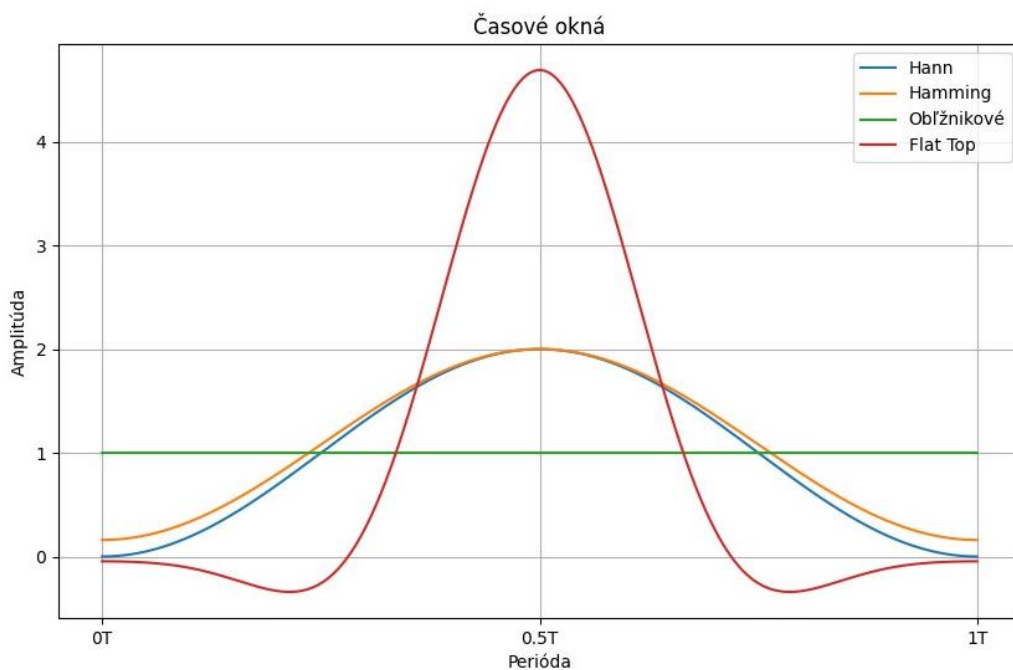
Každé z nasledujúcich okien vid' Obr. 3.16 je v matematická funkcia zadaná predpisom. Boli vybrané len základné okná, existuje ich oveľa viac. Obdĺžnikové okno, taktiež známe ako Dirichletovo okno, je jednotné v celom intervale pozorovania a možno si ho predstaviť ako

postupnosť aplikovanú na údaje tak, aby mali konečný rozsah. Pri hranách nie sú modifikácie, spektrum je rozťahnuté a amplitúda spektra pre hlavnú frekvenciu je skreslená.

Okno Hann využíva funkciu cosinus, kvôli hodnotám na krajoch blízky nule. Patrí medzi najpoužívanejšie okná. Správny názov okna je Hann, nie Hanning.

Modifikáciou okna Hann vznikne okno Hamming. Má väčší vplyv na vyrušenie postranných pásiem. Je definované pomocou koeficientov. Perfektne vyrušuje prvé postranné pásmo.

Pri praktickom meraní periodických a kváziperiodických signálov sa využíva vo veľkej miere okno Flat top. Aj toto okno ma plynulý prechod z nuly na začiatku a na konci. Váha signálu je na okrajoch dokonca záporná a v strede je najväčšia. To má za následok, že je braný do úvahy hlavne stred signálu a nie jeho napojenie na začiatku a konci výrezu. Na Obr 3.16. je vidno porovnanie doteraz zmienených časových okien.

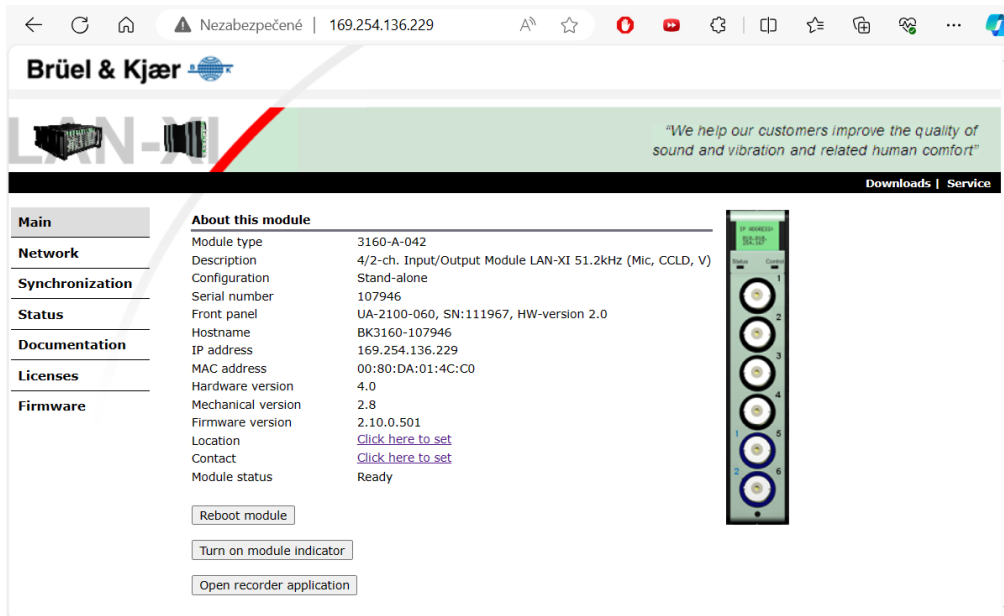


Obr. 3.16 Porovnanie časových okien

4 TVORBA DIAGNOSTICKÉHO NÁSTROJA

Modul LAN-XI od firmy Hottinger Brüel and Kjær (HBK) ponúka veľkú mieru prispôsobenia od konektorov na prednej strane až po samotné spracovanie nameraných dát. Spojenie medzi počítačom a modulom je zabezpečené pomocou Ethernet kábla, ktorý slúži nielen na prenos dát. V tomto prípade je modul používaný len ako A/D prevodník, ale je možné ho použiť aj v opačnom smere ako generátor. Je použitý model analyzátoru 3160-A-042, teda obsahuje 4 vstupy a 2 výstupy ako generátor. Softvérové a grafické rozhranie je zabezpečené cez Windows aplikáciu BK Connect. Avšak, môže sa stať, že pre individuálne potreby nie je táto aplikácia vhodná, ako napríklad pre užívateľov operačných systémov Linux alebo OS X od Apple, alebo ak je potrebné spracovanie signálu a zber dát pomocou vlastného softvéru – ako aj v tomto prípade. Na to slúži open API, čo znamená, že protokol aplikačného rozhrania je voľne dostupný pre verejnosť, teda je open source a je možné ho využiť vo vlastnom programe. [17]

Open API pre LAN-XI je protokol založený na REST API a JSON technológii známej z webov. Keďže kontrola, riadenie modulu a nastavenie TCP socketov je založená na základe príkazov z HTTP (GET, PUT a POST), môžu byť použité rôzne programovacie jazyky ako napríklad C#, C/C++ alebo Python. TCP (Transmission Control Protokol) je základný protokol vďaka ktorému sa v počítačových sieťach môžu medzi sebou vytvárať spojenia a tým posielat' dáta. LAN XI ma aj webové rozhranie vid' Obr. 4.1. kde je možné nastaviť meranie, ukladanie dát a získať informácie o snímačoch. Toto rozhranie je však limitované



Obr. 4.1 Webové rozhranie analyzátoru LAN-XI

a nedá sa v ňom nastaviť všetko potrebné pre bezproblémové meranie a získanie dát. Preto bolo potrebné ovládať analyzátor cez príkazy založené na technológii REST API priamo cez Python. Samotné streamovanie z alebo do LAN-XI je už zložitejšie, preto sú na stránke GitHub - github.com/hbk-world firmy HBK nahrané príklady kódu napísaného v jazykoch C# a v Python. Sú potrebné hlavne na demonštráciu toho, ako sa interpretujú binárne dáta získané priamo z analyzátoru na čísla a text, ktorému človek rozumie a vie s ním ďalej pracovať. Taktiež je zabezpečené spojenie medzi sieťovým adaptérom počítača a analyzátorom. Túto

časť softvéru nebolo potrebné vyvíjať od začiatku, ale stačilo ju upraviť pre konkrétne potreby tejto práce. [17]

Na začiatku práce s LAN-XI bolo potrebné pochopiť podstatu a povahu prenosu dát. Taktiež to, ako a kam program napísaný v jazyku Python dáta ukladá, akým spôsobom ich zobrať a uložiť na disk vo formáte v akom je potrebné pre rýchly zápis a čítanie. Bolo potrebné pochopiť základy dátových formátov a formátov na prenos a ukladanie nastavenia analyzátora alebo snímačov, vo formáte JSON. V nasledujúcej kapitole je popísaný postup pri vytváraní tohto softvéru, použitý programovací jazyk, knižnice a jednotlivé funkcie. Taktiež formát uloženého súboru, jeho štruktúra a dôvody využitia práve tohto riešenia.

4.1 PROGRAMOVACÍ JAZYK A KNIŽNICE

Ako už bolo vyššie spomenuté, tento softvér bol napísaný v jazyku Python. Je to objektovo orientovaný, (ďalej len OOP) vysokoúrovňový programovací jazyk. OOP znamená, že v programe vieme zadefinovať triedu a ďalej ju využiť v programe viac krát v inej podobe a tak vytvoríme nový objekt. Takáto trieda má svoje vlastnosti, ale aj metódy (funkcie). Tým že v OOP objekty dedia vlastnosti a funkcie, je možné vytvárať nové inštancie. Napríklad z triedy auto vieme vytvoriť viac objektov áut konkrétnej značky výrobcu, ktoré vedú rovnako jazdiť dopredu a dozadu, ale majú inú farbu a iný výkon. Táto vlastnosť zjednodušuje a urýchľuje písanie softvéru, ale pre jej plnohodnotné využitie je potrebná značná prax.

Ďalšou výhodou jazyku Python je, že syntax je pomerne jednoduchá a podobná anglickému jazyku, takže kód je čitateľný a dá sa v ňom dobre orientovať. Je možné využívať rozšírenia – knižnice. Tieto knižnice slúžia v rôznych oblastiach použitia na zvýšenie produktivity pri vývoji. Je možné používať knižnice v širokom spektre napríklad na vývoj webov – *Django*, *Flask*, vývoj desktopových aplikácií – *PyQT*, *Tkinter* na analýzu dát – *Pandas*, *Matplotlib*, *NumPy* alebo v oblasti umelej inteligencii – *PyTorch*, *TensorFlow*.

Pri vývoji boli použité rôzne dátové štruktúry pre manipuláciu a ukladanie dát. Medzi základne z nich patria integer – číslo, float – číslo s desatinným miestom. String – text, je ďalším typom dát a je ohraničený zátvorkami a úvodzovkami („...“). Veľký význam má v tejto práci aj štruktúra *list*. Je to dátový typ na zoskupenie iných hodnôt. Zapisuje sa doňho zoznam hodnôt (objektov) oddelených čiarkami medzi hranatými zátvorkami $[x,y,z]$. Dôležitou vlastnosťou je, že môže byť delený a každá hodnota má index, podľa ktorého sa dá vybrať. Ďalší dátový typ je *slovník* (*dictionary*), predstavujúci asociatívnu pamäť. Na rozdiel od listu, ktorý je indexovaný rozsahom čísel, *slovník* je indexovaný *klúčmi*, ktoré môžu byť ľubovoľného nemenného typu. *Slovník* je skupina *klúč:hodnota* (*key:value*) párov s podmienkou, že *klúč* je unikátny. Hlavnou úlohou je uloženie hodnoty s nejakým kľúčom a následné získanie hodnoty späť podľa daného kľúču. [25]

Podobným štýlom je realizovaná štruktúra formátu JSON. Je to skratka pre „JavaScript object notation“. Umožňuje kompaktný popis dátových štruktúr a ich súvislostí medzi sebou. Slúži ako formát na výmenu údajov. Ľahko sa číta a upravuje manuálne ale aj strojovo. Je nezávislý od programovacieho jazyka. [17]

XML alebo Extensible Markup Language je ďalšia metóda na popis dátových štruktúr. Je sebaopisný a môžu byť doňho vkladané alebo odoberané informácie. Je taktiež ľahko čitateľný človekom alebo počítačom. Slúži hlavne na výmenu dát a konfiguračných súborov ako je to aj v tomto prípade. [17]

4.2 STANOVENIE POŽIADAVIEK

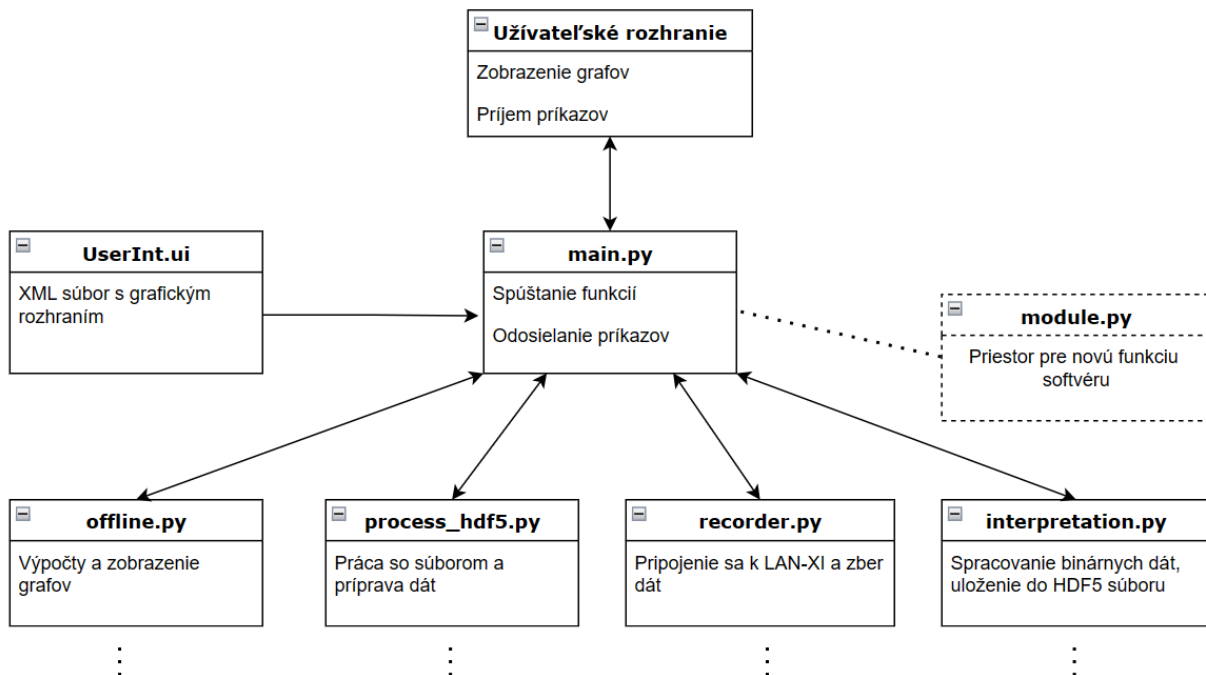
Na začiatku boli stanovené požiadavky pre softvér. Má umožňovať získanie dát, ich ukladanie a následné spracovanie. Taktiež zobrazenie teoretických frekvencií na základe vstupov od užívateľa a v ideálnom prípade aj zobrazenie dát v reálnom čase. Toto všetko by malo byť rýchlejšie ako komerčne dostupný program BK Connctet. Z toho vyplynuli hlavné časti riešenia. Prvá časť riešenia, nazvaná offline, je zameraná na spracovanie už uloženého súboru z merania, alebo z poskytnutých dát a zobrazenie v grafoch. Obsahuje podkategóriu, ktorá sa venuje diagnostike a zobrazovaniu teoretických frekvencií v nameraných dátach a slúži pre ich porovnanie a validáciu. Skladá sa z viacerých funkcií, ktoré majú nasledujúce úlohy:

- Získanie otáčok
- Výpočet teoretických frekvencií
- Načítanie dát z experimentu
- Porovnanie frekvencií v signáli s experimentom a teoretickými hodnotami
- Grafické zobrazenie

Ďalšia časť je zameraná na nadviazanie spojenia s modulom LAN-XI a pochopenie podstaty prenosu dát. Prenesené dáta je potrebné ďalej spracovať a uložiť na disk aby mohli byť zobrazené ako v prvej časti programu. V nasledujúcom texte sú detailne rozobrané jednotlivé časti programu tak, ako boli v priebehu vývoja riešené.

4.3 LOGIKA KÓDU

Vo fáze plánovania bolo potrebné mať určitú predstavu o fungovaní a prepojení kódu. Každá časť softvéru má vlastný súbor s triedou a funkciami v nej vid' Obr. 4.2. Centrálnou časťou



Obr. 4.2 Základná schéma logiky kódu

kódu je *main.py*, odkadiaľ sa spúšťa grafické užívateľské rozhranie. Taktiež sa tu načítava súbor *UserInt.ui*. Je vo formáte XML s údajmi o grafickom užívateľskom rozhraní. XML typ súboru slúži na uchovanie a transport dát, hlavne na konfiguráciu a výmenu informácií medzi rôznymi systémami. V tomto prípade je to program *QtDesigner*, v ktorom sa vytváralo užívateľské rozhranie a programovací jazyk *Python*. Sú tu prijímané príkazy a na základe toho ďalej rozposielané do jednotlivých častí programu. Ako je vidieť na Obr. 4.2 jednotlivé čiary znázorňujú interakciu medzi modulmi kódu pomocou šípok. Všetky moduly medzi sebou komunikujú a vymieňajú si informácie cez argumenty funkcií alebo tried. Pomocou príkazu *import* sa postupne do *main.py* vložia všetky triedy z daných modulov. Vtedy je možné zavolať príkazy mimo daného modulu priamo v *main.py* a tak ovládať všetky funkcie softvéru. Bolo snahou zachovať určitú modularitu, v zmysle aby sa v budúcnosti dali jednoducho pridať ďalšie časti softvéru. Takýmto spôsobom môžu byť rozšírené už existujúce funkcie a tým zabezpečiť aj možnosť aktualizácií pre novšie verzie.

4.4 SPRACOVANIE DÁT

Bez vhodného spracovania nemajú samotné dáta žiadnu výpovednú hodnotu. Preto je potrebné uložené dáta spracovať a graficky ich zobrazit'. Cieľom bolo vytvoriť grafické prostredie s tromi oknami a to konkrétne graf v časovej doméne, frekvenčnej doméne a spektrogram.

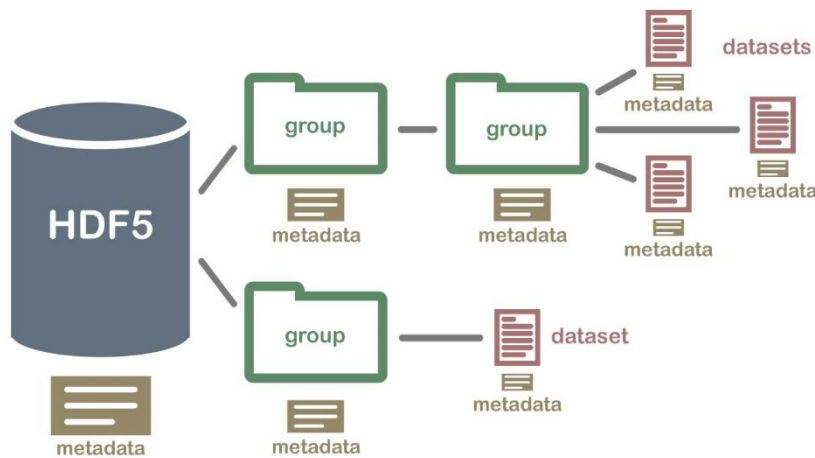
V počiatku spracovávaní boli preskúvané možnosti zobrazovania grafov pomocou jazyka *Python*. Existuje mnoho knižníc umožňujúce pracovať s grafmi ako napríklad *Matplotlib*, *Seaborn*, *Plotly* alebo *PyQtGraph*. Po porovnaní zložitosti syntaxe, možnosti prispôsobenia plynulosti a rýchlosti spracovania, sa ako najvhodnejšia možnosť javila knižnica *PyQtGraph*. Na druhom mieste bola knižnica *Matplotlib*, v ktorej bol spočiatku zobrazovaný spektrogram. Nakoniec však knižnica *PyQtGraph* prevýšila výhody nad nevýhodami a všetky okná sú zobrazované v tejto knižnici. Pre prácu s vektormi a hodnotami dát, bola použitá knižnica *NumPy* a pre spracovanie signálu, FFT a STFT knižnica *SciPy* a *NumPy*. Obe tieto knižnice sú napísané v jazyku C a C++ , takže sú spojené výhody týchto jazykov a jazyku *Python*, čo je rýchlosť spracovania, práca s pamäťou počítača a sú eliminované nevýhody ako zložitosť syntaxe a časová náročnosť písania kódu.

4.4.1 NAČÍTANIE DÁT

Výber správneho formátu pre prácu s dátami predstavuje kľúčový faktor v rýchlosti zápisu a čítania dát. Spočiatku bol využívaný formát *.csv* (čiarkou oddelené hodnoty), ktorý má využitie v množstve aplikácií a analýze dát. Pomocou knižnice *Pandas* boli súbory otvárané, ale časová náročnosť rástla s veľkosťou dát. Taktiež bolo náročné dostať sa k popisu daného súboru a meraných kanálov inak nazývané aj metadáta. V domnienke, že vytvorením robustného algoritmu na vynechanie a následné spracovanie iba meraných dát sa rýchlosť zvýši, bol napísaný takýto kód, ktorý dokázal vynechať metadáta na začiatku datasetu, ale bolo náročné zovšeobecniť toto riešenie aj pre iné súbory dát. Rýchlosť to však nezlepšilo. Taktiež nebolo možné jednoducho otvoriť nameraný súbor a skontrolovať jeho hodnoty, alebo metadáta. Preto bol zvolený iný prístup a iný formát dát.

V tomto prípade bolo potrebné nájsť formát, ktorý by umožňoval rýchlejší zápis a čítanie veľkého množstva dát a taktiež uľahčil prácu s popisom kanálov a meraných veličín. Hierarchický dátový formát (HDF5) s koncovkou *.h5* je správnou voľbou pre takýto účel. Tento formát má jednoduchú štruktúru, kde hlavný súbor obsahuje skupiny a tieto skupiny môžu obsahovať buď ďalšie podskupiny alebo len dataseť.

Každá spomenutá úroveň má pri sebe metadáta a je možné ju popísať vid' Obr. 4.3. Preto je HDF súbor seba popisný. Je vhodný pre dáta časovej rady, keďže využíva *data slicing* – to znamená, že nie je potrebné načítať celý súbor do RAM počítača, ale len potrebnú časť na spracovanie. Taktiež využíva rôzne algoritmy pre kompresiu, ktoré značne znižujú potrebné miesto na disku. Je možné paralelne čítať aj zapisovať dáta, keďže môže využiť viacero jadier



Obr. 4.3 Štruktúra formátu HDF5 [36]

procesora naraz. Toto urýchľuje zápis a čítanie *.h5* súborov oproti *.csv*. Pri formáte 1 milión riadkov a 10 stĺpcov je rozdiel pri čítaní formátu *HDF5* pomocou knižnice *NumPy* a *Pandas* približne 6 sekúnd a pri zápise až 9 sekúnd. Taktiež umožňuje zobrazenie datasetu pomocou aplikácie *HDF View*, čo uľahčuje prácu a zvyšuje prehľadnosť. Užívateľ vidí celú štruktúru a obsah súboru vrátane metadát. Otvorenie takéhoto súboru a kontrola dát trvá iba pár sekúnd. Pri formáte *.csv* bolo potrebné využiť program Excel, kde načítanie trvalo značne dlhšiu dobu. [18]

Dáta na spracovanie bolo nutné najprv načítať z pamäte počítača. Pri načítaní dát je potrebné dodržať určitú štruktúru bez ktorej by nebolo možné otvárať súbory z merania. Bola využitá knižnica *h5py*, ktorá je rozhraním formátu *HDF5* pre prácu v jazyku *Python*. Pomocou nej, je možné otvárať a načítavať dáta tohto typu. Pre tento účel bola vytvorená trieda s názvom *process_data* ako je vidieť v nasledujúcej ukážke kódu. Kód bol niekoľko krát v procese vývoja upravený, pre dosiahnutie najvyššej výkonnosti. Niekedy sa súbor otváral viac krát pri každom procese s nameraným súborom, čo bolo z časového hľadiska neúsporné. Aktuálna verzia pracuje s nasledujúcim základným kódom pre získanie dát z merania a uloženie do premennej *dataset*. [36]

```
class process_data():
    def __init__(self, path):
        self.path=path
        self.dataset=None
        self.hdf5_file=h5py.File(self.path, 'r')
```

```

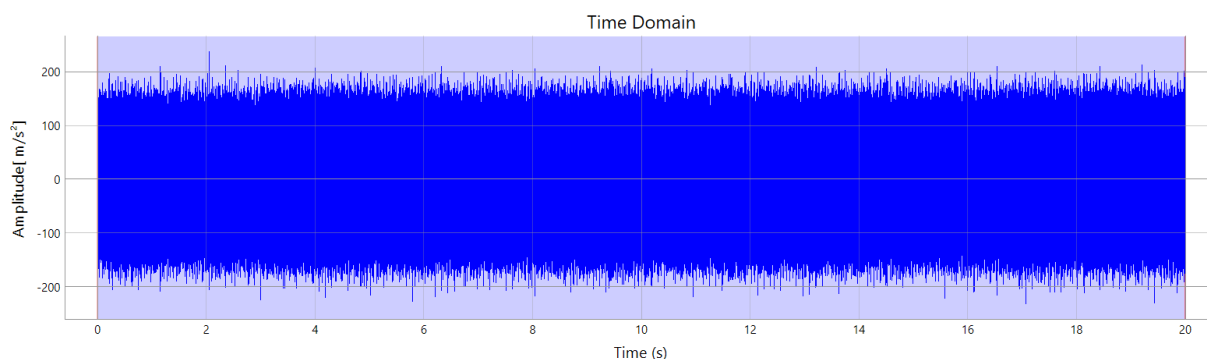
self.group=self.hdf5_file['Table1']
def get_data(self,name_of_dataset):
    if name_of_dataset in self.group:
        self.dataset = self.group[name_of_dataset]
        print(f"Dataset '{name_of_dataset}' opened successfully.")
    else:
        print(f"Dataset '{name_of_dataset}' does not exist.")
    if self.dataset is not None:
        data=np.array(self.dataset)
        return data
    else:
        print('"Dataset is not available.")

```

Táto trieda obsahuje ešte ďalšie funkcie napríklad na získanie názvu všetkých kanálov, získanie metadát, vzorkovacej frekvencie, otáčok alebo teoretických frekvencií ako bude vysvetlené neskôr. V prvom riadku je definovaná trieda a premenná *path*, do ktorej užívateľ zadá adresu cesty k danému súboru. Je definovaná aj premenná *dataset*, ktorá slúži na uchovanie informácie o aktuálnom spracovávanom datase pre ďalšie funkcie, a tak sa nemusí súbor znova otvárať. Na začiatku je jej hodnota nastavená na *None*, pre správnu funkciu, keďže ešte nemá odkiaľ zobrať dáta. Otvorenie súboru je realizované pomocou funkcie *h5py.file* so vstupom *self.path*. Vstup je definovaný užívateľom, pre maximálnu mieru univerzálnosti softvéru. Následne sa načítajú dáta z konkrétnej skupiny, v tomto prípade je to iba jedna s názvom *Table1*. Takto sa pripraví dáta zo súboru a sú pripravené všetky kanály merania. Prvou funkciou je *get_data*, ktorá berie argument *name_of_dataset*. Užívateľ si zvolí, ktorý kanál chce spracovať a výsledkom je vektor dát *data* vo formáte *np.array* pre lepšiu manipuláciu s dátami. To sa však stane len vtedy ak taký kanál existuje a prejde cez zadanú podmienku existencie takéhoto datasetu.

4.4.2 ČASOVÁ DOMÉNA

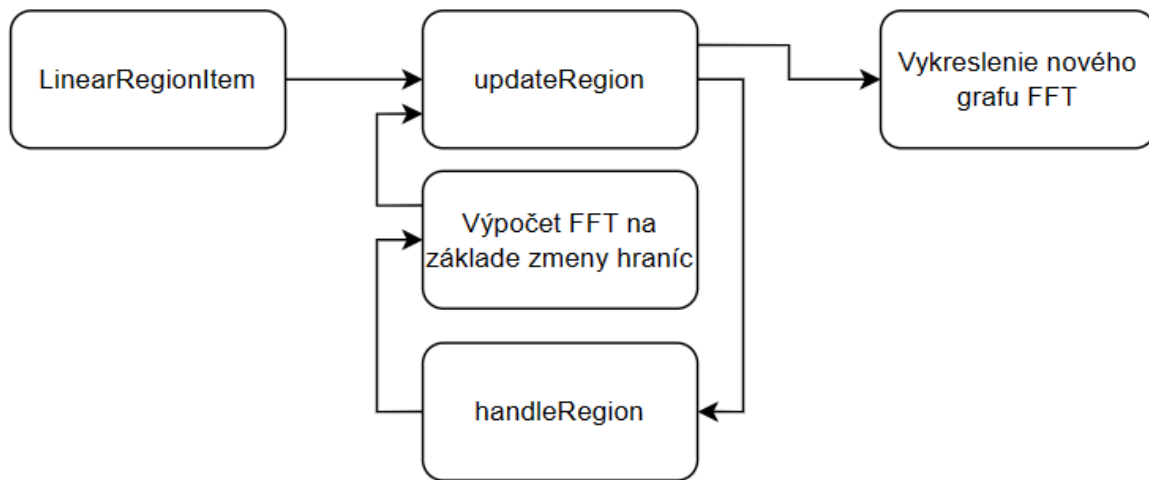
Pri vytváraní grafu frekvenčnej domény bolo od začiatku jasné, že sa využije knižnica *PyQtGraph*. Dovoľuje plynulé priblíženie signálu, jeho natiahnutie v smere x aj y a tým umožní prehľadnejšie zobrazenie. Na Obr. 4.4 je vidieť príklad zobrazenia signálu v čase. Okrem samotného zobrazenia časovej domény, slúži toto okno aj na vymedzenie hodnôt pre výpočet FFT. Slabo modrá farba na pozadí značí, že je braná celá oblasť. Je to tak nastavené



Obr. 4.4 Zobrazenie signálu v čase z pripravovaného softvéru

ako počiatočné zobrazenie. Posunutím sa dá ľubovoľne v časovom rozmedzí signálu nastaviť okno v ktorom sa FFT bude vykonávať. Bola využitá funkcia *LinearRegionItem*, ktorá zbiera a odosiela aktuálnu polohu koncových bodov tohto okna. Kvôli tomuto účelu boli vytvorené 2 funkcie *updateRegion* a *handleRegion*. Ich úlohou je komunikovať s výberom v oblasti časového okna, získať hodnoty indexov na začiatku a na konci. Podľa toho priradiť hodnoty

dát. Následne sa vybrané dáta spracujú pomocou FFT. Funkcia *handleRegion* slúži len ako sprostredkovateľ medzi samotným *LinearRegionItem* ako je vidieť na Obr. 4.5.

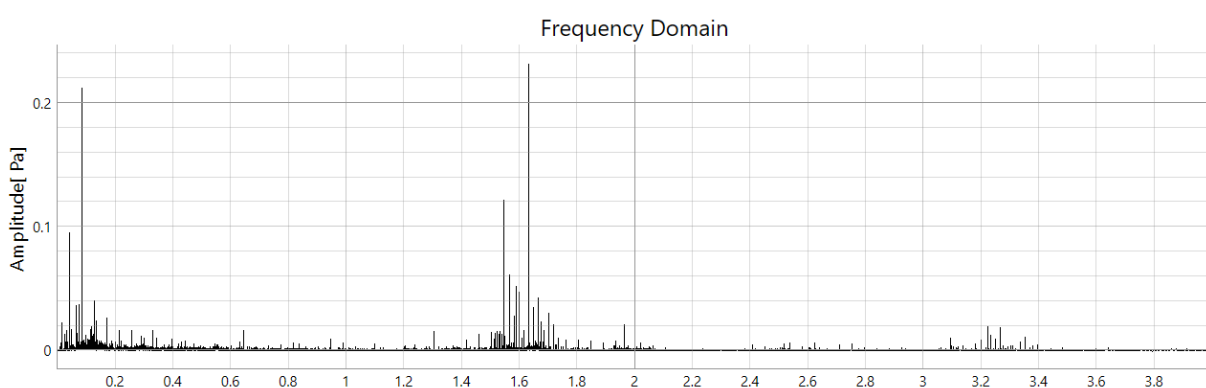


Obr. 4.5 Schéma funkcie výberu dát z časového okna

Bolo potrebné spojiť túto funkciu s ďalším oknom – frekvenčnou doménou. Posúvaním sa neustále prepočítava frekvenčné spektrum a zobrazuje jeho nové hodnoty, čo môže mať význam pri neustálených dejoch. Nevýhodou je, že toto neustále prepočítavanie spomaľuje program, ale len pri veľkých dátach. V tom prípade by bolo vhodné využiť iné riešenie, ktoré najprv nastaví polohu okna a až potom prepočíta frekvenčné spektrum. Pri veľkosti dát 1.6 GB program síce reaguje, ale je značne spomalený. Toto meranie je z dobehu prevodovky a má dĺžku približne 360 sekúnd. Vtedy však samotné frekvenčné spektrum nemá zmysel, keďže sú zobrazované frekvencie z celej doby merania a tieto frekvencie sa časom menia. Preto je vhodné pozerat' sa na spektrogram a frekvenčné spektrum využiť len v malom časovom rozmedzí a využiť možnosť posúvania sa pozdĺž časovej osy. Môže to byť praktické pri sledovaní určitého časového výseku v spektrograme a porovnávať tieto hodnoty vo frekvenčnej doméne vo väčšom detaile.

4.4.3 FREKVENČNÁ DOMÉNA

Okno frekvenčného spektra vid' Obr. 4.6 je výstupom funkcie s názvom *dbfft*, do ktorej vstupuje vektor časového signálu. Ako už bolo spomenuté, tento vektor sa mení podľa toho,

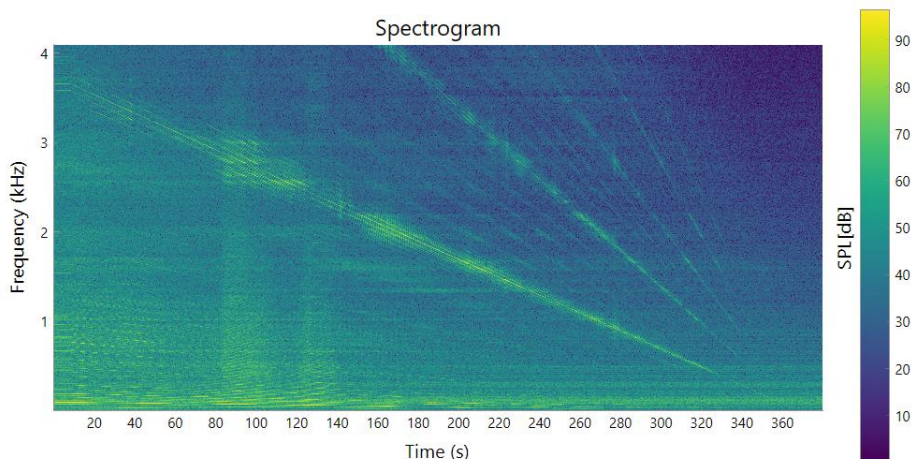


Obr. 4.6 Zobrazenie signálu v vo frekvenčnom spektre z pripravovaného softvéru

kde sa momentálne nachádza rozmedzie okna *LinearRegionItem*. Výstupom tejto funkcie je vektor frekvencií a vektor amplitúd. Výsledným grafom je závislosť hodnoty amplitúdy na frekvencii. Pomocou algoritmu rýchlej Fourierovej transformácie, ktorá je súčasťou knižnice *NumPy* sú získané tieto vektory. Bolo potrebné dať si pozor na nasledujúce problémy. FFT počíta reálne ale aj komplexné hodnoty, ale pre tento účel je dostačujúce pracovať len s reálnymi. V knižnici *NumPy* existuje jednoduché riešenie, ktoré túto skutočnosť obchádza a sú získané len reálne hodnoty pomocou funkcie *rfft*. Má to za následok o polovicu menej výpočtov a tým pádom ešte rýchlejšie spracovanie dát. Taktiež je potrebné, aby boli vektor hodnôt amplitúdy a vektor s hodnotami frekvencií rovnako dlhé, inak by došlo k chybe pri vytváraní grafu. Tento vektor je získaný funkciou *rfftfreq*. Vstupmi do tejto funkcie sú dĺžka vektora spracovávaných dát a perióda vzorkovania – prevrátená hodnota vzorkovacej frekvencie. [35]

4.4.4 SPEKTROGRAM

Knižnica *Matplotlib* umožňuje jednoduché spracovanie spektrogramu doslova v jednom riadku kódu. Je nutné iba zadať vektor dát a vzorkovaciu frekvenciu, ostatné parametre

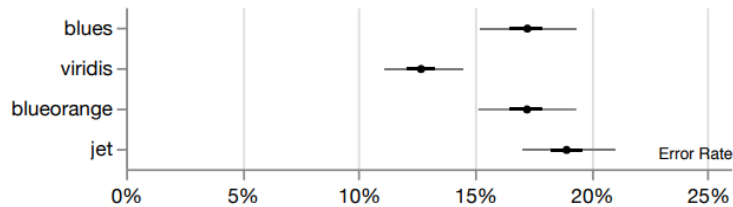


Obr. 4.7 Zobrazenie signálu v spektrograme z pripravovaného softvéru

nie sú povinné, avšak pre správne zobrazenie sú odporúčané. Taktiež je možné si vybrať zo širokej škály farebných máp. Pre problémy s veľkosťou okna a mierky osy Z – intenzity, bolo nakoniec potrebné využiť knižnicu *PyQtGraph*. Táto knižnica je pre prácu s veľkým množstvom dát rýchlejšia a miera prispôsobenia je taktiež väčšia. Má nevýhodu v tom, že dokumentácia kódu nie je tak jasná ako v druhom prípade a taktiež nie je tak rozšírená medzi užívateľmi a vo vedeckej sfére ako *Matplotlib*. Kvôli tomuto bola práca zložitá a vyžadovala značne viac času. [33],[34].

Na druhej strane však bolo dosiahnuté to, že graf sa pri pohybe v okne prispôsobuje a je stále vidno celý jeho obsah. S kolieskom myši je možné plynule približovať a oddiaľovať zobrazenie. Túto funkciu neumožňovala knižnica *Matplotlib* a pre analýzu signálu je to značné obmedzenie. Toto boli hlavné problémy implementácie grafu z *Matplotlib*. Taktiež sa pri spracovaní signálu z mikrofónu podarilo pridať logaritmickú os s referenčným tlakom 20 μPa vid' Obr. 4.7, čo neumožňovala knižnica *Matplotlib*, keďže mala už dopredu stanovené možnosti zobrazenia na osách. Je tu využitá iná farebná mapa – *viridis*, oproti farebnej mape

rainbow. V štúdiu sa zistilo že mapa *viridis* má najmenšiu mieru chybovosti a farebná mapa *jet*, ktorá má podobné vlastnosti ako *rainbow*, má naopak najvyššiu mieru chybovosti vid' Obr. 4.8. Problémom je ľudské vnímanie farieb a kontrastu medzi farbami. Farebná mapa *jet*,



Obr. 4.8 Chybovosť jednotlivých farebných máp [19]

alebo *rainbow* mátie užívateľa nedostatočnou usporiadanosťou a tým zakrýva údaje, kvôli nekontrolovanej variácii jasú. Prechod z tmavej farby na jasnejšiu má byť plynulý, to sa však nedá dosiahnuť pri farebnej škále typu *rainbow* a *jet* ako je vidieť na Obr. 4.9 hore, kde je farebná mapa *rainbow* a dole *viridis*. [19] [20]



Obr. 4.9 Farebná mapa *jet* (*rainbow*) hore a *viridis* dole

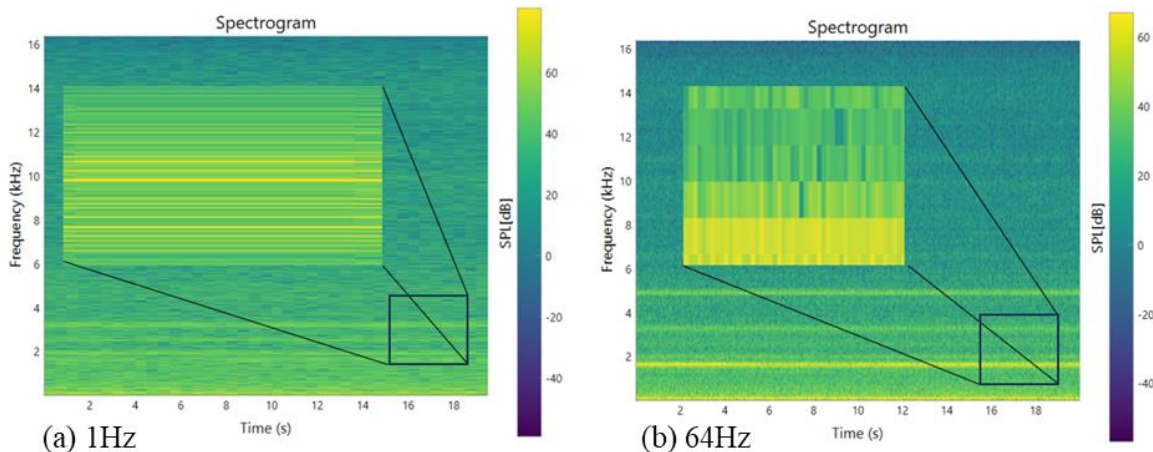
Pri vytváraní funkcie pre zobrazenie spektrogramu *plot_spectrogram*, bola využitá aj knižnica *SciPy* a konkrétne funkcia *spectrogram*, ktorej výstupmi sú vektor frekvencií, vektor času a dvoj dimenzionálny vektor *Sxx*, ktorý obsahuje jednu hodnotu amplitúdy pre dvojicu frekvencia a čas. Vstupmi sú vektor hodnôt zo signálu a vzorkovacia frekvencia. Bolo potrebné nastaviť mód na „*magnitude*“ aby boli získané absolútne hodnoty veľkosti amplitúdy vo vektore *Sxx* ako je vidno v ukážke kódu nižšie.

```
def plot_spectrogram_pyqt(self):
    #Počet vzoriek pre FFT - kontroluje sa tým frekvenčné rozlíšenie
    nfft=self.fs[0]/self.resolution
    #Výpočet STFT pre spektrogram
    f, t, Sxx =
signal.spectrogram(x=self.x, fs=self.fs, window=np.hanning(nfft), nfft=nfft, no
verlap=nfft/2, mode="magnitude")
    # Interpretácia obrazových údajov ako riadkových namiesto stĺpcových
    pg.setConfigOptions(imageAxisOrder='row-major')
    # Vloženie obrazu do grafického okna
    img = pg.ImageItem()
    self.spectrogram.addItem(img)
    # Pridanie histogramu kvôli farebnej mape
    hist = pg.HistogramLUTItem()
    # Spojenie histogramu s obrazom
    hist.setImageItem(img)
    hist.gradient.loadPreset(self.cmap)
```

Tieto hodnoty sú následne načítané a zobrazené ako obraz. Bez zvolenej farebnej mapy, by boli hodnoty iba čierno-biele a nebolo by jasne vidieť rozdiely v hodnotách, preto je pomocou funkcie *HistogramLUTItem* načítaná farebná mapa *viridis* a následne zobrazená spolu

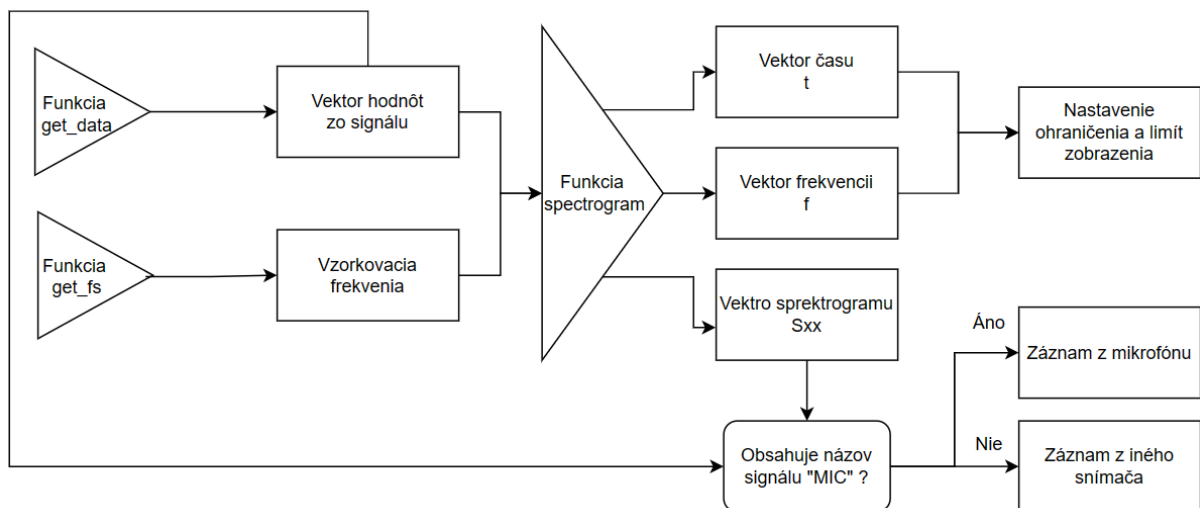
s mierkou z funkcie *ColorBarItem*. Podľa hodnoty amplitúdy sa farba mení a z tmavo modrej cez zelenú až po najvyššie hodnoty kde je vidieť svetlo žltú farbu.

Podstatou spektrogramu je vytvorenie obrazu pomocou pixelov v tvare obdĺžnika. Kľúčovou vlastnosťou je možnosť meniť rozlíšenie zobrazovania frekvencií. To je dosiahnuté práve veľkosťou jedného pixela v obraze. Výška tohto pixela udáva frekvenčné rozlíšenie a je kontrolované pomocou počtu vzoriek spracovaných v jednom výpočte FFT. Ak je počet rovný veľkosti vzorkovacej frekvencie, rozlíšenie je 1 Hz. Voľbou rozlíšenia v tvare 2^n teda mocniny čísla 2 je možné získať rozlíšenie napríklad 8, 32 alebo 256 Hz. Mocniny čísla 2 sú zvolené, ako už bolo spomenuté, kvôli rýchlosti výpočtu FFT. Práve pri mocninách čísla 2 je



Obr. 4.10 Vplyv rozlíšenia frekvencií na časové rozlíšenie

výpočet najefektívnejší. Druhý rozmer pixelu, jeho dĺžka, udáva časové rozlíšenie. To nie je možné v tomto softvéri priamo zvoliť. Ako už bolo vysvetlené v kapitole 3.3.5 existuje určitý pomer medzi presnosťou zobrazenia času a frekvencie. Ak je potrebné vidieť frekvencie vo vysokom rozlíšení je zvolená hodnota rozlíšenia napríklad 1, ale to má za následok, že nie je



Obr. 4.11 Schéma funkcie na posúdenie obsahu signálu

vidieť rýchle zmeny frekvencie v čase. Na Obr. 4.10 je vidieť rovnaký signál s rôznym rozlíšením. V spektrograme s frekvenčným rozlíšením 1 Hz je pomerne jasne vidieť rozťahnuté obdĺžniky s časovým rozlíšením len 0.5 s. Oproti tomu spektrogram vedľa s frekvenčným rozlíšením 64Hz má časové rozlíšenie až 0.008 s.

Ďalej bolo potrebné rozlíšiť, či sa jedná o signál z mikrofónu alebo nie, keďže pri zvuku je vhodné zobrazit' dáta ako hladinu akustického tlaku v decibeloch. Na to bolo potrebné skontrolovať názov kanálu, ako je vidieť na Obr. 4.11. Ak názov obsahoval „MIC“ tak sa dáta spracovali podľa algoritmu pre hladinu akustického tlaku ak nie, tak sa použila klasická logaritmická mierka. Alternatívou by bolo sledovanie jednotky zo snímača, ak by obsahovala Pa ako Pascal, tak sa jedná o signál z mikrofónu. Následne sa dáta pomocou zvolenej mierky upravujú a pošlú sa ako argument do funkcií na vkladanie obrazu spektrogramu a obrazu mierky vedľa neho.

4.5 DIAGNOSTIKA

Pridanou hodnotou tohto softvéru je zobrazovanie teoretických frekvencií, získané numerickým výpočtom a taktiež zobrazenie vlastných frekvencií meranej štruktúry. Princíp diagnostiky sa skladá z nasledujúcich častí:

4.5.1 OTÁČKY

Pre výpočet aktuálnych teoretických frekvencií bolo potrebné získať otáčky z daného merania. Poskytnuté vzorové dáta obsahujú kanál s názvom *DS01-TACHO*, kde sa nachádza signál z optickej sondy merania otáčok. Keďže sa jedná o čistý signál v podobe hodnoty napätia, bolo potrebné vymyslieť funkciu na získanie otáčok z tohto signálu.

Úlohou funkcie je dostať sa k perióde otáčania. Tá je získaná tak, že je odčítaná hodnota vyznačená medzi dvomi výchylkami napätia v signáli vid' Obr. 4.12. Pre ľudské oko je to pomerne jednoduchá úloha. Pre zvolenú hranicu napätia, je odčítaná hodnota času prvej výchylky, tak isto sa odčíta hodnota napätia v tretej výchylke v poradí. Je to tak z dôvodu podstaty funkcie sondy. Prvý skok značí začiatok zaznamenania reflexnej nálepky, druhý jej koniec a až tretí skok značí ďalší začiatok reflexnej nálepky a práve vtedy sa vykonala jedna celá otáčka. Týmto je získaná perióda otáčania

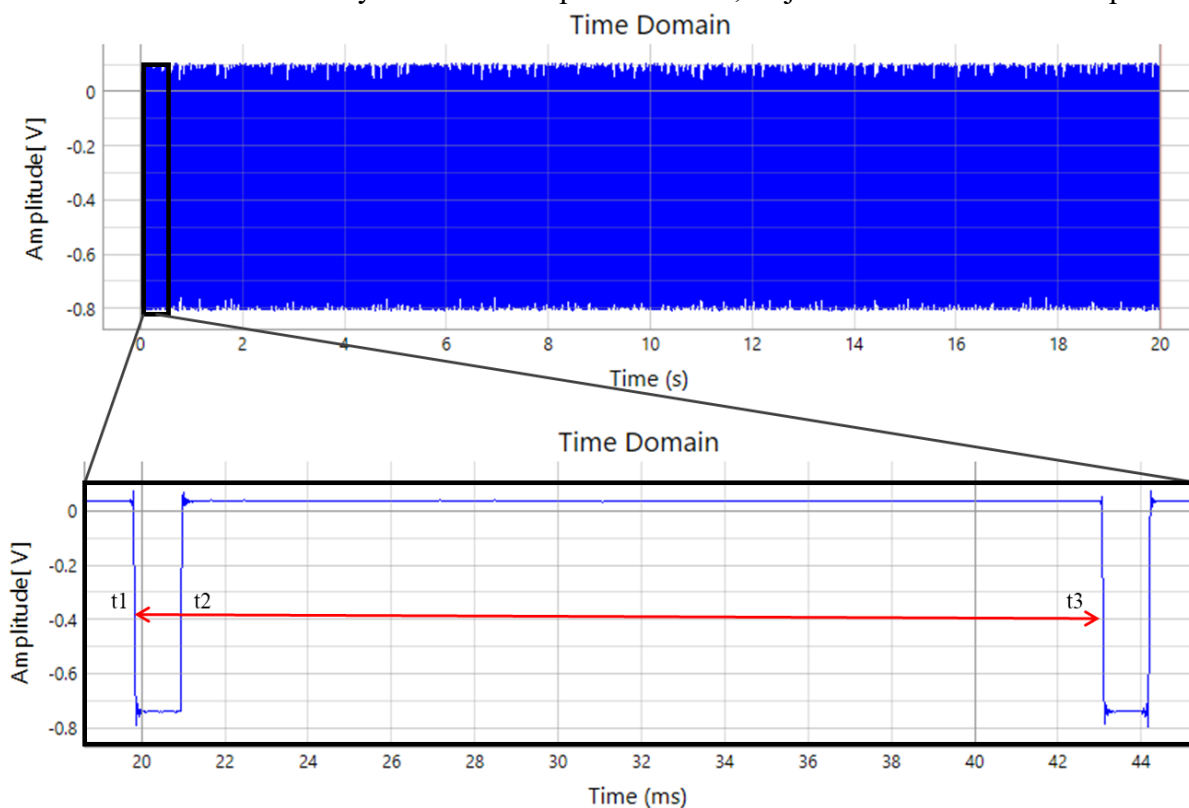
$$T = t_3 - t_1, \quad (41)$$

Kde t_3 je čas v bode 3 a t_1 čas v bode 1.

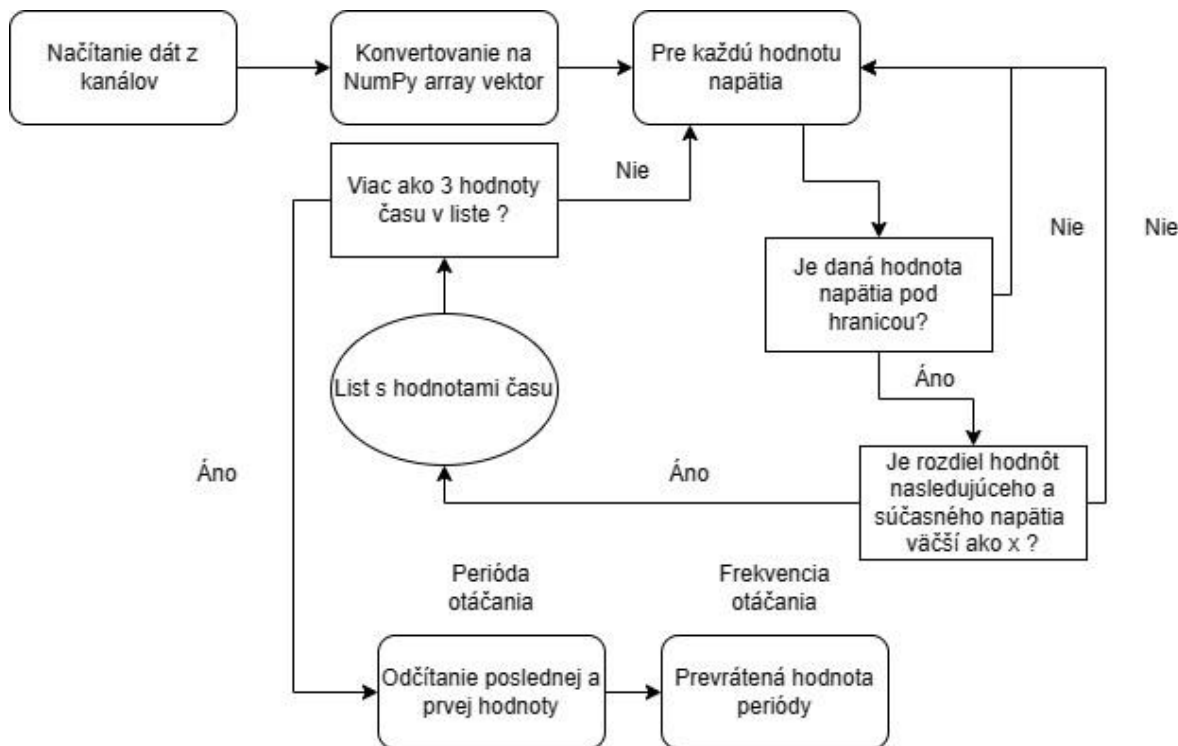
Prevrátenou hodnotou je získaná hodnota frekvencie otáčania v Hz, ktorá je použitá v ďalších výpočtoch.

Tento zdanlivo jednoduchý postup bolo potrebné zovšeobecniť a premeniť na kód, ktorý zaručí, že prepojí hodnoty napätia v danom bode s hodnotou času. Ako je vidieť na schéme logiky kódu vid' Obr. 4.13, najprv bolo potrebné načítať potrebné kanály a to konkrétne kanál času a tachu sondy. Taktiež z dôvodu lepšej manipulácie a operáciami s dátami sa previedli dáta do vektora pomocou knižnice *NumPy*. Následne sa v slučke skontrolovali hodnoty napätia. Ak bola daná hodnota nad hranicou, nasledovala ďalšia hodnota až kým neprišla na rad hodnota napätia, nachádzajúca sa pod červenou čiarou z Obr. 4.12 Táto hodnota musela

splniť ešte ďalšiu podmienku. Aby sa neukladali všetky hodnoty napätia v čase prechodu reflexnej nálepky, rozdiel hodnôt predchádzajúcej a aktuálnej hodnoty napätia musí byť väčší ako limitná hodnota. Vtedy sa uloží iba prvá hodnota, najbližšie k času začiatku prechodu



Obr. 4.12 Signál z otáčkovej sondy



Obr. 4.13 Schéma logiky algoritmu na získanie otáčok zo signálu otáčkovej sondy

reflexnej nálepky, posledná hodnota kedy sa reflexná nálepka ešte nachádza v dosahu senzora a následne zase prvá hodnota, kedy sa nálepka znova objaví. Tieto tri hodnoty sú zaručené treťou podmienkou, ktorá kontroluje maximálny počet troch hodnôt v liste. Ak je dosiahnutý tento počet, znamená to, že bola vykonaná jedna celá otáčka a môže sa vykonať spomínané odčítanie hodnôt času pre získanie periódy a obrátenou hodnotou je získaná frekvencia otáčania.

4.5.2 VÝPOČET TEORETICKÝCH FREKVENCÍ

Táto časť softvéru zahŕňa vzťahy na výpočet frekvencií prejavu vád ložísk a zubových frekvencií z literatúry, vysvetlené v kapitolách 1.2 a 1.3. Bolo potrebné prepojiť užívateľské rozhranie s funkciou na výpočet frekvencií a následne uložiť tieto hodnoty aj s príslušným názvom frekvencie. Počítajú sa tu frekvencia prejavu vady vonkajšej dráhy, frekvencia prejavu vady vnútornej dráhy, frekvencia prejavu vady valivého elementu, frekvencia otáčania klietky a zubová frekvencia ozubených kolies. Pre prácu s výslednými frekvenciami bol zvolený dátový typ *slovník (dictionary)*. Umožňuje zaznamenať dáta v podobe kľúču, ku ktorému je priradená hodnota. V tomto prípade je to názov frekvencie a jej hodnota.

4.5.3 NAČÍTANIE DÁT Z EXPERIMENTU

Ak si užívateľ praje zobrazit' aj iné frekvencie získané iným spôsobom ako výpočtom, je možné ich zapísať do pripravenej tabuľky v programe Excel. Bol zvolený tento typ programu, keďže je veľmi rozšírený a pre vkladanie číselných údajov je rýchlejší a pohodlnejší pre užívateľa ako vkladanie čísel do samotného užívateľského rozhrania. Táto funkcia je využiteľná napríklad pre porovnanie získaných hodnôt vlastných frekvencií z MKP výpočtov alebo z experimentu. Takto budú frekvencie zobrazené spolu s reálnymi nameranými dátami. V tejto verzii softvéru je možné zadať ľubovoľný počet frekvencií avšak len pre 3 rôzne telesá.

Na prácu a manipuláciu s dátami bola využitá knižnica *Pandas*, ktorá je využívaná v oblasti dátovej analýzy. Bola využitá preto, lebo na tento účel je dostatočne rýchla, je jednoduchá na ovládanie a pomocou pár riadkov je možné načítať súbor a manipulovať s jeho obsahom. Pre správny vstup dát bolo potrebné predísť načítaniu dát bez hodnoty s označením NaN.

4.5.4 FILTROVANIE FREKVENCÍ

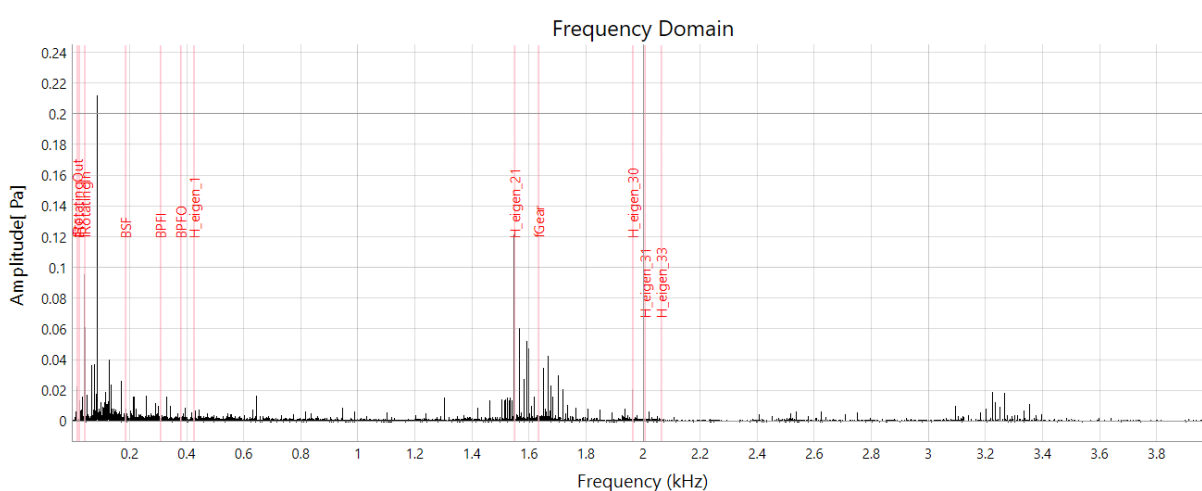
Po načítaní externých dát z formátu *.xlsx* bolo potrebné vyselektovať iba také frekvencie, ktoré sú relevantné a to nasledovne. Najprv sa získala maximálna hodnota amplitúdy sledovaného signálu vo frekvenčnej doméne. Prvým filtrom prešli hodnoty frekvencií, ktoré sú väčšie ako 1% z amplitúdy daného kanála. Najprv boli získané ich indexy poradia v dátovom type *list* a následne podľa týchto hodnôt vybrané a uložené frekvencie, ktoré majú hodnotu amplitúdy vyššiu ako podmienka. Tým sa vyradili frekvencie s nízkou intenzitou.

Ďalšia časť sa zameriava na zistenie, či sa v určitom okolí vyselektovaných frekvencií nachádza jedna z frekvencií zadanej užívateľom. Tento postup je cyklicky opakovaný pre každú vyselektovanú frekvenciu. Princíp je založený na odčítaní hodnoty frekvencie zo

signálu od hodnoty zadanej užívateľom a porovnaní získanej absolútnej hodnoty s premennou *accuracy*. Táto premenná je definovaná na hodnotu 2 Hz, ale užívateľ vie túto hodnotu prepísať a zvoliť prísnejšie alebo menej prísne ohraničenie. To znamená, že ak je v okolí ± 2 Hz od hodnoty frekvencie zadanej užívateľom dostatočne veľká výchylka, táto frekvencia je relevantná a je uložená pre ďalšie spracovanie. Relevantné frekvencie sú pridané do rovnakej dátovej štruktúry *slovník*, aby bol uložený ich názov a príslušná hodnota frekvencie. Názov spočíva z údajov zapísaného v pripravenej tabuľke Excel. Spočíva zo skratky H, O a P ako *Housing*, *Output shaft*, *Pinion* podľa toho, z ktorého stĺpca bola hodnota vybraná. Ďalej nasleduje skratka *eigen* s významom vlastná frekvencia a číselné poradie tejto frekvencie. Tento postup skladania názvu je tiež zakomponovaný v kóde.

4.5.5 GRAFICKÉ ZOBRAZENIE

Nasledujúci a posledný krok je zobrazenie vypočítaných a získaných frekvencií vid' Obr. 4.14. V knižnici *PyQtGraph* existuje príkaz s názvom *InfiniteLine*. Táto funkcia pridá do grafu čiaru, v definovanej polohe, pod konkrétnym uhlom, so zvolenou farbou a zadaným nápisom. Práve pre tento účel je vhodné mať dáta na zobrazenie uložené v dátovom type *slovník*, pretože pomocou slučky je cyklicky prejdený a pre každú frekvenciu je vytvorená inštancia *InfiniteLine* s inou polohou a názvom definovaným v *slovníku*. V tejto slučke sa každá inštancia uloží a taktiež pridá do grafu. Užívateľ po kliknutí na tlačidlo uvidí zvislé čiary vo frekvenčnom spektre s príslušným názvom danej frekvencie v polohe danou samotnou hodnotou frekvencie. Ostatné spomenuté operácie prebehnú v priebehu pár sekúnd na pozadí.



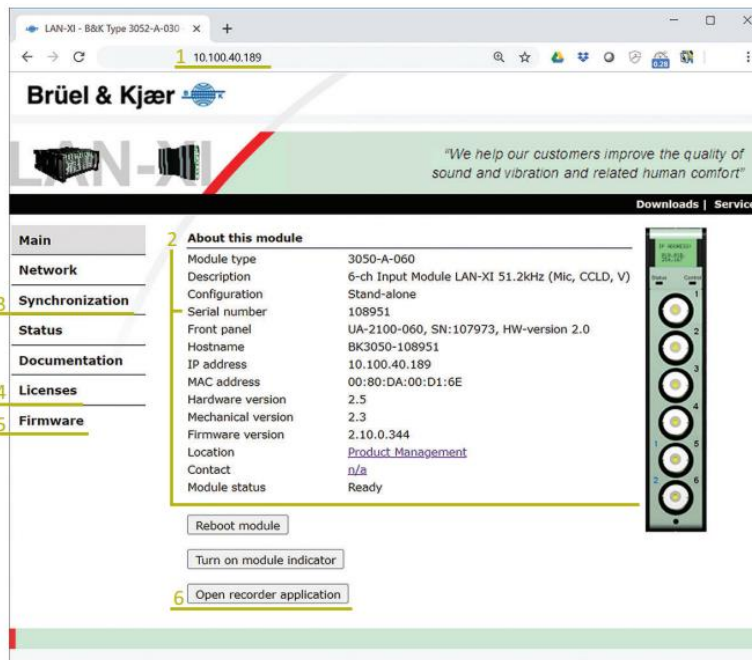
Obr. 4.14 Zobrazenie konkrétnych frekvencií vo frekvenčnom spektre z pripravovaného softvéru

4.6 ONLINE ZBER DÁT

Aby bol tento softvér schopný nahradiť komerčne dostupný program, je neoddeliteľnou súčasťou aj získavanie dát. Kvôli tomu, bolo potrebné nadviazať spojenie s analyzátorom LAN-XI a pomocou vhodných príkazov nastaviť vlastnosti snímačov, miesto pre ukladanie a spustiť samotný zber dát. Detailný popis vývoju je v nasledujúcich podkapitolách.

4.6.1 KOMUNIKÁCIA S ANALYZÁTOROM LAN-XI

Ako už bolo spomenuté, pre komunikáciu s modulom LAN-XI je možné využiť aj webovú domovskú stránku modulu, ktorá obsahuje niekoľko relevantných informácií. Spočiatku nebolo možné pripojiť sa na IP adresu zadanú do prehliadača, ako sa neskôr zistilo, je potrebné mať nainštalované ovládače od HBK alebo len resetovať LAN nastavenia analyzátoru. Potom už PC vie komunikovať cez Ethernet kábel a zobrazí sa rozhranie ako na Obr. 4.15 [17]



Obr. 4.15 Webové rozhranie ovládania analyzátoru [17]

V tomto grafickom rozhraní je vidieť nasledujúce informácie :

- 1) IP adresa – dá sa získať z displeja na module
- 2) O tomto module – základné vlastnosti modulu ako typ, popis, sériové číslo alebo stav používania
- 3) Synchronizácia- udáva informácie o PTP stave – synchronizuje čas
- 4) Licencie – je možné vložiť licencie, v tomto prípade je licencia už dostupná
- 5) Firmvér – je možné aktualizovať verziu firmvéru
- 6) Otvoriť aplikáciu nahrávania- otvorí nahrávanie

Práve bod číslo 6 je najdôležitejší z hľadiska nahrávania a streamovania dát. Po otvorení nebolo možné nastaviť nahrávanie tak, aby sa dáta ukladali do pamäte PC, alebo aby sa dali nastaviť vlastnosti snímačov manuálne. Z tohto dôvodu bolo jedinou možnosťou komunikácia cez Python a príkazy REST. Pomocou manuálu k analyzátoru a open API od firmy HBK a taktiež kódu z GitHub repozitára firmy HBK, bolo možné postupne spustiť nahrávanie a ukladať dáta priamo do PC.

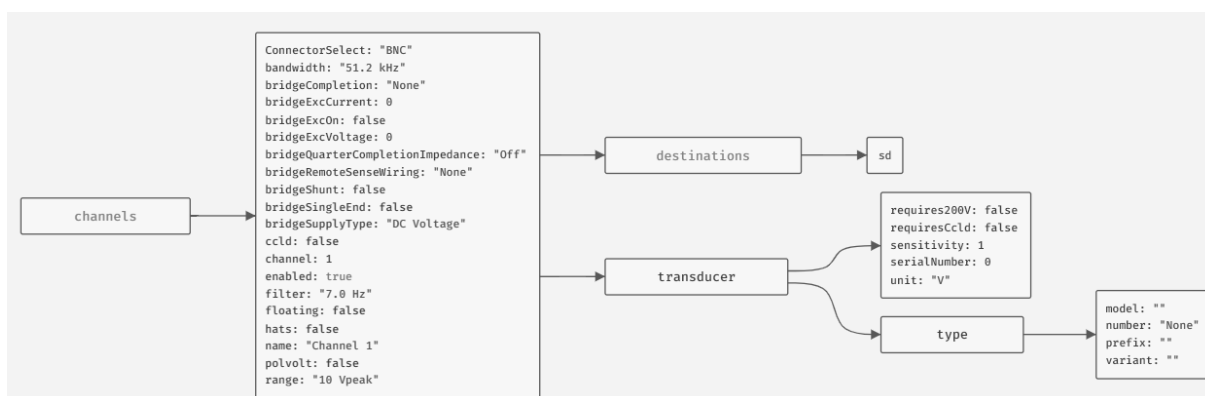
Do prehliadača je možné napísať nasledujúci REST príkaz a informácie zobrazené vo webovom rozhraní sa dajú získať aj v JSON formáte ako výstup tohto príkazu.

`http://169.254.136.229/rest/rec/module/info/`

Prvá časť je IP adresa konkrétneho modulu, nasleduje `/rest/rec/`, ktoré musí byť v príkaze stále prítomné aby bol možný prístup k LAN XI Open API. Ďalej sa cez `/module/info/` dá dostať do objektu modul a ďalej do informácií o ňom. Informácie sú zapísané ako JSON typ súboru, ktorý umožňuje výmenu dát a je čitateľný a prehľadný aj pre človeka, keďže je štruktúrovaný pomocou zátvoriek. [17]

Ďalší dôležitý príkaz nižšie, vráti JSON súbor s informáciami a základným nastavením kanálov analyzátoru a taktiež samotných snímačov. Je to potrebné pre pripojenie snímačov s TEDS - Transducer Electronic Datasheet (digitálny list snímaču s jeho charakteristikou), ale hlavne pre snímače bez TEDS. Keďže analyzátor by nemal bez týchto informácií potrebné dáta pre spustenie nahrávania, nebolo by ho možné spustiť. Preto boli na začiatku vývoja používané iba snímače s TEDS. Časť z JSON súboru môže vyzeráť po formátovaní v online editore nasledovne ako na Obr. 4.16

`http://169.254.136.229/rest/rec/channels/input/default`



Obr. 4.16 Príklad obsahu súboru s nastavením analyzátoru vo formáte JSON, zobrazený pomocou online editora JSON Crack

Na Obr. 4.16 je vidieť názov `channels`, ktorý v skutočnosti obsahuje 4 kanály ale pre jednoduchosť je zobrazený iba jeden. Nasleduje skupina s vlastnosťami kanálu ako napríklad vzorkovacia frekvencia, názov kanálu alebo hodnota `True/False` pre povolenie aktivity kanálu. Ďalej sa v skupine nachádza `destinations` a nastavuje sa tam, kam budú dáta odosielané, ako je vidieť v základnom nastavení je to pamäťová karta a nie sieťový adaptér. Poslednou skupinou je samotný snímač, ktorý obsahuje podskupiny s vlastnosťami. Najdôležitejšou z nich je `sensitivity`, kde sa nastavuje citlivosť snímača. To má za následok následnú získanú hodnotu amplitúdy signálu, keďže hodnota je prepočítavaná.

Keďže firma HBK pripravila skripty pre nadviazanie komunikácie a získanie dát, tento kód bol využitý a upravený pre konkrétne potreby. V ukážke kódu od HBK je vidieť nastavenie a získanie IP adresy so základným príkazom automaticky.

```

ip_addr = script_config['addr'].split("%")[0]
addr_family = socket.getaddrinfo(ip_addr, port=0)[0][0]
base_url = f"http://{ '[' + script_config['addr'] + ' ]' " if addr_family ==
socket.AF_INET6 else f"http://{script_config['addr']}"
base_url = base_url + "/rest/rec"
  
```

Typický postup, pre nadviazanie spojenia a získanie binárnych dát môže vyzeráť nasledovne [17]:

- 1) Pripojenie sa k LAN-XI a spustenie aplikácie Recorder – nahrávanie
- 2) Spustenie detekcie snímačov – automatická detekcia snímačov s TEDS a manuálne zadanie vlastnosti snímačov bez TEDS
- 3) Nastavenie vzorkovacej frekvencie a filtrov
- 4) Nastavenie toku dát do PC, predvolené miesto je SD karta v analyzátoch, preto treba v konfigurácii zmeniť SD za Socket
- 5) Vytvorenie „zásuvky“ (Socket) a prijímanie blokov údajov
- 6) Rozbor hlavičky správy kvôli nájdeniu metadát, popisu kanálu a mierky
- 7) Ukladanie dát do vyrovnávacej pamäte - buffer a odtiaľ vo zvolenom formáte do súboru na disku

4.6.2 SPUSTENIE NAHRÁVANIA

Najprv bolo pripojenie sa k analyzátoch a nahrávanie spúšťané cez jednu triedu - *Recorder* a sériu príkazov z kódu od HBK. Slúžila na pripojenie sa k analyzátoch, spustenie detekcie snímačov s TEDS a vytvorením streamu dát cez sieťový adaptér a ethernetový kábel. Argumentami pre vstup do triedy boli názov pre vytvorenie súboru s meraním, dĺžka snímania, vzorkovacia frekvencia a IP adresa analyzátoch. Celá trieda sa skladala hlavne z príkazov GET pre získanie informácií z analyzátoch alebo PUT pre poslanie informácií do analyzátoch.

Neskôr bolo spozorovaných niekoľko problémov pri tomto postupe.

- Snímače neboli pripravené na meranie od prvej sekundy merania
- Podpora iba snímačov s TEDS

Preto bolo potrebné postup pripájania sa k analyzátoch zmeniť a rozdeliť ho do dvoch fáz. V prvej fáze sa vytvorí spojenie medzi PC a LAN-XI a vymenia sa dáta o aktuálnom nastavení a pripojených snímačoch. Je využitá trieda *Lanxi* predpripravená od HBK. Analyzátoch zostane pripravený na meranie, avšak samotné meranie bude spustené až v ďalšom kroku.

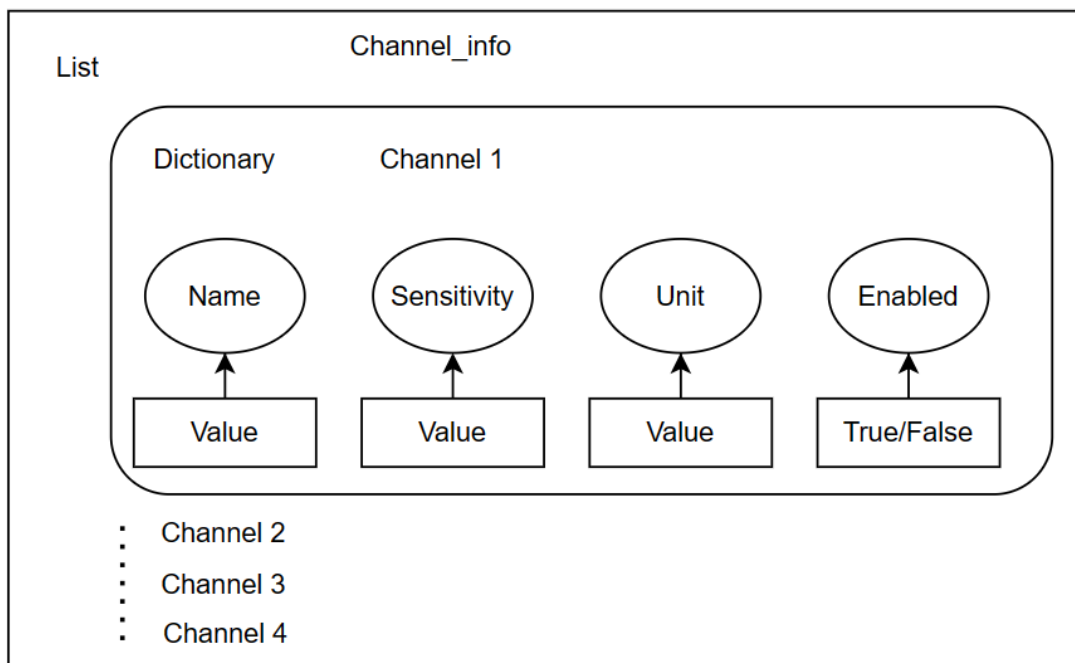
Prvá fáza

Ako už bolo spomenuté vyššie, analyzátoch LAN-XI 3160-A-042 pracuje bez problémov so snímačmi, ktoré obsahujú TEDS a teda po pripojení sa tieto dáta o snímači načítajú a vložia do predpripraveného JSON súboru s počiatočným nastavením merania. Pre úplnú kompatibilitu so snímačmi a zaistenie bezproblémového merania bolo potrebné upraviť softvér tak, aby vedel pracovať aj so snímačmi bez TEDS. Do tejto chvíle nebolo potrebné meniť kód alebo nastavenia, ale kvôli tejto požiadavke bolo nutné hlbšie pochopiť, ako funguje nadviazanie spojenia a štruktúru formátu JSON.

Štúdiom užívateľskej príručky a zdrojového kódu bolo zistené, že hlavnou podstatou komunikácie je JSON súbor so zvoleným nastavením. Ako už bolo vysvetlené v kapitole 4.6.1. tento súbor informácií je možné získať a v tomto prípade bol uložený do premennej *setup*. Podľa vstupu od užívateľa tu je možné prepísať vzorkovaciu frekvenciu a tým zaistiť správne meranie. Taktiež sa tu mení miesto pre ukladanie dát na *socket* teda dáta už prúdia do PC a nie do SD karty. Neskôr bolo zistené, že aj vlastnosti snímaču sa dajú ľubovoľne prepisovať. Tým je možné vložiť informácie o snímačoch bez TEDS aj manuálne. Nasleduje konfigurácia na základe TEDS. Ak je prítomný snímač s TEDS, jeho vlastnosti sú zapísané do JSON súboru a tým je pripravený na snímanie. Tento krok je nadradený manuálnemu vstupu

od užívateľa. Teda ak má snímač TEDS, budú použité tieto dáta a nie od užívateľa. Týmto bol vyriešený problém s TEDS a softvér je schopný pracovať so všetkými pripojenými snímačmi.

Kvôli tomuto účelu bol vytvorený súbor informácií v dátovej štruktúre *list* s vlastnosťami snímačov bez TEDS s názvom *channel_info*. Tieto informácie musí užívateľ manuálne zadať pred začiatkom merania. Tento *list* obsahuje 4 slovníky pre každý kanál. Obsahuje 4 atribúty a to meno merania, citlivosť snímaču, jednotku meranej veličiny a to, či je kanál aktivovaný alebo nie vid' Obr. 4.17.



Obr. 4.17 Štruktúra formátu pre ukladanie informácií o snímačoch

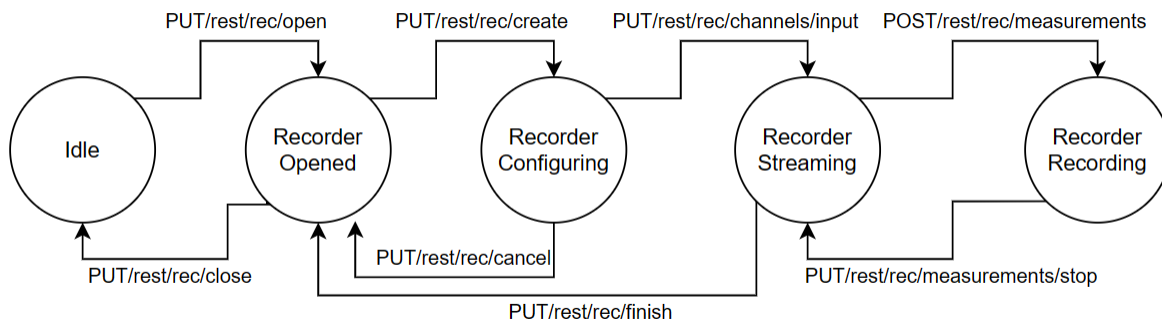
Pomocou jednoduchšej zmeny v kóde bolo dosiahnuté veľké rozšírenie využitia tohto softvéru a približuje sa tak k reálnemu využitiu v praxi. Pridaný kód vyzerá nasledovne.

```
for i,ch in enumerate(self.setup['channels']):
    if self.channel_info[i]['enabled']:
        ch['transducer'] = {
            "requires200V": 0,
            "requiresCclD": 1,
            "sensitivity": self.channel_info[i]['sensitivity'],
            "unit":self.channel_info[i]['unit'] }
        self.setup["channels"][i]["enabled"] = True
        self.setup['channels'][i]['cclD'] = True
        self.setup['channels'][i]['polvolt'] = 0
        self.setup['channels'][i]['name']=self.channel_info[i]['name']
        print(f"Channel {i+1} name is {self.channel_info[i]['name']}")
    else:
        self.setup['channels'][i]['enabled'] = False
```

V súbore *setup* sa nachádza niekoľko podkategórií a to konkrétne 4 možné kanály. Predstavuje ich premenná *ch* a vďaka funkcii *enumerate* je vedený aj počet opakovaní

v slučke *i*. Teda pre každý kanál platí, ak je hodnota *enabled*, zadanej užívateľom, *True*, tak sa nastaví vlastnosti snímača podľa toho, ako ich užívateľ zadal a kanál je aktívny. Taktiež je možné pridať vlastné meno kanálu napríklad *MIC01* ak sa jedná o záznam z mikrofónu, čím sa vyriešil ďalší problém z nasledujúcej kapitoly, kde nebolo možné získať vlastný názov kanálu. Nakoniec sa pre kanály ktoré, nie sú aktívne zmení hodnota pre povolenie kanálu na *False*. Takto upravený JSON súbor je poslaný do analyzátora ako nové nastavenie. Ďalšími príkazmi, je vytvorené spojenie a umožnené prúdenie dát do PC.

Pomocou užívateľskej príručky LAN-XI Open API a kódu od firmy HBK sa podarilo upraviť kód tak, aby sa analyzátor po pripojení k PC dostal z *idle* módu do módu, kedy je pripravený na zber dát ale nahrávanie nezačne ako je vidno na Obr. 4.18. Bolo to docielené tromi príkazmi v tomto poradí */open*, */create* a */channels/input* a vytvorením novej funkcie *connect_to_lanxi*. V triede *Lanxi* bola týmto otvorená aplikácia *Recorder*, v nej vytvorené nové meranie a odoslaná konfigurácia nastavenia analyzátora a pripojených snímačov ako bolo vysvetlené vyššie .



Obr. 4.18 Schéma postupnosti módov analyzátora pri pripájaní sa k snímačom

Analyzátor a snímače sú týmto v režime *RecorderStreaming* vid' Obr. 4.18, kedy je kedykoľvek možné spustiť nahrávanie s danými parametrami. Ak by užívateľ chcel pripojenie prerušiť, bola vytvorená funkcia *disconnect_from_lanxi*. V tejto funkcii sa prejde zo stavu *RecorderStreaming* do *Idle*, pomocou príkazov */finish* a */close*. V tomto stave je možné zmeniť nastavenia alebo vlastnosti snímačov a znova odoslať príkaz na pripojenie sa k analyzátoru cez funkciu *connect_to_lanxi*.

Druhá fáza

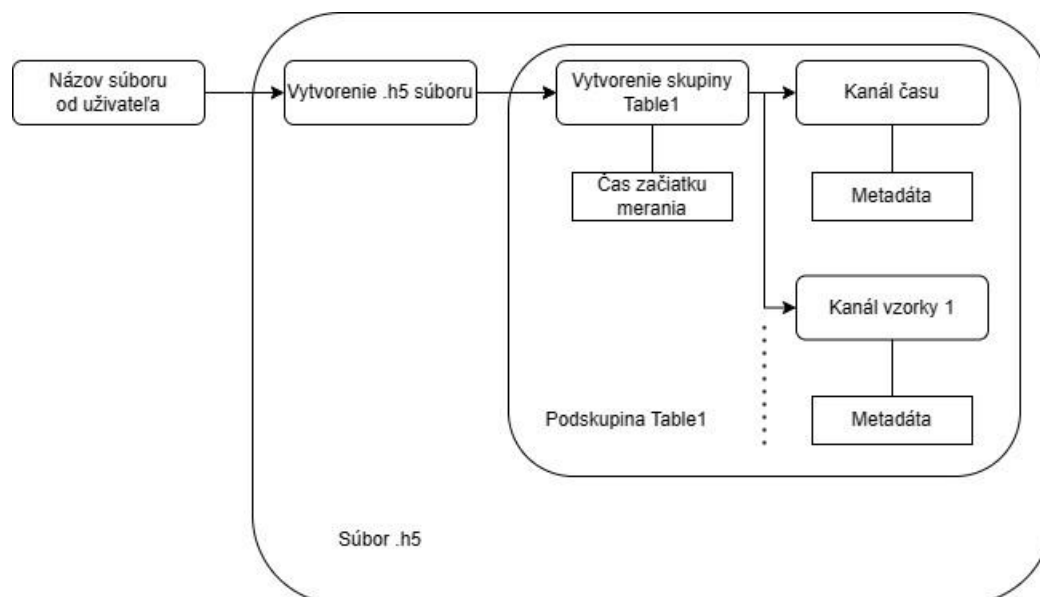
V tejto fáze bola využitá pôvodná trieda *Recorder* s pôvodnými argumentami. Je v nej však už iba spúšťané samotné nahrávanie. Zmenou je aj to, že boli vynechané príkazy na ukončenie streamovania */finish*, */close*, ktoré boli presunuté do funkcie na odpojenie sa od analyzátora. Touto zmenou je možné vytvárať ľubovoľný počet meraní s pripravenými snímačmi až pokiaľ ich užívateľ neodpojí pomocou funkcie *disconnect_from_lanxi*.

4.6.3 UKLADANIE DÁT

Aj pri algoritme na ukladanie dát vznikali časom problémy, ktoré bránili správnej funkcii a využiteľnosti programu. Hlavným z nich boli nasledujúce:

- Získané dáta nezodpovedali reálnym nameraným hodnotám
- Názvy kanálov nezodpovedali zadaným názvom od užívateľa

Pre ukladanie už preložených dát z binárneho formátu na písmená alebo čísla, bolo potrebné na základe poskytnutých dát napísať kód, ktorý vytvorí *.h5* súbor s rovnakým členením skupín a datasetov. Je to súčasť triedy s názvom *Parser* v ktorej sa dáta aj prekladajú. Kvôli možnému následnému spracovaniu poskytnutých vzorových dát ale aj nových nameraných dát pomocou tohto softvéru, je potrebné dodržať rovnakú štruktúru. Na tento účel bola znova použitá knižnica *h5py*. Logika štruktúry je vidieť na Obr. 4.19 a v ukážke kódu nižšie.



Obr. 4.19 Schéma logiky tvorby nového súboru s nameranými dátami

Práve v tejto časti kódu sa skrývali spomenuté problémy. Tým, že bolo zadefinované ako zdroj dát `value["samples"]`, boli zapisované nepripravené „surové“ dáta priamo z analyzátoru. Riešením tohto problému bolo zabezpečiť vhodnú úpravu dát pred ich ukladaním. V užívateľskej príručke bolo vysvetlené ako sa dopracovať k správnym kalibrovaným hodnotám nasledujúcim postupom.

$$kHodnota = Vzorka \cdot scaleFactor \quad (42)$$

kde *kHodnota* je výsledná kalibrovaná hodnota a *Vzorka* je získaná hodnota z analyzátoru.

Pre dosiahnutie správnej funkcie je nevyhnutné vynásobiť dáta pomocou *scaleFactoru*. Tento faktor zohľadňuje rôzne parametre, ako je rozsah vstupného signálu, headroom a citlivosť senzora. Headroom je faktor používaný analyzátorom na ochranu proti clippingu signálu. Je to forma skreslenia signálu, ktorá sa vyskytuje, keď je zosilňovač preťažený a snaží sa dodávať výstupné napätie alebo prúd nad svoje maximálne možnosti. Tento jav môže viesť k strate informácií a deformácii signálu, čo môže ovplyvniť presnosť a spoľahlivosť meraní. Obvykle je to 1,5 dB alebo $10^{\frac{1,5}{20}}$, čo je približne 1,1885. Scale faktor sa vypočíta na základe týchto parametrov a je následne použitý na kalibráciu vzoriek z analyzátoru.

$$scaleFactor = rozsahSignálu \cdot \frac{headroom}{citlivost} \quad (43)$$

Rozsah vstupného signálu je nastavený v tomto prípade na 10V a citlivosť buď podľa TEDS alebo odčítaná užívateľom z kalibračného listu v obale snímača, napríklad $9,755 \text{ mV} / \text{m} \cdot \text{s}^{-2}$ pre snímač zrýchlenia.

Avšak len násobenie dát scale faktorom nie je postačujúce. Následne je dôležité dáta normalizovať, aby sa zabezpečilo ich konzistentné spracovanie a interpretácia, pomocou delenia hodnotou 2^{23} . Táto hodnota je vybraná v súvislosti s použitím 24-bitového dátového formátu, ktorý je v dnešnej dobe široko používaný aj v iných A/D prevodníkoch. Po normalizácii sa hodnoty dát zmenšia na rozsah od -1 do 1. Hodnota 1 reprezentuje maximálnu pozitívnu hodnotu, zatiaľ čo hodnota -1 reprezentuje maximálnu negatívnu hodnotu. Tento normalizačný proces zabezpečuje, že dáta majú konzistentnú škálu. Umožňuje to ich efektívne ukladanie a ďalšie spracovanie, bez straty informácií alebo deformácie signálu. [21] [17]

```
self.active_channels=[]
for channel in self.channel_info:
    if channel['enabled']:
        self.active_channels.append(channel)
hdf5_filename = f'{self.name_of_file}.h5'
with h5py.File(hdf5_filename, 'w') as hdf5_file:
    signal_group = hdf5_file.create_group('Table1')
    signal_group.attrs['start_time'] = str(value["start_time"])
    # Zistí dĺžku nahrávky
    length_of_recording = len(samples) * sample_period
    # Vytvorí časový vektor s požadovanou dĺžkou a pridá ho do datasetu
    time = np.linspace(0, length_of_recording, len(samples))
    signal_group.create_dataset('Ds01-Time', data=time)
    for index, (key, value) in enumerate(data.items()):
        print(f"index nr {index}")
        print(f"key {key}")
        print(f"value {value}")

        samples = value["samples"]
        scale_factor =
value[OpenapiMessage.Interpretation.EDescriptorType.scale_factor]
        scaled_samples = (samples * scale_factor) / 2 ** 23
        # Uloží vzorky
        print(f"Sample nr {index+1} name is
{self.active_channels[index]['name']}")
        samples_dataset =
signal_group.create_dataset(f'{self.active_channels[index]["name"]}',
data=scaled_samples)
        # Uloží metadáta pre každý signál
        # Signal ID
        samples_dataset.attrs['signal_id'] = key
        # Jednotka
        samples_dataset.attrs['InterpretationUnit'] =
self.active_channels[index]["unit"]
        # Vzorkovacia perióda
        channel_info_sampling_period =
self.calc_time(value[OpenapiMessage.Interpretation.EDescriptorType.period_t
ime])
        samples_dataset.attrs['ChannelInformationSamplingPeriod'] =
```

```
np.array([channel_info_sampling_period])
    #scale_factor
    samples_dataset.attrs['InterpretationScaleFactor'] = scale_factor
```

Postup ukladania je nasledovný. Najprv sa vytvorí prázdny súbor s názvom, ktorý zadá užívateľ, potom sa podľa predlohy vytvorí prvá skupina s názvom *Table1*. Tejto skupine sa priradia metadáta.

Aby bolo možné pracovať s dátami časovej rady ako sú tieto, je potrebné mať prehľad o čase. Preto je potrebné vytvoriť vektor času. Na to je nutné poznať dĺžku nahrávky a počet vzoriek. Dĺžka sa zistí vynásobením počtu vzoriek so vzorkovacou periódou tak ako je to vysvetlené v rovnici 31 v kapitole 3.4.3. Pomocou funkcie *linspace* z knižnice *NumPy* je vytvorený vektor hodnôt času od 0 po dĺžku signálu s rovnomerným rozostupom a rovnakým počtom záznamov ako je vzoriek meraného signálu. Nakoniec je uložený do datasetu *Ds01-Time*. Ďalej sa cyklicky prejdú všetky hodnoty zo získaných dát a vzorky sa uložia spolu s príslušnými metadátami do jednotlivých kanálov snímačov.

Tento typ modulu podporuje maximálne 4 kanály vzoriek. Spočiatku nebolo možné, mať možnosť vybrať konkrétny názov kanálu, ale v tejto verzii softvéru sa podarilo získať vstupom od užívateľa vlastné názvy kanálov cez *channel_info*. Vyskytol sa však problém pri pripojení sa k analyzátoru v náhodnom poradí, napríklad ak boli pripojené kanály 1 a 3. Vtedy sa brali hodnoty z kanálu 2, kvôli tomu, že boli povolené iba 2 kanály a teda nesedeli počty iterácií v slučkách. Preto bola vytvorená premenná *active_channels*, a podmienka na vybratie kanálov ktoré sú aktívne. Takto bol zjednotený počet aktívnych kanálov s počtom objektov v *data.items()* kde sa nachádzali namerané hodnoty.

4.6.4 ONLINE ZOBRAZENIE DÁT

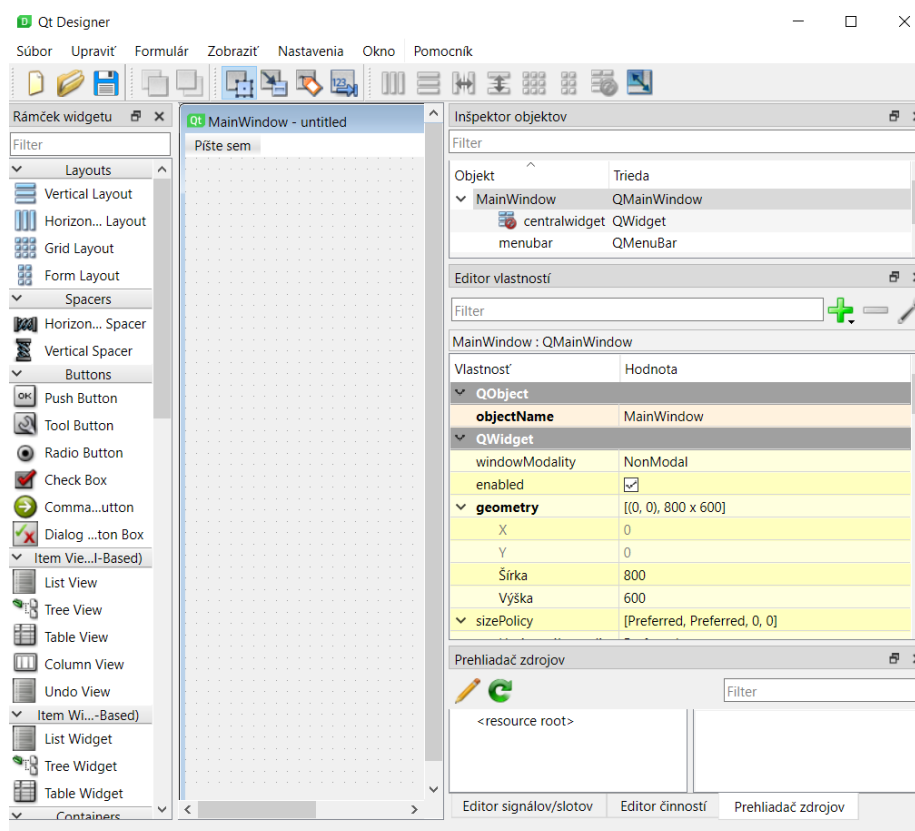
Zmenou zdroja spracovávaných dát bolo možné dosiahnuť zobrazenie grafov v reálnom čase. Tým bolo spracovanie dát rozšírené aj o možnosť online monitoringu, kedy sú dáta brané priamo z analyzátora a nie zo súboru z pamäte počítača. Dáta sa najprv ukladali do takzvaného buffera, kde sa uchovávala na krátku dobu časť dát potrebná na vykonanie výpočtov a následne boli nahradené novými. Takto sa cyklicky menil vstup pre výpočet a tým pádom aj zobrazované hodnoty a vznikla z toho animácia v podobe kontinuálne sa meniaceho obrazu. Bolo potrebné upraviť funkcie na zobrazovanie grafov tak, aby sa aktualizovali ich hodnoty po každom novom výpočte. Toto zobrazenie je obmedzené iba na jeden snímač. Keďže sa jednalo o funkciu, ktorá je nad rámec tejto diplomovej práce, nebola ďalej rozširovaná. Je možné však potvrdiť, že s týmto vybavením sa dajú snímať a zobrazovať dáta aj v reálnom čase.

4.7 UŽÍVATEĽSKÉ ROZHRAŇIE

Keďže cieľom bolo vytvoriť nástroj, pre urýchlenie práce hlavne s nameranými dátami, bolo potrebné vytvoriť ovládanie programu, ktoré je jednoduché a rýchle. Umožní rýchlejšie prehliadanie a voľbu dát v počítači. Taktiež nie je potrebné písať ďalší kód, keďže sa počíta s tým, že užívateľ nemusí vedieť programovať a tým je možné cieľiť na širšiu skupinu ľudí.

Pre tieto potreby existuje v prostredí *Python* niekoľko možností knižníc, najznámejšie z nich sú *Tkinter* alebo *PyQt*. Keďže pri zobrazovaní grafov je použitá knižnica *PyQtGraph* bola zvolená najaktuálnejšia knižnica *PyQt6*. Je možné do nej bez problémov integrovať grafy

a okná z predošlých kapitol. Taktiež je veľkou výhodou, že obsahuje nástroj pre vytváranie grafického užívateľského rozhrania (GUI) s názvom *Qt Designer* vid'. Obr. 4.20. Pri vytváraní GUI je možné vkladať do okna rôzne grafické prvky ako napríklad tlačidlo, nadpis, odklikávací box alebo rolovacie okno. Po vložení a umiestnení týchto prvkov, je vidieť, ako bude pri spustení programu GUI naozaj vyzerat'. Je možné priamo pri vývoji testovať rôzne rozlíšenia a štýly zobrazenia. Výstupom je XML súbor, ktorý by inak bolo potrebné zdĺhavo písať, pretože aj takto je možné upravovať a vytvárať GUI v *PyQt6*, ale z dôvodu úprav do budúcnosti sa to neodporúča. Mohlo by sa stať, že sa naruší členenie a štruktúra tohto súboru. Pri každom spustení softvéru, sa tento text, načíta a vytvorí sa podľa neho predpripravené GUI s danými objektami.



Obr. 4.20 Prostredie programu *Qt Designer* pre tvorbu GUI

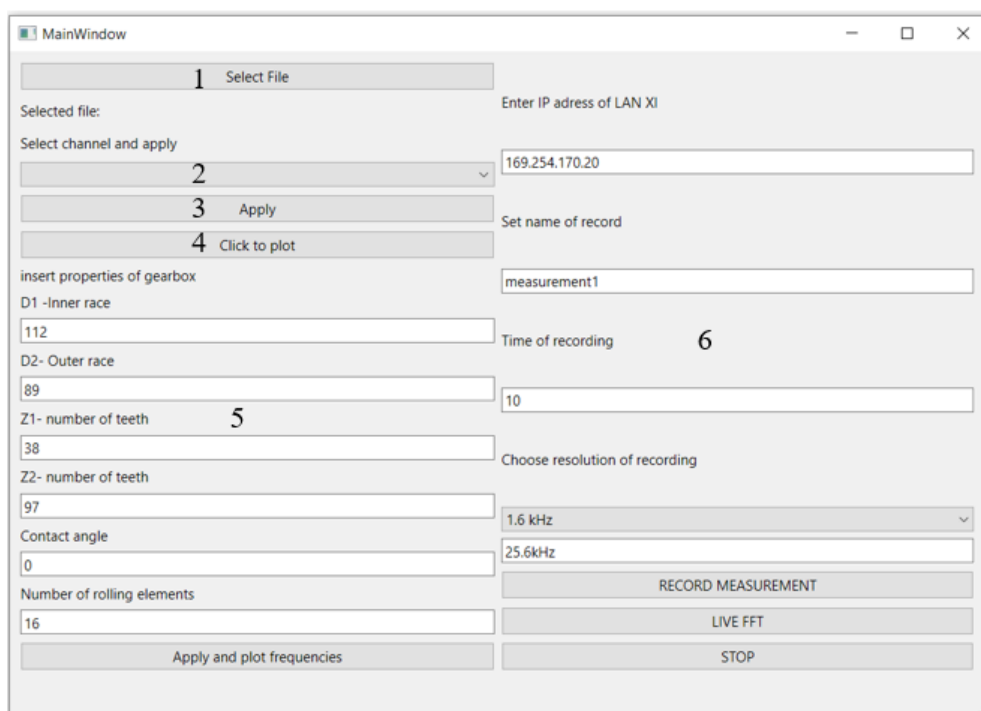
4.7.1 NÁVRH

Pri spustení softvéru sa zobrazí okno, ktoré vyzerá podobne ako na Obr. 4.21, keďže v tejto kapitole sa pojednáva o návrhu a logike GUI, nie je to finálne riešenie. Bolo potrebné myslieť na dobrú funkcionálnosť ale zároveň zväziť možnosti a rozsiahlosť tejto knižnice.

Ako prvé pri spracovaní dát, je potrebné vybrať spracovávaný súbor. Pomocou objektu *QPushButton* bolo vytvorené tlačidlo a bolo prepojené s funkciou, ktorá otvorí prehliadač súborov v danom operačnom systéme. Pre zníženie rizika chyby, je dovolené vybrať iba *.h5* súbory. Po vybraní súboru, bola získaná jeho adresa, ktorá je použitá ako argument do triedy *process_data*. Ďalej sa v tejto triede nachádza funkcia, ktorá prejde vo vybranom súbore

skupiny, ktoré predstavujú jednotlivé kanály merania a uložia sa do dátového formátu *list*. Nakoniec sú tieto informácie, teda text s názvom kanálov pridané do rolovacieho okna (2), kde si užívateľ môže vybrať kanál, ktorý chce spracovať. Podobným spôsobom bolo potrebné pokračovať ďalej, pri tlačidle *Apply* (3), ktoré potvrdí voľbu kanálu a tento názov uloží ako vstup do triedy *Analyser*. Neskôr bolo zistené, že toto tlačidlo je zbytočné a funkciu pre získanie textu z rolovacieho okna a jej uloženie do premennej je možné implementovať priamo do tlačidla *Click to plot* (4), takže v ďalšej verzii softvéru toto tlačidlo už nebude.

Najrozsiahljšou triedou je trieda *Analyser*, ktorá je zodpovedná za príjem číselných dát a vrátenie grafického okna s tromi grafmi. Ako už bolo spomenuté, je prepojená s tlačidlom *Click to plot*. Logika fungovania tejto triedy je nasledovná. V súbore *main.py* sú získané argumenty ako adresa súboru, kanál merania, frekvenčné rozlíšenie spektrogramu, farebná mapa spektrogramu a voľba grafov, ktoré užívateľ chce zobraziť. Týmto spôsobom je možné interagovať medzi GUI a samotnými grafmi, pretože údaje sa medzi funkciami vymieňajú. V rozsahu vyše 300 riadkov kódu sa nachádzajú príkazy pre vytvorenie grafického okna, rozloženie v ňom, samotné grafy, ktoré sú z počiatku prázdne a až neskôr vyplnené údajmi. Je tu zadefinovaný formát písma a nadpisy, ktoré vidí užívateľ a taktiež všetky grafické prvky ako osy, farba pozadia atď. Bolo vybrané rozloženie s funkciou, kde je možné ukazovateľom myši rozšíriť alebo zúžiť zobrazenie grafov. Taktiež tu sú funkcie na vytvorenie spektrogramu



Obr. 4.21 Návrh GUI

alebo prvok výberu ohraničenia v časovej doméne z predošlej kapitoly, ktorých podstata už bola vysvetlená.

Toto užívateľské rozhranie slúži vo všeobecnosti ako vstup pre dáta, ktoré sú ďalej spracované. Ako je vidieť, v oblasti 5 sa vkladajú geometrické rozmery ložísk a počet zubov ozubených kolies prítomných v meranej prevodovke. Tieto čísla slúžia ako vstup do funkcie na výpočet frekvencií prejavu väd a zubových frekvencií popísaných v kapitole 1.1 a 1.2.

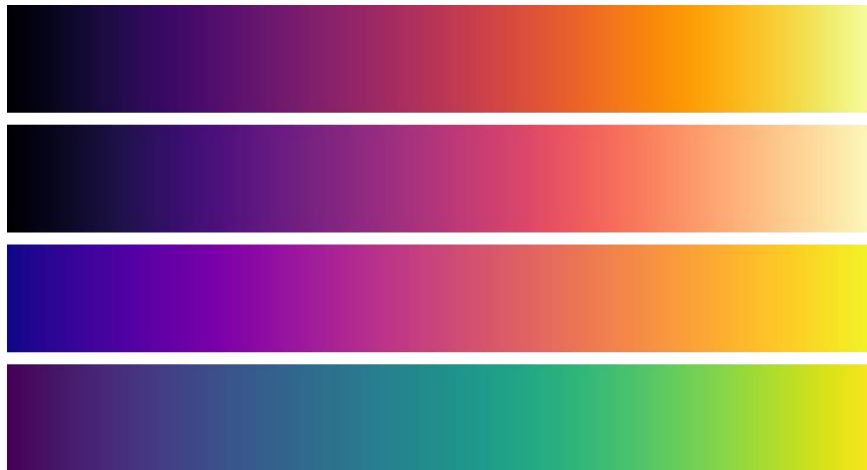
Celá oblasť 6 slúži na nastavenie merania cez analyzátor LAN-XI. V tejto verzii softvéru, ešte nebolo správne vyriešená komunikácia medzi analyzátorom a GUI. Vo finálnej verzii je detailnejšie popísaná aj táto časť GUI.

4.7.2 FINÁLNA VERZIA

Offline časť

Táto verzia softvéru je pripravená na reálne zbieranie dát a ovládanie analyzátoru so snímačmi tak, aby boli dáta využiteľné a mohli byť následne spracované taktiež týmto softvérom.

Oproti predošlej verzii tu pribudlo veľa funkcií a niektoré boli odobrané alebo vylepšené ako napríklad spomenuté tlačidlo *Click to plot*. Avšak predtým ako sa užívateľ rozhodne vykresliť grafy, môže si vybrať ktoré okná chce vidieť, pomocou odškrtnutých políčok. Tak isto je možné vybrať logaritmické zobrazenie v ose Y pre frekvenčné spektrum. V ose X je zase možné nastaviť limity zobrazenia ako pre frekvenčné spektrum tak aj pre spektrogram. V kóde je ošetrené aj to, aby sa pri prekročení limitnej hodnoty, nastavila maximálna možná hodnota. Tá je daná vzorkovacou frekvenciou signálu. Pre filtrovanie frekvencií bolo pridané



Obr. 4.22 V poradí *Inferno*, *magma*, *plasma*, *viridis* [33]

okno, kde užívateľ zadá požadovanú presnosť okolia, ktoré chce kontrolovať v okolí každej vybranej frekvencii. Pre spektrogram tu boli pridané dve rolovacie okná s možnosťami výberu rozlíšenia zobrazenia frekvencií a taktiež možnosť zmeny farebnej mapy vid' Obr. 4.22 z *viridis* na *inferno*, *magma* alebo *plasma*, ktoré splňujú podmienky vysvetlené v kapitole 4.4.4.

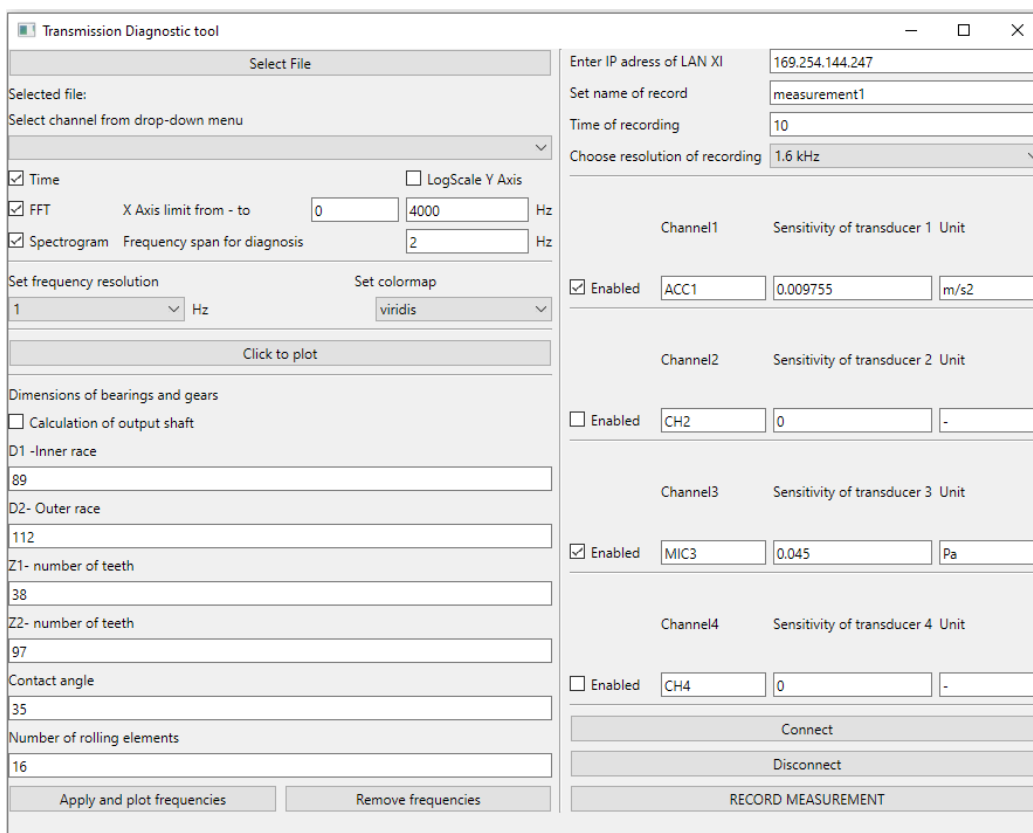
Nakoniec bolo pridané tlačidlo pre vmazanie už vykreslených teoretických frekvencií závad ložísk a frekvencií zadaných užívateľom.

Online časť

Táto časť GUI vid' Obr. 4.23 prešla značnou zmenou, hlavne čo sa týka funkcií, keďže v predošlej verzii nebola pripravená na zber dát. Teraz je však už plnohodnotnou náhradou programu *BK Connect*.

Prvé riadky neprešli zmenou, užívateľ sem zadá IP adresu analyzátoru, ktorá je vidieť po zapnutí na obrazovke. Niekedy je potrebné v module manuálne reštartovať LAN nastavenia až vtedy je IP adresa platná. V druhom riadku sa nastaví názov merania, ďalej jeho dĺžka a rozlíšenie nahrávania. Nenastavuje sa tu priamo vzorkovacia frekvencia, ale tá je následne

prepočítaná a teda zvolí sa taká aby bolo možné zobrazit' zvolené rozlíšenie bez aliasingu. V ďalšej časti je nastavenie snímačov. Každý riadok predstavuje jeden snímač. Je možné



Obr. 4.23 Finálna verzia GUI pre softvér na diagnostiku prevodových ústrojenstiev

zakliknúť políčko *Enabled*, čo má za následok povolenie alebo zamedzenie tohto kanálu. Ďalej je potrebné zadať jeho názov, hodnotu citlivosti snímača v základných jednotkách a vybrať jednotku, ktorá bude zobrazená v grafoch. Takto pripravené údaje sa pomocou tlačítka *Connect* spracujú a pošlú do analyzátora, kde prebehne nastavenie a analyzátor prejde od stavu kedy je pripravený snímať. Kliknutím na tlačidlo *Disconnect* sa zase odpoja kanály a je možné znova nakonfigurovať meranie alebo snímače. Nakoniec pomocou tlačidla *Record measurement* užívateľ zapne meranie podľa aktuálnej konfigurácie a je uložené do úložiska počítača. Odtiaľ je možné ihneď otvoriť tento súbor s meraním a vykonať potrebné kroky k analýze tohto signálu.

4.8 VALIDÁCIA MERANÝCH DÁT

K tomu aby bol softvér použiteľný, je potrebné zobrazovať validované hodnoty voči etalónu. Na to boli použité kalibrátory vibrácií a hluku od firmy Brüel & Kjær. Kalibrátor hladiny akustického tlaku typ 4230 so signálom s frekvenciou 1000 Hz a hodnotou hladiny akustického tlaku 93 dB. Kalibrátor vibrácií typ 4294 s frekvenciou 159,15 Hz a amplitúdou zrýchlenia $10 \text{ m}\cdot\text{s}^{-2}$.

Najprv boli namerané hodnoty s programom BK Connect a vyexportované vo formáte *.h5* aby ich bolo možné zobrazit' v tomto softvéry. Obdobným spôsobom bolo vykonané meranie cez tento softvér. Ukázalo sa, že meranie je spoľahlivé a namerané dáta sú rovnaké ako pri konvenčnom postupe s programom BK Connect. Efektívna hodnota amplitúdy zrýchlenia

vyšla $9,94 \text{ m}\cdot\text{s}^{-2}$ s frekvenciou 159,16 Hz. Pri validácii hladiny akustického tlaku hluku vyšla amplitúda s hodnotou 93,6 dB s frekvenciou 994,7 Hz. Tieto výsledky je možné považovať za presné a tým pádom je tento softvér vhodný aj pre reálne technické experimenty.

4.9 MOŽNOSTI VYLEPŠENIA

Počas testovania softvéru, sa zistilo, že chýba tlačidlo na zastavenie merania skôr ako po uplynutí nastaveného času. Pri pokuse o implementáciu tohto tlačidla, nebolo možné softvér ovládať, pravdepodobne z dôvodu priebehu procesov na pozadí.

Z grafického pohľadu by bolo vhodné vylepšiť zobrazenie vypočítaných frekvencií, pretože názvy sa miestami prekrývajú. Taktiež by bolo vhodné pridať stále rozmery grafických okien, s daným pomerom strán, napríklad pre účely prezentácie alebo do dokumentov a vedeckých článkov.

Možnosťou pre urýchlenie práce pri validácii meraní, experimentu a MKP výpočtov, by mohlo byť prepojenie tohto softvéru s programom Ansys. V tejto verzii softvéru sa využíva manuálne vkladanie získaných hodnôt zo simulácie do predpripravenej tabuľky v programe Excel. Tento proces by mohol byť taktiež automatizovaný a užívateľ by mal možnosť zvoliť si konkrétne dáta zo simulácie, ktoré chce použiť už v prostredí Ansys. Tie by boli automaticky vložené do programu Excel.

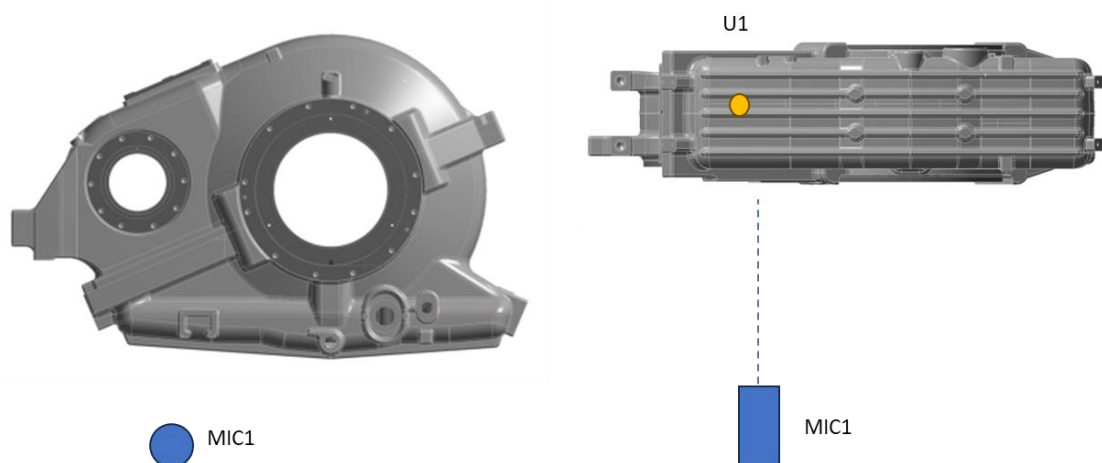
Ďalším predmetom skúmania, ako už bolo predtým spomenuté v kapitole 3.4.5 o STFT, je spracovanie obrazu pomocou strojového učenia. Tento softvér by mohol slúžiť ako základ pre vývoj algoritmu na rozpoznanie zmien a závad v strojoch kde je potrebné monitorovať rotačné časti, ložiská a ozubené kolesá. Keďže väčšina priemyslu je a bude viac automatizovaná, ľudsky faktor kontroly neprichádza do úvahy. V prípadoch, kde je potrebná neustále fungujúca výroba a nečakaný výpadok by mal za následok finančné straty, by mohla mať táto investícia význam.

Taktiež pri dlhodobých skúškach prevodoviek, kedy nie je potrebné ukladať celý priebeh merania, by bolo možné nastaviť začiatok ukladania na základe zmien chovania prevodovky. Softvér by zbieral a vyhodnocoval dáta po celú dobu, ale meranie by spustil až keď by vyhodnotil začiatok zmeny v chovaní.

Softvér má stále obrovské možnosti vylepšenia, ktoré sa do tohto pilotného projektu nepodarilo zakomponovať. Za zmienku stojí, využitie pamäte počítača a ovládanie využitia výkonu procesora, alebo grafickej karty na zobrazovanie dát. To sa v tejto práci nerieši, ale mohlo by to byť do budúcnosti predmetom skúmania.

5 ANALÝZA DÁT Z PREVODOVKY

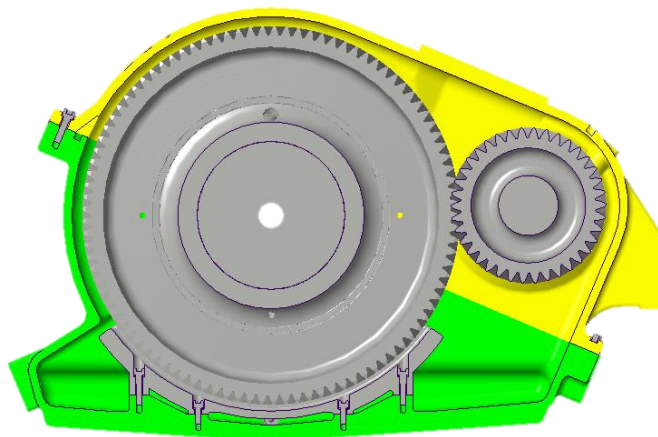
Ako už bolo spomenuté v predošlých kapitolách, v prevodových ústrojenstvách vznikajú počas chodu rôzne druhy vibrácií a hluku. Táto kapitola sa zaoberá diagnostikou konkrétnej vlakovej prevodovky od firmy VOITH, na ktorej boli experimentom získané dáta zo snímača zrýchlenia a mikrofónu vid' Obr. 5.1. Účelom tejto práce, je vytvorenie softvéru na diagnostiku, preto je potrebné dodať, že tieto testovacie dáta boli dodané a experiment bol robený v minulosti mimo súvislosť s touto diplomovou prácou. Táto kapitola slúži na znázornenie využiteľnosti a funkcií tohto softvéru.



Obr. 5.1 CAD model skrine prevodovky so znázorneným mikrofónom MIC1 z bočnej strany a snímačom zrýchlenia U1 zo spodnej strany

5.1 ANALYZOVANÉ ZDROJE VIBRÁCIÍ

Pre analýzu boli použité testovacie dáta z jednostupňovej vlakovej prevodovky vid' Obr. 5.2. Jedná sa o prevodovku so šikmým čelným ozubením a obsahuje 2 ozubené kolesá. Hnané ozubené koleso má 97 zubov a hnacie má 38. To spolu dáva prevodový pomer 2,55. Vstupný hriadeľ je osadený na dvoch valčekových ložiskách NU215ECML a jednom guľôčkovom QJ215N2MA pre zachytenie axiálnych síl. Výstupný hriadeľ je uložený na 2 ložiskách T2DC220, ktoré boli pravdepodobne upravené pre konkrétnu prevodovku.



Obr. 5.2 Pohľad na prevodovku v reze

Potrebné rozmery pre diagnostiku, ktoré boli zadané do softvéru sú uvedene v Tabuľke 2 nižšie.

Tabuľka 2.

Názov ložiska	D1- Priemer vnútornej dráhy [mm]	D2-Priemer vonkajšej dráhy[mm]	Kontaktný uhol[°]	Počet valivých elementov
<i>NU215ECML</i>	88.5	113.9	0	18
<i>QJ215N2MA</i>	88.5	112	35	16
<i>T2DC220</i>	260	290	15.8	38

Podľa vzťahov z kapitoly 1.1 a 1.2, boli softvérom vypočítané teoretické frekvencie prejavu vady v ložiskách vid' Tabuľka 3.

Tabuľka 3.

Názov ložiska	BPFI [Hz]	BPFO [Hz]	FTF [Hz]	BSF [Hz]
<i>NU215ECML</i>	436	338	24	168
<i>QJ215N2MA</i>	308	308	19	181
<i>T2DC220</i>	337	303	8	153

Taktiež boli v softvéry podľa rovnice 1 a rovnice 2 vypočítané otáčková a zubová frekvencia. Pre signál s konštantnými otáčkami 2562 min^{-1} vyšli hodnoty otáčkových frekvencií 43 Hz a pre druhú hriadeľ vyšla hodnota 17 Hz. Zubová frekvencia potom vyšla 1634 Hz. Tieto hodnoty frekvencií bude možné v nasledujúcich kapitolách vidieť aj v grafickom zobrazení vo frekvenčnej doméne spolu s vlastnými frekvenciami. Taktiež je možné vidieť postranné pásma a harmonické násobky frekvencií.

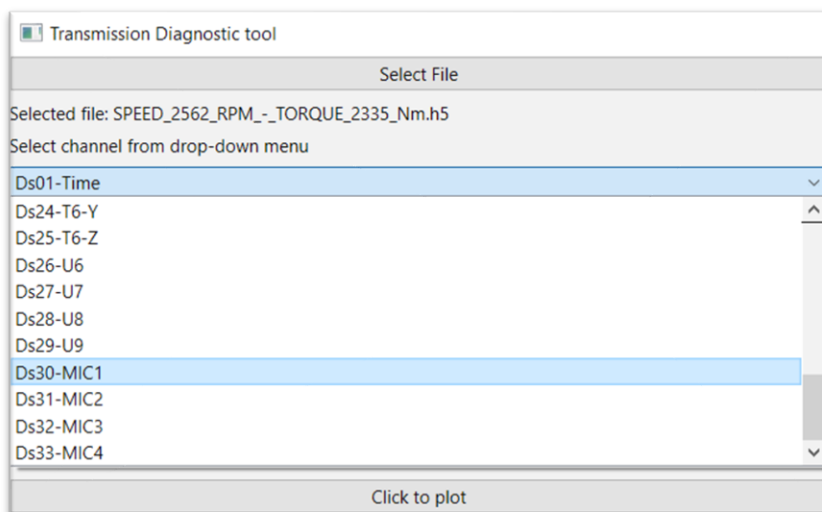
5.2 DEFINÍCIA DÁT Z MERANIA

Boli vybrané 2 testovacie sady dát. Jedna pri konštantných otáčkach 2562 min^{-1} a zaťažení krútiacim momentom o veľkosti 2335 Nm. Druhá sada testovacích dát je meranie dobehovej skúšky prevodovky. Tento dataset obsahuje nahrávky s dĺžkou vyše 6 minút s vyše 12 miliónov hodnôt pre každý z 33 kanálov. Spolu majú tieto dáta viac ako 1,6 GB. Meranie s takýmto rozsahom trvalo otvoriť pomocou tohto softvéru iba približne 11 sekúnd.

V dátach z merania s konštantnými otáčkami bolo demonštrované, kde sa vo frekvenčnom spektre nachádza zubová frekvencia a teoretické frekvencie prejavu vady v ložiskách. Boli analyzované signály z kanálu MIC1 teda mikrofón a U1 ako uniaxiálny akcelerometer. Mikrofón bol umiestnený 1270 mm od pozdĺžnej roviny prevodovky a akcelerometer bol umiestnený na spodnej strane skrine prevodovky. V dátach z dobehovej skúšky má význam pozerať sa hlavne na spektrogram, kde bolo jasne vidieť priebeh frekvencií v čase.

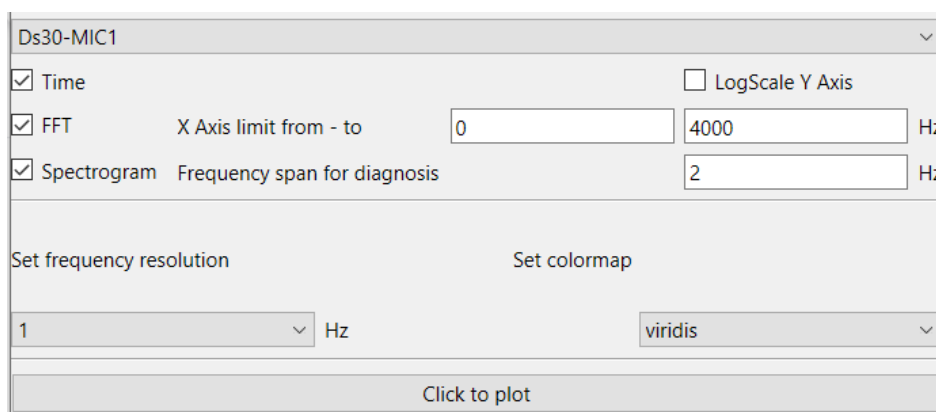
5.3 POSTUP PRI DIAGNOSTIKE

Pri diagnostike dát s konštantnými otáčkami, bol vybraný súbor a konkrétny kanál merania. V tomto prípade bol zvolený kanál s názvom *MIC1* vid' Obr. 5.3, takže sa bude analyzovať akustický tlak nameraný pomocou mikrofónu. Taktiež sa bude analyzovať zrýchlenie zo snímaču zrýchlenia U1.



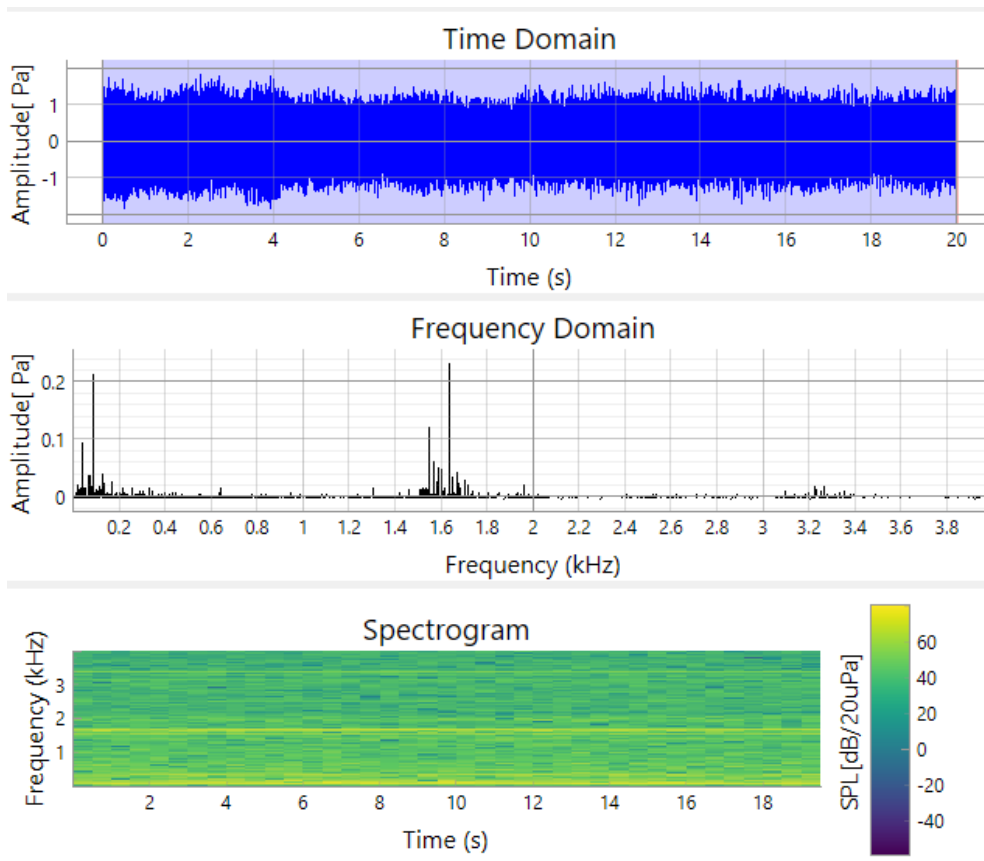
Obr. 5.3 Ukážka rolovacieho okna s dostupnými kanálmi merania

Zaškrtnuté políčka *Time*, *FFT* a *Spectrogram* značia, že budú vykreslené všetky 3 grafy. Boli nastavené hranice pre zobrazenie frekvencií od 0 do 4000 Hz. V kolónke pre rozlíšenie frekvencií v spektrograme sa vybrala hodnota 1 Hz a predvolená farebná mapa viridis vid' Obr. 5.4.



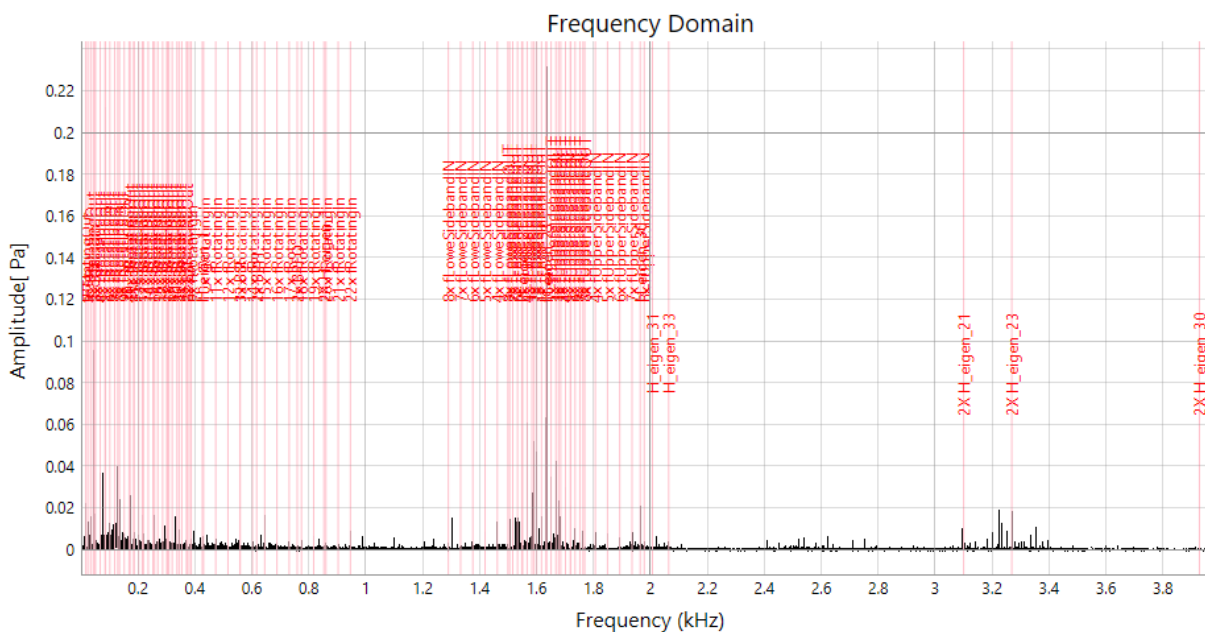
Obr. 5.4 Nastavenie zobrazenia frekvencií

Po kliknutí na tlačidlo *Click to plot* sa zobrazili 3 grafy vid' Obr. 5.5. Od vrchu v poradí najprv časová doména so signálom, pod tým frekvenčná doména a ako posledný je spektrogram. Aby bolo grafy lepšie vidieť, softvér obsahuje funkciu, ktorá umožňuje grafy roziahnuť a zmeniť ich veľkosť podľa potreby. Taktiež je možné ich vykresliť po jednom pomocou zaškrtnutia príslušného políčka. Táto funkcia bude využitá vo frekvenčnom spektre pre prehľadnejšie zobrazenie, keďže obsahuje veľa informácií.



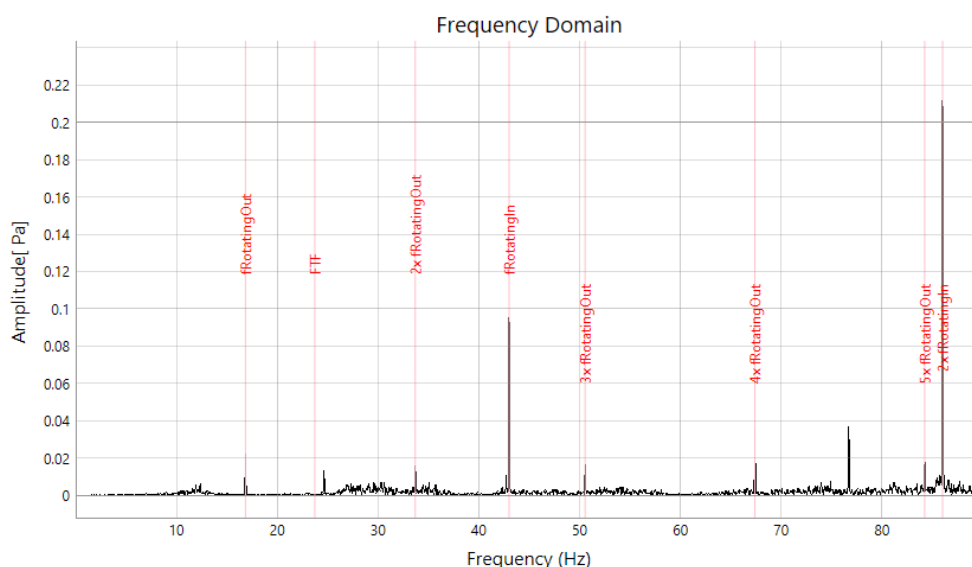
Obr. 5.5 Výsledné zobrazenie grafov v okne

V tomto type signálu ma zmysel zamerať sa čisto na frekvenčnú doménu a zobraziť aj zubové, otáčkové a frekvencie prejavu vady v ložiskách spolu s vlastnými frekvenciami získaných z technického experimentu vid' Obr. 5.6. Je zobrazený akustický tlak v lineárnej stupnici. V grafe bolo vykreslených veľa frekvencií, preto je potrebné priblížiť ich zobrazenie.

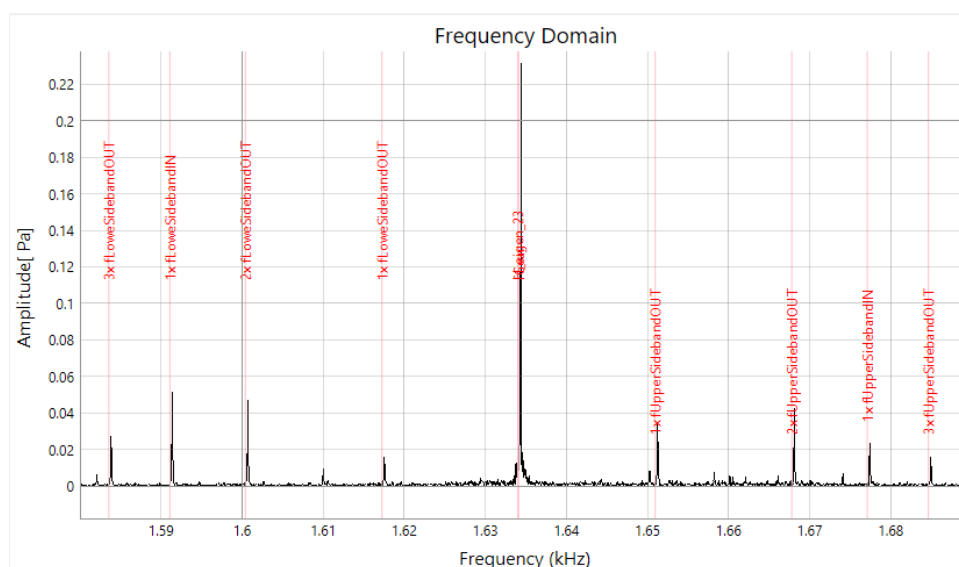


Obr. 5.6 Frekvenčné spektrum z mikrofónu MIC1

V grafe na Obr. 5.7 je po priblížení vidieť typické frekvencie spomínané v predošlých kapitolách. Konkrétne otáčková frekvencia vstupného hriadeľa s hodnotou 43 Hz a jej harmonický násobok s hodnotou 86 Hz, ktorý má vyššiu amplitúdu. To môže byť spôsobené práve tým, ako sa zvuk z ozubenia šíri štruktúrou stien prevodovky a tie ďalej prechádzajú vzduchom až k mikrofónu. Taktiež je tu vidieť otáčkovú frekvenciu výstupnej hriadele s hodnotou približne 17 Hz a jej násobky. Pomer medzi otáčkovými frekvenciami je daný pomerom prevodu teda približne 2,55. Frekvencia s hodnotou 77 Hz zostala nepopísaná, nie je to žiadna z ložiskových frekvencií ani žiaden harmonický násobok. Softvér nemá o tejto frekvencii žiadne informácie. Keďže sa jedná o signál z mikrofónu, môže sa jednať aj o hluk generovaný externým agregátom mazania alebo ventilátorom.



Obr. 5.7 Priblížení na otáčkové frekvencie z mikrofónu MIC1

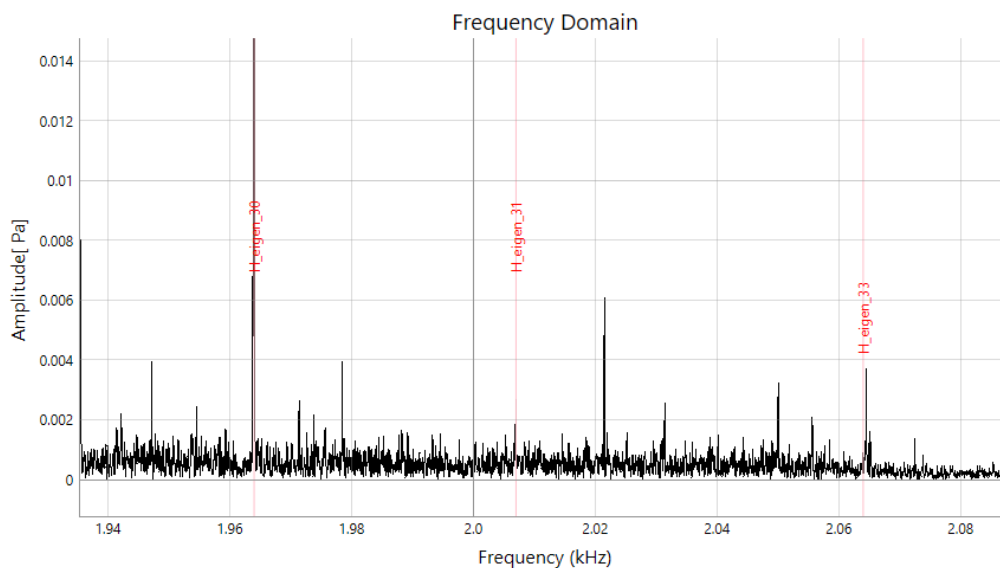


Obr. 5.8 Zubová frekvencia a postranné pásma z mikrofónu MIC1

Ďalej je vidieť menej výrazné frekvencie prejavu väd ložiska *QJ215N2MA* konkrétne frekvenciu klietky – FTF. Frekvencie vonkajšej dráhy ložiska - BPFO a vnútornej dráhy ložiska – BPFI nie je vidno. To že nie sú výrazné, je správne, pretože sa jednalo o novú

prevodovku, takže sa nepredpokladalo poškodenie ložísk. Frekvencie ostatných ložísk neboli vykreslené, pre zachovanie prehľadnosti. Na druhej strane, zubová frekvencia je vždy výrazná a je vidieť že sa nachádza v okolí 1640 Hz. Presnejšie ju je vidieť na Obr. 5.8 po priblížení, kde je vidieť aj postranné pásma. Ako už bolo vysvetlené, tu je prakticky vidieť, že sú vzdialené od zubovej frekvencii o hodnotu približne 43 Hz, čo predstavuje práve otáčkovú frekvenciu vstupného hriadeľa a taktiež o hodnotu 17 Hz čo predstavuje otáčkovú frekvenciu výstupného hriadeľa. Softvér vyhodnotil tieto frekvencie správne a korešpondujú so signálom.

Taktiež je možné si všimnúť 3 vlastné frekvencie v okolí 2000 Hz. Konkrétne ide o 30. 31. a 33. vlastnú frekvenciu skrine prevodovky vid' Obr. 5.9. V zobrazení chýba 32. vlastná frekvencia, to je však správne, keďže softvér vyhodnotil amplitúdu za príliš malú a tak nezobrazil túto frekvenciu. Podľa výsledkov z experimentu, by sa mala nachádzať na frekvencii 2043 Hz, avšak nie je tu vidieť žiadne viditeľné zvýšenie amplitúdy na rozdiel od 33. vlastnej frekvencii. To potvrdzuje aj 32. a 33. vlastný tvar kmitania na Obr. 5.10.



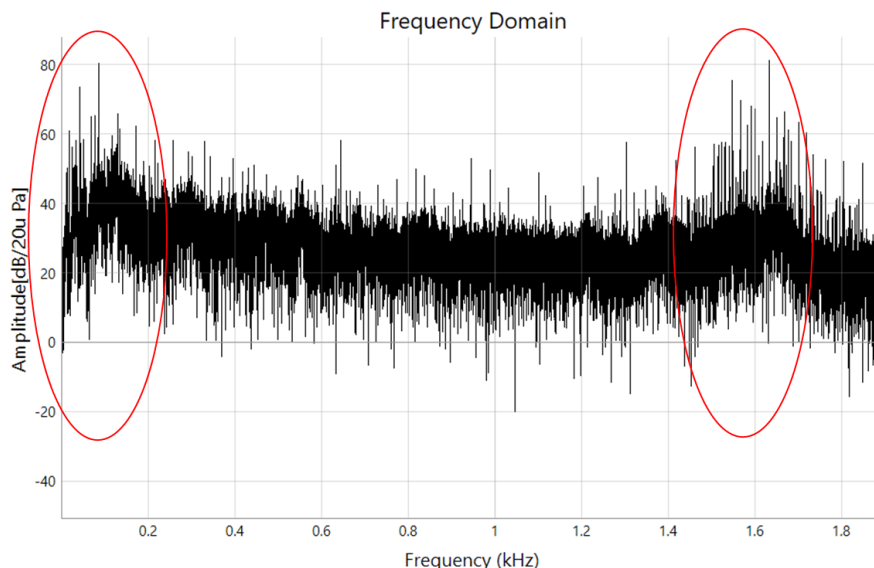
Obr. 5.9 Odozva na snímači MIC1, zobrazujúca lokálne navýšenie amplitúdy akustického tlaku vo frekvenčnej oblasti blízkej frekvencii 30.31. a 33. vlastného tvaru kmitu skrine prevodovky



Obr. 5.10 32. a 33. vlastný tvar kmitu

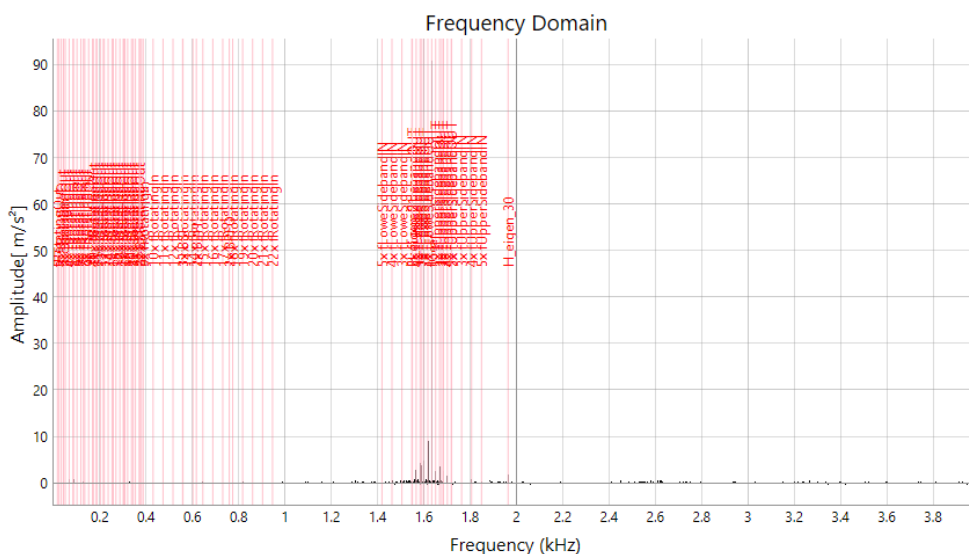
Signál z mikrofónu bol vybraný aj kvôli tomu, že je na ňom možné znázorniť funkciu pre výpočet hladiny akustického tlaku. Pri tomto meraní boli dosiahnuté najvyššie hodnoty v oblasti zubovej frekvencii a v oblasti otáčkových frekvencií. Ich hodnoty boli v oboch

prípadoch približne 80 dB. Znova sa potvrdilo to, že je budená pravdepodobne skriňa prevodovky a tak je maximálna amplitúda až na 2. harmonickom násobku otáčkovej frekvencii pri 86 Hz a nie pri 43 Hz vid' Obr. 5.11.



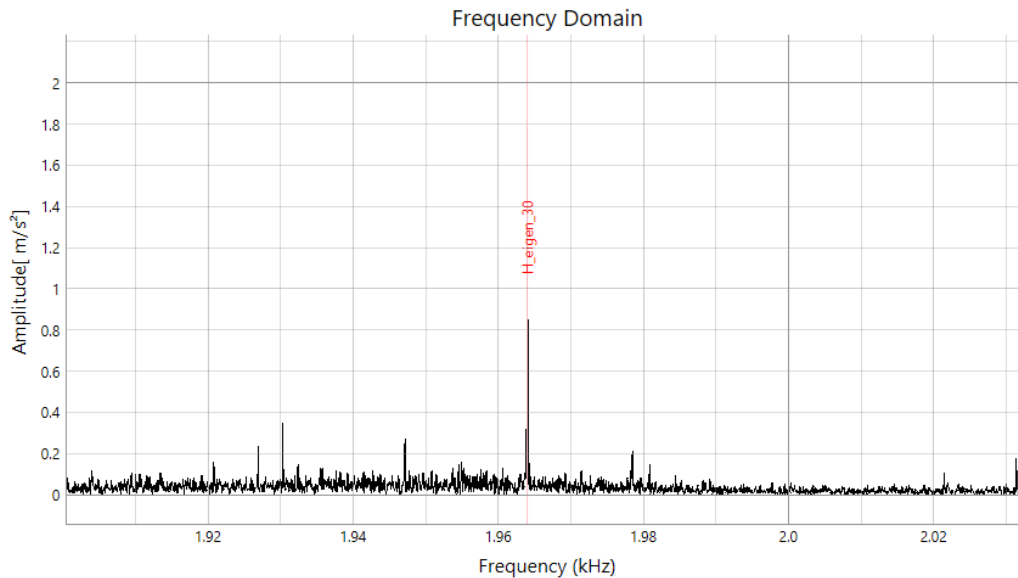
Obr. 5.11 Frekvenčné spektrum z mikrofónu MIC1 zobrazené v decibeloch

Vlastné tvary kmitania skrine prevodovky je možné zobraziť aj zo snímača zrýchlenia. Bol vybraný snímač U1 ktorý sa nachádza na spodnej strane skrine. Ak sa v signáli nachádza významné lokálne maximum na mieste kde bola experimentom zaznamenaná vlastná frekvencia, softvér ju zobrazí. Na Obr. 5.12 je vidno frekvenčné spektrum z tohto kanála v lineárnej stupnici aj s vyznačenými frekvenciami. Sú tu zase dominantné otáčkové frekvencie a ich násobky so zubovou frekvenciou a jej postranné pásma. Je možné si všimnúť aj 30. vlastnú frekvenciu skrine prevodovky s lokálnym navýšením amplitúdy. Toto zobrazenie je neprehľadné, takže je potrebné obraz priblížiť na konkrétne miesto s vlastnou frekvenciou.

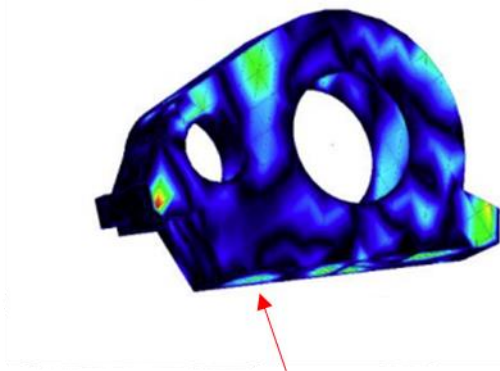


Obr. 5.12 Frekvenčné spektrum zo snímača zrýchlenia U1

Po priblížení je na Obr. 5.13 jasne vidieť dominantnú vlastnú frekvenciu. Keďže snímač bol umiestnený na spodok skrine, dá sa očakávať práve v tomto mieste odozva na budenie. To potvrdzuje aj Obr. 5.14 kde je vidieť 30. vlastný tvar kmitu skrine. Na spodnej strane je vidieť lokálne zvýšenie amplitúdy zrýchlenia práve v oblasti umiestnenia akcelerometru.



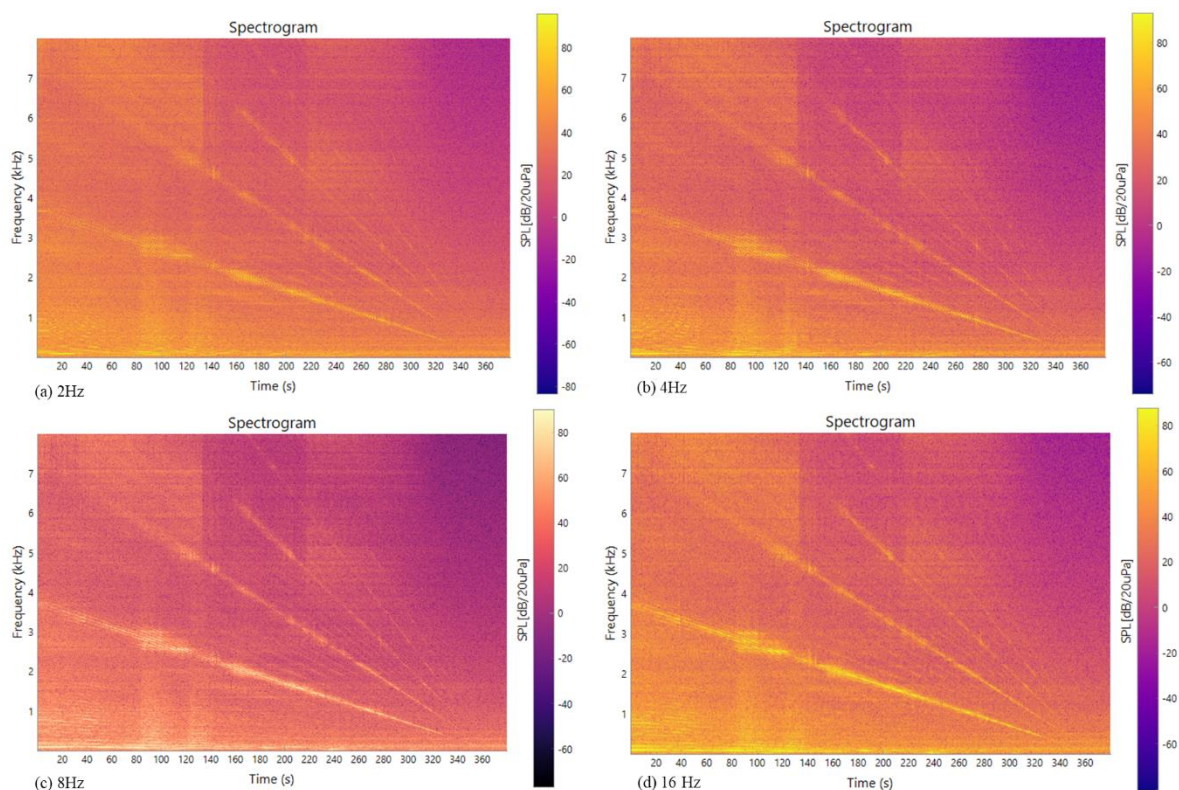
Obr. 5.13 Odozva na snímači U1, zobrazujúca lokálne navýšenie amplitúdy zrýchlenia vo frekvenčnej oblasti blízkej frekvencii 30. vlastného tvaru kmitu skrine prevodovky



Obr. 5.14 30. vlastný tvar kmitu skrine prevodovky – 1964 Hz

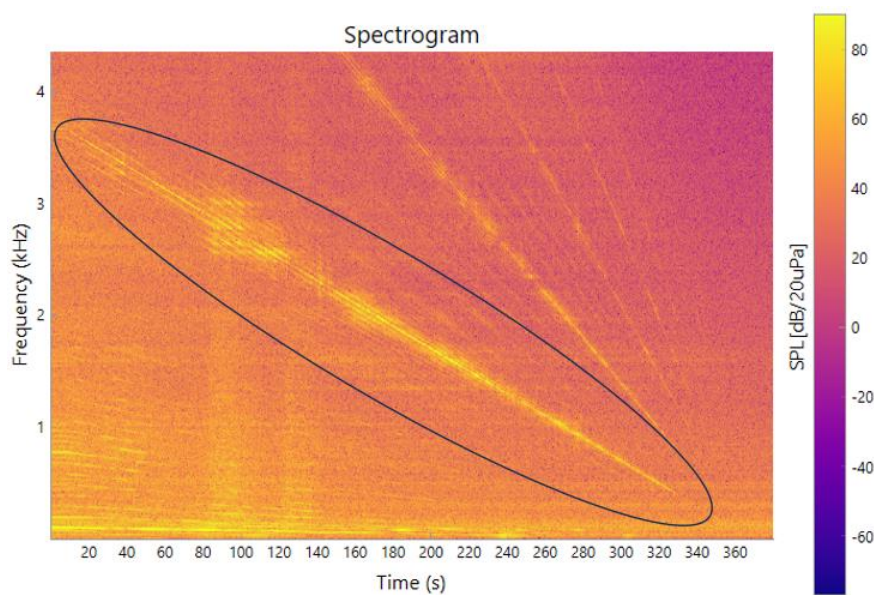
Pri druhom datasete z dobovej skúšky je vhodné zamerať sa na demonštráciu 2 funkcií a to konkrétne možnosť vybrať si pásmo, z ktorého sa bude počítať FFT a samotný spektrogram.

V spektrograme sa zobrazujú dáta z kanálu *MIC1*. Je potrebné zvoliť vhodné rozlíšenie frekvencií. Ak by bolo príliš jemné, znamenalo by to, že časové rozlíšenie je zase príliš široké. Na Obr. 5.15 vidieť 4 rôzne frekvenčné rozlíšenia - 2 Hz, 4 Hz, 8 Hz a 16 Hz. Taktiež bola na ukážku použitá iná farebná mapa a to *plasma*. Najvhodnejšie sa zdá frekvenčné rozlíšenie 8 Hz, takto je dosiahnuté časové rozlíšenie 0,5 sekundy. Jednotlivé frekvencie sú dobre rozlíšiteľné a nezlievajú sa dokopy ako to je pri vyššom alebo nižšom rozlíšení.



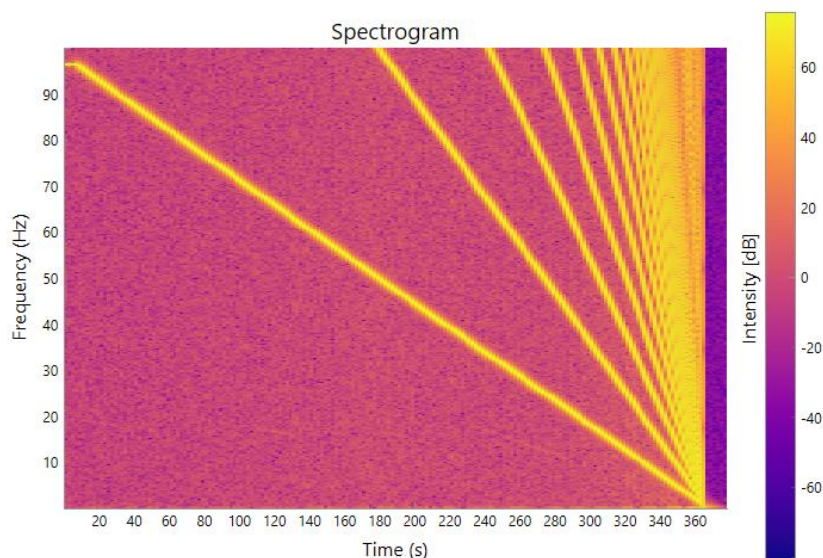
Obr. 5.15 Rôzne rozlíšenie spektrogramu podľa šírky jedného pixela v závislosti na frekvencii

Zubová frekvencia označená na Obr. 5.16 je znovu dominantným javom v spektrograme. Začína na frekvencii okolo 3600 Hz až do 0 keďže sa jedná o dobohovú skúšku. Taktiež je tu vidieť jemné postranné pásma v okolí zubovej frekvencii a jej harmonické násobky.



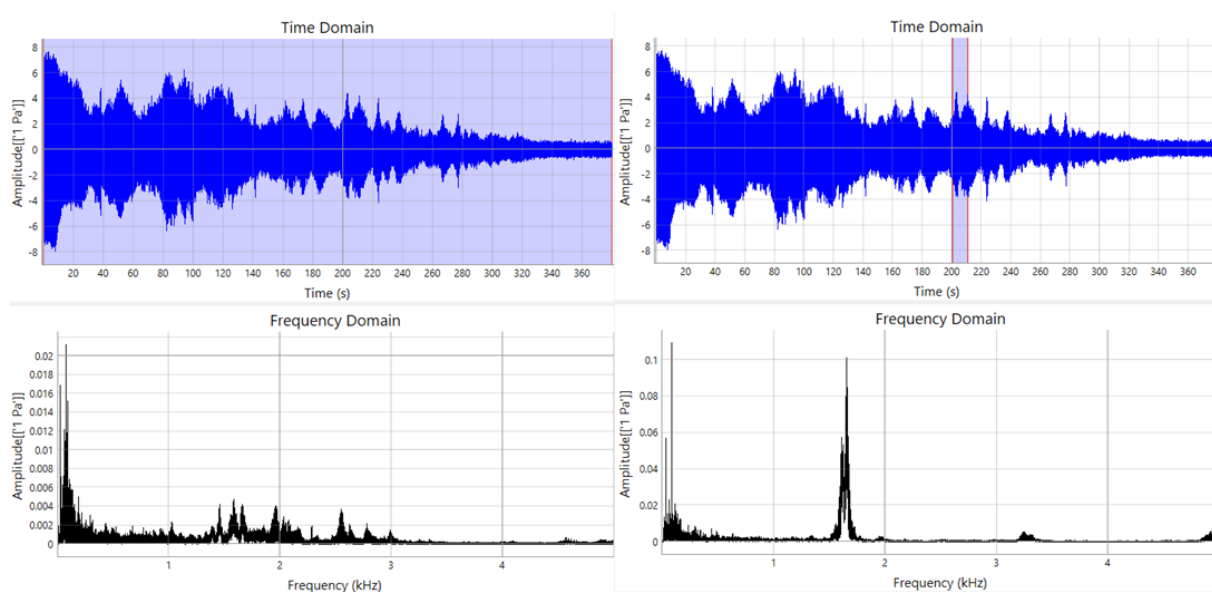
Obr. 5.16 Výrazná zubová frekvencia v spektrograme

Funkciu pre rozlíšenie spektrogramu je vhodné popísať na kanále signálu z tachy sondy, pre určenie aktuálnych otáčok na Obr. 5.17. Bolo zvolené najjemnejšie rozlíšenie 0,25 Hz vo frekvenčnej doméne a z toho vyšlo rozlíšenie 1 sekunda v časovej doméne. Taktiež sa obraz priblížil na prvú amplitúdu od 96 Hz. Keďže otáčky sa každou sekundou konštantne menia, je potrebné mať prehľad o konkrétnej frekvencii na určenie aktuálnych otáčok. Podľa tohto signálu začalo meranie na otáčkach 96 Hz čo je 5760 min^{-1} a zotrvali tam počas 8 sekúnd. Potom začali kontinuálne klesať až do nuly. Toto zobrazenie slúži len ako ukážka zmeny rozlíšenia, technicky nemá toto zobrazenie veľký význam až na prvý rád ktorý zobrazuje aktuálnu frekvenciu otáčania.



Obr. 5.17 Zmena frekvencie v čase v signáli z otáčkovej sondy

Časová doména takéhoto signálu vyzerá nasledovne ako na Obr. 5.18 Nemá však veľký význam pozerat' sa na frekvenčné spektrum, pretože sú tu pomiešané frekvencie z celého spektra otáčok od 0 až po 5760 min^{-1} .



Obr. 5.18 Porovnanie výpočtu FFT pre celý signál a pre jeho časť

Významnejšie môže byť označenie iba určitej oblasti v časovej doméne kde prebehne výpočet FFT iba s dátami z označenej oblasti. Stále je tam nepresnosť, keďže aj počas 10 sekúnd sa zmenia otáčky a tým pádom aj všetky charakteristiky, ale užívateľ má lepší prehľad o dianí v danej oblasti.

Týmto boli znázornené funkcie softvéru priamo na reálnych dátach z technického experimentu a potvrdené niektoré predpoklady fungovania, ako napríklad označenie významných frekvencií.

ZÁVER

Táto diplomová práca sa zameriava hlavne na vývoj softvéru pre diagnostiku prevodových ústrojenstiev, ktorý môže slúžiť pre analýzu vibroakustického prejavu prevodoviek a tak isto na zber dát z technických experimentov. Kľúčovým prínosom tohto softvéru je možnosť ovládať analyzátor LAN-XI od firmy HBK. Taktiež jeho prispôsobenie rôznym potrebám a možnosť pridávať ďalšie funkcie vďaka jeho modularite.

Pred samotným návrhom tohto softvéru, bolo potrebné spraviť rešerš v oblasti prevodových ústrojenstiev a vibroakustiky. V tejto časti sa vysvetlilo, ako vznikajú vibrácie v prevodových ústrojenstvách a taktiež to, ako sa merajú a získavajú dáta pomocou analyzátora. Vďaka následnému popisu základných meraných veličín, bolo možné prejsť do oblasti digitálneho spracovania signálov, kde sa pojednávalo o typoch signálov v časovej doméne. Keďže bolo potrebné získať informácie o frekvenčnom spektre, bol tu detailne popísaný výpočet diskkrétnej Fourierovej transformácie a aj to, prečo sa na tento výpočet používa algoritmus rýchlej Fourierovej transformácie. Následne bolo možné zamerať sa na jednotlivé časti budúceho softvéru, ako je zobrazenie signálu v čase, vo frekvenčnom spektre a v ich spojení ako spektrogram.

Pri vývoji softvéru bol vybraný programovací jazyk Python, keďže disponuje obrovským množstvom rôznych knižníc a širokou užívateľskou podporou. Preskúmali sa knižnice pre zobrazovanie grafov a boli porovnané ich výhody. Taktisto sa postupovalo až pokým neboli vybrané knižnice pre prácu s grafmi, matematické operácie, manipulácie s dátami a ich čítanie a ukladanie na disk.

Druhou časťou pri vývoji softvéru bola funkcia na získavanie dát zo snímačov. V tejto časti bol vysvetlený princíp komunikácie analyzátora s počítačom a to ako sa pri vývoji postupovalo. Bolo nutné zabezpečiť nastavenie analyzátora a aj vlastnosti samotných snímačov pre správny zber dát. Takisto tu bolo vysvetlené, aký formát sa použil pre ukladanie dát na disk, a to akou formou to bolo realizované.

Po úspešnom snímaní dát z kalibrátorov, bolo potrebné upraviť získavané dáta, pretože amplitúdy v signáloch nezodpovedali realite. Pomocou užívateľskej príručky boli chyby odstránené a tak vzniklo finálne riešenie tohto softvéru. Bolo potrebné ho zaobaliť do užívateľského rozhrania, pre lepšiu využiteľnosť a opakovateľnosť, či už pri testovaní alebo vyhodnocovaní nameraných dát. Nakoniec boli vyhodnotené namerané dáta z vlakovej prevodovky a názorne ukázané všetky funkcie tohto softvéru.

Ako bolo povedané, vďaka rôznym druhom knižníc a možnosti jednoducho pridávať nové funkcie je možné tento softvér v budúcnosti použiť ako základ pre softvéry na princípe zberu a vyhodnocovania dát. Môže zohrávať dôležitú úlohu ako súčasť priemyslu 4.0, ktorým cieľom je automatizovať a digitalizovať procesy.

POUŽITÉ INFORMAČNÉ ZDROJE

- [1] BILOŠ, Jan a Alena BILOŠOVÁ. Aplikovaný mechanik jako součást týmů konstruktérů a vývojářů: Vibrační diagnostika [online]. 1. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, Fakulta strojní, 2012 [cit. 2023-12-15]. Dostupné z: <http://projekty.fs.vsb.cz/147/ucebniopory/978-80-248-2755-1.pdf>
- [2] BRÜEL & KJÆR. Piezoelectric Accelerometer Types 4507 and 4508 [online]. Nærum Denmark, 2020. [cit. 2024-03-08] Dostupné z: <https://www.bksv.com/-/media/literature/Product-Data/bp1841.ashx>.
- [3] BRÜEL & KJÆR. Measurement vibration [online]. Nærum Denmark, 1982. [cit. 2024-08-03] Dostupné z: <https://www.bksv.com/media/doc/br0094.pdf>
- [4] TŮMA, Jiří. Zpracování signálů získaných z mechanických systémů užitím FFT. 1. Praha: Sdělovací technika, 1997
- [5] BRUNTON Steven, KUTZ Nathan. Data driven Science&Engineering : Machine Learning, Dynamical Systems, and Control. [online]. University of Washington 2017.[cit. 2023-12-15]. Dostupné z: <http://databookuw.com/databook.pdf>
- [6] YAN Du, AIMING Wang, SHUAI Wang, BAOMEI He, GUOYING Meng, "Fault Diagnosis under Variable Working Conditions Based on STFT and Transfer Deep Residual Network", *Shock and Vibration*, vol. 2020, Article ID 1274380, 18 pages, 2020. <https://doi.org/10.1155/2020/1274380>
- [7] HONGMEI Liu, LIANFENG Li, JIAN Ma, "Rolling Bearing Fault Diagnosis Based on STFT-Deep Learning and Sound Signals", *Shock and Vibration*, vol. 2016, Article ID 6127479, 12 pages, 2016. <https://doi.org/10.1155/2016/6127479>
- [8] THE MATHWORKS, Inc. Documentation [online]. [cit. 2024-02-15]. Dostupné z: <https://de.mathworks.com/help/dsp/ref/dsp.stft.html>
- [9] ASUNCION Arthur, Signal Processing Applications of Wavelets. [online] University of California, Irvine. [cit. 2024-02-15]. Dostupné z : https://ics.uci.edu/~asuncion/180/Asuncion_Wavelet_Final.pdf
- [10] NOVOTNÝ, Pavel. Experimentální metody: Základní principy využívané při měření vibrací. Brno.2022 Prednáška. Vysoké učení technické v Brně, Fakulta strojního inženýrství.
- [11] EKOSOFTWARE: Měření hluku a vibrací [online]. Liberec: DPOINT.CZ, 2017 [cit. 2024-4-30]. Dostupné z: <https://www.ekosoftware.cz/mereni-otacek>
- [12] HEINZEL, G., RÜDIGER, A., & SCHILLING, R. Spectrum and spectral density estimation by the Discrete Fourier transform (DFT), including a comprehensive list of window functions and some new at-top windows. [online] AEI-Hannover (2002) [cit. 2024-02-15] Dostupné z : <https://hdl.handle.net/11858/00-001M-0000-0013-557A-5>

- [13] F. J. HARRIS, "On the use of windows for harmonic analysis with the discrete Fourier transform," in Proceedings of the IEEE, vol. 66, no. 1, pp. 51-83, Jan. 1978, doi: 10.1109/PROC.1978.10837.
- [14] PANZA MARIA Antonietta, . A Review of Experimental Techniques for NVH Analysis on a Commercial Vehicle. Energy Procedia. 70th Conference of the Italian Thermal Machines Engineering Association, ATI2015, 2015, (82), 1017-1023. [cit. 2024-08-04] Dostupné z: <https://doi.org/10.1016/j.egypro.2015.11.861>
- [15] NOVÝ, Richard. Hluk a chvění. Vyd. 3. V Praze: České vysoké učení technické, 2009. ISBN 978-80-01-04347-9.
- [16] NOVOTNÝ, Pavel. NVH of vehicles: Vibroacoustic signals. Brno 2022. Přednáška. Vysoké učení technické v Brně, Fakulta strojního inženýrství.
- [17] BRÜEL & KJÆR, LAN-XI Open API User Guide. [online], Nærum Denmark, 2021, [cit. 2024-04-15]. Dostupné z: User Guide: LAN-XI Open API (be1872) (bksv.com)
- [18] SATURN CLOUD, What Is the Fastest File Format for ReadWrite Operations with Pandas andor Numpy. [online], 2023, [cit. 2024-04-15]. Dostupné z: <https://saturncloud.io/blog/what-is-the-fastest-file-format-for-readwrite-operations-with-pandas-andor-numpy/>
- [19] YANG Liu, HEER Jeffrey, Somewhere Over the Rainbow: An Empirical Assessment of Quantitative Colormaps. [online], 2018, [cit. 2024-04-16]. Dostupné z: <https://idl.cs.washington.edu/files/2018-QuantitativeColor-CHI.pdf>
- [20] HARVEY Lisa, A dangerous rainbow: Why colormaps matter. [online], 2018, [cit. 2024-04-16]. Dostupné z: <https://blogs.mathworks.com/headlines/2018/10/10/a-dangerous-rainbow-why-colormaps-matter/>
- [21] BRÜEL & KJÆR, LAN-XI Data Acquisition Hardware Frames, Modules and Front Panels. [online], 2021, [cit. 2024-04-20]. Dostupné z: <https://www.bksv.com/media/doc/bp2215.pdf>
- [22] DOČKAL, Aleš. Konstrukční optimalizace ozubené převodovky s ohledem na snižování hlukové emise. V Brně: Vysoké učení technické, Fakulta strojního inženýrství, Ústav konstruování, 2003. Vědecké spisy Vysokého učení technického v Brně. PhD Thesis. ISBN 80-214-2439-7.
- [23] SMITH, Derek J. Gear noise and vibration. 2nd edition. New York: Marcel Dekker, 2003, 297 s. ISBN 824741293.
- [24] TŮMA, Jiří. Diagnostika strojů: Frekvence projevu vad ozubených kol. Ostrava. Přednáška. Technická univerzita Ostrava, Fakulta strojní.
- [25] VAN ROSSUM, Guido and the Python development team. Python Tutorial 3.7.0. [online] 2018. [cit 2024-01-05] Dostupné z : https://bugs.python.org/file47781/Tutorial_EDIT.pdf

- [26] BÄKER, Bernard. Fahrzeugelektronik: Elektronische Systeme im Kraftfahrzeug. Dresden 2023. Přednáška. Technische Universität Dresden, Professur für Fahrzeugmechatronik.
- [27] TRAMPE BROCH, Jens. Mechanical vibration and shock Measurements. Nærum, Denmark, 1984. ISBN 87-87355-34-5
- [28] BRÜEL & KJÆR, Kurtosis in Random Vibration Control. [online], 2009, [cit. 2024-05-01]. Dostupné z: <https://www.bksv.com/media/doc/bo0510.pdf>
- [29] SMITH Grant Maloy, Measuring Shock and Vibration with Accelerometer Sensors. [online], 2023, [cit. 2024-05-01]. Dostupné z: <https://dewesoft.com/blog/measure-shock-vibration-with-accelerometers>
- [30] POLYTEC, Industrial Vibration Sensor. [cit. 2024-08-05] Dostupné z: <https://www.polytec.com/eu/vibrometry/products/single-point-vibrometers/ivs-500-industrial-vibration-sensor>
- [31] PROKOP, Aleš. MODELOVÁNÍ VIBRACÍ POHONNÝCH JEDNOTEK APLIKACÍ VIRTUÁLNÍCH PROTOTYPŮ. Brno, 2016, 160 s. Dizertační práce. Vysoké učení technické v Brně, Fakulta strojíního inženýrství. Vedoucí práce Doc. Ing. Pavel Novotný, Ph.D.
- [32] BRÜEL & KJÆR, Measurement microphones, [cit. 2024-10-05] Dostupné z : <https://www.bksv.com/en/transducers/acoustic/microphones>
- [33] J. D. HUNTER, "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.
- [34] PyQtGraph - Scientific Graphics and GUI Library for Python. [cit. 2024-10-05] Dostupné z : <https://www.pyqtgraph.org/>
- [35] HARRIS, C.R., MILLMAN, K.J., van der WALT, S.J. et al. Array programming with NumPy. Nature 585, 357–362 (2020). DOI: 10.1038/s41586-020-2649-2.
- [36] COLLETTE, A., 2013. Python and HDF5, O'Reilly.
- [37] MATUS, G. Časové okno pro frekvenční analýzu signálů využívající hladkých funkcí. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 56 s. Vedoucí bakalářské práce prof. Ing. Pavel Jura, CSc.

Zoznam použitých skratiek a symbolov

<i>ABS</i>	Anti- lock braking system
<i>API</i>	Application programming interface
<i>BPFI</i>	Ball pass frequency inner race
<i>BPFO</i>	Ball pass frequency outer race
<i>BSF</i>	Ball spin frequency
<i>CAD</i>	Computer aided engineering
<i>DAQ</i>	Data acquisition
<i>DFT</i>	Diskrétna Fourierova transformácia
<i>Eigen</i>	Vlastná frekvencia
<i>FFT</i>	rýchla Fourierova transformácia
<i>FTF</i>	Fundamental train frequency
<i>GUI</i>	Graphical user interface
<i>H</i>	Housing
<i>HBK</i>	Hottinger Brüel and Kjær
<i>HDF5</i>	Hierarchický dátový formát
<i>IP</i>	Internet protocol
<i>JSON</i>	JavaScript object notation
<i>LAN</i>	Local area network
<i>MKP</i>	Metóda konečných prvkov
<i>NaN</i>	Not a number
<i>O</i>	Output shaft
<i>OOB</i>	Objektovo orientované programovanie
<i>P</i>	Pinion
<i>PC</i>	Personal computer
<i>PTP</i>	Precision time protocol
<i>RAM</i>	Random access memory
<i>REST</i>	Representational State transfer
<i>SD</i>	Secure digital
<i>STFT</i>	Short time Fourier transform
<i>TCP</i>	Transmission control protokol
<i>TE</i>	Transmission Error
<i>TEDS</i>	Transducer electronics datasheet

<i>TFI</i>		Time frequency images
<i>XML</i>		Extensible markup language
<i>a</i>	[m·s ⁻²]	Zrýchlenie
<i>c</i>	[m·s ⁻¹]	Rýchlosť šírenia zvuku v danom prostredí
<i>c</i>	[m·s ⁻¹]	Rýchlosť šírenia zvuku v danom prostredí
<i>CF</i>	[-]	Crest factor – činiteľ výkmitu
<i>D1</i>	[m]	Priemer vonkajšej dráhy
<i>D2</i>	[m]	Priemer vnútornej dráhy
<i>DV</i>	[m]	Priemer valivého členu
<i>F</i>	[N]	Sila
<i>f</i>	[Hz]	Frekvencia
<i>f₀</i>	[Hz]	Základná frekvencia snímania záznamu
<i>f_{BPFI}</i>	[Hz]	Frekvencia prejavu vady na vnútornej dráhe
<i>f_{BPFO}</i>	[Hz]	Frekvencia prejavu vady na vonkajšej dráhe
<i>f_{BSF}</i>	[Hz]	Frekvencia prejavu vady valivého členu
<i>f_{BSF}</i>	[Hz]	Frekvencia prejavu vady valivého členu
<i>f_c</i>	[Hz]	Frekvencia otáčania klietky
<i>F_{FTF}</i>	[Hz]	Frekvencia otáčania klietky
<i>F_{FTF}</i>	[Hz]	Frekvencia otáčania klietky
<i>f_i</i>	[Hz]	Frekvencia otáčania vnútorného krúžku
<i>F_k</i>	[-]	Koeficient Fourierovej rady
<i>f_o</i>	[Hz]	Frekvencia otáčania vonkajšieho krúžku
<i>f_{ot}</i>	[Hz]	Otáčková frekvencia
<i>f_{pp}</i>	[Hz]	Postranné pásma zubovej frekvencie
<i>f_r</i>	[Hz]	Frekvencia otáčania valivého členu
<i>f_s</i>	[Hz]	Vzorkovacia frekvencia
<i>f_z</i>	[Hz]	Zubová frekvencia
<i>H</i>	[-]	Veľkosť skoku medzi segmentmi STFT
<i>I₀</i>	[W·m ²]	Hodnota referenčnej intenzity
<i>K</i>	[-]	Číslo rádu harmonickej frekvencie
<i>K</i>	[-]	Kurtosis – špicatosť
<i>k_s</i>	[N·m ⁻¹]	Tuhosť
<i>L</i>	[-]	Číslo postranného pásma

L_I	[dB]	Hladina akustickej intenzity
L_p	[dB]	Hladina akustického tlaku
$L_{p(A)}$	[dB]	Hladina akustického tlaku, vážená filtrom A
L_W	[dB]	Hladina akustického výkonu
m	[kg]	Hmotnosť
n	[-]	Počet valivých členov
p_0	[Pa]	Hodnota referenčného tlaku
P_0	[W]	Hodnota referenčného výkonu
PD	[m]	Roztečený priemer
RD	[m]	Priemer valčeka
s	[m]	Dráha
s	[m]	Dráha
t	[s]	Čas
T	[s]	Periódna
v	[m·s ⁻¹]	Rýchlosť
X	[m]	Amplitúda vynúteného kmitania
x_{ef}	[m]	efektívna hodnota harmonického signálu výchylky
x_{str}	[m]	stredná hodnota harmonického signálu výchylky
z	[-]	Počet zubov
β	[°]	Stykový uhol
λ	[m]	Vlnová dĺžka
φ	[rad·s ⁻¹]	Fázový posun výchylky za budiacou silou
φ_F	[rad·s ⁻¹]	Počiatočný fázový posun budiacej sily
ω	[rad·s ⁻¹]	Uhlová rýchlosť
Ω	[Hz]	Vlastná frekvencia