



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV VÝROBNÍCH STROJŮ, SYSTÉMŮ A ROBOTIKY

INSTITUTE OF PRODUCTION MACHINES, SYSTEMS AND ROBOTICS

TVORBA SW PRO GENEROVÁNÍ SIGNÁLU SIMULUJÍCÍ ZÁVADY ROTAČNÍCH SYSTÉMŮ

SW FOR SIGNAL GENERATION SIMULATING ROTARY SYSTEM FAULTS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Marek Martínek

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Daniel Zuth, Ph.D.

BRNO 2021

Zadání diplomové práce

| | |
|-------------------|--|
| Ústav: | Ústav výrobních strojů, systémů a robotiky |
| Student: | Bc. Marek Martínek |
| Studijní program: | Strojní inženýrství |
| Studijní obor: | Kvalita, spolehlivost a bezpečnost |
| Vedoucí práce: | Ing. Daniel Zuth, Ph.D. |
| Akademický rok: | 2020/21 |

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:

Tvorba SW pro generování signálu simulující závady rotačních systémů

Stručná charakteristika problematiky úkolu:

Práce se bude zabývat tvorbou algoritmu pro generování simulovaného signálu z vibrodiagnostického zařízení. SW bude generovat jak data v časové oblasti, tak frekvenční spektra, dle navolené závady rotačních strojů a to v podobě číselných dat i grafického zobrazení. SW bude schopen generovat signály simulující nevývahu, nesouosost a také závady ložisek včetně jakékoliv kombinace.

Cíle diplomové práce:

Rešerše v oblasti SW pro matematické/kinematické simulace a oblasti vibrodiagnostiky – projevy závad rotačních strojů.

Vybrat vhodný SW.

Popsat naprogramované algoritmy.

Popsat simulace jednotlivých poruch včetně grafických výstupů.

Vytvořit dokumentaci k navrženému SW.

Vlastní závěr a doporučení pro praxi.

Seznam doporučené literatury:

BLATA, Jan a Janusz JURASZEK. Metody technické diagnostiky: teorie a praxe = Metody diagnostyki technicznej : teorie a praktyka. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2013. ISBN 978-80-248-2997-5.

BROCH, Jens Trampe. Mechanical Vibration and Shock Measurements. 2nd Edition. Naerum: Brüel & Kjaer, 1984. ISBN 8787355361.

DANIŠ, Stanislav. Základy programování v prostředí Octave a Matlab. Praha: Matfyzpress, 2009. ISBN 978-80-7378-082-1.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2020/21

V Brně, dne

L. S.

doc. Ing. Petr Blecha, Ph.D.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

ABSTRAKT

Tato diplomová práce se věnuje návrhu a tvorbě algoritmu pro generování simulovaného signálu z vibrodiagnostického zařízení. První část je zaměřená na teoretické seznámení se s vibrodiagnostikou a charakteristikou jednotlivých závad rotačních strojů. Další část se zabývá možnostmi matematických a kinematických simulací za pomoci počítačových programů. Hlavní část této práce se věnuje tvorbě programu pro generování simulovaného signálu. V poslední části je názorně předveden princip simulace konkrétních závad rotačních strojů.

ABSTRACT

This diploma thesis deals with the design and creation of an algorithm for generating simulated signal data from a vibration diagnostics device. The first part is focused on theoretical acquaintance with vibration diagnostics and characteristics of individual defects of rotary machines. The next part deals with the possibilities of mathematical and kinematic simulations using a computer software. The main part of this work is dedicated to design and creation of software for generating simulated signal data. In the last part, the principle of simulation of specific defects of rotary machines is clearly demonstrated.

KLÍČOVÁ SLOVA

Vibrodiagnostika, frekvenční analýza, závady rotačních strojů, závady ložisek, simulace poruchových signálů, Python

KEYWORDS

Vibration diagnostics, frequency analysis, faults of rotating machines, bearing faults, simulation of fault signals, Python

BIBLIOGRAFICKÁ CITACE

MARTÍNEK, Marek. *Tvorba SW pro generování signálu simulující závady rotačních systémů*. Brno, 2021 Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/132673>. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav výrobních strojů, systémů a robotiky. Vedoucí práce Ing. Daniel Zuth Ph.D.

PODĚKOVÁNÍ

Děkuji mému vedoucímu diplomové práce Ing. Danielovi Zuthovi Ph.D. za odborné vedení, připomínky a rady. Rovněž bych chtěl poděkovat svým rodičům za podporu během studia.

ČESTNÉ PROHLÁŠENÍ

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. Daniela Zutha Ph.D. a s použitím literatury uvedené v seznamu.

V Brně dne 21. 5. 2021

.....

Bc. Martínek Marek

OBSAH

| | | |
|----------|--|-----------|
| 1 | ÚVOD | 15 |
| 2 | VIBRODIAGNOSTIKA | 17 |
| 2.1 | Měřené veličiny charakterizující vibrace..... | 17 |
| 2.1.1 | Amplituda | 17 |
| 2.1.2 | Frekvence..... | 17 |
| 2.1.3 | Fáze..... | 17 |
| 2.1.4 | Výchylka..... | 17 |
| 2.1.5 | Rychlost | 18 |
| 2.1.6 | Zrychlení..... | 18 |
| 2.2 | Snímače vibrací..... | 19 |
| 2.2.1 | Snímače výchylky..... | 19 |
| 2.2.2 | Snímače rychlosti | 20 |
| 2.2.3 | Snímače zrychlení..... | 21 |
| 2.3 | Vibrodiagnostický signál | 22 |
| 2.3.1 | Analýza vibrodiagnostického signálu v časové oblasti | 22 |
| 2.3.2 | Analýza vibrodiagnostického signálu ve frekvenční oblasti | 25 |
| 2.4 | Závady rotačních strojů | 27 |
| 2.4.1 | Nevyváženost..... | 28 |
| 2.4.2 | Nesouosost..... | 31 |
| 2.4.3 | Ohnutý hřídel | 33 |
| 2.4.4 | Mechanické uvolnění..... | 33 |
| 2.4.5 | Ozubené a řemenové převody | 33 |
| 2.4.6 | Valivá ložiska | 35 |
| 2.4.7 | Rezonance..... | 38 |
| 3 | SOFTWARE PRO SIMULACI MECHANICKÝCH SOUSTAV..... | 39 |
| 3.1 | Obecný postup pro simulaci metodou konečných prvků..... | 39 |
| 3.1.1 | Inženýrský postup..... | 39 |
| 3.1.2 | Softwarový postup..... | 41 |
| 3.1.3 | Prvky..... | 42 |
| 3.1.4 | Solver – řešitel úlohy | 43 |
| 3.1.5 | Vibrační analýza | 43 |
| 3.2 | Přehled vybraných programů pro simulaci mechanických soustav | 43 |
| 3.2.1 | Ansys Mechanical..... | 43 |
| 3.2.2 | Abaqus/CAE..... | 44 |
| 3.2.3 | SolidWorks | 44 |
| 4 | PRAKTICKÁ ČÁST | 45 |
| 4.1 | Použitý software a programovací jazyky..... | 45 |
| 4.1.1 | Python..... | 45 |
| 4.1.2 | PyQt5 | 46 |
| 4.1.3 | PyCharm | 47 |
| 4.2 | Použité knihovny | 48 |
| 4.2.1 | NumPy | 48 |
| 4.2.2 | Matplotlib | 48 |
| 4.2.3 | PyInstaller..... | 48 |
| 4.2.4 | Cgitb | 49 |

| | | |
|----------|--|-----------|
| 4.3 | Program Generátor signálů..... | 50 |
| 4.3.1 | Účel programu..... | 50 |
| 4.3.2 | Rozvržení uživatelského rozhraní..... | 50 |
| 4.3.3 | Panel vstupních parametrů..... | 51 |
| 4.3.4 | Panel grafu a pruh nástrojů grafu..... | 55 |
| 4.3.5 | Spuštění simulace..... | 56 |
| 4.4 | Princip funkce programu a jeho struktura..... | 58 |
| 4.4.1 | Multithreading..... | 58 |
| 4.4.2 | Soubor main.py..... | 59 |
| 4.4.3 | Soubor fault_methods.py..... | 60 |
| 4.4.4 | Soubor mplwidget.py..... | 60 |
| 4.4.5 | Soubor gui.ui..... | 61 |
| 4.4.6 | Instalace..... | 61 |
| 4.5 | Samostatný skript..... | 61 |
| 4.6 | Tvorba vibrodiagnostického signálu..... | 62 |
| 4.6.1 | Závady ložisek..... | 62 |
| 4.6.2 | Závady ozubení..... | 67 |
| 4.6.3 | Nevyváženost a nesouosost..... | 73 |
| 5 | ZHODNOCENÍ A DISKUZE..... | 75 |
| 6 | ZÁVĚR..... | 77 |
| 7 | SEZNAM POUŽITÝCH ZDROJŮ..... | 79 |
| 8 | SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK..... | 83 |
| 8.1 | Seznam tabulek..... | 83 |
| 8.2 | Seznam obrázků..... | 83 |
| 8.3 | Seznam použitých zkratk a symbolů..... | 85 |
| 9 | SEZNAM ELEKTRONICKÝCH PŘÍLOH..... | 87 |

1 ÚVOD

Vývoj technologií v oblasti průmyslové údržby hraje důležitou roli při zvládnání výzev, které tento obor doprovází. Jednou ze zásadních výzev je včasné odhalení možných poruch a minimalizace či kompletní eliminace neočekávaných prostojů ve výrobě. Pro uspokojení zákazníka je důležité dodávat produkty včas a v požadované kvalitě. Tento vývoj má za následek stále rostoucí význam technické diagnostiky. [1]

Moderní diagnostika a údržba je v dnešní době ve fázi, kdy už nestačí předcházet a předvídat selhání jednotlivých strojů, ale je třeba vytvořit komplexní systém proaktivního monitorování stavu, kde i ty nejmenší korekce jsou provedeny včas. [2]

Pro včasné odhalení snížené kvality výroby, nebo poruchy, lze využít různé metody technické diagnostiky. Jednou z nejčastěji používaných metod technické diagnostiky je právě vibrodiagnostika. To je dáno především tím, že velké množství nejen výrobních strojů obsahuje rotační komponenty. Vibrodiagnostika ale nemusí sloužit pouze k zajištění výrobní kvality, může být použita i k zajištění bezpečnosti. Jako příklad lze uvést monitorování vibrací větrné turbíny. [2]

Pozorováním vibrací částí strojů lze nalézt vztah mezi rozsahem přirozených frekvencí a poruchovým režimem. Měřením a analýzou nasbíraných dat o vibracích je technik schopen získat cenné informace o stavu zařízení, nalézt příčiny selhání a naplánovat včasný zásah údržby. Pozorováním trendu frekvenčního spektra lze také zjistit informace o typu poruchy, její závažnosti a rovněž lze stanovit zbývající životnost vadné součásti. Ke správnému určení jednotlivých závad je třeba, aby byl technik dobře seznámen s monitorovaným zařízením z technické stránky a aby znal projevy jednotlivých závad, které se mohou zařízení týkat. [1]

Cílem této práce je provést rešerši v oblasti vibrodiagnostiky, popsat jednotlivé závady rotačních strojů a informovat o možnostech matematických a kinematických simulací pomocí počítače. Druhá část diplomové práce se zabývá tvorbou algoritmu pro generování simulovaného signálů z vibrodiagnostického zařízení.

Diplomová práce je rozdělena na úvod, čtyři kapitoly a závěr. První kapitola představuje teoretické základy vibrodiagnostiky a charakteristiky vibrací. Dále se kapitola věnuje jednotlivým typům snímačů vibrací a závad rotačních strojů. Konec kapitoly je zaměřen na analýzu vibrodiagnostického signálu.

Druhá kapitola se věnuje softwarovým řešením simulace mechanických soustav metodou konečných prvků. Následuje krátký přehled programů pro simulaci mechanických soustav.

Třetí kapitola dokumentuje tvorbu programu pro generování simulovaného signálu v programovacím jazyce Python. V první části je popsán software použitý pro tvorbu programu. V další části je popsán princip funkce programu. Následuje představení programu a návod k obsluze grafického rozhraní. V poslední části je popsán princip simulace jednotlivých poruch včetně grafického výstupu. Celá kapitola je doplněna vybranými částmi kódu pro ilustraci funkce programu.

V poslední kapitole jsou shrnuty dosažené cíle v rámci návrhu a realizace aplikace a návrhy na její další rozvoj.

2 VIBRODIAGNOSTIKA

Různé druhy diagnostiky a metody monitorování stavu strojů jsou široce používány téměř v každém průmyslovém odvětví s aplikací v automatizaci, prediktivní údržbě a kontrole kvality. Nejčastěji používané techniky jsou založené na analýze opotřebení, a častěji na analýze vibrací. Vibrodiagnostika je běžně používaná metoda pro diagnostiku stavu rotačních strojů s pevnou základnou díky relativně rychlému sběru dat a jejich interpretaci ve srovnání s ostatními off-line metodami. Vzhledem k tomu, že jsou data shromažďována jako digitálně vzorkované signály v časové doméně, lze tyto data, pomocí různých metod analýzy vibračních signálů, dále upravovat za pomoci počítače. Tímto vibrodiagnostika kontrastuje s analýzou olejů a opotřebení, která často spoléhá na složitou a rozsáhlou chemickou analýzu a interpretaci těchto dat zkušeným technologem. [1]

2.1 Měřené veličiny charakterizující vibrace

Jakoukoli vibraci popisují tři charakteristické hodnoty: jak moc (amplituda), jak rychle (frekvence) a její fáze. Podle typu stroje či zařízení se zvolí, zdali se bude pozorovat výchylka, rychlost či zrychlení. Jelikož jsou všechny tři parametry spolu matematicky provázány, lze měřením jedné veličiny vyvodit hodnoty ostatních dvou. [2]

2.1.1 Amplituda

Termín amplituda je všeobecně uznáván jako termín používaný k popisu „jak moc“ bez ohledu na měřený parametr. Amplituda vibrací se měří třemi hlavními veličinami: Výchylka, rychlost a zrychlení. [3]

2.1.2 Frekvence

Množství času potřebného k dokončení jednoho celého cyklu vibrace se nazývá perioda vibrace. Pokud například stroj dokončí jeden celý cyklus vibrace za 1/60 sekundy, pak je perioda vibrace považována za 1/60 sekundy. [3]

Frekvence vibrací je jednoduše měřítkem počtu úplných cyklů, ke kterým dojde ve specifikovaném časovém úseku, jako jsou například cyklů za sekundu či cyklů za minutu. [3]

2.1.3 Fáze

Třetí charakteristikou nutnou k popisu jakékoli vibrace je fáze. Fáze je definována jako úhel mezi okamžitou polohou vibrující části a pevnou referenční pozicí. [3]

Pokud mluvíme o fázi ve smyslu měření fáze pak se fázovým úhlem rozumí úhel, o který se otočí hřídel od počátku měření až po bod, ve kterém naměří snímač maximální kladnou odezvu (výchylka, rychlost, zrychlení). [3]

2.1.4 Výchylka

Výchylkou vibrací se u měřeného objektu rozumí jeho vzdálenost nebo poloha vzhledem k referenční poloze. Periodickým, pravidelným vychýlením z referenční polohy vzniká tzv. harmonické kmitání. Rovnici harmonického kmitání pak popisuje rovnice (1). [2]

$$y = A \cdot \sin(2\pi \cdot f \cdot t) \quad (1)$$

2.1.5 Rychlost

Rychlost je definována jako změna výchylky za čas. V případě jednoduchého harmonického pohybu se rychlost neustále mění a kmitá stejně tak jako výchylka. Když je hodnota výchylky maximální, rychlost je nulová. Když je výchylka nulová, rychlost je maximální. Rychlost je pak dána vztahem (2). [4]

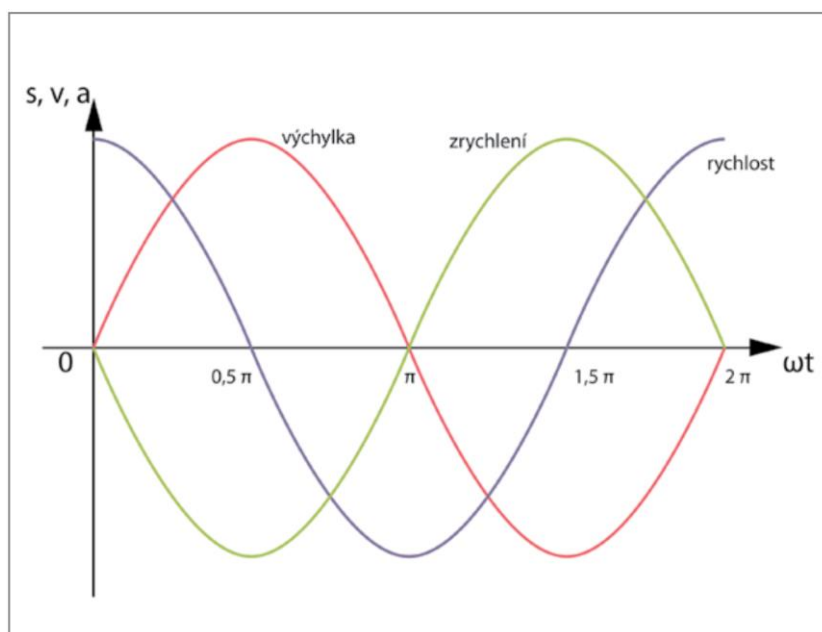
$$v = \frac{dy}{dt} = 2\pi \cdot f \cdot A \cdot \cos(2\pi \cdot f \cdot t) \quad (2)$$

2.1.6 Zrychlení

Zrychlení je definováno jako změna rychlosti za čas a také osciluje jednoduchým harmonickým pohybem. Uvažujme hmotu na pružině. Je-li výchylka nulová, je zrychlení také nulové. Jakmile je výchylka maximální, zrychlení je také maximální, protože pružina působí maximální silou. Síla vyvíjená pružinou působí v opačném směru oproti výchylce. Zrychlení je pak dáno vztahem (3). [4]

$$a = \frac{dv}{dt} = -(2\pi \cdot f)^2 \cdot A \cdot \sin(2\pi \cdot f \cdot t) \quad (3)$$

Jak výchylka, tak i rychlosti a zrychlení mají tvar sinusoidy. Liší se ale fázovým posunem, ten u rychlosti činí 90° nebo $\pi/2$ a u zrychlení 180° nebo π vzhledem k výchylce, jak lze vidět na Obr. 1). [5]



Obr. 1) Vztah mezi výchylkou, rychlostí a zrychlení s ohledem na čas [1]

2.2 Snímače vibrací

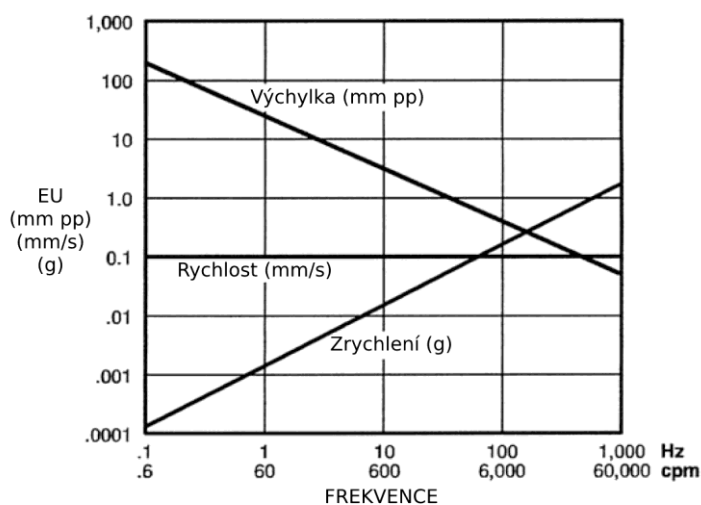
První krok, který je při výběru vhodného typu snímače nutné provést je určit, zdali se jedná o vibrace relativní nebo absolutní. V případě absolutních vibrací se z pravidla jedná o pohyb sledovaného tělesa (stroje či zařízení) vůči zemi. Relativní vibrace popisují pohyb objektu vůči uměle vytvořené, či vhodně zvolené základně. Často volenou základnou je rám stroje, ten se sám nadále může hýbat vůči zemi. [2]

Dalším důležitým kritériem je dostupnost samotné sledované části stroje. Ta je buďto přímá, nebo nepřímá s využitím přenosu chvění na přístupnější místo. Toto hraje roli například v případě kmitů hřídele uložené v kluzných ložiscích, kde mazivo utlumí kmity hřídele. V případě takového stroje je třeba zvolit relativní snímač kmitů hřídele proti rámu, případně může být vhodné navíc měřit i absolutní vibrace rámu. [2]

Mechanické kmitání je mimo frekvenci určeno společně výchylkou, rychlostí a zrychlením, které jsou navzájem závislé. Princip funkce snímače vibrací je založen na pohybu seismické hmoty o hmotnosti m vůči objektu o hmotnosti M , jehož vibrace se měří. Tuto interakci lze popsat vztahem (4). [6]

$$m\ddot{y} + b\dot{y} + ky = F_b = M\ddot{x} \quad (4)$$

Při volbě vhodné veličiny k měření se lze řídit grafem níže (Obr. 2)



Obr. 2) Výběrová kritéria pro sledování dynamických účinků různých rozsahů rychlostí (frekvencí) [2]

2.2.1 Snímače výchylky

Existuje mnoho typů snímačů výchylky. Lze je rozdělit dvěma způsoby, buďto na kontaktní a bezkontaktní anebo podle principu jejich fungování. Snímače výchylky nachází uplatnění zejména u strojů pracujících za velmi nízkých otáček z pravidla v rozmezí 0-300 Hz. [2]

Kontaktní senzor je založen na principu snímání změny výchylky přímým kontaktem senzoru s měřeným objektem. Jsou náchylné na opotřebení a citlivé na kvalitu měřeného povrchu. [2]

Bezkontaktní senzor – nedochází k přímému kontaktu senzoru s měřeným předmětem. Nejčastější typ senzoru, se kterým se lze v praxi setkat, pracuje na principu sledování změn v elektrickém nebo magnetickém poli. U tohoto typu senzorů nedochází k opotřebení. [2]

Kapacitní senzor měří výchylku, vzdálenost či polohu jakéhokoli elektricky vodivého předmětu. Ke své funkci vyžaduje dostatečnou čistotu provozu. Umožňuje měření ve vysokém rozlišení a disponuje dobrou tepelnou stabilitou. Typický rozsah měření je 0,05 až 10 mm. [2]

Magneticko-indukční snímače fungují na principu vířivých proudů (Eddy-Current). Pracují s magnetickým polem a jsou schopny měřit polohu či změnu polohy jakéhokoli vodivého předmětu. Změna polohy vytváří střídavý proud ve snímací cívkě sondy. Střídavé magnetické pole poté ve vodivém materiálu indukuje malé proudy (vířivé proudy). Interakce magnetických polí poté představuje změnu polohy mezi sondou a objektem. Jak se poloha mění, senzor zaznamenává změnu v interakci polí a generuje napětí úměrné vzdálenosti. Indukčnostní senzor disponuje o něco horším rozlišením měření v porovnání s kapacitními senzory. Jejich výhodou je schopnost pracovat ve velmi znečištěném a náročném prostředí. Typický rozsah měření je mezi 0,5 až 80 mm. [2]

Laserový senzor s principem zaostření funguje na principu triangulace. Světelný bod se promítá na měřený povrch pomocí laseru. Prostřednictvím přijímací optiky je odraz mapován na světlocitlivý prvek. Vzdálenost od objektu je určena na základě určení polohy světelného bodu. Laserové senzory měří výchylku, rozměr, polohu a pozici objektu z jakéhokoli materiálu. Dokáží měřit na vzdálenost mezi 2 až 2000 mm. Vyžadují čisté pracovní prostředí.

Laserový snímač vzdálenosti funguje na principu měření doby letu laserového paprsku k objektu a zpět. Tento princip umožňuje měření na velké vzdálenosti a to až 250 m, ale je nevhodný pro velmi přesné měření. [2]

Ultrazvukový senzor generuje krátký ultrazvukový signál, který se po odrazu od měřeného objektu vrací zpět k senzoru. Vzdálenost mezi senzorem a objektem je vypočítána z doby, kterou signál putoval tam a zpět a média ve kterém se pohyboval. Tento typ senzoru vyžaduje minimální vzdálenost od měřeného objektu. Typický měřicí rozsah je mezi 0,1 m až několik metrů. Mají nízkou přesnost a vyžadují velkou plochu měřeného objektu. [2]

2.2.2 Snímače rychlosti

V dnešní době se s tímto typem snímače již téměř nesetkáváme. Ve většině případů bývá nahrazen akcelerometrem, kde rychlost kmitavého pohybu je určena integrací signálu z akcelerometru. Případně lze rychlost získat také derivací signálu snímače výchylky. Uplatnění pro velocimetry lze najít například ve velmi specifických podmínkách, jako jsou teploty přesahující 500°C. Obecně platí, že rychlost měřená v rozmezí 10-1000 Hz vykazuje nejlepší známky závažnosti poruchy. [7]

Elektromotorické – Magnet senzoru rychlosti, sloužící jako seismická hmota, je zavěšen na pružině, která se pohybuje měřicí cívkou. V důsledku vibrací se cívka pohybuje v magnetickém poli permanentního magnetu a generuje elektromotorickou sílu. Takto cívka generuje napětí, které je úměrné vibracím a rychlosti. [6]

Piezoelektrické – Jedná se o akcelerometry s vnitřním integračním obvodem. Jsou vhodné zejména u zařízení, kde se setkáváme s vysokofrekvenčním šumem (například kavitace). Princip piezoelektrických akcelerometrů je popsán níže v kapitole 2.2.3. [8]

2.2.3 Snímače zrychlení

Ke snímání zrychlení slouží akcelerometry. Existuje mnoho různých typů akcelerometrů pro průmyslové a vědecké aplikace. Toto umožňuje technikům používat širokou škálu akcelerometrů pro různé účely, například pro monitorování stavu stroje nebo ke krátkodobým testům. Jeden z nejdůležitějších parametrů akcelerometru je typ odezvy signálu. Dle toho dělíme akcelerometry na typ AC, nebo typ DC. Z pravidla se zrychlení měří u strojů s vysokou rychlostí a to až 20 kHz. Nicméně samotné akcelerometry jsou schopné měřit v celém rozsahu 0-20 kHz. V praxi se jedná o nejčastěji zastoupený typ senzorů. [7]

Typ AC nejsou schopné měřit statické zrychlení, proto jsou vhodnou volbou pro dynamické testy. Jsou také užitečné pro výpočet rychlosti a polohy testovaného objektu za pomoci integrace křivky zrychlení. Nejčastějším typem AC akcelerometrů, jsou akcelerometry s piezoelektrickými prvky. [2]

Typ DC jsou schopné měřit konstantní zrychlení jako je například gravitace. Tím pádem jsou teoreticky schopny měřit vibrace až na hranici 0 Hz. Nejčastějším typem DC akcelerometrů jsou kapacitní MEMS a piezorezistentní senzory. [2]

Další možný způsob dělení akcelerometrů, je na základně jejich fyzikálních funkcí.

Piezoelektrický akcelerometr využívá piezoelektrického jevu. Jedná se o schopnost piezoelektrického materiálu (piezokrystalu) generovat elektrické napětí při své deformaci. Generovaný náboj je přímo úměrný zrychlení, které na senzor působí. Vyprodukovaný náboj se obvykle převádí na napětí pomocí integrované elektroniky – nábojový zesilovač a obvod pro úpravu signálu. Základna snímače je oddělená od vlastního snímacího mechanismu, to umožňuje vyrábět snímače vhodné pro použití při vyšších teplotách či s vyšší mechanickou odolností. Jedná se o akcelerometry typu AC. Frekvenční rozsah senzorů je od 0,1 Hz až po cca 40 kHz. [2]

Piezorezistivní akcelerometr využívá piezorezistivního jevu, který souvisí se změnou odporu polovodiče v důsledku aplikace mechanického napětí. Tento typ akcelerometrů používá piezorezistivní substrát namísto piezoelektrického krystalu. Síla vzniklá působením seismické hmoty, mění odpor krystalů, které jsou obvykle uspořádány do Wheatstoneova můstku. Změna odporu je úměrná zrychlení. Jedná se o akcelerometry typu DC. Frekvenční rozsah bývá tedy od 0 Hz až do 7 kHz. Dynamický rozsah je až 2000 g. Jsou tedy preferovanou volbou pro aplikace vysokými rázy. [9]

Kapacitní akcelerometry snímají změnu elektrické kapacity s ohledem na zrychlení. Převod mezi mechanickou energií na elektrickou se nejčastěji realizuje elektrostaticky. Mikrometrová mechanická soustava je sestavena z miniaturních proměnných kondenzátorů, kde jedna elektroda je pevná a druhá je připevněna na detekční část senzoru, kterou je v případě akcelerometrů seismická hmota uložená na pružinách. Obvykle je mechanická i elektronická (vyhodnocovací) část senzoru uložena v jednom mikročipu. Jedná se o akcelerometry typu DC, tudíž umí měřit frekvence od 0 Hz, nicméně frekvenční rozsah je poměrně nízký v řádu stovek Hz. Nejčastěji se jedná o řešení vhodné pro nízkorychlostní stroje. [2]

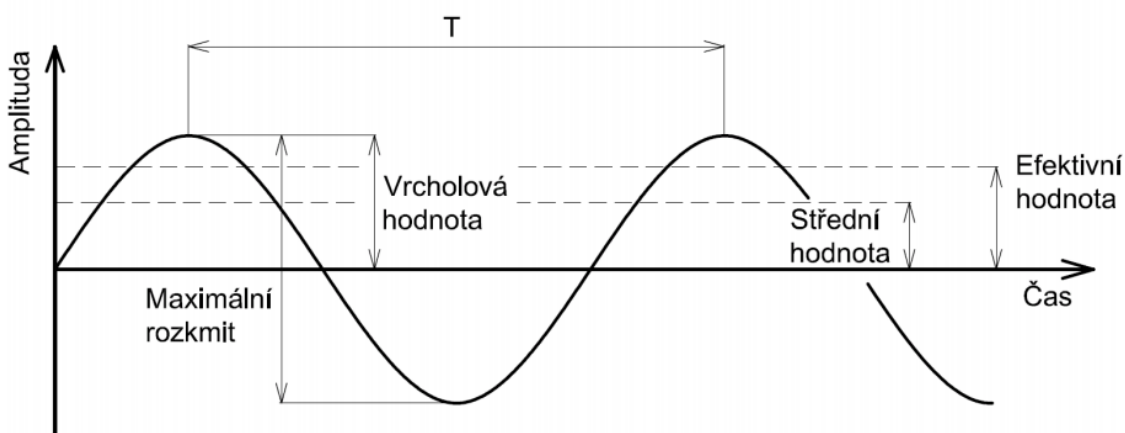
2.3 Vibrodiagnostický signál

Vzhledem k rozsahu měření a komplexitě měřeného systému je třeba vhodně zvolit i způsob jakým se bude analýza provádět. První možnost je analyzovat v časové oblasti, tato možnost poslouží v případě, že existuje jediný nebo alespoň dominantní zdroj vibrací. V případě, že je zdrojů vibrací více či dochází k rušení, je třeba přistoupit k analýze signálu ve frekvenční oblasti, kde jsou možnosti omezené pouze zvoleným diagnostickým systémem a zkušenostmi technologa. [3]

2.3.1 Analýza vibrodiagnostického signálu v časové oblasti

Vyhodnocování vibrací v časové oblasti (amplituda vůči času), je při kvantifikaci síly vibračního profilu omezeno na několik parametrů: amplituda, rozkmit (špička-špička, peak-to-peak) a efektivní hodnota (root mean square, RMS). Obrázek (Obr. 3) zobrazuje jednoduchou sinusovou vlnu s těmito parametry. Ideální harmonický signál je popsán vztahem (5). [6]

$$X(t) = X \cdot \sin(\omega t) \quad (5)$$

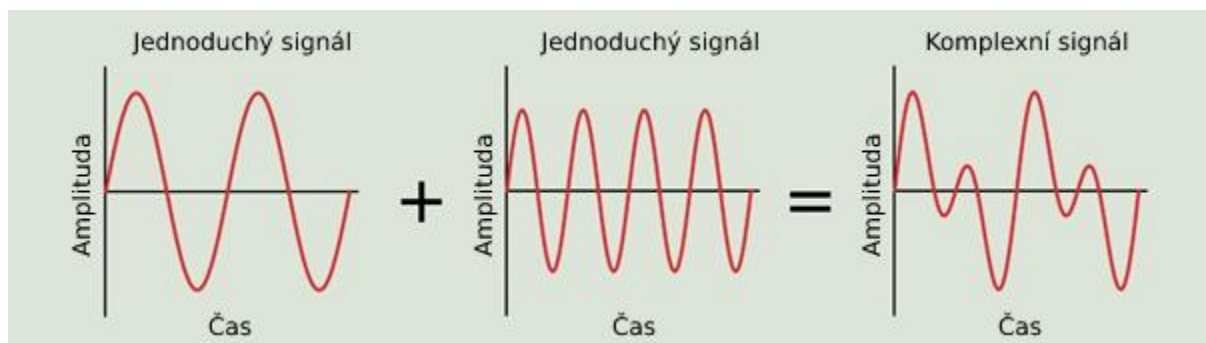


Obr. 3) Časová charakteristika vibrace s parametry [10]

Vrcholová hodnota nebo amplituda je užitečná informace pro šokové události (rázy), ale nebere v úvahu čas, a tedy ani energii v daném momentě. To stejné platí i pro hodnotu špička-špička s přidanou hodnotou spočívající ve zjištění maximální odchylky od referenční polohy, což je užitečné při pohledu na informace týkajících se posunutí, konkrétně vůle. [10]

Hodnota RMS je obecně nejužitečnější, jelikož je přímo úměrná energii vibrace, a tudíž i destruktivní schopnosti vibrace. Efektivní hodnota také bere v potaz historický časový průběh vlny. [10]

Složená vibrace (Obr. 4) je součet několika jednoduchých vibrací. Příčinou vzniku těchto jednoduchých vibrací jsou jednotné pohyblivé i nepohyblivé části stroje. S ohledem na tuto skutečnost je jasné, že vibrační křivka stroje není sinusoida, nýbrž naopak se může jednat o velmi komplexní signál. [11]

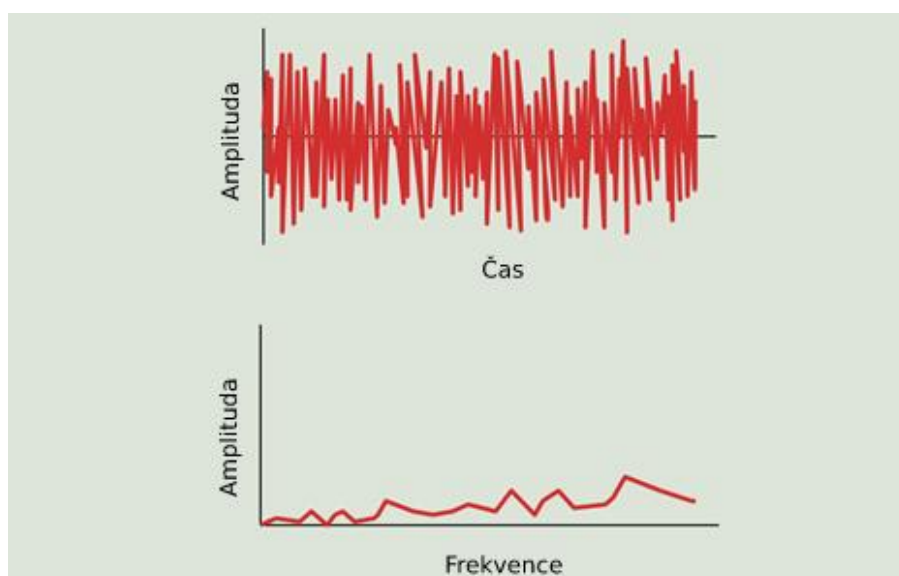


Obr. 4) Skládání jednoduchých vibrací v časové doméně [11]

Jak lze vidět na obrázku (Obr. 4), sloučením dvou jednoduchých signálů s různou frekvencí vznikne komplexní složená vibrace. Již i u takto jednoduchého případu, není jednoduché z výsledné vlny určit amplitudu a frekvenci jednotlivých složek. Absolutní většina vibračních signálů, se kterou se setkáváme v praxi je mnohem složitější a může být velmi obtížné je správně interpretovat. [11]

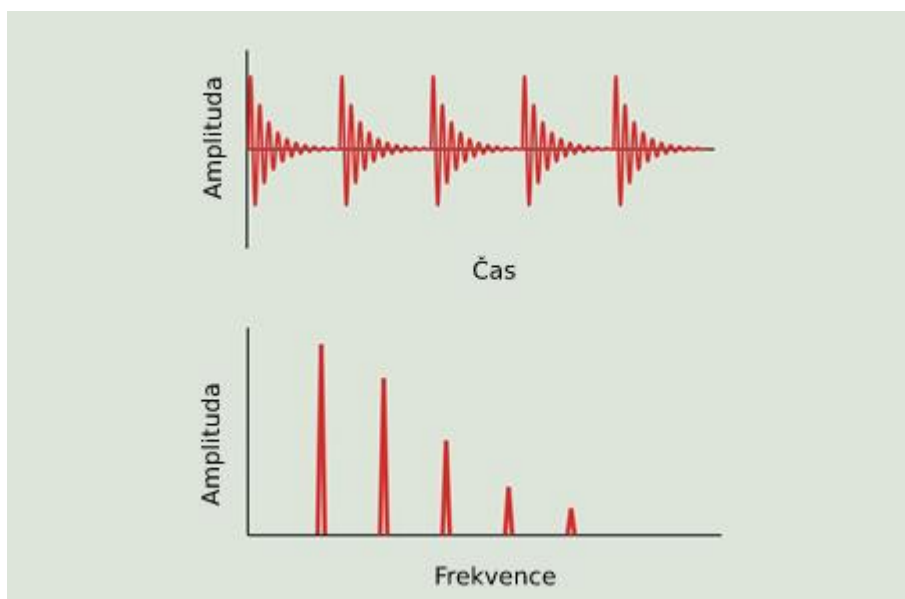
U jednoduchých sinusových vln lze frekvenci vibrací určit sledováním tvaru vlny v časové doméně. S přidáním dalších frekvenčních složek a šumu, je pro získání jasnějšího pohledu na věc nutné provedení frekvenční analýzy. Nicméně existují případy, kdy analýza vlnění v časové oblasti může být velmi užitečná. [11]

Náhodné vibrace (Obr. 5) nespĺňujú žádné kritéria opakujícího se signálu nebo je příliš obtížné zjistit kde cyklus začíná a kde končí. Tento typ vibrací je obecně spojován s turbulencí v dmychadlech a čerpadlech, v mazacích soustavách, při styku kov na kov valivých prvků anebo kavitace v čerpadlech. Tento typ vibrací se lépe interpretuje ve frekvenční oblasti než v časové. [11]



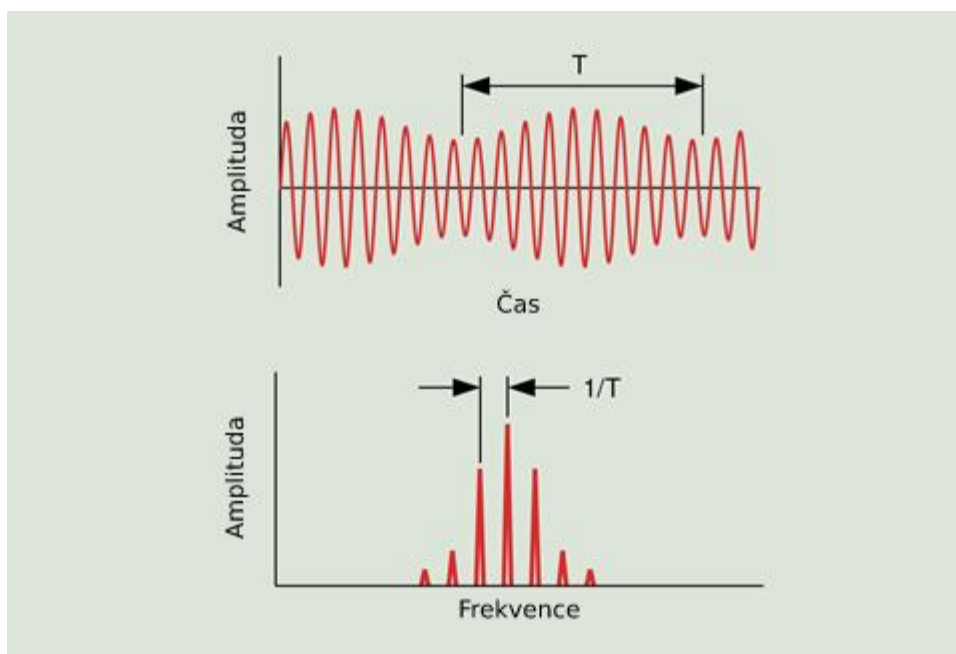
Obr. 5) Náhodná vibrace [11]

Přerušované rytmy (Obr. 6) jsou spojovány s kontinuálními rázy, které vytvářejí opakující se signál. Nejčastěji se vyskytují v převodech, valivých ložiscích, lopatek ventilátoru apod. Tento typ signálu má tendenci utíchat v důsledku tlumení okolním médiem. [11]



Obr. 6) Přerušované rytmy [11]

Amplitudová modulace (Obr. 7) je typ signálu, u kterého dochází vlivem jiného signálu, obecně na nižší frekvenci, k variaci síly amplitudy. Frekvence, která je modulovaná se nazývá nosná frekvence. Ve frekvenčním zobrazení je nosič nejdůležitější složkou a ostatní složky, které vypadají jako by byly harmonické frekvence se nazývá frekvenční postranní pásmo (sideband). Toto postranní pásmo je symetricky na obou stranách od nosiče a jejich vzdálenost je rovna modulované frekvenci. Nejčastěji se lze s tímto typem vibrací setkat u převodovek. [11]

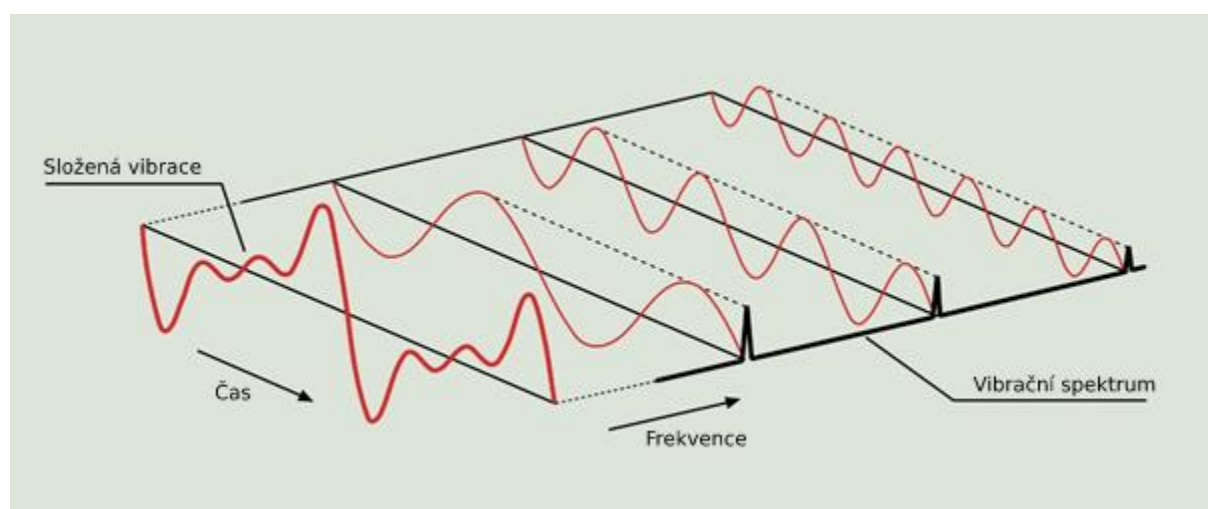


Obr. 7) Amplitudová modulace [11]

2.3.2 Analýza vibrodiagnostického signálu ve frekvenční oblasti

Jakýkoli signál je ve skutečnosti jen součet vícero jednoduchých sinusoid s různou frekvencí, amplitudou a fází. Fourierova řada je souhrn těchto sinusových vln. Za pomoci frekvenční analýzy dojde k dekonstrukci signálu na jeho jednotlivé složky. Výsledkem je amplituda zrychlení/vibrace jako funkce frekvence, což umožňuje provádět analýzu ve frekvenční doméně (či spektru), k získání lepšího porozumění o vibračním profilu. Většina vibračních analýz se provádí ve frekvenční oblasti. [10]

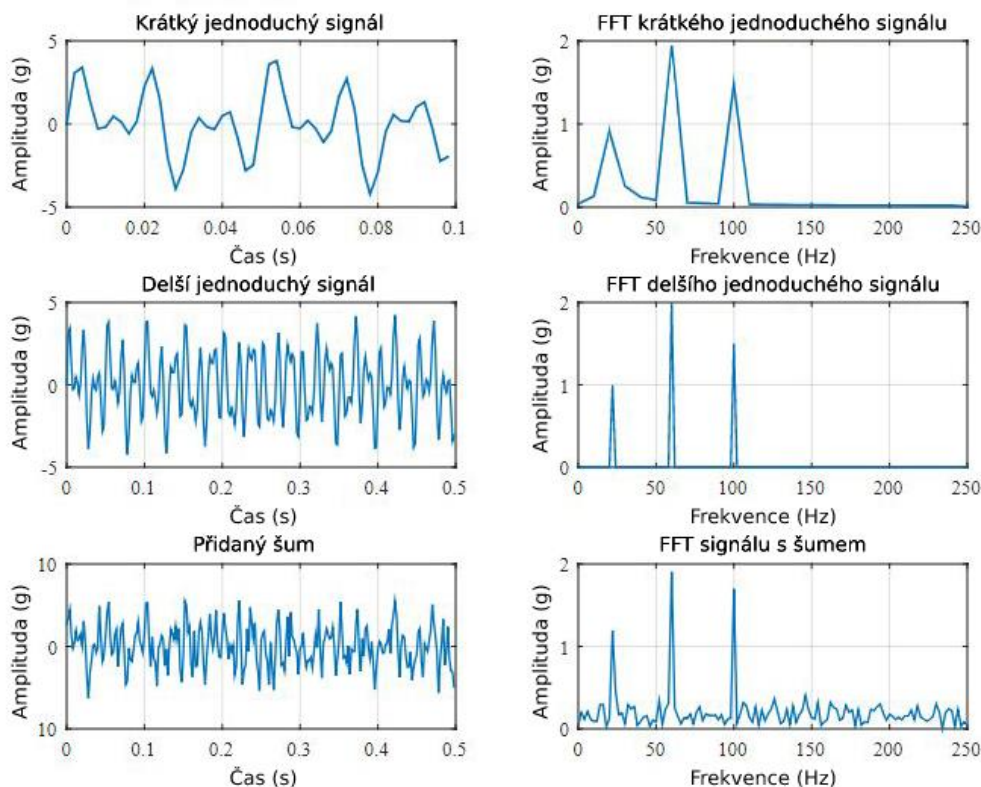
K získání frekvenčního spektra z časové oblasti (Obr. 8), slouží diskrétní Fourierova transformace (DFT). V dnešní době se ale spíše setkáváme s rychlou Fourierovou transformací (FFT). Jedná se totiž o efektivnější algoritmus DFT. [10]



Obr. 8) FFT zpracování komplexního vibračního signálu [11]

Příklad principu FFT

K ilustraci, jak lze FFT využít, je předveden na jednoduché křivce, složené ze tří různých komponent o frekvencích 22 Hz, 60 Hz a 100 Hz. Tyto frekvence mají amplitudu 1 g, 2 g a 1,5 g. Na grafech (Obr. 9) lze vidět, že i jednoduchá křivka může vypadat v časové doméně komplikovaně a také to, jakou roli hraje doba signálu. [10]



Obr. 9) Křivka s frekvenčními složkami 22, 60 a 100 Hz s různou délkou vzorků a přidaným šumem pro ilustraci užitečnosti FFT analýzy [10]

Pokud je vlnění vzorkováno na 500 Hz (500 vzorků za sekundu) a FFT se vytvoří z prvních 50 vzorků, bude výsledná FFT „zubatá“ a nepřesná. Amplitudy jednotlivých složek jsou navíc také poněkud nízké. Pokud se ale počet vzorků rozšíří na 250, může FFT vypočítat frekvence a amplitudy jednotlivých složek mnohem přesněji. Po přidání šumu se křivka v časové oblasti stane ještě více nepřehlednou. Nejsilnější stránkou FFT analýzy, je tedy schopnost identifikovat hlavní frekvence a při analýze dopomocť k určení původu jednotlivých vibrací. [10]

2.4 Závady rotačních strojů

Plánovaná údržba a údržba po poruše jsou běžně používané nástroje v průmyslu, nicméně s sebou nesou vysoké náklady. Za pomoci prediktivní údržby a údržby na základě technického stavu lze předpovědět a předejít neplánovaným poruchám na zařízení. Pozorováním vibrací lze nalézt vztah mezi rozsahem přirozených frekvencí a poruchovým režimem. Měřením a analýzou naměřených dat o vibracích je technik schopen získat cenné informace o stavu zařízení, nalézt příčiny selhání a naplánovat včasný zásah údržby. Pro prodloužení doby mezi poruchami monitorovaného zařízení je třeba pravidelně sledovat trend vibrací ve frekvenční oblasti. Trend frekvenčního spektra poskytuje informace o tom, o jaké typy poruchy se jedná a jejich závažnost. Využitím těchto dat lze stanovit životnost vadné součásti. [2]

Porozuměním údajů o vibracích umožňuje technikům detekovat a předvídat poruchy způsobené poškozením zařízení nebo zhoršením stavu, jakým je například nevyvážení v rotorech, vady ložisek, nedostatečné mazání a nesouosost předtím, než dojde k fatální poruše. K pochopení, jak lze k identifikaci poruch zařízení použít analýzu vibrací, je třeba nejprve nutné porozumět tomu, že všechny mechanické systémy vibrují. Každá vibrace si zachovává svůj jedinečný podpis, který na základě správné analýzy může technikovi sdělit, jak systém reaguje na své provozní podmínky. Změna provozních podmínek může vyvolat jinou odezvu s jiným podpisem, zároveň se ale může opět projevit specifický vzor vibrací naznačující konkrétní problém v systému. V průběhu času se může tento vzor vibrací stát zjevnějším, což naznačuje blížící se selhání, pokud nedojde k nápravě. Rozpoznání a kategorizace těchto vzorů před selháním zařízení je cílem detekce poruch a prediktivní údržby a umožňuje společností snížit výdaje na opravy či výměnu zařízení. [12]

Jednotlivé příčiny vzniku vibrací, které známe jsou následující:

- Nevyváženost rotujících součástí
- Excentricita součástí
- Ohnutý hřídel
- Mechanické uvolnění
- Opotřebené nebo poškozené převody
- Poškozené hnací řemeny nebo řetězy
- Poškození valivých ložisek
- Změny v točivém momentu
- Elektromagnetické síly
- Aerodynamické síly
- Hydraulické síly
- Rezonance
- Tření

Všechny tyto příčiny lze zredukovat na jednu nebo více z pěti různých kategorií. Tyto kategorie jsou nevyváženost, nesouosost, mechanické uvolnění, excentricita nebo reakce na vnější sílu. [3]

2.4.1 Nevyváženost

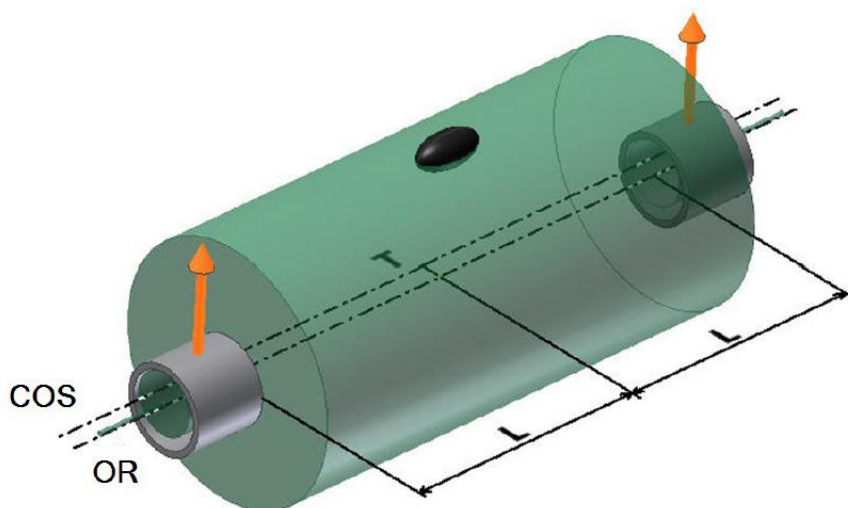
Nevyváženost patří mezi nejčastější důvod vzniku mechanického kmitání a zároveň má za následek velké namáhání celého stroje. Vznik nevyváženosti má příčinu již při samotné výrobě zařízení. Příkladem může být nerovnoměrná hustota materiálu součásti, neobrobené plochy, výrobní úchytky apod. Další možností vzniku nevyváženosti jsou funkční příčiny. Mezi ty patří nerovnoměrné opotřebení rotoru, deformace či samotná funkce stroje, například odstředivka. Následkem nevyváženosti je zejména vznik velkého dynamického namáhání a snížení trvanlivosti ložisek, což razantně snižuje životnost celého zařízení. Vibrace způsobené nevyvážeností jsou také velmi výrazně závislé na otáčkách, jejich hodnota roste s druhou mocninou otáček. Nevyváženost rozdělujeme na statickou, momentovou a dynamickou. [1]

Projevy nevyváženosti ve frekvenčním spektru

Vibrace způsobené nevyvážeností mají sinusový průběh s jednou vlnou za otáčku. Projevem ve frekvenčním spektru je tedy zvýšení hodnoty amplitudy vibrací na otáčkové frekvenci. Zvýšené hodnoty amplitudy na otáčkové frekvenci se mohou projevit i u jiných závad, nicméně ostatní závady jsou z pravidla doprovázeny vyššími hodnotami amplitudy i na dalších harmonických frekvencích. V praxi tedy platí, že pokud se v signálu vyskytují další harmonické frekvence, které jsou násobky základní otáčkové frekvence, pak se nejedná o nevyváženost. [3]

Statická nevyváženost

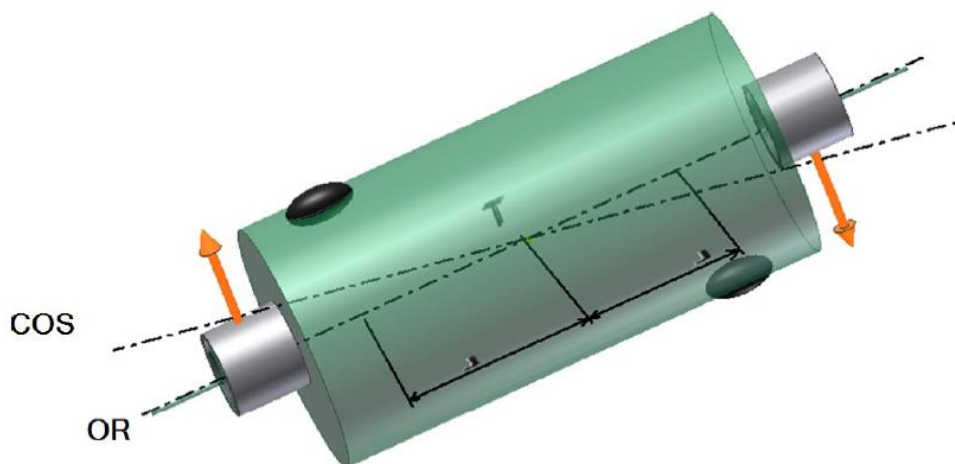
Při této závadě dochází k paralelnímu posunutí centrální osy setrvačnosti (COS) vůči ose rotace (OR), vzájemně jsou spolu tedy rovnoběžné (Obr. 10). Projevem statické nevyváženosti ve frekvenčním spektru je jedna výrazná amplituda na 1x otáčkové frekvenci. Ve většině případů se jedná o dominantní amplitudu (Obr. 12). Ke zjištění velikosti nevyváženosti poslouží měření vibrací na ložiskových domcích. [3]



Obr. 10) Statická nevyváženost [1]

Momentová nevyváženost

Centrální osa setrvačnosti (COS) a osa rotace (OR) jsou v tomto případě různoběžné, ale protínají se v těžišti rotoru (Obr. 11). Při otáčení rotoru působí dvě totožné síly proti sobě (pootočený o 180°) a způsobují vibrace. Proto lze tento typ poruchy pozorovat pouze za provozu, jelikož v klidu se systém jeví jako vyvážený. I v tomto případě je projevem ve frekvenčním spektru jedna výrazná amplituda na 1x otáčkové frekvenci. A i zde se velmi často jedná o dominantní amplitudu (Obr. 12). Měření vibrací se rovněž provádí na ložiskových domech. [3]



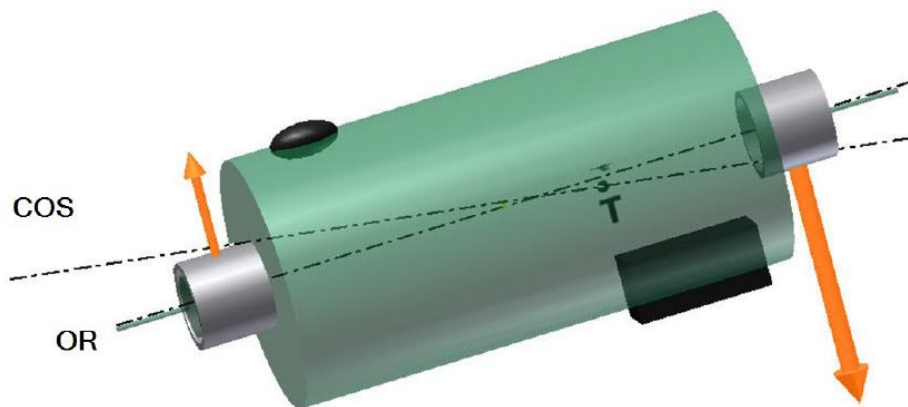
Obr. 11) Momentová nevyváženost [1]



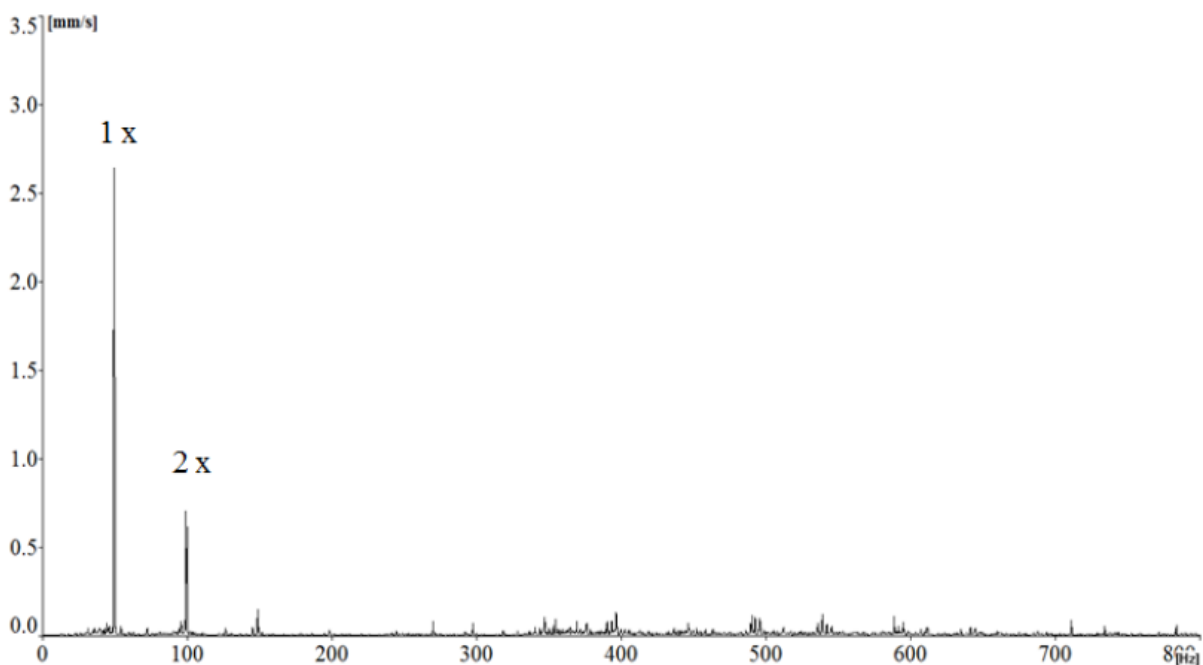
Obr. 12) Statická nebo momentová nevyváženost ve frekvenčním spektru [1]

Dynamická nevyváženost

Vzhledem k výrobním tolerancím se lze u rotorů zpravidla setkat s nevyvážeností statickou i momentovou (Obr. 13). Tuto kombinaci pak nazýváme dynamická nevyváženost. K průniku hlavní osy setrvačnosti s osou rotace nedochází v těžišti rotoru, nýbrž mimo něj. U jednoduchých strojů bývá statická nevyváženost výraznější než momentová nevyváženost. Naopak u složitějších strojů bývá momentová nevyváženost dominantní. I v tomto případě je amplituda na 1x otáčkové frekvenci dominantní (Obr. 14). V případě velké nevyváženosti se může vytvořit amplituda i na 2x otáčkové frekvenci. Měření je opět prováděno na ložiskových domcích. [3]



Obr. 13) Dynamická nevyváženost [1]



Obr. 14) Dynamická nevyváženost ve frekvenčním spektru [1]

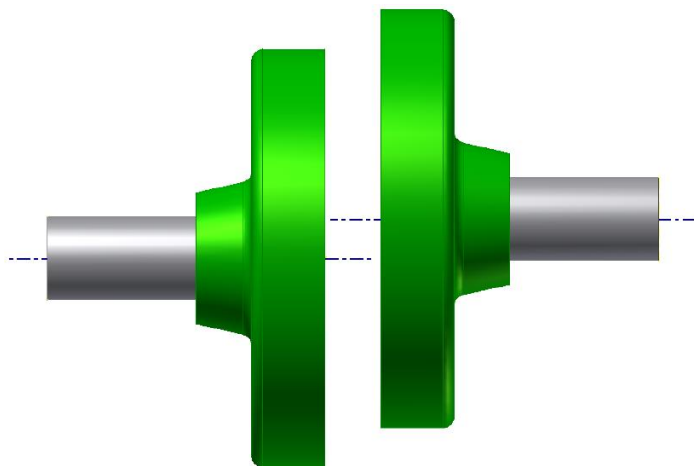
2.4.2 Nesouosost

V praxi jsou výrobní zařízení složené z jednotlivých strojů. Spojení jednotlivých strojů, respektive jejich hřídelí je uskutečněno spojky. Nepřesné ustavení či špatné smontování spojek vede ke vzniku vynucených sil a kmitů, jež vedou k dodatečnému zatížení hřídelů, ložisek, spojek a základů. Při překročení přípustných mezí ve vyrovnávání os, dochází k poškození spojek, ložisek, základů a trvalým deformacím hřídelů. [3]

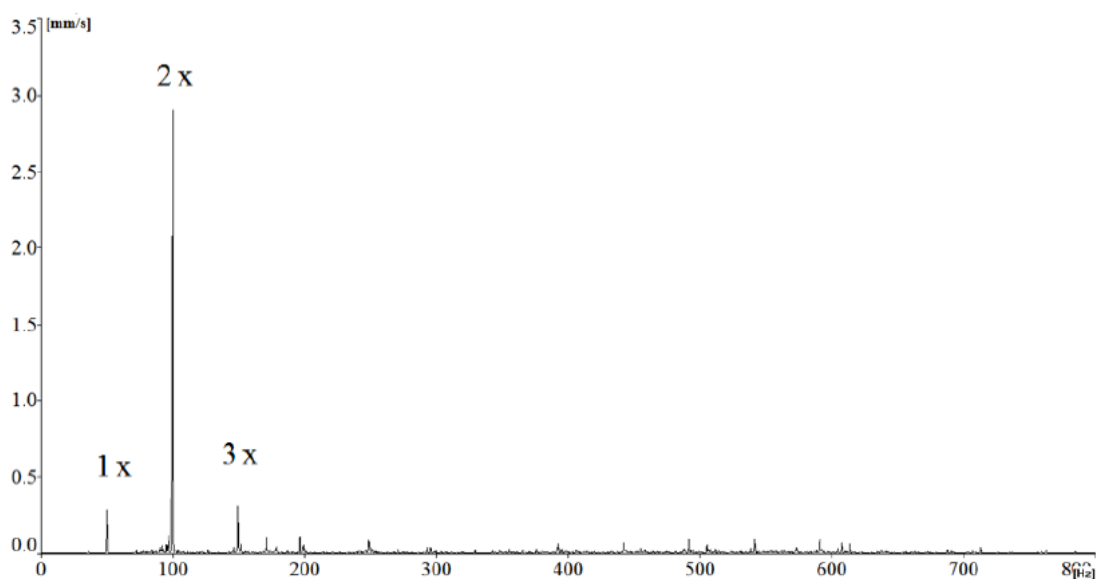
Druhou nejčastější příčinou poškození strojů ihned po vyvážení rotorů, jsou právě závady spojek a chyby ve vyrovnávání os. [3]

Rovnoběžná nesouosost

Osy rotací jednotlivých hřídelí před smontováním jsou rovnoběžné (Obr. 15). Projevem jsou velké radiální vibrace u spojky. $2x$ otáčková frekvence bývá často vyšší než $1x$, nicméně její velikost vůči první harmonické složce je často dána typem spojky (Obr. 16). Při velkém přesazení se generují vyšší amplitudy s čtvrtým až osmým harmonickým násobkem otáčkové frekvence. [3]



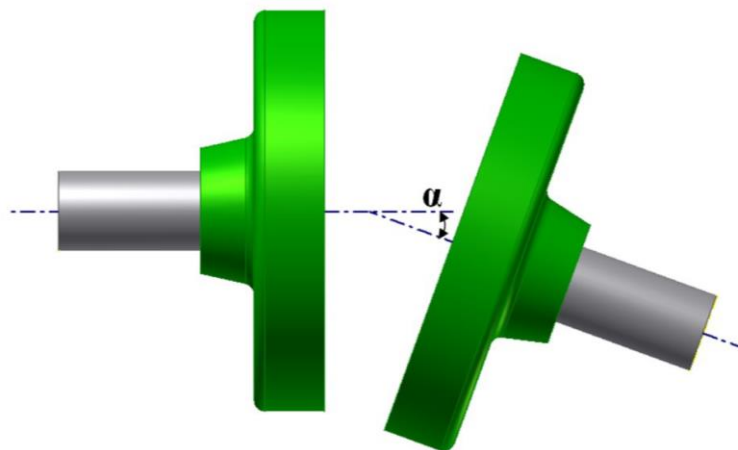
Obr. 15) Rovnoběžná nesouosost [1]



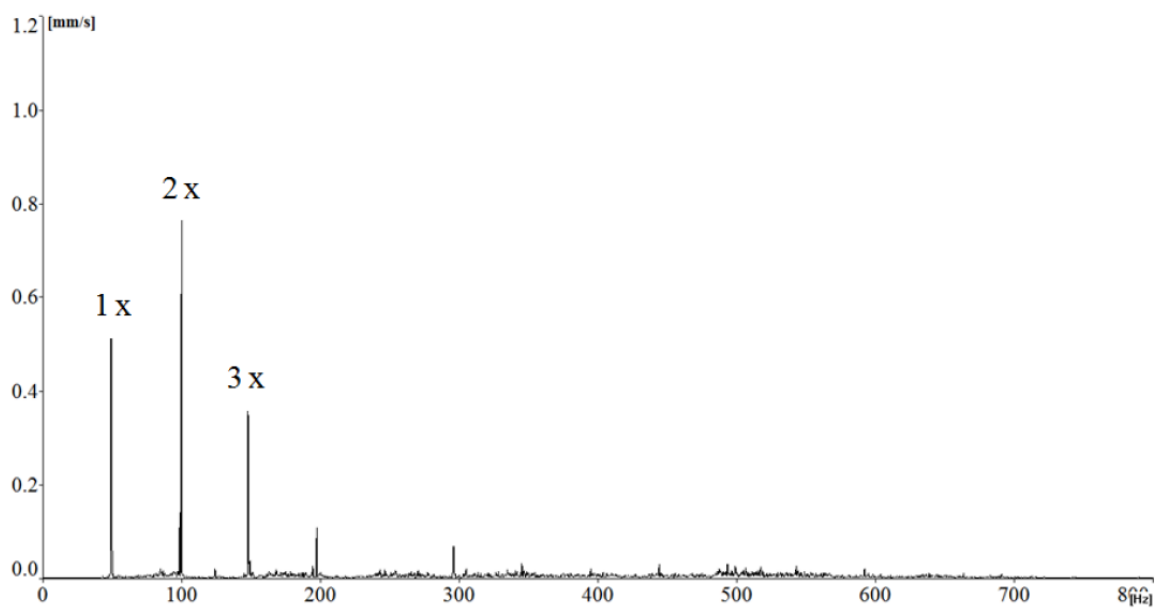
Obr. 16) Rovnoběžná nesouosost v radiálním směru s vyznačením otáčkové složky ve frekvenčním spektru [1]

Úhlová nesouosost

Osy rotací jednotlivých hřídel před smontováním nejsou rovnoběžné, ale protínají se (Obr. 17). Projevem jsou velké axiální vibrace. Ve frekvenčním spektru typicky pozorujeme velké axiální vibrace na obou složkách 1x a 2x. Není obvyklé, že by 1x, 2x nebo 3x otáčková frekvence dominovala (Obr. 18). [3]



Obr. 17) Úhlová nesouosost [1]

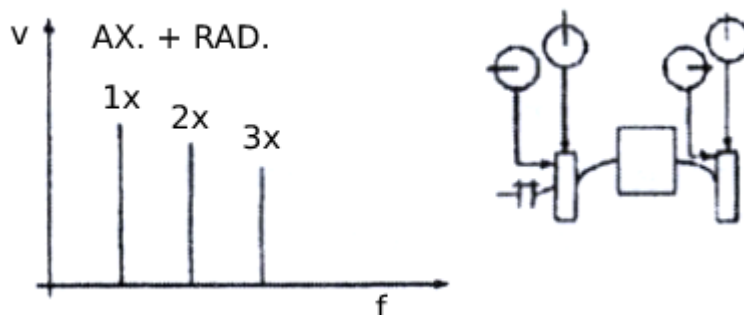


Obr. 18) Úhlová nesouosost v axiálním směru s vyznačením otáčkové složky ve frekvenčním spektru [1]

2.4.3 Ohnutý hřídel

Mezi důvody ohnutí hřídele, které nejsou spojeny se samotným provozem lze zařadit ohnutí za studena. Jako příklady lze uvést nesprávnou manipulaci během transportu, prohnutí hřídele důsledkem působení teploty nebo průhyb vzniklý vlastní vahou. Tak jako u nevyváženosti je důsledkem ohnutí hřídele nadměrné dynamické zatěžování ložisek. To vede k namáhání materiálu a následným defektům ložisek. [3]

Ohnutí hřídele se u frekvenční analýzy projevuje identicky jako nesouosost (Obr. 19). K rozlišení těchto dvou závad je nutné provést měření fáze. Ve frekvenčním spektru se dají pozorovat vyšší hodnoty amplitud pro 1x a 2x otáčkovou frekvenci. Amplituda druhého násobku otáčkové frekvence se pohybuje v rozmezí 30 až 200% amplitudy základní otáčkové frekvence. Může se objevovat i třetí násobek základní otáčkové frekvence. Fáze v radiálním směru je obvykle „ve fázi“ a fáze v axiálním směru v protifázi (rozdíl fází přibližně 180°). [3]



Obr. 19) Ohnutý hřídel [3]

2.4.4 Mechanické uvolnění

Projevem mechanického uvolnění je typicky dlouhý sled nezvykle vysokých amplitud vyšších harmonických složek (2x, 3x, 4x, ...), interharmonických složek (1,5x, 2,5x, ...) a subharmonických složek (2/3x, 1/2x, 1/3x, ...). Velikost těchto amplitud by měla dosahovat alespoň 20% amplitudy základní otáčkové frekvence. K přesnému zjištění, kde se závada nachází slouží metoda zviditelnění provozních tvarů a kmitů. [3]

2.4.5 Ozubené a řemenové převody

Různé typy převodů jsou přítomny v téměř každém průmyslovém stroji. Zároveň se jedná o nepřetržitě namáhanou součást. Poškození ozubených kol, řemenových převodů i ostatních součástí sloužících ke přenosu točivého pohybu může mít negativní vliv jak na součást samotnou, tak i na ostatní komponenty stroje. [3]

Ozubená soukolí

Pokud se vezmou v potaz všechny fyzikální jevy, které provázejí záběr ozubených kol, jedná se o velmi komplikovaný děj. Tento fakt navíc komplikuje to, že samotná převodovka je z konstrukčního hlediska složena z dalších ložisek, hřídelů apod. Poškození zubů se dělí do dvou kategorií, na lokální a na distribuované. U lokálních závad se jedná o konkrétní závadu na jednom zubu, například pitting. Zatímco distribuovaná porucha působí na všechny zuby, například nesouosost ozubení. Poškození zubů a nesprávná montáž ozubení se projevuje na zubové otáčkové frekvenci, pro kterou platí vztah (6)

$$f_{z12} = z_1 \cdot f_{R1} = z_2 \cdot f_{R2} \quad (6)$$

a také na 2x a 3x harmonické zubové frekvenci, pro které platí vztah (7) a (8). [3]

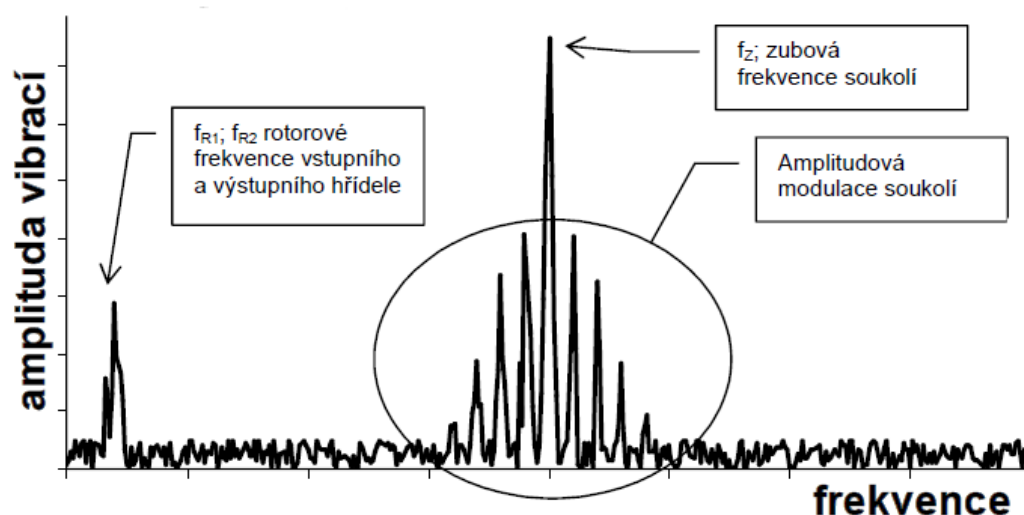
$$f_{2Hz} = 2 \cdot f_z \quad (7)$$

$$f_{3Hz} = 3 \cdot f_z \quad (8)$$

Jedním z projevů poškození zubů je nárůst amplitud složek f_{2Hz} a f_{3Hz} . Se zhoršením závady se harmonické zubové frekvence projevují výrazněji než základní zubová frekvence f_z , proto je nutné dbát na to, aby frekvenční rozsah byl vyšší než f_{3Hz} . [3]

Poškození boků zubů se typicky projevuje výskytem postranních pásem kolem frekvencí f_z , f_{2Hz} , a f_{3Hz} (Obr. 20). Velikost postranního pásma je dána velikostí rotorových frekvencí. Postranní pásmo pro zubovou frekvenci f_z je dáno vzorcem (9). [3]

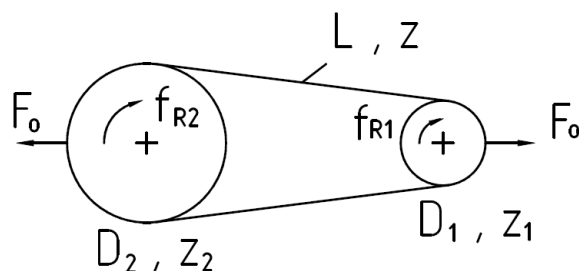
$$f_z \pm f_r \quad (9)$$



Obr. 20) Frekvenční spektrum ozubeného převodu [13]

Řemenové převody

Jedná se o praktické řešení v případech, kdy je mezi hnací a hnanou hřídelí větší vzdálenost. V praxi je pozorována řemenová frekvence (10), zubová frekvence (u ozubených řemenů) (11) a vlastní frekvenci řemene.



Obr. 21) Řemenový převod [3]

$$f_B = \frac{\pi \cdot D_1}{L} \cdot f_{R1} = \frac{\pi \cdot D_2}{L} \cdot f_{R2} \quad (10)$$

$$f_{BZ} = z_1 \cdot f_{R1} = z_2 \cdot f_{R2} \quad (11)$$

Vlastní frekvence f_0 je závislá na parametrech L a F_0 .

Vznik vibrací u řemenových převodů lze připsat několika příčinám. Mezi časté problémy se řadí opotřebené a volné řemeny. Dále lze uvést také nesouosost a excentricitu řemenic. Nelze opomenout ani samotný řemen, v případě že bude vlastní frekvence řemene blízka s frekvencí hnací či hnané hřídele může dojít k rezonanci a vzniku velmi silných vibrací. Vibrace se mění spolu se stavem řemene. Na ten má vliv jeho stáří, počet rozběhů, čistotě a teplotě pracovního prostředí. [13]

Při poruše ve frekvenčním spektru zvyšují amplitudy rotorových a řemenových frekvencí včetně jejich harmonických násobků.

2.4.6 Valivá ložiska

Valivá ložiska jsou dnes nedílnou součástí průmyslových strojů. Základní funkcí ložiska je přenos síly mezi rotující částí a konstrukcí stroje a také snížení tření v soustavě. V dnešní době se klade velký důraz na kvalitu zpracování ložisek, což vede k zvýšení jejich životnosti, nicméně stále platí to, že i nepatrné poškození ložiska může v krátké době vést ke katastrofické poruše. [3]

Faktorů, které ovlivňují životnost ložisek je mnoho, mezi ně například patří nesprávné mazání (použití špatného maziva, příliš málo nebo naopak příliš mnoho maziva), znečištěné mazivo, povrchová únava materiálu, nesprávné zacházení nebo montáž, větší než předepsaná zátěž (způsobená také například jinými poruchami jako nevyváženost, nesouosost, ohnutý hřídel atd.) a další. [3]

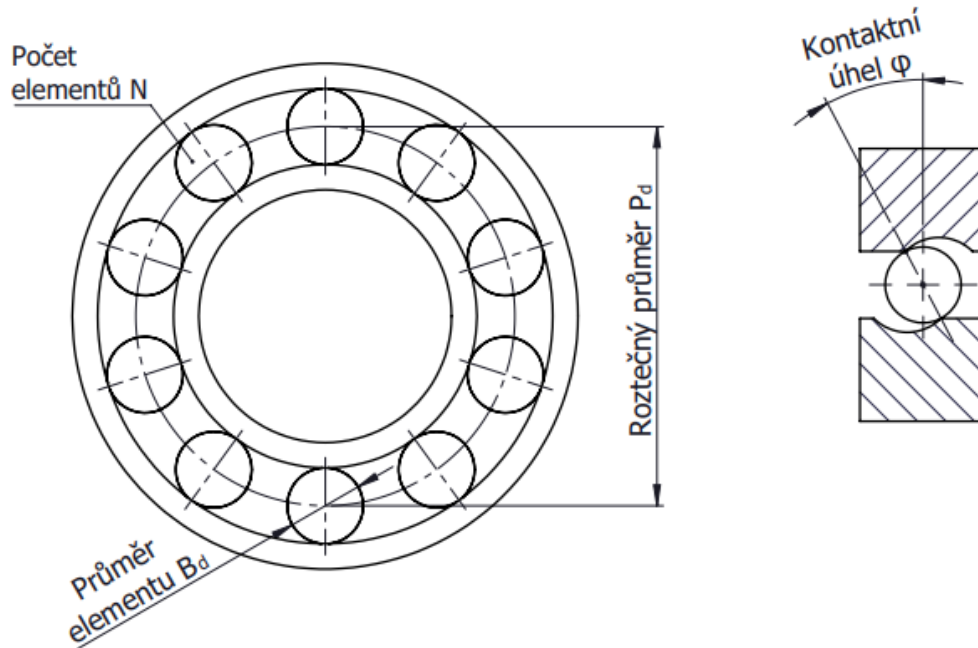
K počáteční únavě ložisek dochází působením smykových napětí, která působí bezprostředně pod plochou nesoucí zátěž. V delším časovém horizontu má působení tohoto napětí za důsledek vznik trhlin, které se šíří k povrchu. Přejdem valivých prvků přes tyto trhliny dochází k ulomování částí materiálu. Tento jev je známý jako vydrolování (pitting). S postupem času se vydrolování zvětšuje, čímž může dojít až k vyrazení ložiska. Další možné poškození ložiska je iniciováno poškozením plochy. Trhlina se v tomto případě šíří do materiálu. K poškození plochy může dojít nadměrným zatížením nebo špatným mazáním. [13]

Oba typy poruchy jsou doprovázeny zvukovým a vibračním signálem. Ten lze detekovat a po jeho správné analýze zajistit nápravu příčiny problému ložiska. Tím lze dosáhnout prodloužení životnosti ložiska nebo zabránit fatálnímu selhání. [13]

Samotná výměna ložiska často není dostačujícím opatřením při údržbě. Nejprve je nutné určit původní příčinu poškození ložiska. Tou je nejčastěji jiný strojní problém (nevyváženost, nesouosost atd.) nebo jiný problém procesu (např. nedostatečné mazání). Tímto se předejde k opakovanému poškození ložiska. [13]

Zjišťování závad valivého ložiska

Jakmile dojde vlivem nadměrného přetížení ke vzniku prasklin nebo vyštípnutí, dojde ke generování impulsních sil, jak se každý valivý prvek valí přes prasklinu na povrchu. Podle místa (Obr. 23), kde se závada vyskytuje, rozlišujeme na valivém ložisku čtyři typy závad. Každé z těchto závad odpovídá právě jedna poruchová frekvence ložiska (12) až (15), kterou je možné na základně parametrů ložiska (Obr. 22) a otáčkové frekvence spočítat. [13]



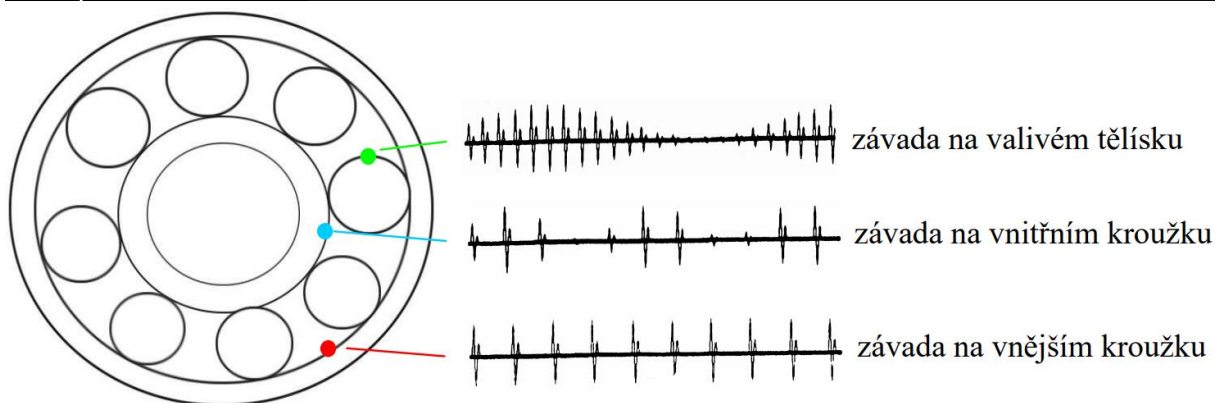
Obr. 22) Geometrie ložiska [3]

$$\text{Poškození vnitřního kroužku (BPFI)} \quad \frac{N}{2} \cdot \left(1 - \frac{B_d}{P_d} \cdot \cos \varphi\right) \cdot n \quad (12)$$

$$\text{Poškození vnějšího kroužku (BPFO)} \quad \frac{N}{2} \cdot \left(1 + \frac{B_d}{P_d} \cdot \cos \varphi\right) \cdot n \quad (13)$$

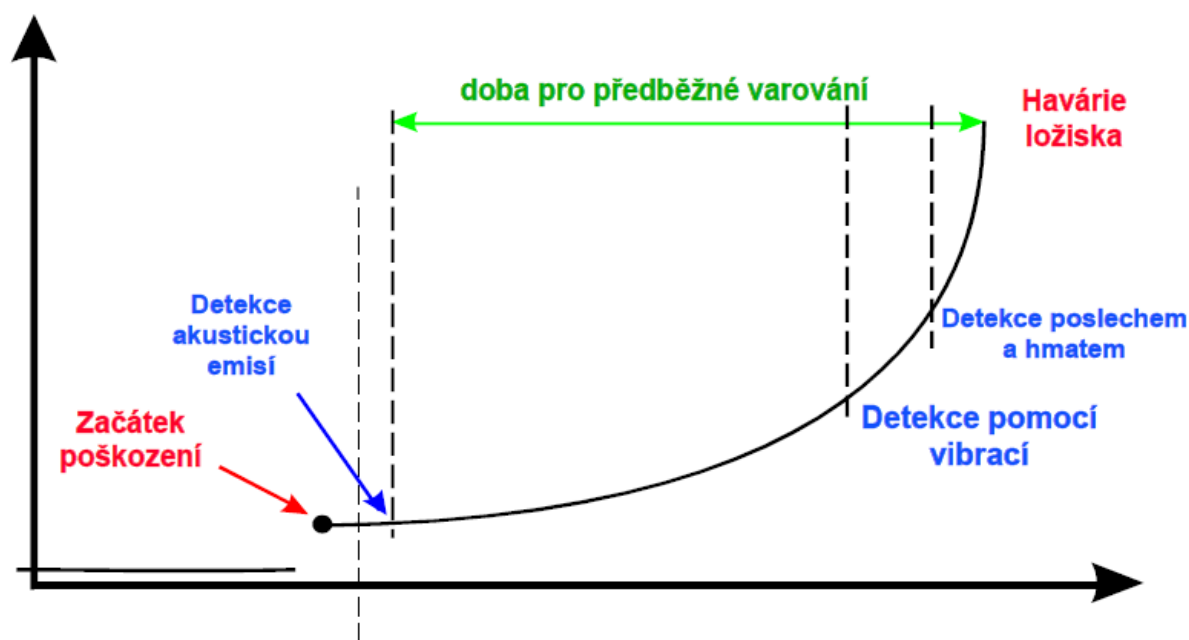
$$\text{Poškození valivých elementů (BSF)} \quad \frac{P_d}{2B_d} \cdot \left(1 - \left(\frac{B_d}{P_d} \cdot \cos \varphi\right)^2\right) \cdot n \quad (14)$$

$$\text{Poškození klece (FTF)} \quad \frac{1}{2} \cdot \left(1 - \frac{B_d}{P_d} \cdot \cos \varphi\right) \cdot n \quad (15)$$



Obr. 23) Frekvence ložiskových závad a jejich poloha [14]

Na rozdíl od ostatních poruchových frekvencí se poruchové frekvence ložisek liší tím, že se jedná o frekvence existujících závad. Pokud závada neexistuje, nejsou poruchové frekvence ložisek přítomny. Jakmile jsou přítomny, indikuje to, že existuje vznikající problém. V tomto okamžiku je vhodné zvýšit četnost sběru dat tak, aby šlo lépe určit spolehlivost stroje. K tomu také poslouží trendování vibrací (Obr. 24). Jehož smyslem je detekovat nárůst poškození v čase a předejít tak závažné poruše. [13]



Obr. 24) Trendování měřených vibrací valivých ložisek [1]

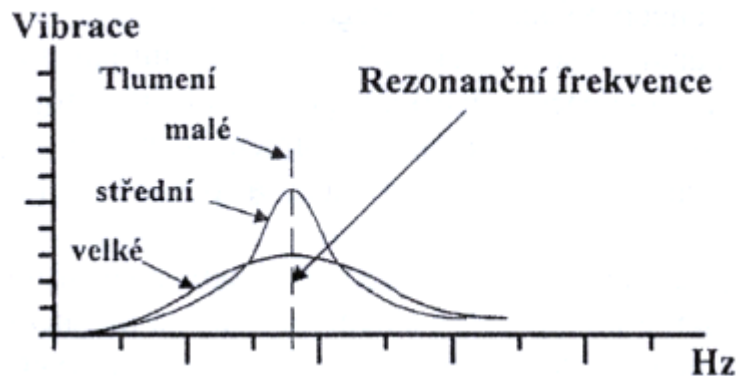
V ranných stádiích poruchy ložiska jsou frekvenční složky odpovídající závadám velmi malé a jsou jen velmi obtížně detekovatelné standardními metodami měření vibrací. K určení opotřebení ložiska v takto ranných stádiích slouží například metody celkové hodnoty zrychlení, Spike Energy, HFD, Crest Factor, Shock Pulse Meter, filtrování signálu zrychlení v daném pásmu a další. Většina těchto metod je založena na trendování dané hodnoty ke zjištění stavu poškozeného ložiska. Tyto metody ale nemusí vždycky poukazovat na poškození ložiska, podobnou odezvu totiž má také například problém ozubených kol. [3]

2.4.7 Rezonance

Rezonance sama o sobě nezpůsobuje vibrace, pouze je zesiluje, a to až do vysokých hodnot. Dalším důsledkem je vysoká citlivost stroje na silové působení s rezonanční frekvencí. Obecně se jedná o velmi častý zdroj velkých vibrací jejichž příčina není vždy známá. [3]

Každá strojní součást má svou hmotnost a tuhost, to znamená, že silovým působením dojde k jejich rozkmitání. Doba kmitání závisí na jejich tuhosti, hmotnosti a také na tom na jakou frekvenci silového působení je konkrétní součást citlivá. V praxi to znamená, že přiblíží-li se otáčková frekvence stroje k této citlivé frekvenci (rezonanční frekvenci), dojde ke vzniku vibračních problémů – rezonance. Důležitou roli hraje také tření, tím že redukuje vibrace při rezonanci, jinými slovy dochází ke tlumení. Velikost tlumení má významný vliv na velikost vibrací v důsledku rezonance, jak je možné vidět na Obr. 25). [3]

Konkrétní případ, se kterým se v praxi často setkáváme je rezonance na hřídeli známá také jako kritické otáčky. Je to stav kdy se otáčková frekvence hřídele přiblíží jeho rezonanční frekvenci.



Obr. 25) Vliv tlumení na velikost vibrací

Vzhledem k tomu, že přesná úroveň tlumení není nikdy známá, je vhodné řídit se doporučením, že pracovní otáčky by měly mít odstup přibližně 30 % od rezonanční frekvence. V praxi je řešením tohoto problému tzv. dynamické ladění strojů. Toho se docílí například změnou pracovních otáček, změnou tuhosti konstrukce nebo konstrukčním řešením, které má za cíl lepší tlumení vibrací. [3]

3 SOFTWARE PRO SIMULACI MECHANICKÝCH SOUSTAV

Nejčastější metodou pro simulaci mechanických soustav v dnešní době je metoda konečných prvků (MKP, FEM – z angličtiny Finite Elements Method). Jedná se o pohodlnou metodu pro inženýry, jak provádět složité výpočty v mechanice a v jiných fyzikálních oblastech. Umožňuje provádět přesné odhady napětí, namáhání, frekvencí a mnoho dalšího. Je možné simulovat jak samostatné díly, tak i větší části strojů nebo i kompletní sestavy. V praxi se lze setkat i s kompletní simulací nárazu automobilu. [15]

Analýza metodou konečných prvků (FEA – z pro Finite Elements Analysis) je dnes neocenitelným nástrojem při vývoji všech různých produktů. Analýzu lze aplikovat již ve fázi konceptu a dále s ní pracovat při dalším vývoji součástí. To má významný pozitivní vliv na cenu i čas vývoje, jelikož lze významně snížit rozpočet jinak vynaložený na testování a prototypy. Ve spolupráci s CADem (z angličtiny Computer-aided design) se z metody konečných prvků stává velmi silný nástroj v rukou zkušeného inženýra. Na druhou stranu ale není vhodné v MKP slepě důvěřovat bez základního porozumění mechaniky a principů FEM. Jelikož počítačem kompletně vypočítaná struktura nemusí v reálných podmínkách obstát a při selhání pak může dojít k fatálním následkům, za které pak nese zodpovědnost výpočetní technik. [15]

V dnešní době je na trhu řada MKP simulačních programů. Mezi v praxi běžný software patří ANSYS, ABAQUS, NASTRAN, I-DEAS a další. Tyto programy jsou schopné provádět simulace prakticky bez omezení, co se týče rozsahu a reprodukce reality. Na trhu jsou ale k dispozici i nesespecializované programy, které kombinují 3D CAD spolu s analýzou metodou konečných prvků jako je například SolidWorks nebo AutoDesk Inventor.

3.1 Obecný postup pro simulaci metodou konečných prvků

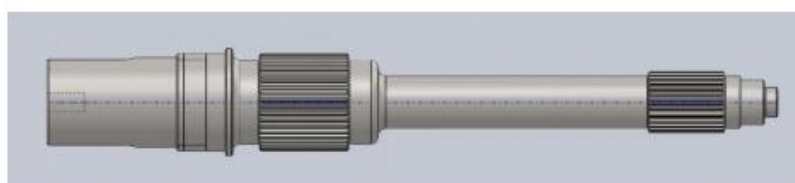
Při tvorbě simulace MKP je třeba dodržovat vhodný postup, ten se netýká pouze simulace samotné ale i kroků předcházejících.

3.1.1 Inženýrský postup

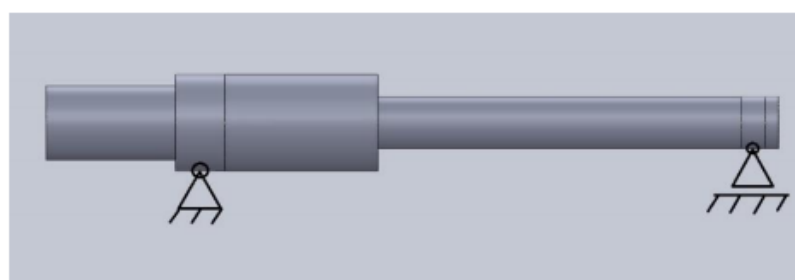
Před provedením simulace MKP je nejprve třeba vytvořit CAD model. V závislosti na solveru (řešič úlohy) a jeho kompatibilitě s CAD modely, může být nutné provést propojení CADu se solverem. K tomu slouží standardizované rozhraní jako například IGES nebo dnes novější STEP. Většina simulačních programů nabízí i možnost tvorby vlastních modelů. Nicméně je většinou časově úspornější použití originálních CAD dat. Následující procedura je příklad obecného postupu a může se místy lišit dle typu analýzy. Body 1-5 lze shrnout pod názvem pre processing (před zpracování). [16]

1. Úprava 3D geometrie – snížení geometrických detailů jako jsou malé otvory, poloměry atd. tak, aby došlo ke snížení počtu chyb ve výpočtu. Je to dáno tím, že tyto malé prvky zvyšují počet špatných elementů a zkreslují výsledky. (Obr. 26)
2. Volba správného materiálu – většina simulátorů má k dispozici rozsáhlé materiálové knihovny, případně možnost vytvořit nové na základně fyzikálních vlastností.

3. Volba typu prvku – jedná se o jeden z nejdůležitějších kroků simulace. V závislosti na typu analýzy je nutné zvolit vhodný typ prvku s ohledem na očekávané / požadované výsledky.
4. Vytvoření sítě konečných prvků (Meshing) – základním principem MKP je nahradit spojitý model, modelem složeným z konečného počtu prvků. Úkolem inženýra je také zamyslet se, zdali je možné použití symetrie na část modelu, což sníží náročnost simulace. Dále je vhodné doladit například detaily děr, pokud je jejich funkce zásadní pro simulaci. (Obr. 27)
5. Okrajové podmínky – volba zatížení a výběr bodů kde zatížení působí.
6. Výpočet – pokud byly předchozí kroky v pořádku, může solver vypočítat výsledky
7. Kontrola výsledků – po ukončení simulace je důležité porovnat výsledky simulace s výsledky získanými jinou analytickou metodou a prověřit, že oba výsledky jsou si podobné. Kontrolou deformace simulovaného objektu lze také objevit chyby v okrajových podmínkách.

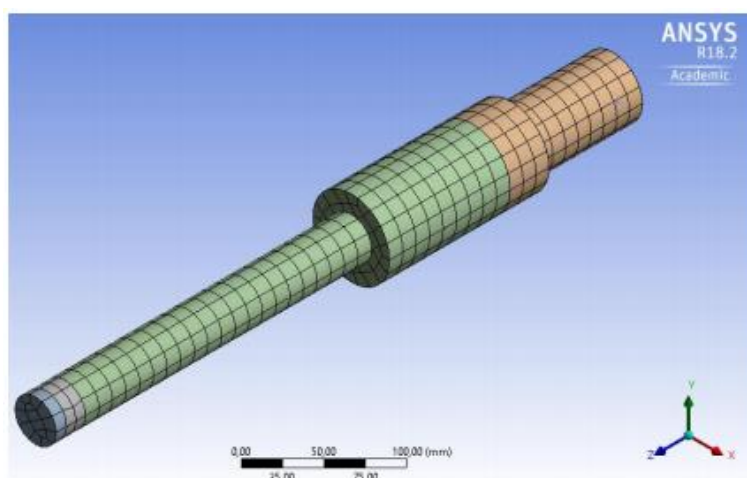


(a)



(b)

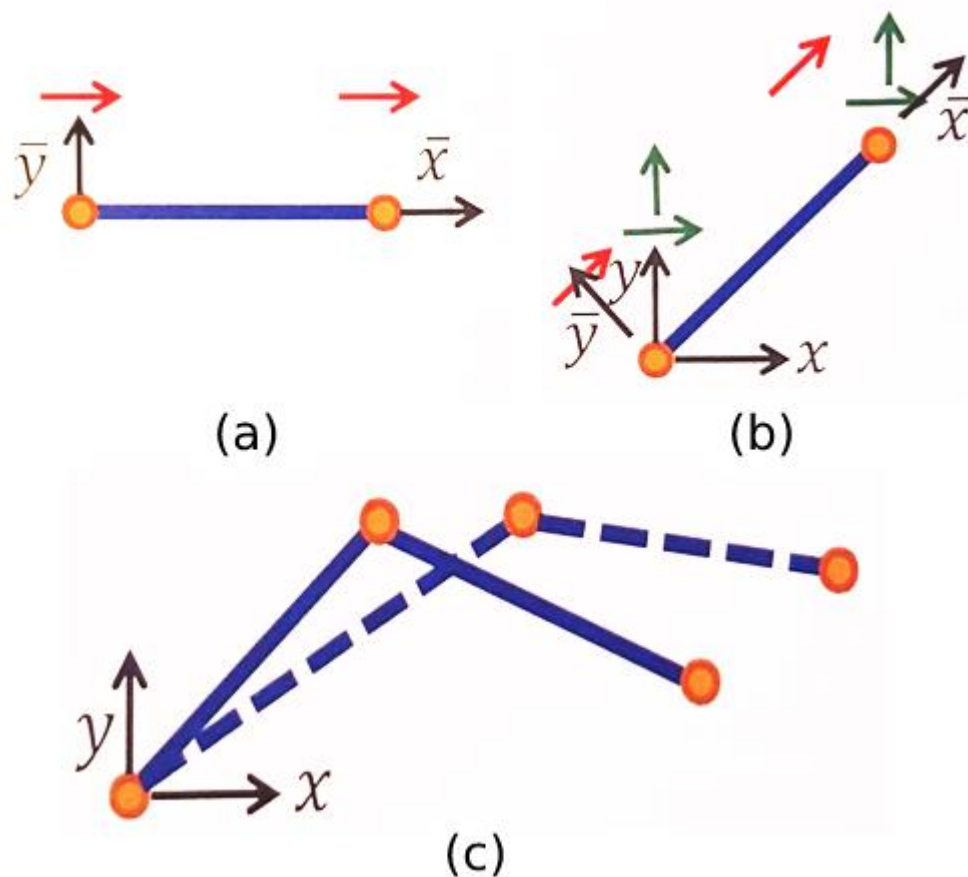
Obr. 26) Originální 3D model hřídele (a), očištěný model hřídele pro simulaci (b) [15]



Obr. 27) Síť konečných prvků [15]

3.1.2 Softwarový postup

Následující kroky popisují obecný postup simulačního softwaru při lineární statické analýze. Tato podkapitola poskytuje představu o tom, jak software obecně provádí simulaci. Postup je ukázán na prutu, ale funguje podobně i pro všechny ostatní prvky. [15]



Obr. 28) (a) lokální souřadnicový systém, (b) globální souřadnicový systém, (c) Složená struktura [15]

1. Výpočet matice tuhosti pro každý prvek v lokálním souřadnicovém systému (\bar{x}, \bar{y}) (Obr. 28) (a)
2. Transformace do globálního souřadnicového systému (x, y) (Obr. 28) (b)
3. Sestavení všech prvků dle sítě konečných prvků (Obr. 28) (c)
4. Přidání okrajových podmínek – zatížení a míst kde působí
5. Řešení soustavy rovnic pro síly, posunutí, mechanické napětí
6. Post processing – následné zpracování (vybarvení prvků podle hodnot mechanického napětí)




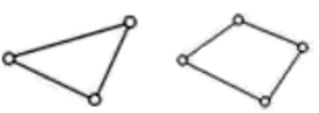

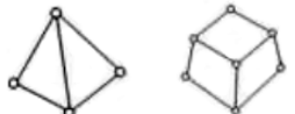
3.1.3 Prvky

1D prvky jsou rozděleny na dva druhy – pruty a nosníky (Obr. 29). Zatímco pruty mohou přenášet sílu pouze po jejich podélné ose, nosníky mohou navíc přenášet i příčné síly a momenty. Pruty mají jeden stupeň volnosti, nosníky pak dva. 1D prvky slouží k simulaci rámových konstrukcí a hřídelů. [17]

2D skořepinové prvky jsou rozděleny na trojúhelníkové a čtvercové. Čtvercové elementy jsou preferovanou volbou tam, kde to je možné. Například kvádr složený z trojúhelníkových tvarů má celkově trojnásobný počet elementů oproti čtvercovým tvarům. Dále se dělí na skořepiny, které se používají pouze pro simulaci napětí a pro simulaci napětí a ohybu. Liší se počtem stupňů volnosti. 2D skořepiny jsou používány k simulaci plechových konstrukcí. [17]

3D objemové prvky mají tři stupně volnosti v posunu a žádný momentový. I zde platí, že čtvercové tvary jsou preferovány. Objemové elementy se používají k simulaci silnostěnných konstrukcí jako je například motor. [17]

Výše popsané prvky se automaticky spojí dohromady do jedné velké sítě konečných prvků po celém tvaru součásti. Nicméně automaticky generovaná síť není vždy dokonalá a dle zvolené analýzy může být nutná manuální úprava. [17]

| Typ | Skutečný model | | Další požadavky (výpočet skutečného objemu) |
|-----|---|---|--|
| 1D |  Prut Nosník |  Délka (L) | Plocha (S, průřez tvarem) $V = L \cdot S$ |
| 2D |  Skořepina, rovinné napětí / namáhání, osová souměrnost atd. |  Plocha (S) | Tloušťka (t) $V = S \cdot t$ |
| 3D |  Pevný objekt |  Objem (V) | Žádné |

Obr. 29) Druhy elementů [18]

3.1.4 Solver – řešitel úlohy

V metodě konečných prvků mají řešitelé zvláštní význam. Například pokud existuje struktura s 1000 stupni volnosti, znamená to, že je třeba vyřešit matici s proměnnými o rozměrech $1000 \cdot 1000$. To je v dnešní době poměrně jednoduchý problém. Pokud se ale struktura v trojrozměrném problému zvětší, exponenciálně se zvyšuje i počet stupňů volnosti. Z toho je jasné, že analytické řešení je pro analýzu konečných prvků neúčinné a téměř nemožné. Vzhledem k velikostem soustav rovnic se k řešení rovnic používají numerické metody. Rozlišujeme metody na přímé a iterační a v dynamice také na redukční. K řešení vibračních úloh pak slouží speciální varianty iteračních i přímých řešitelů v závislosti na komplexitě problému. [19]

Volbu řešitele provádí software ve většině případů automaticky. V případě, kdy má smysl vybrat řešitele ručně, se volba v zásadě řídí celkovým počtem stupňů volnosti, tak aby byl zajištěn co nejefektivnější proces. [19]

3.1.5 Vibrační analýza

S vibracemi se můžeme setkat u mnoha různých technických aplikacích a zejména v konstrukcích, které jsou v pohybu nebo jsou vystaveny dynamickému zatížení. Existuje mnoho druhů mechanických vibrací a hodnot, které je charakterizují. Dvě hodnoty, které jsou pro konstruktéra zvláště zajímavé a lze je simulovat metodou konečných prvků, jsou vlastní frekvence a vlastní mód. Co je to vlastní frekvence bylo popsáno v kapitole 2.4.7. [19]

Nejběžnějším typem vibrací, které mohou na hřídel působit jsou ohybové vibrace a torzní vibrace. Ohybové vibrace se na hřídeli vyskytují kvůli nerovnoměrnému rozložení hmotnosti. Hřídele nemají dokonalý kulatý tvar a osa otáčení přesně neodpovídá středové ose hřídele. Jakmile se hřídel začne otáčet a zrychlovat kolem své osy, dosáhne bodu, kde dochází k vibracím. Tento bod se nazývá vlastní kruhová frekvence. Pokud zrychlení pokračuje, vibrace postupně zmizí. K torzním vibracím dochází, pokud je k hřídeli připojena hmota jako je například řemenice nebo ozubené kolo. Tím pak dochází ke zkroucení kolem středové osy. [19]

3.2 Přehled vybraných programů pro simulaci mechanických soustav

Na trhu existuje velké množství programů zaměřených na analýzu konečných prvků. Volba konkrétního programu je velmi individuální proces závislý na konkrétních potřebách daného projektu. Roli může hrát s jakou přesností nebo naopak rychlostí je nutné analýzy provádět, vzhledem k tomu, že tyto simulace jsou hardwarově velmi náročné. Některé programy jsou zaměřené pouze pro konkrétní typ analýzy (například analýzy přenosu tepla nebo proudění), zatímco jiné nabízí kompletní sadu funkcí. Proto je zde zastoupeno pouze několik programů vzhledem k jejich popularitě. [20]

3.2.1 Ansys Mechanical

Ansys je jedním z nejčastěji používaných produktů na trhu pro analýzu metodou koncových prvků. Je vyvíjen společností Ansys, Inc. jež vyvíjí širokou škálu inženýrských CAD produktů, nicméně nejznámější je právě Ansys Mechanical. Jedná se o samostatný analytický nástroj, který umožňuje vykonat všechny úkony nutné k provedení analýzy v jednom programu. To znamená příprava modelu, simulace a následné zpracování a vyhodnocení dat. [21]

2D i 3D modely je možné vytvářet přímo v prostředí Ansys nebo je lze importovat z ostatních CAD programů. Ansys nabízí pouze omezené množství materiálů, na druhou stranu nabízí možnost vytvářet úplně nové materiály nebo přidat skutečné materiály na základě jejich fyzikálních vlastností. [21]

Ansys je schopen řešit prakticky jakýkoli typ analýzy, který lze metodou MKP řešit. Je schopen modelovat lineární, nelineární úlohy a přechodně lineární úlohy, výpočty rezonančních frekvencí, akustické simulace, přenos tepla, elektrické/magnetické simulace, proudění tekutin, anizotropní materiály, kompozity, hyperelastické/gumové materiály, plastické deformace, viskoplasticitu/creep, piezoelektrické jevy a další. [22]

Dále nabízí možnost integrace speciálních solverů jako je například FLOTRAN (proudění a přenos tepla). Vzhledem k prakticky neomezeným možnostem může být pro nové uživatele značně komplikovaný. Nedostatky programu tkví v primitivním CAD prostředí pro tvorbu modelů, kde je většinou lepší importovat model z jiného CAD softwaru.

3.2.2 Abaqus/CAE

Abaqus je softwarová sada pro analýzu MKP a počítačové inženýrství. CAE je zkratka pro Complete Abaqus Environment (kompletní prostředí Abaqus). Intuitivní rozhraní integruje modelování, analýzu, správu úloh a vizualizaci výsledků v konzistentním a snadno ovladatelném prostředí, které je snadné k používání pro nové uživatele, a vysoce produktivní pro uživatele zkušené. [23]

Uživatelé mohou vytvářet vlastní modely nebo importovat modely CAD z jiných aplikací. K dispozici je asociativní rozhraní pro CATIA V5, SolidWorks a Pro/ENGINEER, které umožňuje synchronizaci sestav CAD a CAE, umožňující rychlé aktualizace modelu bez ztráty funkcí definovaných uživatelem. [23]

Rozsah možností analýz je obdobný tomu, co je k dispozici u Ansysu. Abaqus vyžaduje konzistentní používání fyzikálních jednotek, čímž se liší od Ansysu i SolidWorksu, které umí automaticky detekovat použité jednotky. Dalším rozdílem jsou solvery, které jednotlivé programy používají. Ty se mohou lišit výkonem dle řešené úlohy. [22]

3.2.3 SolidWorks

SolidWorks je 3D CAD program určený k modelování součástí, tvorbě sestav a výkresů. V tomto směru se jedná o jeden z nejpoužívanějších CAD softwarů současnosti. [24]

Jeho součástí je program pro simulování MKP nazvaný SolidWorks Simulation. Slouží k řešení analýz pevných a skořepinových 3D objektů a 2D úloh zaměřených na napětí a deformace. Je schopen vytvářet strukturální a pohybové simulace, výpočty rezonančních frekvencí, simulovat únavu a životnost materiálu, simulace teplotních změn či provádět optimalizaci návrhu. [22]

Jeho předností je jednoduché ovládání a intuitivní prostředí. Díky tomu, že je integrován přímo do kreslicího programu SolidWorks, najde využití jako jednoduchý nástroj pro rychlé testování při vývoji. Výkonem a možnostmi simulace ale zaostává před profesionálním řešením zaměřeným výhradně na analýzu MKP. Zde pak poslouží jako zdroj modelů pro profesionální simulace. [24]

4 PRAKTICKÁ ČÁST

Cílem práce je vytvoření algoritmu pro generování simulovaného signálu z vibrodiagnostického zařízení. Program je vytvořen ve dvou verzích, jako samostatně spustitelný skript a jako program s grafickým uživatelským rozhraním.

4.1 Použitý software a programovací jazyky

K tvorbě programu byl vybrán programovací jazyk Python vzhledem k jeho jednoduché křivce učení a s ohledem na jeho využití v praxi. Ostatní softwarové komponenty byly voleny tak, aby byly kompatibilní a multiplatformní.

4.1.1 Python

Python je objektově orientovaný interpretovaný vysokoúrovňový programovací jazyk. Poskytuje dynamické typování a automatickou správu paměti. Jedná se o velmi univerzální programovací jazyk. Může například sloužit jako skriptovací jazyk, k vývoji mobilních či webových aplikací nebo jako spojení mezi standardně nekompatibilními komponenty. Python je k dispozici pro většinu běžných platform (Windows, Unix, macOS, Android). Dále se jedná o Open-source projekt pro komerční i nekomerční využití. Díky rozsáhlé uživatelské základně je k dispozici mnoho volně přístupných modulů a knihoven nad rámec základního balíku. [25]

Mezi jeho přednosti dále patří jeho jednoduchost a přehlednost. Na rozdíl od mnoha dalších kódovacích jazyků nepoužívá k vymezení bloků kódu závorky ale odsazení bílými znaky. Blok kódu je tedy vymezen souvislými řádky se stejným odsazením (stejně dlouhým počtem bílých míst). Větší odsazení signalizuje vnořený blok kódu. Tento přístup má jednu velmi užitečnou vlastnost a tou je, že nutí programátora psát lépe čitelný kód. [26]

Nevýhodou jazyka je rychlost samotného programu a vyšší nároky na paměť. Jelikož je Python jazyk interpretovaný, musí být nejprve přeložen do meziformy (tzv. bytecode) a následně spuštěn interpretem. Nicméně při dnešním výpočetním výkonu je rozdíl oproti ostatním jazykům (např. C/C++) zanedbatelný. Na Obr. 30) je zobrazeno logo programovacího jazyka Python. [26]



Obr. 30) Logo programovacího jazyka Python [26]

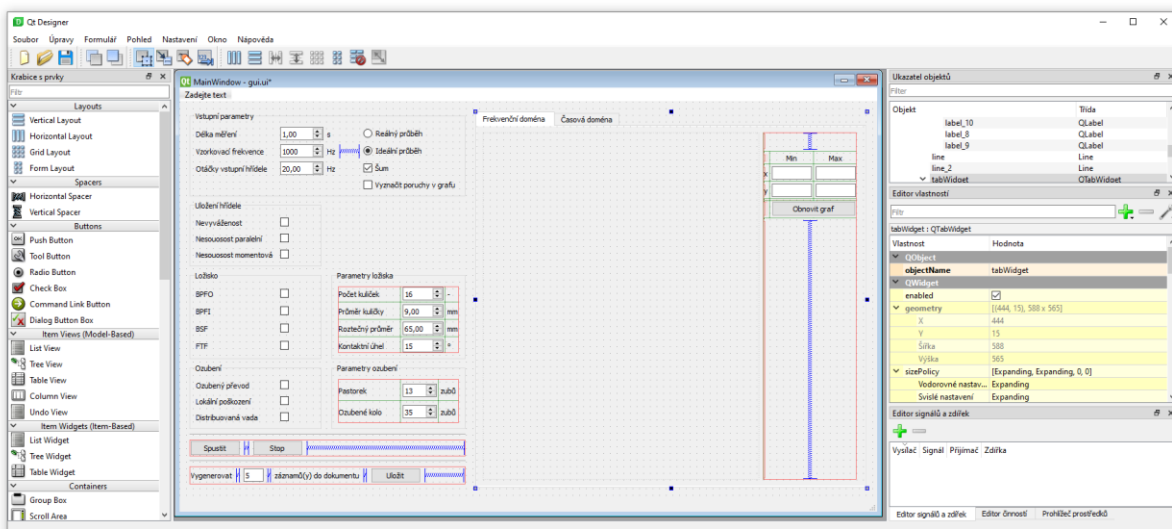
4.1.2 PyQt5

PyQt5 je sada nástrojů pro tvorbu grafických uživatelských rozhraní (GUI) pro práci s SQL databázemi, 2D a 3D grafikou, multimédií, síťovou komunikaci a další. Ve skutečnosti se jedná o sadu vazeb Pythonu, která umožňuje používání aplikačního frameworku Qt od společnosti Qt. Slouží tedy jako most mezi jazykem Python a C++, v němž je Qt vyvíjen. [27]

Qt je multiplatformní sada nástrojů pro vývoj aplikací pro stolní počítače, vestavěné systémy (embedded systems) a mobilní zařízení. Je podporován na většině běžných operačních systémů. [27]

Součástí PyQt je také aplikace Qt Designer (Obr. 31). Jedná se o návrhář grafického uživatelského rozhraní. Ten umožňuje jednoduchý a rychlý způsob tvorby GUI způsobem drag & drop. PyQt je schopen z Qt Designeru vygenerovat Python kód, který pak lze dále upravovat. Dále je také možné vkládat do Qt Designeru nové ovládací prvky grafického uživatelského rozhraní napsané v Pythonu, které defaultně nejsou k dispozici, jako jsou například grafy. [28]

PyQt5 je vydáván pod licencí GPL v3, která vyžaduje kompletní zveřejnění zdrojového kódu, což je pro případ této diplomové práce dostačující.



Obr. 31) GUI návrhář Qt Designer

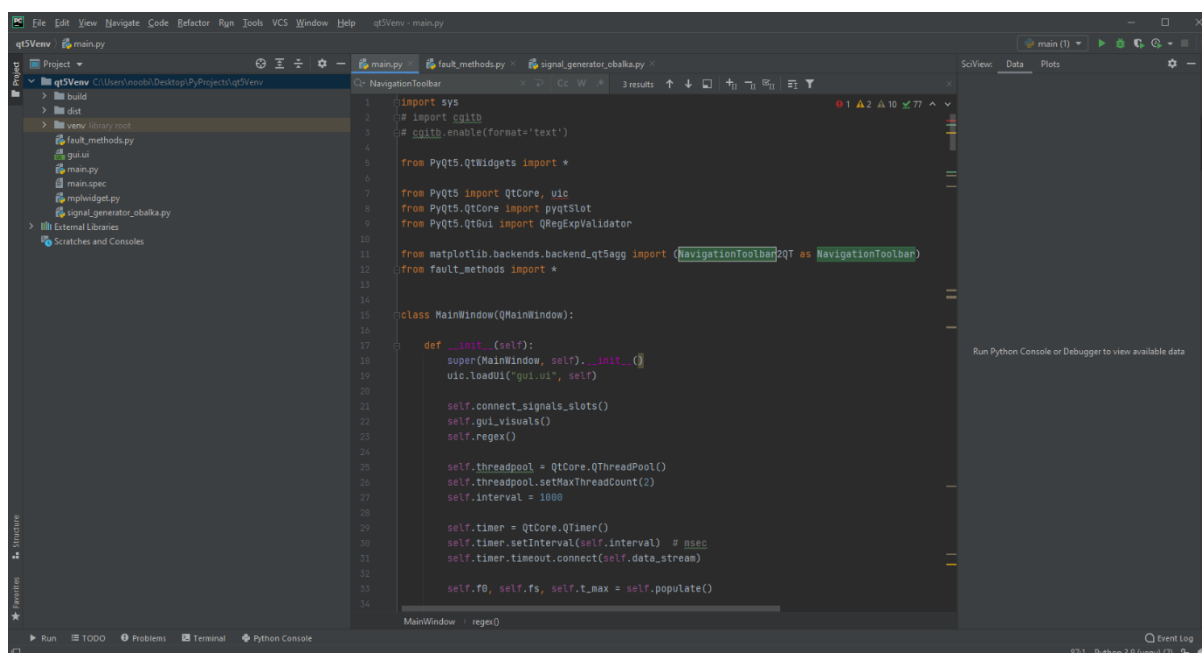
4.1.3 PyCharm

PyCharm (Obr. 32) je integrované vývojové prostředí (IDE) pro Python vyvíjené Českou společností JetBrains. Jedná se o jeden z nejrozšířenějších IDE pro vývoj v Pythonu, adaptovaný mnoha softwarovými společnostmi. Tak jak je u Pythonu zvykem je i tento software multiplatformní. [29]

Smyslem IDE je zefektivnit a zrychlit programování a vývoj aplikací. K tomu PyCharm nabízí řadu nástrojů jako je například inteligentní editor kódu, jež nabízí automatické doplňování a psaní kódu, sadu barevných schémat pro lepší přehlednost kódu, kontrolu pravopisu a konvence stylu psaní, identifikace chyb a další. [29]

Mezi další užitečné nástroje patří možnost tvořit záložky a úkoly přímo v kódu, hledání referencí k jednotlivým metodám bez opouštění IDE, včetně možnosti nahlédnout přímo na zdrojový kód hledané metody. Dále nabízí možnost provádět změny v textu napříč vícero závislými soubory najednou tak aby nedošlo ke ztrátě funkčnosti například přejmenováním proměnné.

PyCharm nabízí i jednoduchý způsob integrace cizích nástrojů skrze jejich API, příkladem může být výše uvedený Qt Designer, který lze nainstalovat a obsluhovat přímo skrze PyCharm. [29]



Obr. 32) Náhled na projekt v prostředí PyCharm

4.2 Použité knihovny

Python knihovna je sbírka funkcí a metod, která umožňuje provádět mnoho různých úkonů bez psaní vlastního kódu. Samotný Python obsahuje více než 200 různých knihoven, které zajišťují jeho funkcionalitu. V některých případech je ale vhodné použití externích knihoven například pro práci s daty nebo jejich vizualizaci. [30]

V této diplomové práci byly použity následující moduly:

4.2.1 NumPy

NumPy je základní balíček pro vědecké výpočty v Pythonu. Jedná se o knihovnu Pythonu, která poskytuje objekt vícerozměrného pole, různé odvozené objekty (například maskovaná pole a matice) a řadu rutin pro rychlé operace s poli, včetně matematické, logické a tvarové manipulace, třídění, výběry, vstupy / výstupy, diskrétní Fourierovy transformace, základní lineární algebru, základní statistické operace, náhodné simulace a další. [31]

4.2.2 Matplotlib

Matplotlib je multiplatformní knihovna pro vizualizaci dat a grafické vykreslování v Pythonu a jeho numerické rozšíření NumPy. Jako takový slouží jako schopná open source alternativa k MATLABu. Matplotlib API (Application Programming Interface) lze také využít pro vkládání grafů do aplikací s GUI (Grafické uživatelské rozhraní). [32]

4.2.3 PyInstaller

PyInstaller slouží k zabalení Python aplikace a všech jejích závislostí (knihovny, moduly, ...) do jednoho balíčku. Uživatel pak může spustit zabalenou aplikaci bez instalace Python interpreteru a jakýchkoli dalších modulů. [33]

PyInstaller přečte zadaný Python skript a analyzuje kód tak, aby zjistil, jaké moduly a knihovny skript potřebuje ke spuštění. Následně vytvoří kopie těchto souborů, včetně Python interpreteru, a umístí je spolu se skriptem do jedné složky nebo v případě že je to možné do jednoho spustitelného souboru. [33]

Balíček pak lze distribuovat dalším uživatelům kteří tak mohou rovnou spustit zabalený program. Pro uživatele se aplikace jeví plně samostatně – nemusí instalovat konkrétní verzi Pythonu ani žádné další moduly.

4.2.4 Cgitb

Modul `cgitb` poskytuje speciální obsluhu výjimek pro skripty Pythonu. Výjimkou se v řeči Pythonu rozumí chyba, která nevznikla v syntaxi kódu, nýbrž se jedná o chybu logického rázu. Příkladem může být dělení nulou nebo když do proměnné která očekává číslo, dosadíme text. [34]

Dojde-li po aktivaci tohoto modulu ke vzniku neočekávané výjimky, zobrazí se podrobný protokol o nastalé situaci. V hlášení jsou pak zobrazeny výňatky ze zdrojového kódu a hodnoty argumentů a místních proměnných pro aktuálně spuštěné funkce, které mohou dopomoci k vyřešení problému. Tyto informace je pak možné volitelně uložit do souboru nebo přímo zobrazit v terminálu. [34]

V běžných případech, pokud dojde k neošetřené výjimce během chodu programu s GUI, dojde k ukončení celého programu bez toho, aby se zobrazila chybová hláška, což je velmi problematické při hledání příčiny. Tento modul namísto toho chybu zachytí, informace o ní uloží a dovolí programu pokračovat dál. [35]

4.3 Program Generátor signálů

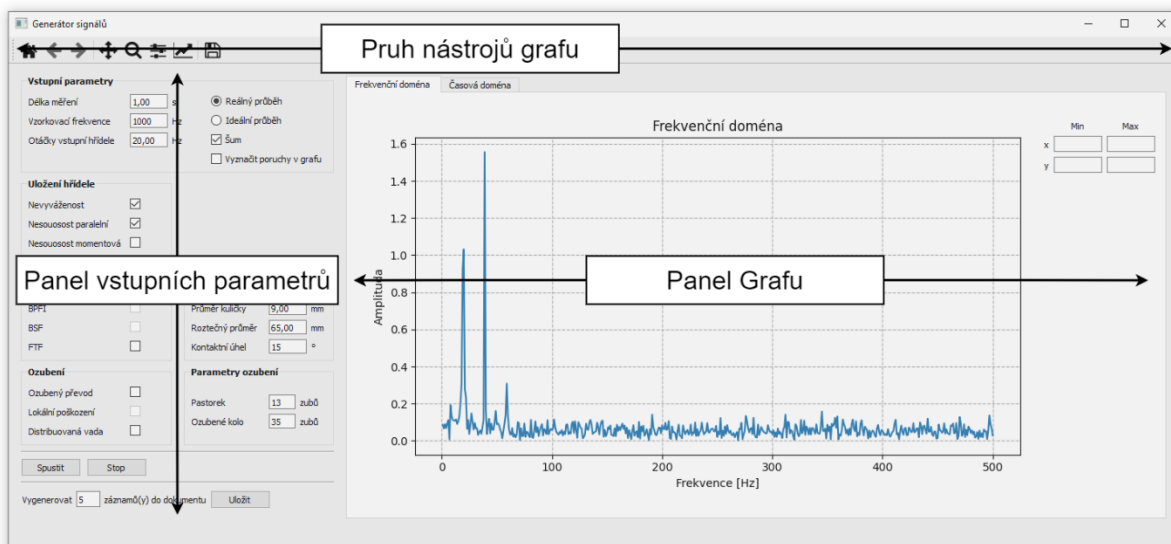
Generátor signálů je desktopová aplikace pro generování poruchových signálů dle uživatelem zvolených parametrů. V následující kapitole jsou popsány jednotlivé funkce programu a jeho ovládání.

4.3.1 Účel programu

Program slouží k simulaci signálů různých typů závad pro potřeby výuky vibrodiagnostiky. Průběhy závad jsou do určité míry idealizované tak, aby bylo při výuce jasně poznat, jak se různé typy poruch projevují. Program je koncipován primárně jako simulátor vibrodiagnostických dat s možností vizualizace generovaných dat. Možnosti analýzy dat jsou proto omezeny pouze na FFT. Uživatel si může data z programu uložit a dále na nich pracovat v programech určených pro analýzu vibrací.

4.3.2 Rozvržení uživatelského rozhraní

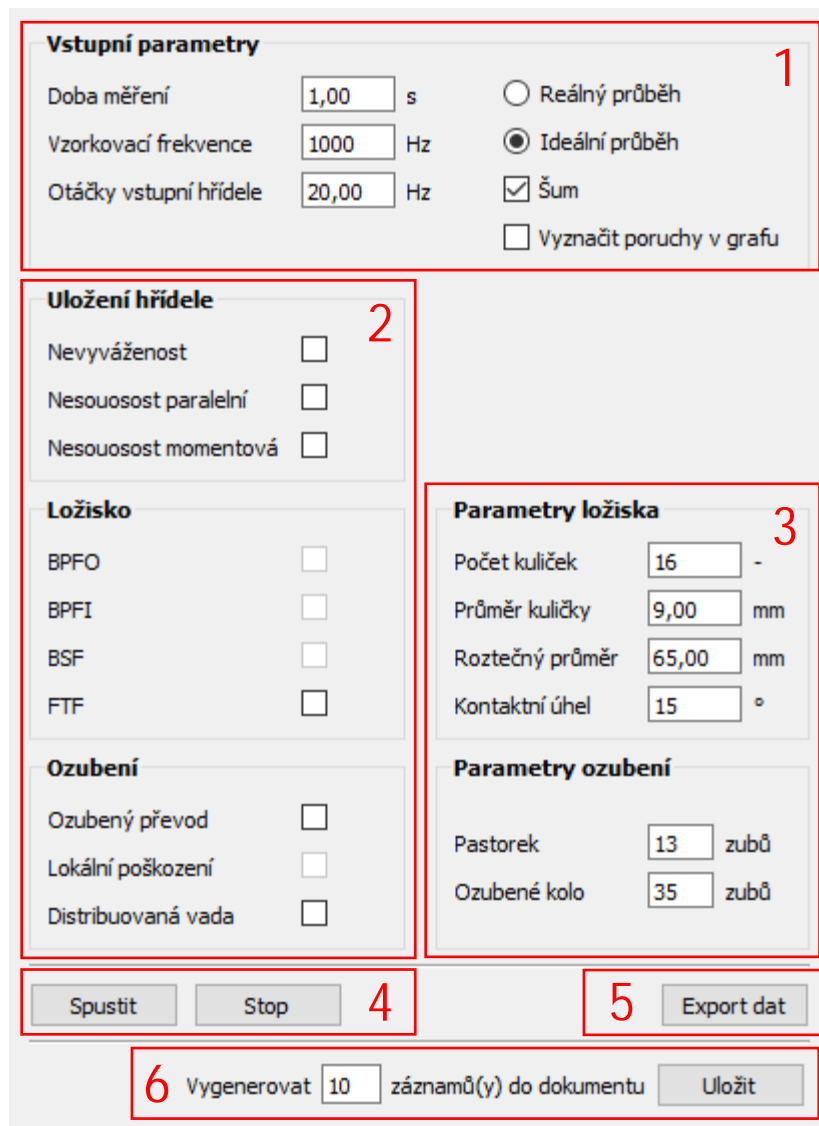
Uživatelské rozhraní je navrženo tak, aby měl uživatel k dispozici všechny ovládací prvky v jednom okně. Na Obr. 33) jsou vyznačeny hlavní ovládací prvky aplikace.



Obr. 33) Hlavní prvky grafického uživatelského rozhraní

4.3.3 Panel vstupních parametrů

Panel vstupních parametrů (Obr. 34), slouží uživateli k zadávání všech vstupních parametrů. Doba měření, vzorkovací frekvence, otáčky vstupní hřídele a parametry ložiska a ozubení jsou po spuštění programu předvyplněny. K pohybu mezi jednotlivými políčky lze použít tabulátor.



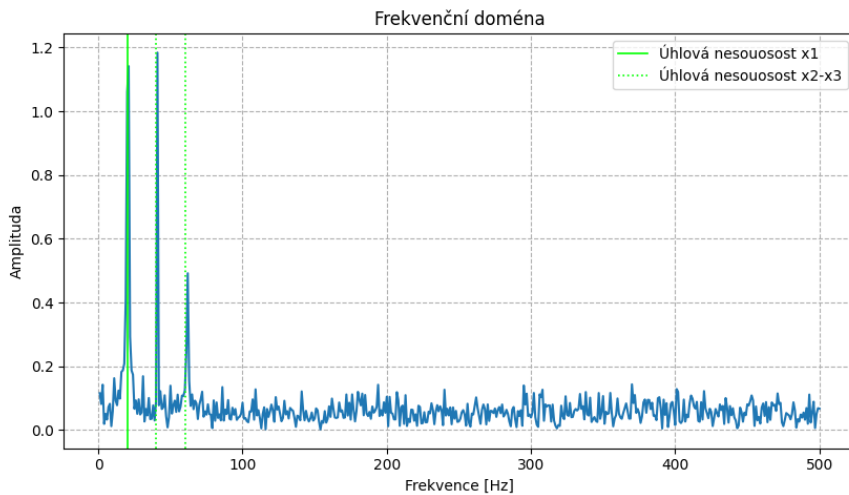
The screenshot shows a software interface for setting input parameters, divided into several sections:

- 1 Vstupní parametry:** Includes input fields for 'Doba měření' (1,00 s), 'Vzorkovací frekvence' (1000 Hz), and 'Otáčky vstupní hřídele' (20,00 Hz). It also features radio buttons for 'Reálný průběh' and 'Ideální průběh' (selected), a checked checkbox for 'Šum', and an unchecked checkbox for 'Vyznačit poruchy v grafu'.
- 2 Uložení hřídele:** Contains three unchecked checkboxes: 'Nevyváženost', 'Nesouosost paralelní', and 'Nesouosost momentová'.
- Ložisko:** Lists four bearing types with unchecked checkboxes: BPFO, BPFI, BSF, and FTF.
- 3 Parametry ložiska:** Includes input fields for 'Počet kuliček' (16), 'Průměr kuličky' (9,00 mm), 'Roztečný průměr' (65,00 mm), and 'Kontaktní úhel' (15 °).
- Ozubení:** Contains three unchecked checkboxes: 'Ozubený převod', 'Lokální poškození', and 'Distribuovaná vada'.
- Parametry ozubení:** Includes input fields for 'Pastorek' (13 zubů) and 'Ozubené kolo' (35 zubů).
- 4 Spustit / Stop:** Two buttons for starting and stopping the process.
- 5 Export dat:** A button for exporting data.
- 6 Vygenerovat 10 záznamů(y) do dokumentu Uložit:** A button to generate 10 records to a document and save them.

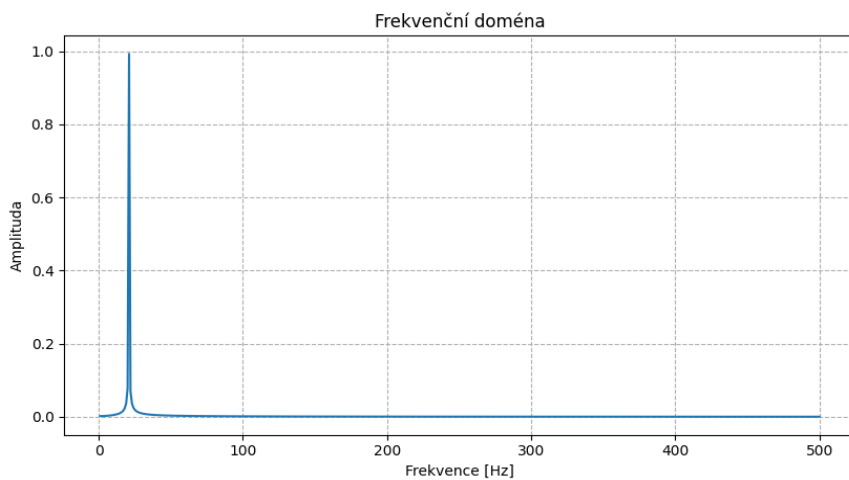
Obr. 34) Panel vstupních parametrů

Vstupní parametry (č. 1)

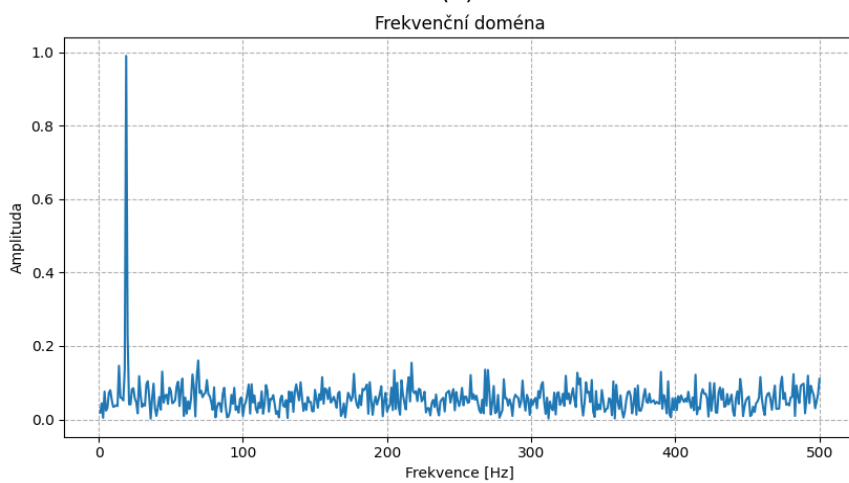
V sekci vstupní parametry se vybírají základní parametry simulace jako je doba simulovaného měření, vzorkovací frekvence a otáčky vstupní hřídele. Dále je zde možnost vyznačit v grafu poruchové frekvence (Obr. 35) a přidat k signálu šum (Obr. 36).



Obr. 35) Vyznačení úhlové nesouososti v grafu s legendou



(a)

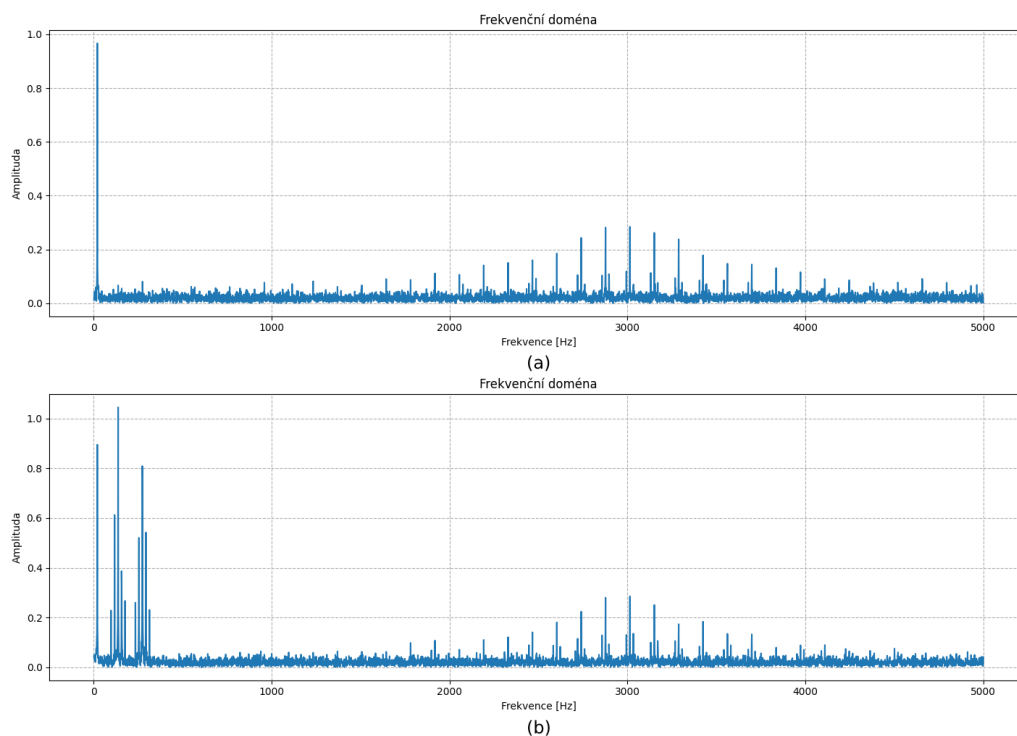


(b)

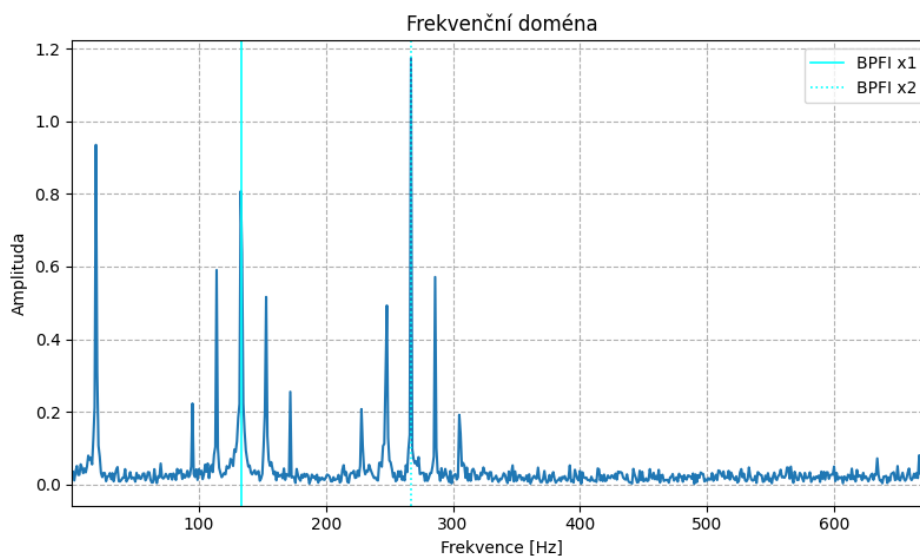
Obr. 36) Porovnání signálu nezatíženého (a) a zatíženého (b) Gaussovským šumem

Vstupní parametry (č. 1) – Přepínač průběhu

Přepínač reálný / ideální průběh slouží k volbě typu simulace. Volba reálný průběh generuje signál v surovém stavu. Tyto hodnoty jsou blíže k reálnému projevu konkrétních závad. Vzhledem k tomu, že program slouží hlavně k simulaci signálu, a ne k jeho analýze, nejsou některé závady zřetelné bez dalšího zpracování dat například filtrováním. Volba ideální průběh proto k signálu přidá další frekvenční složky tak aby byly frekvence závad zřetelné i bez dalších úprav (Obr. 37).



Obr. 37) (a) Reálný, (b) Ideální průběh



Obr. 38) Detail – Ideální průběh závady BPFI s vyznačením poruchových frekvencí

Výběr poruch (č. 2)

V této části panelu vstupních parametrů lze navolit jednotlivé poruchy včetně jejich kombinací. Některé poruchy ložisek a lokální poškození ozubení lze zvolit jen v případě kdy je vzorkovací frekvence dostatečně vysoká. Konkrétní hodnota činí 7680 Hz. Hodnota je dána jako 2,56x vlastní frekvence jednotlivých komponent. Tyto závady nelze při nižších vzorkovacích frekvencích simulovat a následně ani zobrazit v grafu FFT. Pokud uživatel najede myší na zablokovanou položku, zobrazí se popisek s vysvětlivkou tohoto omezení.

Parametry ložiska a ozubení (č. 3)

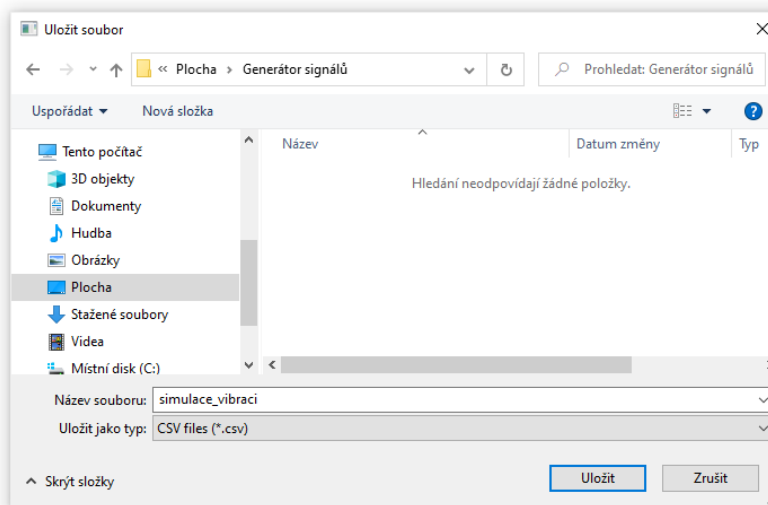
Zde lze nastavit parametry ložiska a ozubení.

Tlačítka Spustit a Stop (č. 4)

Tlačítka Spustit a Stop slouží ke spuštění a zastavení simulace. Spuštěním simulace se začnou generovat a vizualizovat data. Hodnoty v grafu se obnovují ve stejném tempu jako je doba měření. Po zastavení simulace zůstanou vyobrazeny poslední vykreslené data v grafu.

Export aktuálně zobrazených dat (č. 5)

Tlačítko Export dat umožňuje uložit aktuálně zobrazená data do souboru. Kliknutí na tlačítko Export dat otevře standardní ukládací okno Windows (Obr. 39). Uživatel může zvolit umístění dokumentu, jméno a formát. V prvním řádku vygenerovaného dokumentu je uloženo časové pole, v druhém pak aktuálně zobrazená data.



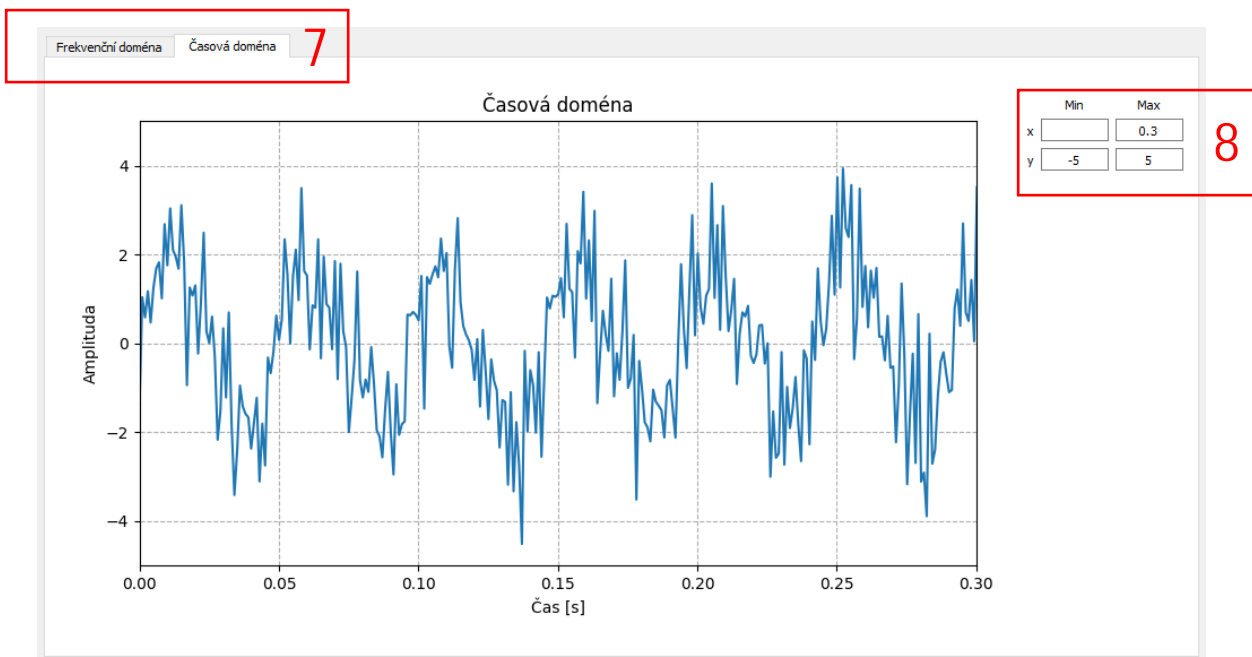
Obr. 39) Export dat do dokumentu

Generování dat do souboru (č. 6)

Program umožňuje vygenerovat více signálů najednou a uložit je do souboru. Uživatel si nejprve navolí vstupní parametry a jednotlivé závady. Pak si vybere, kolik signálů chce vytvořit a klikne Uložit.

4.3.4 Panel grafu a pruh nástrojů grafu

Zvolené závady se po spuštění simulace vykreslí v grafu (Obr. 40). K dispozici je graf ve frekvenční a časové doméně. Mezi grafy se dá přepínat tlačítky v levém horním rohu (č.7). Zmenšením nebo rozšířením okna programu dojde ke změně velikosti okna grafu.



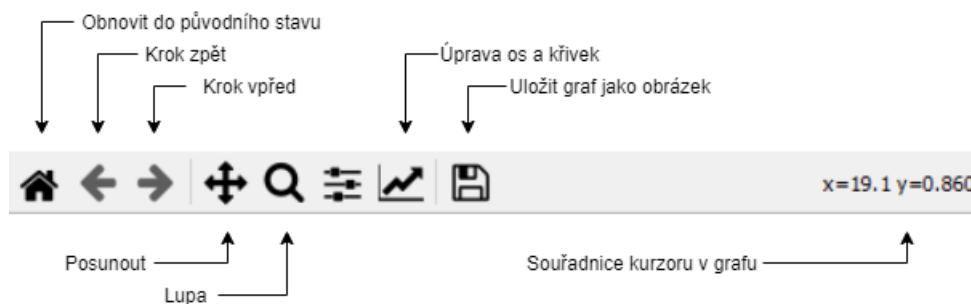
Obr. 40) Ovládací prvky grafu

Limity os (č. 8)

Limity os lze zadávat samostatně nebo v páru Min / Max. Limity ypsilonové osy v časové doméně je nutné zadat v páru. Tak jako v části vstupních parametrů jsou tyto vstupy chráněny před vyplněním neplatných hodnot. Kontrola hodnot je prováděna pomocí regulárních výrazů. Co je to regulární výraz je popsáno v kapitole 4.4.2.

Pruh nástrojů grafu

Další možnosti ovládání grafu jsou v horním panelu aplikace (Obr. 41). Je zde možnost posouvat graf libovolným směrem, přiblížit nebo oddálit vybranou část grafu a upravit barvu a šířku křivky. Dále je zde možnost uložit aktuálně zobrazený graf ve formátu obrázku. V levé horní části panelu se zobrazují aktuální souřadnice polohy kurzoru myši v grafu.



Obr. 41) Pruh nástrojů grafu

Kontrola vstupních hodnot

Všechny textové pole v aplikaci jsou chráněny vůči vyplnění neplatných hodnot. Platnou hodnotou se v tomto případě rozumí pouze kladná čísla. Hodnoty jako počet kuliček, počet zubů nebo počet generovaných záznamů do dokumentu jsou navíc omezeny pouze na celočíselné hodnoty. Pokud uživatel hodnotu v poli smaže, automaticky se nastaví poslední platná hodnota.

4.3.5 Spuštění simulace

Pro navození pocitu, že uživatel sleduje reálný vibrodiagnostický signál z připojeného zařízení se zobrazí data obnovují s časovou prodlevou, která odpovídá délce měření.

Po spuštění simulace lze nadále dynamicky volit jednotlivé poruchy, upravovat osy grafu a měnit parametry ložisek a ozubení. Vstupní parametry doba měření, vzorkovací frekvence a otáčky vstupní hřídele jsou zablokovány a lze je měnit pouze po zastavení simulace (Obr. 42).

The image shows a software control panel for simulation parameters, organized into several sections:

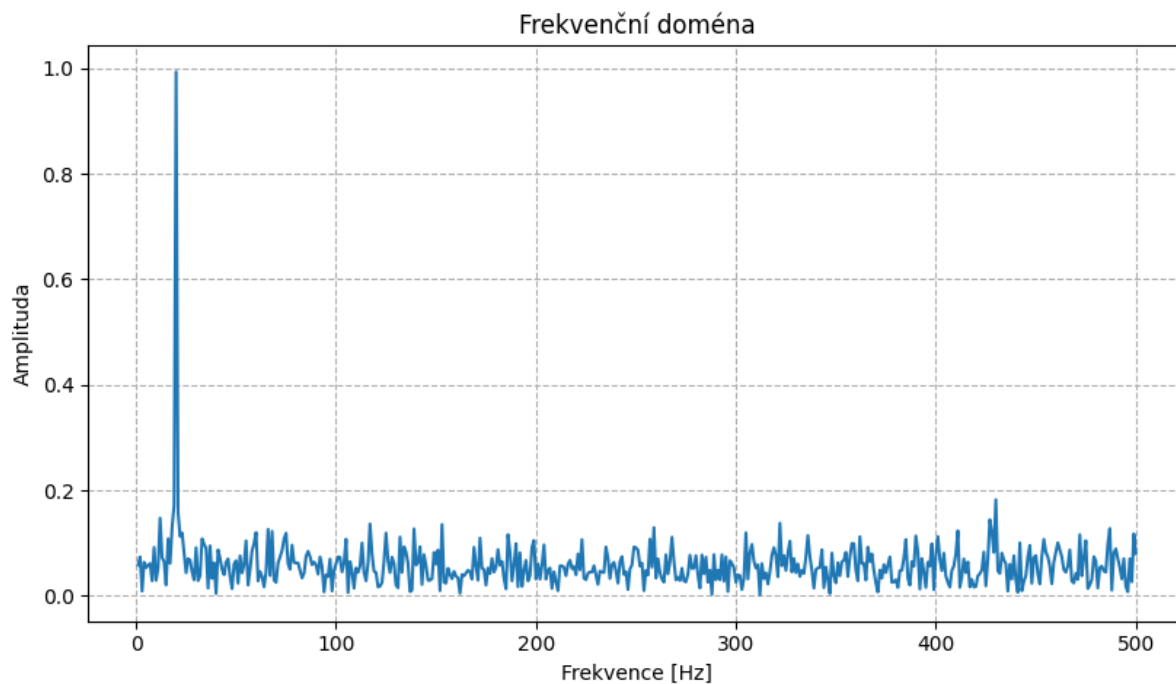
- Vstupní parametry**:
 - Doba měření: 2,00 s
 - Vzorkovací frekvence: 10000 Hz
 - Otáčky vstupní hřídele: 25,00 Hz
 - Radio buttons: Reálný průběh, Ideální průběh
 - Checkboxes: Šum, Vyznačit poruchy v grafu
- Uložení hřídele**:
 - Nevyváženost:
 - Nesouosost paralelní:
 - Nesouosost momentová:
- Ložisko**:
 - BPFO:
 - BPFI:
 - BSF:
 - FTF:
- Parametry ložiska**:
 - Počet kuliček: 16 -
 - Průměr kuličky: 9,00 mm
 - Roztečný průměr: 65,00 mm
 - Kontaktní úhel: 15 °
- Ozubení**:
 - Ozubený převod:
 - Lokální poškození:
 - Distribuovaná vada:
- Parametry ozubení**:
 - Pastorek: 13 zubů
 - Ozubené kolo: 35 zubů

At the bottom, there are control buttons: "Spustit", "Stop", and "Export dat". Below these is a field "Vygenerovat 10 záznamů(y) do dokumentu" and a "Uložit" button.

Obr. 42) Ovládací panel vstupních parametrů po spuštění simulace

Referenční signál pro zařízení bez závad je amplituda o hodnotě 1 základní otáčkové frekvence (Obr. 43) bez přítomnosti jiných dominantních frekvencí s výjimkou šumu. Účelem programu je vygenerovat signál, který by odpovídal odezvě stroje, který by měl vybranou závadu. Nejedná se o fyzikální model konkrétního stroje.

Simulovaná frekvence základní otáčkové frekvence se může lišit $\pm 5\%$ od zadané hodnoty. Toto chování je záměrné vzhledem k tomu, že i v reálném provozu se měřené hodnoty od nastavených mírně liší.



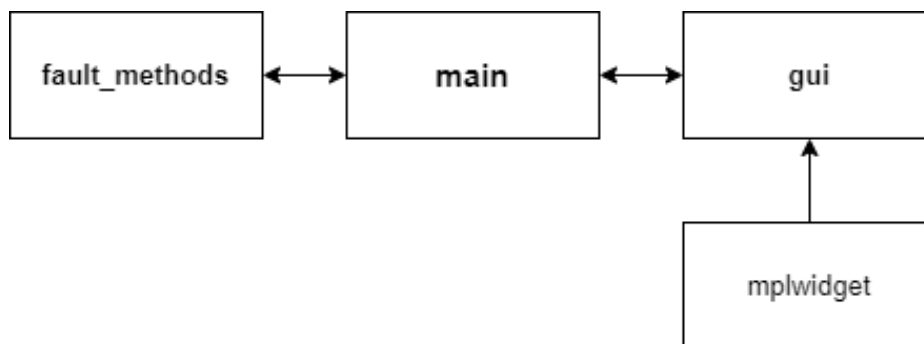
Obr. 43) FFT graf nepoškozeného zařízení

4.4 Princip funkce programu a jeho struktura

Na rozdíl od konzolové aplikace, kde jsou jednotlivé kroky a funkce provedeny postupně, je aplikace, založená na grafickém uživatelském rozhraní, řízena událostmi. Funkce a metody se spouštějí v reakci na akce uživatele, jako je kliknutí na tlačítko, výběr položky ze seznamu, kliknutí myší atd. Těmto akcím se říká události. [36]

V PyQt jsou zdrojem takovýchto událostí widgety (grafické komponenty – tlačítka, textové pole, seznamy, okna atd.), ze kterých je GUI sestaveno. Každý PyQt widget, který je odvozen ze třídy QObject, je navržen tak, aby vydával signál v reakci na jednu nebo více událostí. Samotný signál neprovádí žádnou akci. Místo toho je připojen ke slotu. Slot může být libovolná volaná funkce v Pythonu. [36]

Na Obr. 44) je vyobrazena jednoduchá struktura programu, kde komunikace mezi hlavní částí programu (main) a grafickým rozhraním (gui) probíhá v obou směrech. Gui vysílá signály, main dle typu signálu vytváří odezvu. Komunikace mezi částí programu generující jednotlivé signály závad (fault_methods) a hlavní částí programu probíhá také oběma směry. Main předává vstupní parametry a získává zpátky hotový signál. Mplwidget pak slouží k vykreslování grafů v grafickém rozhraní.



Obr. 44) Souborová struktura programu

4.4.1 Multithreading

Běžným problémem při vytváření aplikací s GUI v Pythonu je „zamrzání“ programu při pokusu o provedení dlouhotrvajících úloh v pozadí.

Jak již bylo řečeno, aplikace vytvořené v Qt (a i většina ostatních GUI) jsou založeny na událostech. To znamená, že provádění úkonů je řízeno v reakci na interakci uživatele, signály a časovače. V aplikaci řízené událostmi vytvoří kliknutí na tlačítko událost, kterou aplikace následně zpracuje. Události jsou vkládány do a odebírány z fronty a postupně zpracovány. [37]

Smyčka událostí, jež se spustí vyvoláním příkazu `.exec_()` běží ve stejném vlákne jako kód Pythonu. Vlákno, které ovládá tuto smyčku, běžně označované jako vlákno GUI, zároveň zpracovává veškerou komunikaci s operačním systémem. Ve výchozím nastavení, jakýkoli úkol spuštěný z této smyčky bude zpracován synchronně s vlákem GUI. To v praxi znamená, že kdykoli aplikace PyQt stráví nějaký čas vykonáváním zadaného úkolu dojde k zamrznutí okna GUI. [37]

Pokud je úkol jednoduchý a vrátí kontrolu na vláknem GUI dostatečně rychle, bude toto zamrznutí pro uživatele nepostřehnutelné. Pokud je ale úkol časově náročný, například čtení dat ze souboru nebo stahování souboru, nastanou problémy. Uživateli se aplikace bude jevit, že nereaguje (protože opravdu nereaguje). Důvodem je to, že aplikace nekomunikuje s operačním systémem a snaží se vyřešit zadaný úkol. [37]

Řešením problému je použití paralelních procesů. Kdy dojde k vytvoření dalšího vlákna, které běží paralelně s vláknem GUI.

4.4.2 Soubor main.py

Spuštěním programu dojde k inicializaci hlavního vlákna GUI. V hlavním vlákně běží smyčka událostí aplikace a ostatní Python kód. Rovněž zpracovává všechny okna, dialogy a zajišťuje komunikaci s operačním systémem.

Po inicializaci hlavního vlákna dojde k načtení přednastavených proměnných z grafického rozhraní. Pak je program připraven a čeká na příchozí signály. Signály lze rozdělit do několika kategorií podle jejich funkce. Lze je dělit na signály, které upravují vzhled grafu, které mění parametry simulace, dále na signál pro zahájení a ukončení simulace a signál pro vygenerování dat do souboru.

Vytváření kontinuálního signálu

Efekt obnovování simulovaných dat v určitém časovém intervalu, popsany v kapitole Obr. 41), je zajištěn pomocí časovačů. Časovač vyšle signál po uplynutí doby, která je rovna délce simulovaného měření. Signál spustí sadu funkcí, které vytvoří a zpracují signál a vloží data do grafu. Tento přístup je možný díky tomu, že vygenerování signálů trvá ve skutečnosti jen velmi krátkou dobu na rozdíl od skutečného měření a nedojde ke zpomalení vlákna GUI. Pokud by byly načítány data z reálného zařízení, bylo by potřeba použít přístup popsany v kapitole 4.4.1.

Ukládání dat do souboru

Funkce slouží k vygenerování více signálů najednou a uložení dat do dokumentu. Spuštěním funkce otevře běžné dialogové okno pro uložení dokumentu, pro daný operační systém. Uživatel může změnit cílové uložení, jméno dokumentu a jeho typ. Výchozí typ dokumentu je .CSV.

Vzhledem k tomu, že uživatel může chtít vytvořit tisíce záznamů najednou a následně je uložit, je nutné tento proces oddělit od hlavního vlákna GUI a proces provádět na pozadí. Pokud by se tak nestalo, bude se program uživateli jevit, jako že neodpovídá, i když ve skutečnosti zpracovává velké množství dat.

Kontrola vstupů - regex

Jedná se o funkci, která má za úkol kontrolu uživatelských vstupů. Ke kontrole používá regulární výrazy (zkráceně regex).

Regex slouží k vyhledávání, nahrazování nebo kontrole textu. Jedná se o speciální řetězec znaků, který odpovídá určitému vzoru. Příklad, jak vytvořit regulární výraz pro kladné nebo záporné číslo s desetinným místem je uveden v Tab 1) . [38]

Tab 1) Regulární výraz pro pozitivní nebo negativní hodnotu s desetinným místem

| Regulární výraz | Význam | Povinné |
|-----------------|---|---------|
| -?\d+(\.\d+)? | Negativní nebo pozitivní číslo s možným desetinným místem | |
| -? | Znaménko mínus | Ne |
| d+ | Libovolný počet čísel 0 - 9 | Ano |
| (\.\d+)? | Desetinná tečka následovaná libovolným počtem čísel 0 - 9 | Ne |

Tento regulární výraz dovolí uživateli vložit hodnotu která může, ale nemusí být záporná, je složena pouze z čísel 0 až 9 a může ale nemusí mít desetinná místa.

Ke kontrole vstupů dochází ihned při zadávání hodnot do patřičných polí. Uživatel proto vůbec nemůže vepsat hodnoty, které nejsou přípustné.

Funkce fft

Funkce vypočítá Fourierovu transformaci ze zadaných hodnot a vrátí hodnoty souřadnic (x, y).

```
def fft(self, wf):
    n = len(wf) # Délka signálu
    xf = np.fft.rfftfreq(n, 1 / self.fs) # Vypočte frekvenční okna
    yf = np.fft.rfft(wf) # Vypočte FFT reálné části spektra
    yf_mag = np.abs(yf) # Absolutní hodnota spektra
    yf_mag = yf_mag * 2 / n # Normalizace hodnot
    return xf, yf_mag
```

4.4.3 Soubor fault_methods.py

Soubor *fault_methods* obsahuje třídu *FaultMethods* v níž jsou naprogramovány funkce k jednotlivým poruchám. Ve většině případů simuluje jedna funkce právě jednu poruchu. Přidání nové poruchy je možné vytvořením nové funkce. Vzhledem k tomu, že jsou všechny vstupní parametry společné pro všechny nebo více než jednu poruchu, vkládají se všechny proměnné najednou v rámci třídy nikoli do jednotlivých funkcí.

Jednotlivé příklady simulace vibrodiagnostického signálu konkrétních poruch, jsou popsány v kapitole 4.6 Tvorba vibrodiagnostického signálu.

Tento soubor je kompatibilní s konzolovým skriptem bez grafického rozhraní.

4.4.4 Soubor mplwidget.py

Tento soubor slouží k vytvoření okna s grafem uvnitř GUI. Vzhledem k tomu, že PyQt5 nemá k dispozici komponenty pro vykreslování grafů je nutné si je vytvořit. V PyQt5 je k dispozici prázdný widget, ve kterém jde spustit libovolný python skript. Spojením tohoto elementu se skriptem *mplwidget.py* dojde k integraci okna s grafem do GUI.

4.4.5 Soubor gui.ui

Soubor gui.ui je automaticky generován z extérní aplikace Qt Designer. Generovaný kód je ve formátu XML. K přeložení na python kódu dochází automaticky při inicializaci programu.

```
def __init__(self):
    super(MainWindow, self).__init__()
    uic.loadUi("gui.ui", self)
```

Výhodou tohoto řešení je možnost dynamicky měnit vzhled aplikace bez nutnosti převádění souboru .ui na soubor .py při každé změně grafického rozhraní.

4.4.6 Instalace

Pokud chce uživatel přidat nové poruchy nebo jakkoli jinak měnit program je nutné mít nainstalovaný Python 3.9, PIP, PyQt5 a ostatní knihovny uvedené v kapitole 1.1. PIP je správce balíčku pro moduly programovacího jazyka Python. PIP slouží k instalaci všech požadovaných knihoven tak, aby je uživatel nemusel instalovat jednu po druhé. Seznam těchto knihoven je uveden v souboru requirements.txt. K jejich instalaci lze použít jeden z následujících příkazů:

```
pip install -r requirements.txt
python3 -m pip install -r requirements.txt
```

Tím se automaticky spustí instalace všech knihoven uvedených v daném souboru. Program je následně připraven k použití.

4.5 Samostatný skript

Jedná se o konzolovou verzi aplikace generátoru signálu. Jeho výstupem je signál dle parametrů, které zadal uživatel. Skript používá stejný modul fault_methods.py, ke generování jednotlivých poruchových signálů. Kód je upraven tak aby přijímal vstupní parametry ve formě slovníku.

Skript kontroluje vstupní hodnoty zadané uživatelem. V případě, že uživatel zadá neplatné hodnoty, zobrazí se chybová hláška specifikující problém a skript se ukončí s Exit kódem 1 – Běžné chyby, chyby typu „dělení nulou“ a další nepřípustné operace.

Skript může sloužit například pro hromadné generování dat, kde každý vygenerovaný záznam bude mít jiné vstupní parametry. Takto vygenerovaná data mohou být použita jako vstup pro strojové učení zaměřené na rozpoznání závad rotačních strojů. Další možností je skript integrovat do programu pro analýzu vibrodiagnostického signálu.

Seznam možných parametrů a jejich formát je uveden na konci skriptu. Návod k instalaci je totožný s předešlou kapitolou 4.4.6. Návod k používání skriptu je obsažen v dokumentaci, která je součástí přílohy.

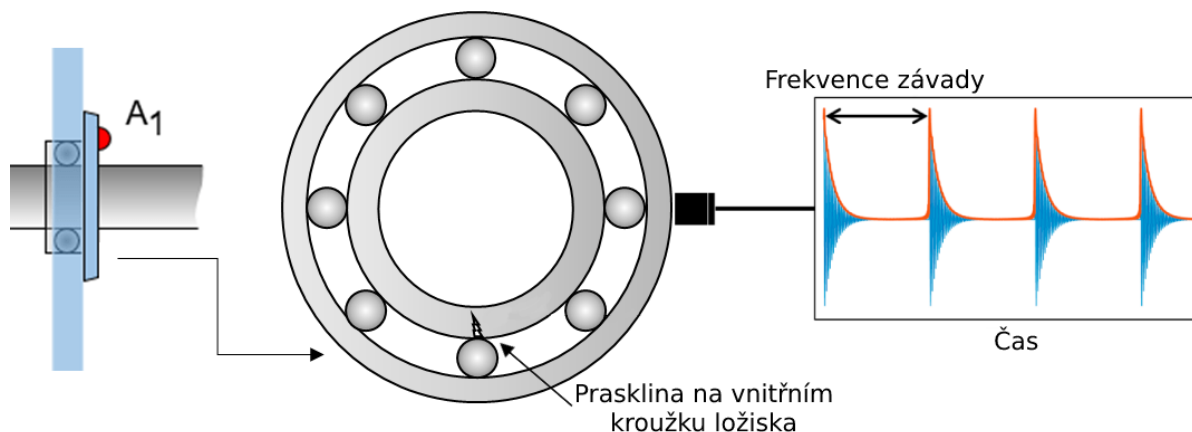
4.6 Tvorba vibrodiagnostického signálu

V následující kapitole je popsán postup generování několika různých poruchových signálů. Kód níže uvedených příkladů se místy liší od výsledné implementace v programu. Jednotlivé příklady jsou napsány tak aby šly spustit samostatně. K jejich funkci je nutná knihovna *NumPy*.

U některých níže uvedených hodnot se jedná pouze o odhad. Například rozpětí rezonančních frekvencí ložisek se může pohybovat přibližně mezi 3 až 50 kHz v závislosti na tvaru a materiálu ložiska. Proto je z důvodu jednodušší vizualizace například zvolena rezonanční frekvence ložiska 3 kHz, která nevyžaduje nastavení příliš vysoké vzorkovací frekvence, jež by jinak měla za důsledek výrazné potlačení šumu ve výsledném produktu, což je nežádoucí jev. Taktéž amplitudy jednotlivých poruch jsou nastaveny tak aby vizuálně imitovaly konkrétní poruchu. Jako bezporuchový stav je pak brán stav kdy amplituda základní otáčkové frekvence je kolem hodnoty 1. [39]

4.6.1 Závady ložisek

Pokaždé, kdy valivý prvek ložiska narazí na lokální závadu (Obr. 45) na vnitřním nebo vnějším kroužku nebo naopak, vyprodukuje tento náraz signál o periodě jedné z poruchových frekvencí ložiska. Každá z těchto poruch je charakterizována vlastní frekvencí, kterou obvykle uvádí výrobce nebo ji lze vypočítat ze specifikací ložiska (12). [40]



Obr. 45) Diagram poškozeného ložiska a poruchový signál [41]

Obvykle jsou nárazové signály na těchto poruchových frekvencích modulovány mnohem vyššími rezonančními frekvencemi ložisek, což vede k signálu ve formě exponenciálně se rozpadajícího kmitání, které se periodicky vyskytuje při poruchových frekvencích, jak můžeme vidět na obrázku (Obr. 45). [40]

Tento jev je matematicky srovnatelný s amplitudovou modulací (AM). Signál závady je vytvořen jako exponenciální funkce $x_1(t)$ (16) (Obr. 46),

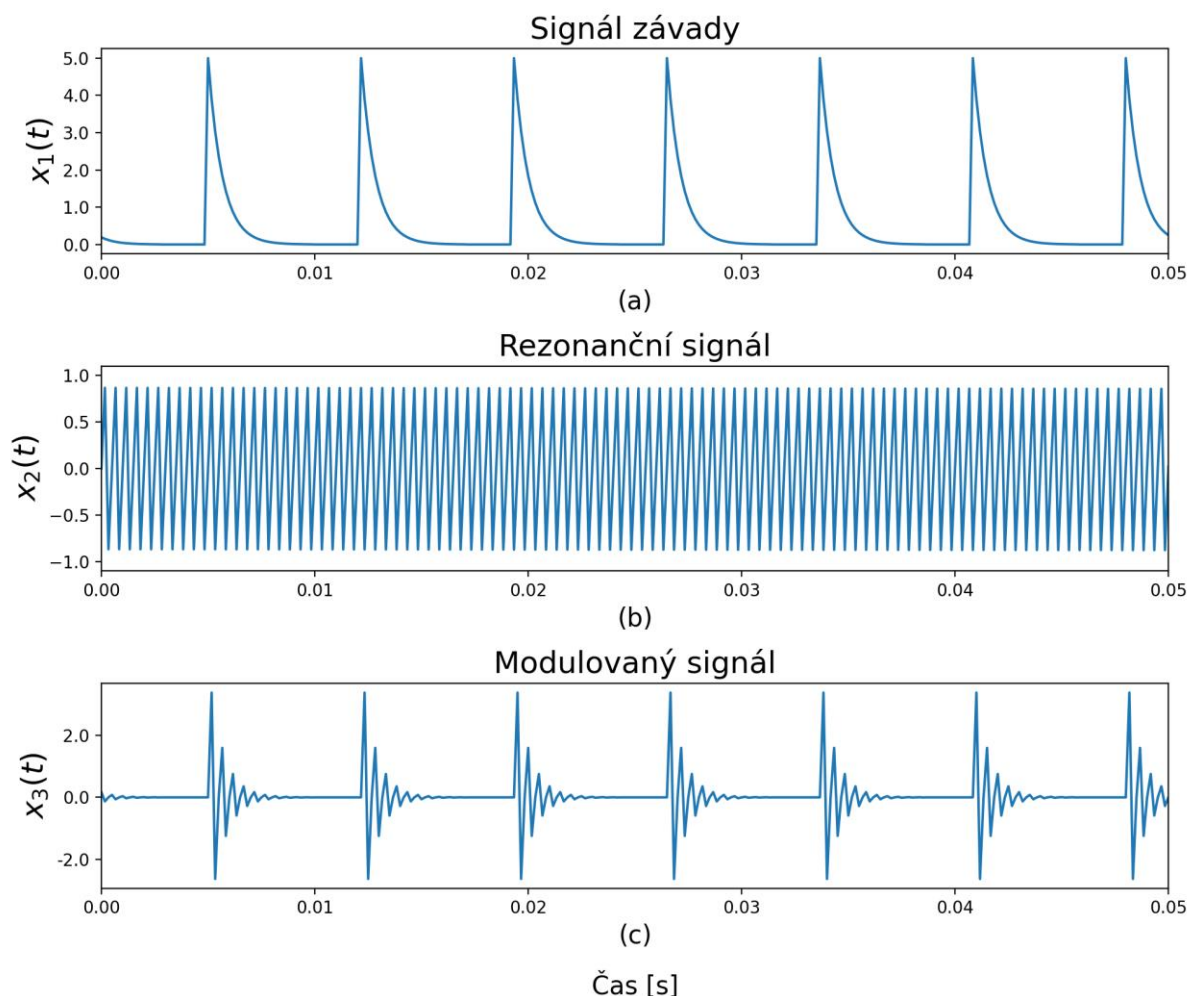
$$x_1(t) = A \cdot \exp(-\alpha \cdot t') \quad (16)$$

kteřá reprezentuje přechodný nárazový signál o frekvenci BPFI, kde t' odpovídá délce jednoho nárazu a A a α představují amplitudu impulzního signálu a parametr tlumení ložiska. Tento signál je pak modulován nosným signálem $x_2(t)$ daným sinusovou funkcí (17) (Obr. 46) (b)

$$x_2(t) = \sin(2 \cdot \pi \cdot f_r \cdot t) \quad (17)$$

s rezonanční frekvencí ložiska $f_r = 2000 \text{ Hz}$, která je mnohem vyšší než frekvence závady. Takto vznikne modulovaný signál (18) (Obr. 46) (c), který nese poruchu s její amplitudou. [40]

$$x_3(t) = x_1(t) * x_2(t) \quad (18)$$



Obr. 46) Modulace simulovaného signálu závady: (a) Signál závady, (b) Rezonanční signál, (c) AM signál, v časové doméně

Vzhledem k tomu, že princip simulace jednotlivých poruch ložiska je velmi podobný a liší se pouze poruchovou frekvencí dle typu závady, je zde pro ilustraci uvedena pouze porucha vnitřního kroužku ložiska – BPFI.

K simulaci závady je třeba nadefinovat vstupní parametry a základní parametry ložiska.

```
tmax = 5          # Doba měření [s]
fs = 10000       # Vzorkovací frekvence [Hz]
f0 = 20          # Frekvence vstupní hřídele [Hz]
```

```
n = 16           # Počet valivých elementů ložiska [-]
d = 0.009        # Průměr valivých elementů ložiska [m]
p = 0.065        # Průměr rozteče ložiska [m]
thetaDeg = 15    # Kontaktní úhel [°]
```

V dalším kroku vypočítáme frekvenci závady BPFI dle rovnice (12).

```
bpfi = n * f0 / 2 * (1 - d / p * np.cos(np.deg2rad(thetaDeg)))
```

Nadefinujeme rezonanční frekvenci ložiska 3000 Hz a délku jednoho nárazu 5 milisekund a vytvoříme jeden simulovaný náraz.

```
fBearing = 3000
tImpact = np.arange(0, 5e-3 - 1 / fs, 1 / fs) # Časové pole o délce 5 ms

A = 5 * np.exp(-tImpact * 1500)                # Signál závady  $x_1(t)$ 
xImpact = np.sin(2 * np.pi * fBearing * tImpact) # Nosný signál  $x_2(t)$ 
xBearingImpact = xImpact * A                   # Modulovaný signál  $x_3(t)$ 
```

Náraz uděláme periodickým, tak že vytvoříme nulové pole o délce t . Do tohoto pole dosadíme hodnotu 1 s periodou BPFI. Konvolucí polí xComb a xBearingImpact získáme opakující se náraz pokaždé kdy valivý element přejede přes prasklinu ve vnitřním kroužku.

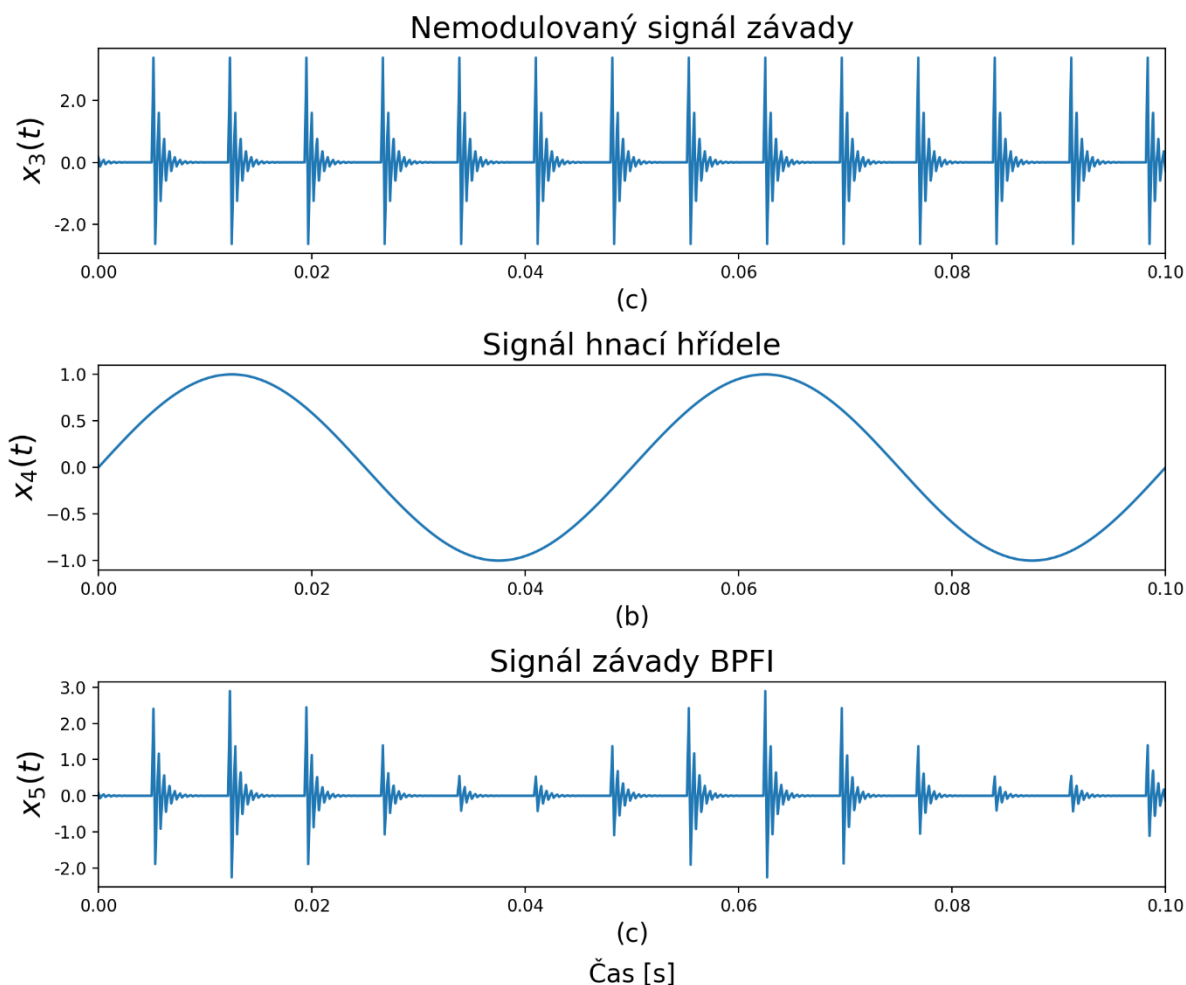
```
t = np.linspace(0.0, tmax, int(fs * tmax)) # Časové pole o délce 5 s

xComb = np.zeros(np.size(t))
xComb[1::int(np.around(fs / bpfi))] = 1
xBper = np.convolve(xComb, xBearingImpact, "same")
```

U závady na vnitřním kroužku ložiska dochází navíc k další modulaci signálem hnací hřídele, na které je ložisko uloženo.

```
fr_in = 1.0 * np.sin(2 * np.pi * f0 * t) # Signál hnací hřídele  $x_4(t)$ 
```

```
modulation_index = 1 / 2
carrier = 0.7 * xBearingImpact
modulator = 1.5 * fr_in
product = 0.7 * (1 + modulation_index * modulator) * carrier
```

Obr. 47) Modulace signálu závady hnacím hřídelem: (a) Nemodulovaný signál závady, (b) Signál hnací hřídele, (c) Signál závady ložiska – BPF1, v časové doméně

Navíc může signál zahrnovat určité diskrétní signály z ostatních komponent jako jsou například hřídele a ozubená kola ale také i šum. V tomto případě jsou reprezentovány nízkofrekvenční sinusoidou $x_4(t)$ (19) s amplitudou 1 a frekvencí 20 Hz a Gaussovským bílým šumem $x_6(t)$ (20), jak je můžeme vidět na Obr. 48) (a), (b) a jejich FFT (d) a (e).

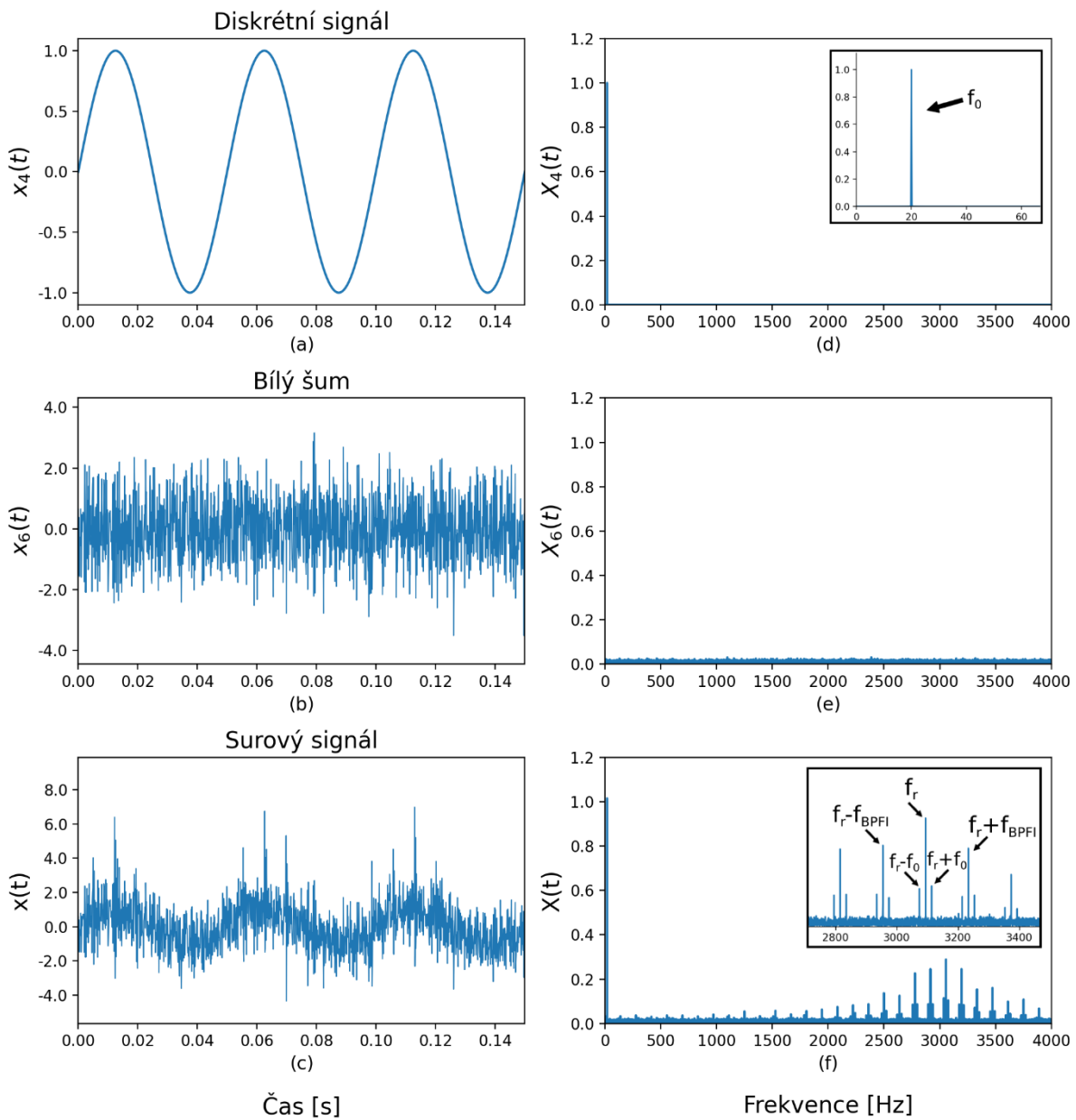
$$x_4(t) = \sin(2 \cdot \pi \cdot f_0 \cdot t) \quad (19)$$

$$x_6(t) \sim N(0,1) \quad (20)$$

Součtem výše uvedených signálů vznikne surový signál $x(t)$, který je na obrázku Obr. 48) (c) a jeho FFT (f) a vypočítá se dle rovnice (21).

$$x(t) = x_4(t) + x_5(t) + x_6(t) \quad (21)$$

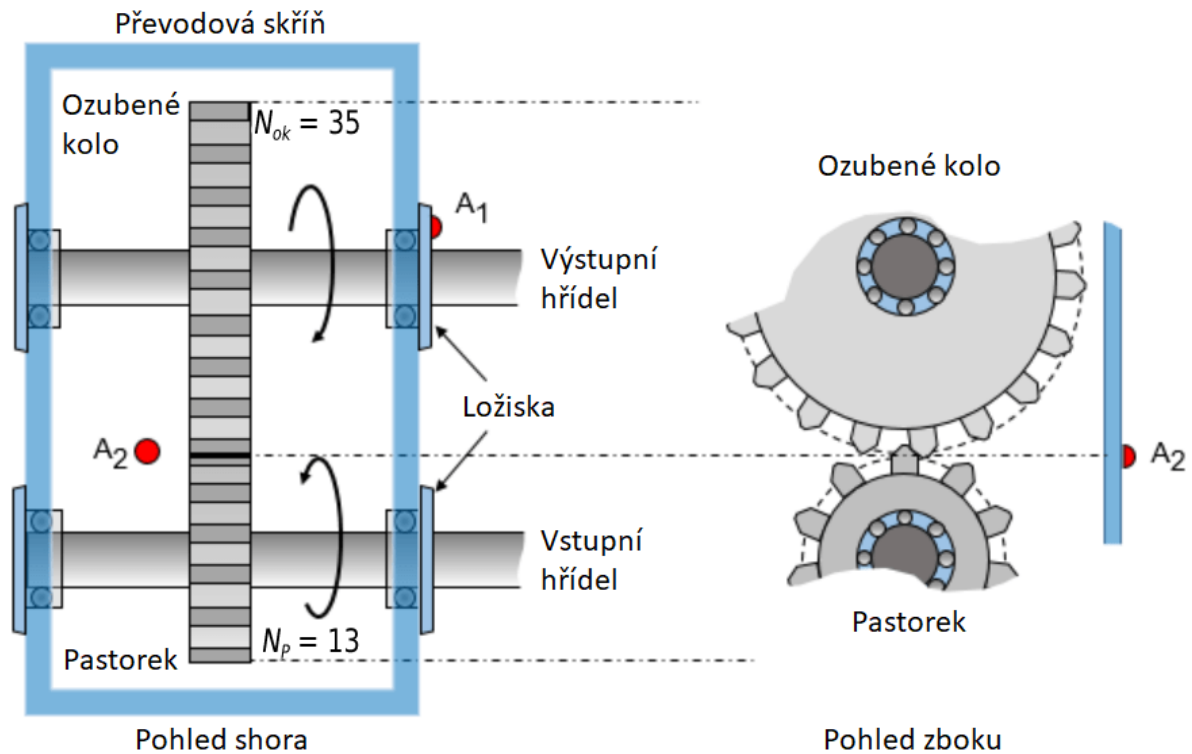
Za povšimnutí stojí to, že dominantním signálem na Obr. 48) (c) je právě diskrétní signál $x_4(t)$. Amplitudy ve vysokofrekvenčním regionu na Obr. 48) (f) reprezentují nosný signál s frekvencí 2000 Hz (rezonance ložiska) a rozestup postranního pásma náleží poruše ložiska. Amplitudy diskrétního a AM signálu jsou 1 a 0,3. To může mít za důsledek to, že se porucha ložiska ztratí mezi nosným signálem a dalšími diskrétními signály, jelikož signály ozubených kol a hřídelů zpravidla vykazují vyšší amplitudy než ty vadného ložiska. [40]



Obr. 48) Simulace závady ložiska: (a) Diskrétní signál, (b) Bílý šum, (c) Surový signál, v časové doméně; (d) Diskrétní signál, (e) Bílý šum, (f) Surový signál, ve frekvenční doméně

4.6.2 Závady ozubení

Pro příklad závad ozubených soukolí budeme uvažovat idealizované soukolí, kde pastorek má 13 zubů a ozubené kolo 35. Pastorek je spojen se vstupním hřídelem připojeným k hnacímu ústrojí. Ozubené kolo je spojeno s výstupním hřídelem. Hřídele jsou uloženy v kuličkových ložiskách v převodové skříni. Na ložisku a převodové skříni jsou umístěny dva akcelerometry A_1 a A_2 . Vzorkovací frekvence akcelerometrů je 10 kHz. [41]



Obr. 49) Převodová skříň [41]

Pastorek se otáčí tempem $f_{\text{Pastorek}} = 20$ Hz. Otáčková frekvence ozubeného kola je dána rovnicí (22). [3]

$$f_{\text{ozub}} = f_{\text{Pastorek}} \cdot \frac{\text{Počet zubů pastorku } (N_p)}{\text{Počet zubů ozubeného kola } (N_{ok})} \quad (22)$$

Zubová frekvence odpovídá tempu, v jakém zabírají zuby ozubeného kola s pastorkem je dána rovnicí (23). [3]

$$f_{\text{Zubová}} = f_{\text{ozub}} \cdot N_{ok} = f_{\text{Pastorek}} \cdot N_p \quad (23)$$

Vstupní parametry:

```
tmax = 5           # Doba měření [s]
fs = 1000          # Vzorkovací frekvence [Hz]
f0 = 20            # Frekvence vstupní hřídele [Hz]
```

Základní parametry ozubení:

```
pinion = 13        # Počet zubů pastorku
gear = 35          # Počet zubů ozubeného kola

f_pin = f0         # Frekvence vstupní hřídele [Hz]
f_gear = f_pin * pinion / gear # Frekvence výstupní hřídele [Hz]
f_mesh = f_pin * pinion # Frekvence ozubení [Hz]
```

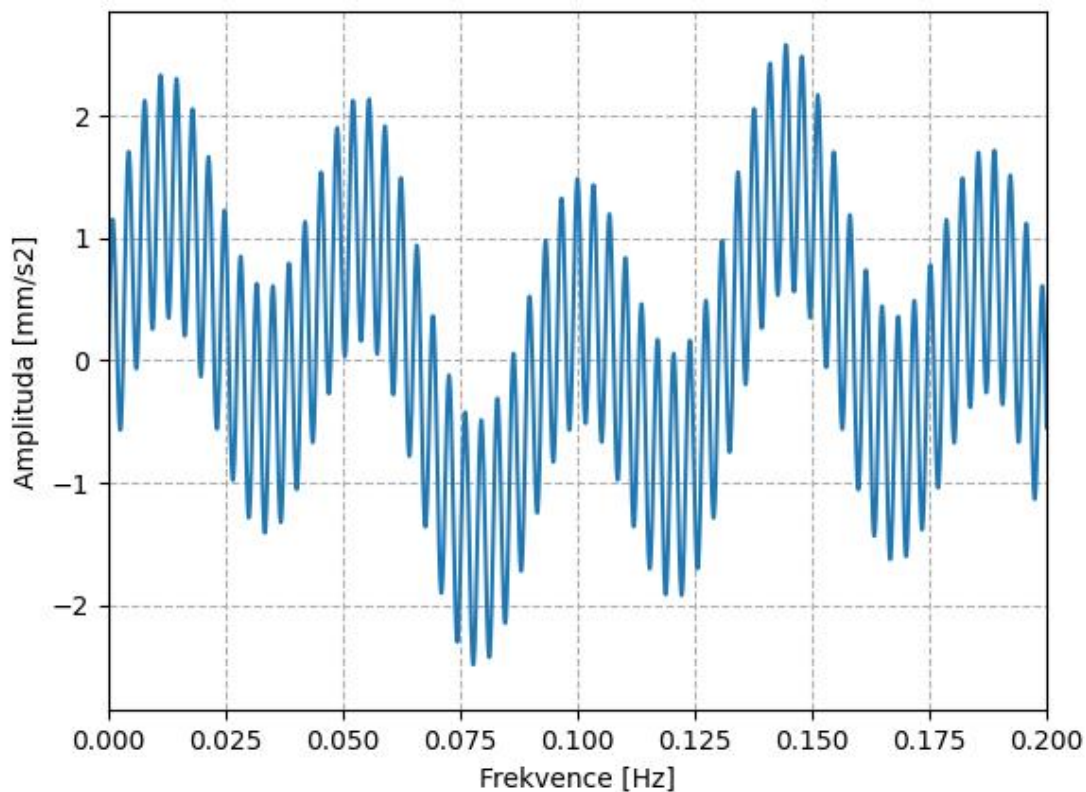
Nyní vygenerujeme vibrační signál pro pastorek, kolo a ozubení (Obr. 50). Vibrace vymodelujeme jako sinusoidy o frekvencích vstupní a výstupní hřídele a ozubení. Budeme analyzovat 5 sekund vibračních dat.

Ozubení je zodpovědné za přenos zatížení tudíž má jeho vibrační křivka nejvyšší amplitudu vibrací. Senzor A2 zaznamenává vibrace vstupní a výstupní hřídele a ozubení. V tomto příkladu budeme zanedbávat vibrace ložisek. Vibrační signál není zatížený šumem.

```
t = np.linspace(0.0, tmax, fs * tmax, endpoint=False)

wfIn = 0.6 * np.sin(2 * np.pi * f_pin * t) # Vibrační křivka pastorku
wfOut = 0.4 * np.sin(2 * np.pi * f_gear * t) # V. křivka ozubeného kola
wfMesh = 1.0 * np.sin(2 * np.pi * f_mesh * t) # V. křivka ozubení

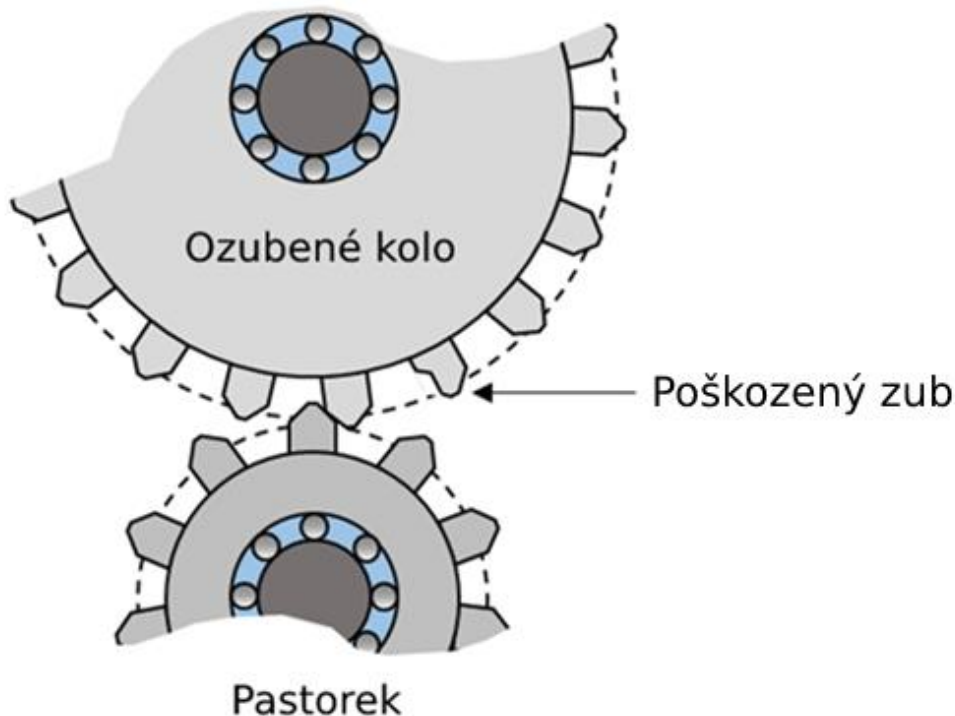
wfGearbox = wfIn + wfOut + wfMesh # Výsledná vibrační křivka
```



Obr. 50) Časový průběh vibrací ozubeného soukolí

Lokální závada – poškozený zub

Předpokládejme, že došlo k poškození jednoho zubu ozubeného kola lokální závadou, důvodem může být například pitting. To má za následek vysokofrekvenční náraz, který se vyskytuje jednou za otáčku ozubeného kola.



Obr. 51) Poškozený zub ozubeného kola [41]

Lokální závada způsobuje náraz, jehož trvání je kratší než doba záběru zubu. Důlek na povrchu zubu generuje po dobu nárazu vysokofrekvenční kmitání. Frekvence nárazu je závislá na fyzikálních vlastnostech ozubeného kola. V našem případě budeme předpokládat, že náraz způsobí vibrace o frekvenci 2 kHz po dobu přibližně 8 % $1/f_{\text{mesh}}$ což je přibližně 0.30 ms. Náraz se opakuje jednou za otáčku ozubeného kola. [41]

```

fImpact = 2000 # Vlastní frekvence kola
tImpact = np.arange(0, 3.0e-4 - 1 / fs, 1 / fs)
xImpact = np.sin(2 * np.pi * fImpact * tImpact)
    
```

Princip, jak vytvořit periodický signál je obdobný principu u ložiska. Výsledkem je opakující se náraz právě jednou za otáčku ozubeného kola (Obr. 52).

```

xComb = np.zeros(np.size(t))
Ind = int(np.around(0.3 * fs / f_mesh))
xComb[Ind::int(np.around(fs / f_gear))] = 1

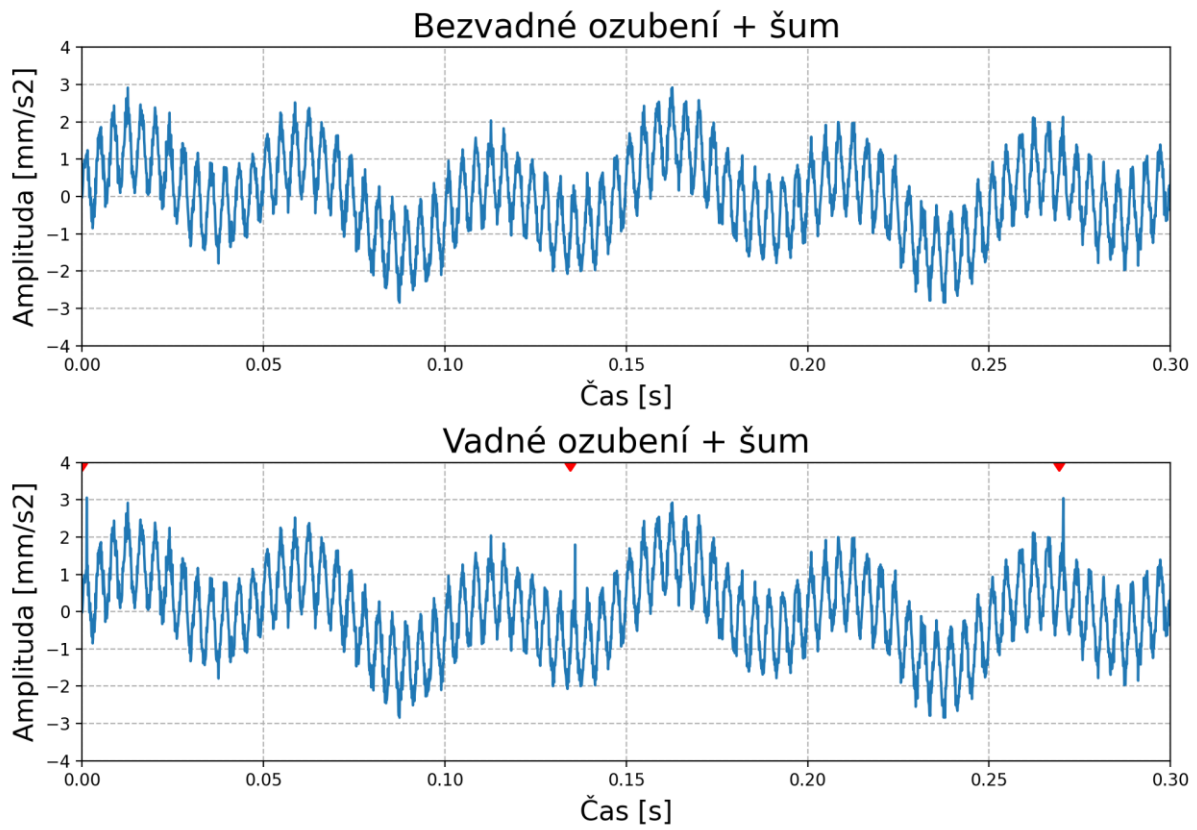
xBper = 2 * np.convolve(xComb, xImpact, "same")
    
```

Přidáme šum a jednotlivé křivky sečteme.

```

wFRand = 0.2 * np.random.standard_normal(np.size(t))

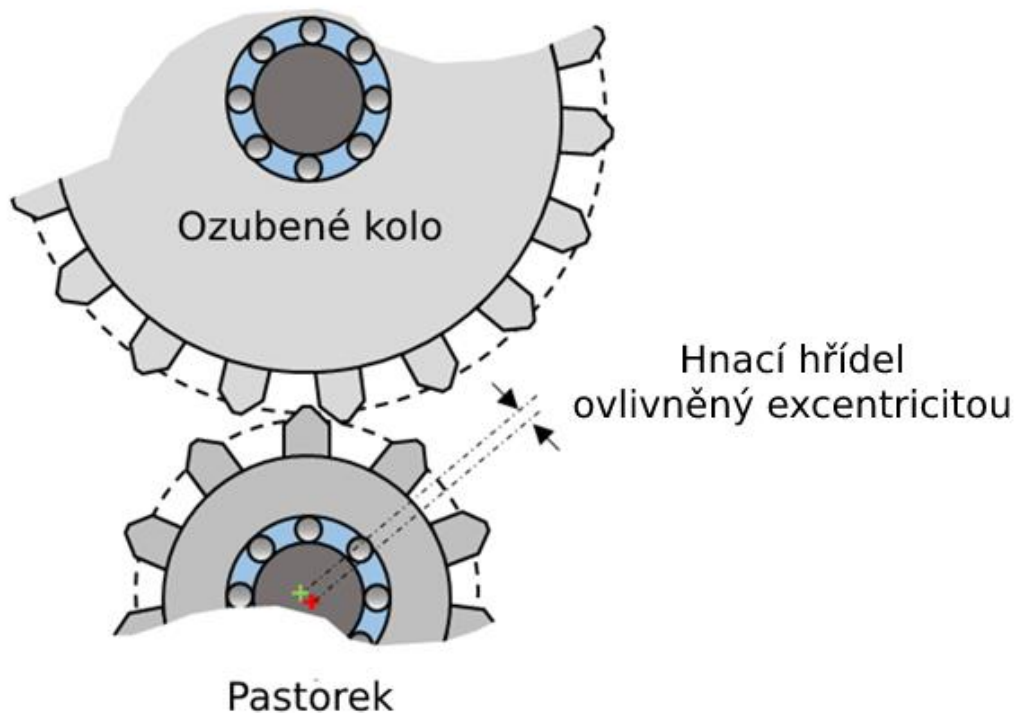
y_healthy = wfGearbox + wFRand
y_faulty = wfGearbox + wFRand + xBper
    
```



Obr. 52) Časový průběh vibrací ozubeného soukolí s lokální závadou

Distribuovaná závada

Ohnutý hřídel nebo excentricita ozubeného kola způsobuje modulaci zubové frekvence. Rozestupy postranních pásem jsou dány frekvencí defektního kola. V případě vážného problému pozorujeme harmonické násobky zubové frekvence a jejich postranní pásma (Obr. 55).



Obr. 53) Excentricita hnací hřídele [41]

Pro náš případ budeme uvažovat závažnou vadu. Nadefinujeme si první tři harmonické složky a jejich amplitudy.

```

wfHarmonicsSum = np.zeros(np.size(t))
harmonics = [1, 2, 3] # 3 Harmonické složky zubové frekvence
harmonics_amp = [1.2, 0.7, 0.6] # Amplitudy harmonických složek
# s dominantní 1X zubovou frekvencí
    
```

V dalším kroku vytvoříme tři složky postranního pásma s klesající amplitudou na každé straně zubové frekvence. Tento krok provedeme v cyklu *for* pro každou harmonickou složku.

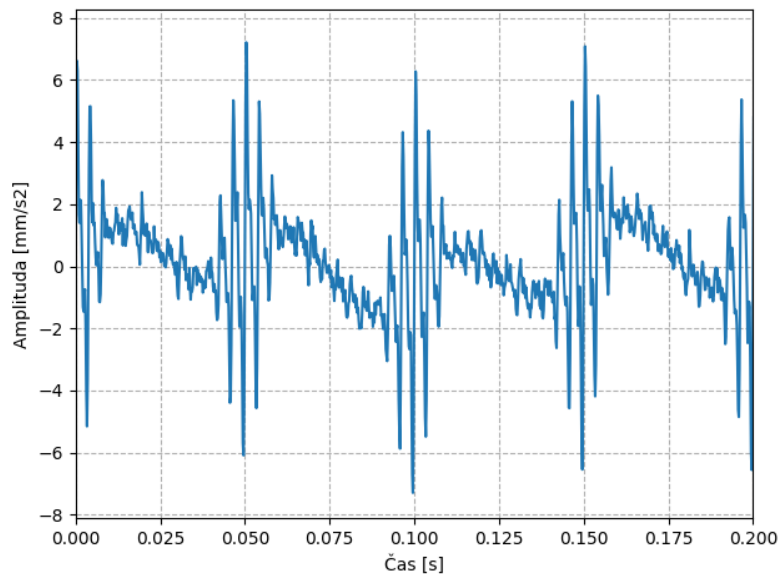
```

for h, ha in zip(harmonics, harmonics_amp):
    sideBands = np.arange(-3, 4)
    sideBandsAmp = ha * np.array([0.2, 0.4, 0.6, 1.0, 0.6, 0.4, 0.2])
# Amplitudy postranního pásma
    sideBandsFreq = h * f_mesh + sideBands * f_pin
# Frekvence postranního pásma

    wfSideBands = sideBandsAmp[:, None] * np.sin(2 * np.pi * t *
        sideBandsFreq[:, None])
    wfSideBandsSum = np.sum(wfSideBands, axis=0)
    wfHarmonicsSum += wfSideBandsSum
    
```

Sečteme křivky vstupní a výstupní hřídele s modulovaným signálem zubové frekvence, a nakonec přidáme šum (Obr. 54).

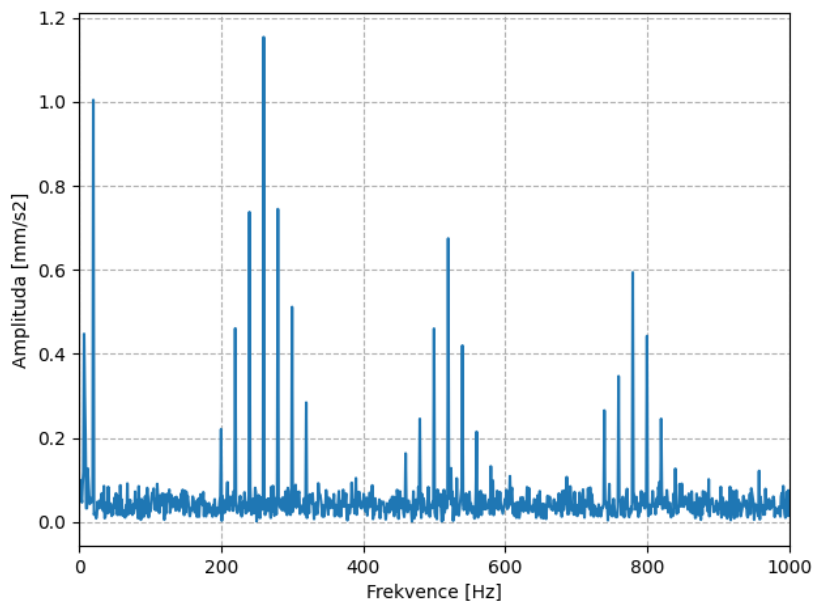
```
y = wfIn + wfOut + wfHarmonicsSum + wfRand
```



Obr. 54) Časový průběh vibrací ozubeného soukolí ovlivněného distribuovanou závadou

K další práci s daty je třeba provést frekvenční analýzu (Obr. 55). Knihovna NumPy obsahuje modul k výpočtu diskrétní Fourierovy transformace spolu s funkcí, která vypočítá odpovídající frekvence v Hz.

```
n = len(y)
xf = np.fft.fftfreq(n, 1 / fs)
yf = np.fft.fft(y)
yf_mag = np.abs(yf)
yf_mag = yf_mag * 2 / n
```



Obr. 55) FFT ozubeného soukolí ovlivněného distribuovanou vadou

4.6.3 Nevyváženost a nesouosost

V případě nevyváženosti a nesouososti je projevem poruchy součet diskrétních signálů ve tvaru sinusoidy jež jsou dány rovnicí (24), kde f_{nX} je harmonický násobek základní otáčkové frekvence hřídele a A_{nX} je jeho amplituda.

$$x_{nX}(t) = A_{nX} \cdot \sin(2 \cdot \pi \cdot f_{nX} \cdot t) \quad (24)$$

Výsledným projevem závady pak je suma těchto signálů (25).

$$x(t) = \sum_1^n A_{1X} \cdot \sin(2 \cdot \pi \cdot f_{1X} \cdot t) + A_{2X} \cdot \sin(2 \cdot \pi \cdot f_{2X} \cdot t) + \dots \quad (25)$$

$$\dots + A_{nX} \cdot \sin(2 \cdot \pi \cdot f_{nX} \cdot t)$$

Projev nevyváženosti a nesouososti se liší počtem harmonických frekvencí a jejich amplitudou. Tento příklad je ilustrován na závadě úhlová nesouosost. Její časový proběh a frekvenční spektrum je vidět na grafu (Obr. 56).

Jako v předešlých příkladech je nejprve nutné zadat vstupní parametry.

```

tmax = 5           # Doba měření [s]
fs = 1000         # Vzorkovací frekvence [Hz]
f0 = 20          # Frekvence vstupní hřídele [Hz]
    
```

Vytvoříme signál vstupního hřídele a šum.

```

t = np.linspace(0.0, tmax, int(fs * tmax))

fr_in = 1.0 * np.sin(2 * np.pi * f0 * t)           # Signál vstupního hřídele
fr_rand = 1.0 * np.random.standard_normal(np.size(t)) # Šum
    
```

Nadefinujeme počet harmonických násobků a jejich amplitudy a vytvoříme signál pro každý z nich. Tyto signály následně sečteme a vložíme do proměnné vHarmonicsSum.

```

harmonics = np.array([1, 2, 3]) # Harmonické násobky
harmonicsFreq = harmonics * f0 # Vypočte frekvence harmonických
                                # násobků
harmonicsAmp = np.array([random.uniform(0.05, 0.1),
                          random.uniform(1.2, 1.7),
                          random.uniform(0.3, 0.5)])
                                # Amplitudy harmonických násobků

vHarmonics = harmonicsAmp[:, None] * np.sin(2 * np.pi *
                                             t * harmonicsFreq[:, None])
                                # Vytvoří signál
                                # pro každou harmonickou frekvenci
vHarmonicsSum = np.sum(vHarmonics, axis=0)
    
```

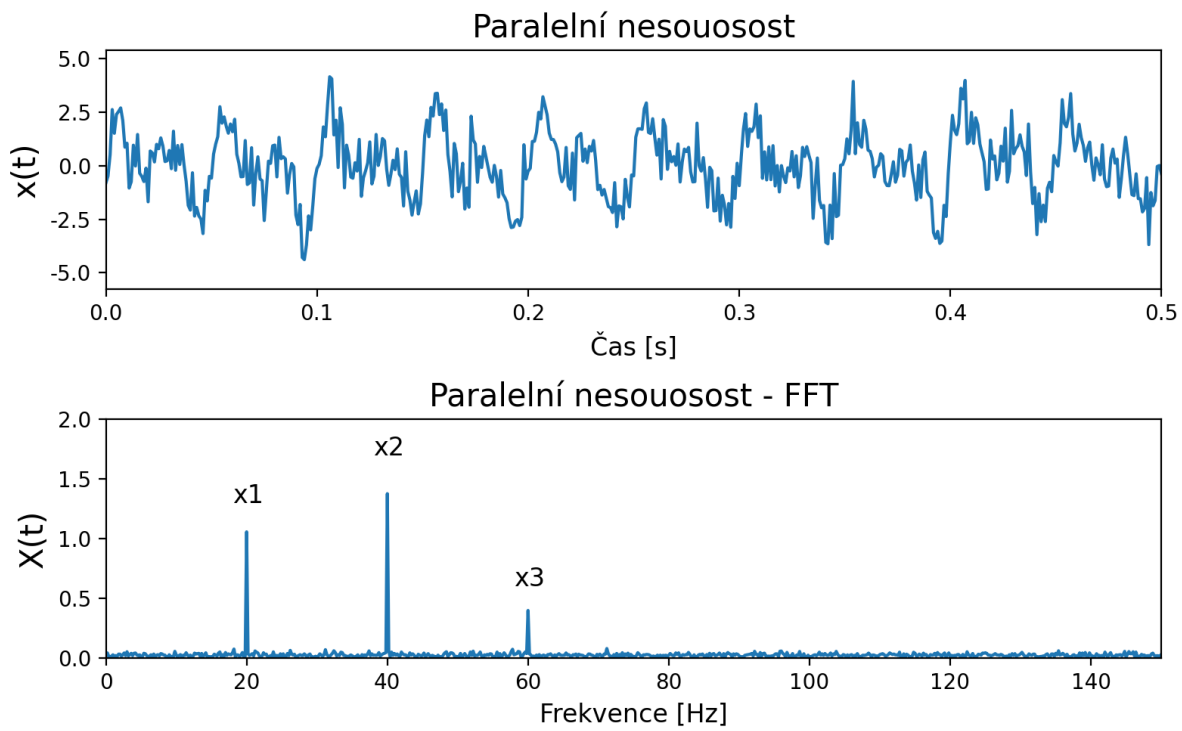
Sečteme signál vstupního hřídele, harmonické signály a šum.

```
y = vHarmonicsSum + fr_in + fr_rand
```

Vypočítáme FFT

```

n = len(y) # Délka signálu
xf = np.fft.rfftfreq(n, 1 / fs) # Vypočte frekvenční okna
yf = np.fft.rfft(y) # Vypočte FFT reálné části spektra
yf_mag = np.abs(yf) # Absolutní hodnota spektra
yf_mag = yf_mag * 2 / n # Normalizuje hodnoty
    
```



Obr. 56) Časový průběh a frekvenční spektrum paralelní nesouososti

5 ZHODNOCENÍ A DISKUZE

V praktické části této práce byl vytvořen algoritmus pro generování vibrodiagnostického signálu. Algoritmus byl následně zakomponován do aplikace s grafickým uživatelským rozhraním. K dispozici je také odlehčená verze ve formě skriptu, určená výhradně ke generování dat. Zmíněná aplikace slouží k simulaci signálů různých typů závad rotačních strojů. Program je koncipován primárně jako simulátor poruch rotačních zařízení s možností vizualizace generovaných dat v grafickém rozhraní.

Účelem programu je rozšířit možnosti výuky vibrodiagnostiky na VUT v Brně. Program může sloužit jako doplnění i náhrada měření vibrací na reálném zařízení vzhledem k tomu, že měřit více různých závad na reálném zařízení je ve školních podmínkách značně komplikované.

Uživatel si může zvolit libovolnou kombinaci z nabízených závad a vizualizovaná data následně exportovat do souboru a zacházet s nimi jako s daty naměřenými na reálném zařízení.

Program generuje signály závad nevyváženosti, nesouososti, jednotlivé druhy závad ložisek a distribuované a lokální závady ozubených kol, přičemž závady ozubených kol byly naprogramovány nad rámec požadovaných cílů. Závady ozubených kol byly přidány, protože ozubená kola jsou velmi častou součástí rotačních strojů a zároveň častým zdrojem poruch.

Program má možnost generovat dva různé typy signálů. První možnost generuje signál, který se snaží co nejdříve napodobit reálné chování poruchového stroje. Tento signál je pak vhodný pro pokročilou vibrační analýzu. Druhá možnost přehledně vizualizuje data v prostředí programu na úkor jejich podobnosti s realitou.

Při vytváření programu byly splněny všechny cíle dané touto diplomovou prací. Nicméně zde stále existuje místo pro zlepšení a rozvoj programu. Oblasti, na které se lze dále zaměřit je naprogramování dalších typů závad a zlepšení autenticity již naprogramovaných závad, použitím přesnějších matematických modelů.

6 ZÁVĚR

První cíl diplomové práce je vytvořit rešerši v oblasti vibrodiagnostiky. Tento cíl je zpracován v kapitole číslo tři. Jsou zde definovány základní veličiny důležité z hlediska vibrodiagnostiky a typy senzorů běžně používaných ve vibrodiagnostice. Následně se věnuje rozboru vibrodiagnostického singálu v časové a frekvenční oblasti, včetně příkladů různých typů vibrací a jejich zdrojů. V poslední části této kapitoly jsou rozebrány důvody vzniku, projevy a možnosti detekce nejvýznamnějších typů závad rotačních strojů.

Druhý cíl diplomové práce je popsat možnosti simulace mechanických soustav. Je zde popsán princip analýzy metodou konečných prvků z pohledu konstruktéra i z pohledu programu. Dále je zde vysvětleno, co je to konečný prvek ve vztahu k MKP a význam řešitele MKP. Vzhledem k tomu, že se praktická část diplomové práce nevěnuje simulaci reálných fyzikálních modelů, není zde uveden matematický princip metody konečných prvků. Na konci kapitoly jsou porovnány výhody a nevýhody několika vybraných programů pro simulaci kinematických soustav.

Třetím cílem diplomové práce je realizace programu pro generování simulovaných signálů z vibrodiagnostického zařízení. Program je napsán v programovacím jazyku Python a používá grafické rozhraní vytvořené v PyQt5. Program simuluje poruchy nevyvážení, nesouososti, závady ložisek a ozubených soukolí včetně libovolných kombinací. Program je vyhotoven ve dvou verzích, s grafickým uživatelským rozhraním a jako samostatný skript. Tento skript může sloužit například jako vstupní data pro strojové učení zaměřené na rozpoznání závad rotačních strojů. Primární využití programu je ale zejména k podpoře výuky vibrodiagnostiky.

Produkt není uzavřený systém. Jedná se totiž v první řadě o výukový modul určený k dalšímu rozvoji. Proto byl při návrhu programu kladen důraz na to, aby bylo možné program dále rozvíjet o nové funkcionality a další typy závad.

V příloze je k dispozici program pro operační systém Windows funkční bez instalace. K dispozici je také samostatný skript, který je určen zejména pro pokročilou analýzu generovaných dat nebo k automatizovanému generování většího množství dat. Tato verze je určena zejména pro pokročilé uživatele. V přílohách je rovněž uveden kompletní zdrojový kód a softwarová dokumentace.

7 SEZNAM POUŽITÝCH ZDROJŮ

- [1] BLATA, Jan a Janusz JURASZEK. *Metody technické diagnostiky: teorie a praxe = Metody diagnostyki technicznej : teorie a praktyka*. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2013. ISBN 978-80-248-2997-5.
- [2] STAMBOLISKA, Zhaklina, Eugeniusz RUSIŃSKI a Przemyslaw MOCZKO. *Proactive Condition Monitoring of LowSpeed Machines*. ISBN 978-3-319-10493-5.
- [3] HELEBRANT, František a Jiří ZIEGLER. *Technická diagnostika a spolehlivost*. Ostrava: VŠB - Technická univerzita, 2004. ISBN 80-248-0650-9.
- [4] Simple harmonic motion. *Boston University* [online]. [cit. 2021-05-13]. Dostupné z: <http://physics.bu.edu/~duffy/py105/SHM.html>
- [5] *Time equation of SMH* [online]. [cit. 2021-05-13]. Dostupné z: <https://www.toppr.com/ask/content/story/amp/time-equation-of-shm-45053/>
- [6] ZUTH, Daniel a František VDOLEČEK. Měření vibrací ve vibrodiagnostice. *Automa* [online]. 2010, 2010(1), 5 [cit. 2021-01-12]. Dostupné z: https://automa.cz/cz/casopis-clanky/mereni-vibraci-ve-vibrodiagnostice-2010_01_40375_5744/
- [7] BROCH, Jens Trampe. *Mechanical Vibration and Shock Measurements*. 2nd edition. Naerum: Brüel & Kjaer, 1984. ISBN 8787355361.
- [8] Industrial Piezo Velocity Vibration Sensor. *VibraSens* [online]. c2003-2021 [cit. 2021-01-14]. Dostupné z: <https://www.vibrasens.com/model-111-piezo-velocity-sensor-7-khz-bandwidth-top-exit/>
- [9] Měření vibrací pro diagnostiku opotřebených strojů. *Automatizace.HW.cz | Elektronika v automatizaci* [online]. 2015 [cit. 2021-01-12]. Dostupné z: <https://automatizace.hw.cz/mereni-a-regulace/mereni-vibraci-pro-diagnostiku-opotrebeni-stroju.html>
- [10] Ultimate guide to vibration analysis. *Design World* [online]. WTW Media, LLC, 2021 [cit. 2021-01-12]. Dostupné z: <https://www.designworldonline.com/the-ultimate-guide-to-vibration-analysis/>
- [11] FERNANDEZ, Alfonso. Industrial machinery vibration. *Power-MI* [online]. [cit. 2021-01-08]. Dostupné z: <https://power-mi.com/content/industrial-machinery-vibration>
- [12] YAGUO, Lei. *Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery* [online]. 1. United Kingdom: Elsevier, 2017 [cit. 2021-01-29]. ISBN 978-0-12-811534-3. Dostupné z: <https://www.sciencedirect.com/book/9780128115343/intelligent-fault-diagnosis-and-remaining-useful-life-prediction-of-rotating-machinery>

- [13] JANEČEK, Pavel a Elias TOMEH. *VIBRAČNÍ DIAGNOSTIKA ZÁKLADNÍCH ZÁVAD STROJŮ*. Liberec, 2010. Technická univerzita v Liberci.
- [14] BILOŠ, Jan a Alena BILOŠOVÁ. *APLIKOVANÝ MECHANIK JAKO SOUČÁST TÝMU KONSTRUKTÉRŮ A VÝVOJÁŘŮ: ČÁST VIBRAČNÍ DIAGNOSTIKA*. Ostrava, 2012. Vysoká škola báňská.
- [15] GRUNWALD, Benjamin. *Vibration Analysis of Shaft in SolidWorks and ANSYS*. Lappeenranta, Finsko, 2018. Saimaa University of Applied Sciences.
- [16] ZHU, Bofang. *The Finite Element Method: Fundamentals and Applications in Civil, Hydraulic, Mechanical and Aeronautical Engineering* [online]. 1st ed. Wiley, 2018 [cit. 2021-05-08]. ISBN 978-1119107316. Dostupné z: <https://www.wiley.com/en-us/The+Finite+Element+Method%3A+Fundamentals+and+Applications+in+Civil%2C+Hydraulic%2C+Mechanical+and+Aeronautical+Engineering-p-9781119107316>
- [17] BOERAEVE, Dr. Ir. P. *Introduction To The Finite Element Method (FEM)* [online]. Institut Gramme, 2010 [cit. 2021-05-08]. Dostupné z: https://kupdf.net/download/finite-element-method_58e5b978dc0d60f24ada97e9_pdf
- [18] Why do FEA engineers use 1D, 2D, 3D elements ?. *FEA for All* [online]. [cit. 2021-05-08]. Dostupné z: <http://feaforall.com/why-do-fea-engineers-use-1d-2d-3d-elements/>
- [19] RIEG, Frank, Reinhard HACKENSCHMIDT a Bettina ALBER-LAUKANT. *Finite Element Analysis for Engineers: Basics and Practical Applications*. Carl Hanser Verlag GmbH & Company KG, 2014. ISBN 9781569904886.
- [20] Finite Element Analysis Software Overview. *TEC* [online]. [cit. 2021-05-09]. Dostupné z: <https://www3.technologyevaluation.com/c/finite-element-analysis-fea>
- [21] MAGOMEDOV, Islam a Z. S. SEBAEVA. Comparative study of finite element analysis software packages. *Journal of Physics: Conference Series* [online]. 2020 [cit. 2021-05-09]. Dostupné z: doi:10.1088/1742-6596/1515/3/032073
- [22] *FEA Compare* [online]. [cit. 2021-05-09]. Dostupné z: <http://feacompare.com/>
- [23] Abaqus CAE. *3DS Design & Engineering Software* [online]. [cit. 2021-05-09]. Dostupné z: <https://www.3ds.com/products-services/simulia/products/abaqus/abaquscae/>
- [24] SolidWorks Simulation - simulace a ověřování návrhů. *SolidVision* [online]. [cit. 2021-05-09]. Dostupné z: <https://www.solidvision.cz/solidworks-simulation/>
- [25] What is Python? Executive Summary. *Python* [online]. [cit. 2021-01-08]. Dostupné z: <https://www.python.org/doc/essays/blurb/>
- [26] What is Python?. *Opensource.com* [online]. [cit. 2021-05-05]. Dostupné z: <https://opensource.com/resources/python>
- [27] WILLMAN, Joshua M. *Beginning PyQt: a hands-on approach to GUI programming*. [United States]: Apress, 2020. ISBN 978-1-4842-5856-9.

- [28] What is PyQt?. *Riverbank Computing* [online]. [cit. 2021-05-05]. Dostupné z: <https://riverbankcomputing.com/software/pyqt>
- [29] *What is PyCharm IDE?* [online]. [cit. 2021-05-06]. Dostupné z: <https://medium.com/@mindfiresolutions.usa/what-is-pycharm-ide-cc0735784f64>
- [30] *Python Libraries – Python Standard Library & List of Important Libraries* [online]. [cit. 2021-05-16]. Dostupné z: <https://data-flair.training/blogs/python-libraries/>
- [31] What is NumPy?: NumPy v1.20 - Manual. *NumPy* [online]. [cit. 2021-05-05]. Dostupné z: <https://numpy.org/doc/stable/user/whatisnumpy.html>
- [32] What Is Matplotlib In Python?. *ActiveState* [online]. [cit. 2021-05-05]. Dostupné z: <https://www.activestate.com/resources/quick-reads/what-is-matplotlib-in-python-how-to-use-it-for-plotting/>
- [33] What PyInstaller Does and How It Does It. *PyInstaller Manual* [online]. [cit. 2021-05-05]. Dostupné z: <https://pyinstaller.readthedocs.io/en/stable/operating-mode.html>
- [34] *Cgitb - Traceback manager for CGI scripts* [online]. [cit. 2021-05-05]. Dostupné z: <https://docs.python.org/3/library/cgitb.html>
- [35] DANIŠ, Stanislav. *Základy programování v prostředí Octave a Matlab* [online]. Praha: Matfyzpress, 2009 [cit. 2021-05-20]. ISBN 978-80-7378-082-1.
- [36] PyQt - Signals & Slots. *Tutorialspoint* [online]. [cit. 2021-05-16]. Dostupné z: https://www.tutorialspoint.com/pyqt/pyqt_signals_and_slots.htm
- [37] Multithreading PyQt applications with QThreadPool. *LearnPyQt* [online]. [cit. 2021-05-17]. Dostupné z: <https://www.mfitzp.com/tutorials/multithreading-pyqt-applications-qthreadpool/>
- [38] PECKA, Miroslav. *Regulární výrazy - Regexp* [online]. [cit. 2021-05-17]. Dostupné z: <https://www.regularnivyrazy.info/>
- [39] GRANEY, Brian P. a Ken STARRY. Rolling Element Bearing Analysis. *Materials Evaluation* [online]. The americal Society for Nondestrucite Testing, Inc., (70), 78-85 [cit. 2021-05-05]. Dostupné z: https://www.maintenance.org/fileSendAction/fcType/0/fcOid/399590942964610394/filePointer/399590942964867002/fodoid/399590942964867000/Rolling_Element_Bearing_Analysis_graney.pdf
- [40] KIM, Seokgoo, Dawn AN a Joo-Ho CHOI. Diagnostics 101: A Tutorial for Fault Diagnostics of Rolling Element Bearing Using Envelope Analysis in MATLAB. *Applied Sciences* [online]. 2020, **10**(20) [cit. 2021-05-03]. ISSN 2076-3417. Dostupné z: doi:10.3390/app10207302
- [41] Vibration Analysis of Rotating Machinery. *MathWorks* [online]. [cit. 2021-05-03]. Dostupné z: <https://www.mathworks.com/help/signal/ug/vibration-analysis-of-rotating-machinery.html?fbclid=IwAR3w-AjRYUItGiA34GdFRqMRs5ZnZXCeTHzUgHF-uZvOFXTcMVyuWeDGwI4>

8 SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK

8.1 Seznam tabulek

Tab 1) Regulární výraz pro pozitivní nebo negativní hodnotu s desetinným místem 60

8.2 Seznam obrázků

| | | |
|----------|--|----|
| Obr. 1) | Vztah mezi výchylkou, rychlostí a zrychlení s ohledem na čas [1] | 18 |
| Obr. 2) | Výběrová kritéria pro sledování dynamických účinků různých rozsahů rychlostí (frekvencí) [2] | 19 |
| Obr. 3) | Časová charakteristika vibrace s parametry [10]..... | 22 |
| Obr. 4) | Skládání jednoduchých vibrací v časové doméně [11] | 23 |
| Obr. 5) | Náhodná vibrace [11] | 23 |
| Obr. 6) | Přerušované rytmy [11] | 24 |
| Obr. 7) | Amplitudová modulace [11]..... | 24 |
| Obr. 8) | FFT zpracování komplexního vibračního signálu [11] | 25 |
| Obr. 9) | Křivka s frekvenčními složkami 22, 60 a 100 Hz s různou délkou vzorků a přidáním šumem pro ilustraci užitečnosti FFT analýzy [10]..... | 26 |
| Obr. 10) | Statická nevyváženost [1] | 28 |
| Obr. 11) | Momentová nevyváženost [1] | 29 |
| Obr. 12) | Statická nebo momentová nevyváženost ve frekvenčním spektru [1]..... | 29 |
| Obr. 13) | Dynamická nevyváženost [1] | 30 |
| Obr. 14) | Dynamická nevyváženost ve frekvenčním spektru [1]..... | 30 |
| Obr. 15) | Rovnoběžná nesouosost [1] | 31 |
| Obr. 16) | Rovnoběžná nesouosost v radiálním směru s vyznačením otáčkové složky ve frekvenčním spektru [1]..... | 31 |
| Obr. 17) | Úhlová nesouosost [1] | 32 |
| Obr. 18) | Úhlová nesouosost v axiálním směru s vyznačením otáčkové složky ve frekvenčním spektru [1]..... | 32 |
| Obr. 19) | Ohnutý hřídel [3] | 33 |
| Obr. 20) | Frekvenční spektrum ozubeného převodu [13] | 34 |
| Obr. 21) | Řemenový převod [3] | 34 |
| Obr. 22) | Geometrie ložiska [3] | 36 |
| Obr. 23) | Frekvence ložiskových závad a jejich poloha [14]..... | 37 |
| Obr. 24) | Trendování měřených vibrací valivých ložisek [1] | 37 |
| Obr. 25) | Vliv tlumení na velikost vibrací | 38 |

| | |
|---|----|
| Obr. 26) Originální 3D model hřídele (a), očištěný model hřídele pro simulaci (b) [15] | 40 |
| Obr. 27) Síť konečných prvků [15] | 40 |
| Obr. 28) (a) lokální souřadnicový systém, (b) globální souřadnicový systém, (c) Složená struktura [15] | 41 |
| Obr. 29) Druhy elementů [18] | 42 |
| Obr. 30) Logo programovacího jazyka Python [26] | 45 |
| Obr. 31) GUI návrhář Qt Designer | 46 |
| Obr. 32) Náhled na projekt v prostředí PyCharm | 47 |
| Obr. 33) Hlavní prvky grafického uživatelského rozhraní | 50 |
| Obr. 34) Panel vstupních parametrů | 51 |
| Obr. 35) Vyznačení úhlové nesouososti v grafu s legendou | 52 |
| Obr. 36) Porovnání signálu nezátíženého (a) a zatíženého (b) Gaussovským šumem | 52 |
| Obr. 37) (a) Reálný, (b) Ideální průběh | 53 |
| Obr. 38) Detail – Ideální průběh závady BPFÍ s vyznačením poruchových frekvencí | 53 |
| Obr. 39) Export dat do dokumentu | 54 |
| Obr. 40) Ovládací prvky grafu | 55 |
| Obr. 41) Pruh nástrojů grafu | 55 |
| Obr. 42) Ovládací panel vstupních parametrů po spuštění simulace | 56 |
| Obr. 43) FFT graf nepoškozeného zařízení | 57 |
| Obr. 44) Souborová struktura programu | 58 |
| Obr. 45) Diagram poškozeného ložiska a poruchový signál [41] | 62 |
| Obr. 46) Modulace simulovaného signálu závady: (a) Signál závady, (b) Rezonanční signál, (c) AM signál, v časové doméně | 63 |
| Obr. 47) Modulace signálu závady hnacím hřídelem: (a) Nemodulovaný signál závady, (b) Signál hnací hřídele, (c) Signál závady ložiska – BPFÍ, v časové doméně | 65 |
| Obr. 48) Simulace závady ložiska: (a) Diskrétní signál, (b) Bílý šum, (c) Surový signál, v časové doméně; (d) Diskrétní signál, (e) Bílý šum, (f) Surový signál, ve frekvenční doméně | 66 |
| Obr. 49) Převodová skříň [41] | 67 |
| Obr. 50) Časový průběh vibrací ozubeného soukolí | 68 |
| Obr. 51) Poškozený zub ozubeného kola [41] | 69 |
| Obr. 52) Časový průběh vibrací ozubeného soukolí s lokální závadou | 70 |
| Obr. 53) Excentricita hnací hřídele [41] | 71 |
| Obr. 54) Časový průběh vibrací ozubeného soukolí ovlivněného distribuovanou závadou | 72 |
| Obr. 55) FFT ozubeného soukolí ovlivněného distribuovanou vadou | 72 |
| Obr. 56) Časový průběh a frekvenční spektrum paralelní nesouososti | 74 |

8.3 Seznam použitých zkratk a symbolů

| | |
|------------------|---|
| y | Výchylka |
| $v; \dot{y}$ | Rychlost |
| $a; \ddot{y}$ | Zrychlení |
| A | Amplituda |
| f | Frekvence |
| t | Čas |
| π | Pí |
| \ddot{x} | Zrychlení objektu |
| m | Hmotnost seismické hmoty |
| M | Hmotnost objektu |
| k | Tuhost pružiny |
| b | Součinitel tlumení |
| F_b | Budící síla |
| AC | Alternating Current / Střídavý proud |
| DC | Direct Current / Stejnoseměrný proud |
| MEMS | Micro Electro Mechanical Systems / Mikro Elektro Mechanické Systémy |
| RMS | Root Mean Square / Efektivní hodnota |
| DFT | Discrete Fourier transform / Diskrétní Fourierova transformace |
| FFT | Fast Fourier transform / Rychlá Fourierova transformace |
| COS | Centrální osa setrvačnosti |
| OR | Osa rotace |
| f_{z12} | Zubová otáčková frekvence |
| f_{2Hz} | Druhá harmonická zubové frekvence |
| f_{3Hz} | Třetí harmonická zubové frekvence |
| z_1, z_2 | Počet zubů pastorku, kola |
| f_{R1}, f_{R2} | Rotorová frekvence pastorku, kola |
| f_z | Zubová frekvence |
| f_r, f_0 | Základní otáčková frekvence |
| f_B | Řemenová frekvence |
| f_{BZ} | Zubová frekvence ozubených řemenů |
| L | Délka řemenu |
| D_1, D_2 | Průměr pastorku, kola |
| HFD | High-frequency detection / Detekce vysokých frekvencí |
| BPFI | Ball Pass Frequency Inner / Frekvence poškození vnitřního kroužku |
| BPFO | Ball Pass Frequency Outer / Frekvence poškození vnějšího kroužku |

| | |
|----------------|---|
| BSF | Ball Spin Frequency / Frekvence poškození valivých elementů |
| FTF | Fundamental Train Frequency / Frekvence poškození klece |
| N | Počet valivých elementů |
| n | Otáčky rotoru (Hz) |
| B_d | Průměr valivého elementu |
| P_d | Roztečný průměr |
| φ | Kontaktní úhel |
| FEA | Finite Elements Analysis / Analýza metodou konečných prvků |
| MKP | Metoda konečných prvků |
| CAD | Počítačem podporované projektování (Computer-aided design) |
| CAE | Kompletní prostředí Abaqus (Complete Abaqus Environment) |
| IDE | Integrated Development Environment / Integrované vývojové prostředí |
| GUI | Grafické uživatelské rozhraní |
| API | Application Programming Interface |
| XML | Extensible Markup Language / Obecný značkovací jazyk |
| t' | Délka nárazu |
| α | Parametr tlumení ložiska |
| f_{Ozub} | Frekvence ozubení |
| $f_{Pastorek}$ | Otáčková frekvence pastorku |
| N_{ok} | Počet zubů ozubeného kola |
| N_p | Počet zubů pastorku |
| f_{nX} | Harmonický násobek základní otáčkové frekvence |
| A_{nX} | Amplituda harmonického násobku základní otáčkové frekvence |

9 SEZNAM ELEKTRONICKÝCH PŘÍLOH

1. Program Generátor signálů.exe
2. Skript main.py
3. Skript fault_methods.py
4. Skript mplwidget.py
5. Soubor gui.ui
6. Skript signal_generator.py
7. Seznam knihoven requirements.txt
8. Dokumentace