

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

ÚSTAV MECHANIKY, BIOMECHANIKY A MECHATRONIKY

Odbor mechaniky a mechatroniky



Diplomová práce

**Kalibrace robotického ramene s využitím
neuro-fuzzy modelu**

Praha, 2021

Jan Hladík

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Hladík** Jméno: **Jan** Osobní číslo: **465355**
Fakulta/ústav: **Fakulta strojní**
Zadávající katedra/ústav: **Ústav mechaniky, biomechaniky a mechatroniky**
Studijní program: **Aplikované vědy ve strojním inženýrství**
Specializace: **Mechatronika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Kalibrace robotického ramene s využitím neuro-fuzzy modelu

Název diplomové práce anglicky:

Pokyny pro vypracování:

- 1) Seznamte se s problematikou kalibrace robotických ramen.
- 2) Seznamte se s postupy vytváření neuro-fuzzy modelů.
- 3) Vybraný algoritmus pro neuro-fuzzy modelování naprogramujte v prostředí Matlab.
- 4) Proveďte kalibraci robotického ramene s využitím neuro-fuzzy modelu.

Seznam doporučené literatury:

- [1] Skopec, T.: Kalibrace paralelních mechanismů s adaptivní složitostí modelu, Ph.D. disertační práce, FS ČVUT v Praze, 2012.
- [2] Hamrle, V.: Kalibrovatelnost a její použití pro návrh paralelních kinematických struktur, Ph.D. disertační práce, FS ČVUT v Praze, 2009.
- [3] Nelles, O.: Nonlinear System Identification with Local Linear Neuro-Fuzzy Models, Ph.D. Thesis, Technische Universitat Darmstadt, 1998.
- [4] Potts, D. and C. Sammut.: Incremental Learning of Linear Model Trees, Machine Learning 61, 2005, pp. 5-48.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Petr Beneš, Ph.D., odbor mechaniky a mechatroniky FS

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **28.04.2021** Termín odevzdání diplomové práce: **13.08.2021**

Platnost zadání diplomové práce: _____

Ing. Petr Beneš, Ph.D.
podpis vedoucí(ho) práce

doc. Ing. Miroslav Španiel, CSc.
podpis vedoucí(ho) ústavu/katedry

prof. Ing. Michael Valášek, DrSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Anotační list

Jméno autora:	Jan Hladík
Název diplomové práce:	Kalibrace robotického ramene s využitím neuro-fuzzy modelu
Anglický název:	Calibration of the robotic arm using a neuro-fuzzy model
Akademický rok:	2020/2021
Obor studia:	Aplikované vědy ve strojním inženýrství – Mechatronika
Ústav/odbor:	Ústav mechaniky, biomechaniky a mechatroniky Odbor mechaniky a mechatroniky
Vedoucí diplomové práce:	Ing. Petr Beneš, Ph.D.
Bibliografické údaje:	Počet stran: 68 Počet obrázků: 52 Počet příloh: 1xCD
Klíčová slova:	Kalibrace, neuro-fuzzy modelování, lokální lineární modelové stromy, RD dělicí pravidlo
Keywords:	Calibration, neuro-fuzzy modelling, local linear model trees, RD splitting rule

Anotace:

Tato diplomová práce se zabývá problematikou použití neuro-fuzzy (NF) modelů a lokálních lineárních modelových stromů (LLMT) pro účel kalibrace a porovnáním těchto metod s tradičními kalibračními přístupy užívanými při kalibraci robotických ramen. Práce se primárně zaměřuje na modely dopředné kinematické úlohy a na dávkový indukční algoritmus využívající RD dělicí pravidlo.

Abstract:

This diploma thesis deals with the use of neuro-fuzzy (NF) models and local linear model trees (LLMT) in comparison with traditional calibration techniques for calibration of robotic arms. The work focuses primarily on a problem of forward kinematic models and a batch induction algorithm using the RD splitting rule.

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

V Praze, dne

.....
Podpis

Poděkování

Rád bych poděkoval mému vedoucímu diplomové práce Ing. Petru Benešovi, Ph.D za neobvykle vstřícný a inspirativní přístup při vedení této práce. Velké poděkování patří také mé rodině, především mým rodičům, kteří mně byli velkou, nejen finanční, oporou v dobách studia na Fakultě strojní ČVUT. Nemohu také opomenout poděkovat všem svým blízkým přátelům, kteří mě v tomto období obšťastňovali na mysli.

Obsah

Anotační list.....	3
Prohlášení.....	4
Poděkování.....	5
Obsah	6
Seznam obrázků	8
1. Úvod	10
2. Cíle práce.....	11
3. Přehled zkoumané problematiky	12
3.1. Průmyslové roboty	12
3.1.1. Klasifikace dle struktury	12
3.2. Měřicí zařízení	13
3.2.1. Laser tracker	14
3.3. Kalibrace.....	15
3.3.1. Základní algoritmus kalibrace.....	16
3.3.2. Kalibrace s uvažováním chyby čidel	17
3.3.3. Složitost a přesnost kalibrace	19
3.4. Neuro-fuzzy modelování	21
3.4.1. Umělé neuronové sítě	21
3.4.2. Fuzzy logika	23
3.4.3. Neuro-fuzzy sítě a modely.....	23
3.4.4. Lineární modelové stromy	25
4. Dávkový indukční algoritmus – Batch-RD	27
4.1. RD dělicí pravidlo.....	28
4.2. Pravidlo pro zastavení dělení	29
4.3. Algoritmus.....	29
4.3.1. Obecná struktura algoritmu	30
4.3.2. Konkrétní realizace algoritmu	30
4.3.3. Testování algoritmu.....	32
5. Modely	37
5.1. Rovinné modely.....	37
5.1.1. Nepoddajný kinematický model.....	37
5.1.2. Poddajný kinematický model	38
5.1.3. Kalibrovaný nepoddajný kinematický model	43
5.1.4. Nepoddajný kinematický model s korekcí realizovanou fuzzy modely.....	44
5.1.5. Poddajný kinematický model realizovaný pouze fuzzy modely	46
5.1.6. Porovnání modelů	48
5.2. Prostorové modely.....	54

5.2.1.	Kinematický popis sledující strukturu robotu	54
5.2.2.	Denavit – Hartenbergova konvence.....	56
6.	Měření na reálném šestiosém robotu.....	60
6.1.	Průmyslový robot Stäubli TX200	60
6.2.	Naměřená data.....	62
6.3.	Aproximace chyby	63
7.	Závěr.....	66
8.	Literatura.....	67

Seznam obrázků

Obr. 1 Struktura a) sériová b) neredundantní paralelní c) redundantní paralelní d) sérioparalelní	12
Obr. 2 Schéma laser trackeru převzato z [3]	14
Obr. 3 Model neuronu převzato z [4]	22
Obr. 4 Struktura vícevrstvé neuronové sítě převzato z [4]	22
Obr. 5 Struktura neuro-fuzzy sítě přejato z [7]	23
Obr. 6 Znárodnění stromové struktury převzato z [8]	25
Obr. 7 Dělení vzorků na podmnožiny převzato z [8]	27
Obr. 8 Struktura indukční funkce (pseudokód)	30
Obr. 9 Vstupní data testovací funkce 1	32
Obr. 10 Vykreslení výstupu algoritmu aproximace funkce 1	32
Obr. 11 Vykreslení bodů aproximace funkce 1 s vyhlazením.....	33
Obr. 12 Porovnání odmocniny kvadrátu chyby modelu bez a s vyhlazením	33
Obr. 13 Testovací funkce 2.....	34
Obr. 14 Vstupní data testovací funkce 2.....	34
Obr. 15 Ukázka rozdělení prostoru algoritmem pro testovací funkci 2.....	35
Obr. 16 Vykreslení výstupu algoritmu aproximace funkce 2	35
Obr. 17 Vykreslení bodů aproximace funkce 2 s vyhlazením.....	36
Obr. 18 Porovnání odmocniny kvadrátu chyby modelu bez a s vyhlazením	36
Obr. 19 Schéma rovinného mechanismu	37
Obr. 20 Polohy koncového bodu z nepoddajného modelu	38
Obr. 21 Uvolnění členu 2.....	39
Obr. 22 Uvolnění členu 3.....	39
Obr. 23 Namáhání vetknutého nosníku 2	41
Obr. 24 Namáhání vetknutého nosníku 3	41
Obr. 25 Polohy koncového bodu z poddajného a nepoddajného modelu	42
Obr. 26 Nahrazení ramene	43
Obr. 27 Vykreslení chyby v ose X a Y	44
Obr. 29 Vykreslení lineárních modelů aproximujících chybu v ose X (LLMT model).....	45
Obr. 28 Vykreslení lineárních modelů aproximujících chybu v ose Y (LLMT model).....	45
Obr. 30 Polohy koncového bodu v ose X.....	46
Obr. 31 Polohy koncového bodu v ose Y	46
Obr. 32 Aproximace pomocí LLMT modelu v ose X.....	47
Obr. 33 Aproximace pomocí LLMT modelu v ose Y.....	47
Obr. 34 Chyba nepoddajného kinematického modelu vůči poddajnému modelu	48
Obr. 35 Celková chyba kinematického modelu vůči poddajnému modelu.....	48
Obr. 36 Chyba kalibrovaného kinematického modelu vůči poddajnému modelu.....	49
Obr. 37 Celková chyba kalibrovaného kinematického modelu vůči poddajnému modelu	49
Obr. 38 Chyba nepoddajného modelu s korekcí realizovanou fuzzy modely vůči poddajnému modelu.....	50
Obr. 39 Celková chyba nepoddajného modelu s korekcí realizovanou fuzzy modely vůči poddajnému modelu.....	50
Obr. 40 Celková chyba modelu LLMT s a bez vyhlazení vůči poddajnému modelu	51
Obr. 41 Chyba modelu LLMT s vyhlazením vůči poddajnému modelu	51
Obr. 42 Chyba modelu LLMT vůči poddajnému modelu	51

Obr. 43 Ukázka připojení souřadnicových systémů ke kloubovému manipulátoru (převzato z [11]).....	55
Obr. 44 Souřadnicové systémy splňující podmínky DH 1 a DH 2 (převzato z [11]).....	57
Obr. 45 Ukázka robotu ze série Stäubli TX200 [13].....	60
Obr. 46 Okótované schéma Stäubli TX200 [13].....	61
Obr. 47 Polohy koncového bodu robotu (robot Stäubli TX200).....	62
Obr. 48 Škálované chyby koncového bodu (robot Stäubli TX200).....	62
Obr. 49 Aproximace chyby v ose X (robot Stäubli TX200).....	63
Obr. 50 Aproximace chyby v ose Y (robot Stäubli TX200).....	64
Obr. 51 Aproximace chyby v ose Z (robot Stäubli TX200).....	64
Obr. 52 Srovnání celkové chyby s a bez korekce LLMT modely (robot Stäubli TX200).....	65

1. Úvod

S narůstajícím počtem lidí obývajících planetu Zemi se stále zvyšuje tlak na objem produkce výrobků. Pro uspokojení poptávky se stává, nejen z časových důvodů, nemožné výrobky vyrábět manuálně. Proto se čím dál tím více výrobků začíná vyrábět ve velkosériové produkci, kde je možné proces výroby maximálně optimalizovat a tím snížit časovou náročnost a cenu výroby. Z toho také vyplývá, že lidský prvek z výroby postupně začal být vytlačován zprvu mechanizací a následně automatizací a robotizací. Automatizace má nesporné množství výhod, jako je nárůst rychlosti, přesnosti a výkonu.

Aby bylo možné stavět různorodé výrobní linky efektivním způsobem, je vhodné, aby se skládaly z poměrně univerzálních členů, které mají rozmanité použití. Takovým vhodným prvkem se ukázaly programovatelné průmyslové roboty, které se v posledních desetiletích začaly hojně užívat. Není s podivem, že s jejich rozšířením se také začaly zvyšovat požadavky na jejich přesnost. Aby byly tyto požadavky splněny, je snahou výrobců stále zlepšovat hardware i software těchto robotů. Nejběžnějším používaným typem průmyslových robotů jsou šestiosé roboty se sériovou strukturou (robotická ramena), jejichž výhodou je relativní jednoduchost, univerzálnost, snadné ovládání a řízení. Jejich nevýhodou je poměrně nízká tuhost i při poměrně robustní konstrukci a s ní spojené vysoké hmotnosti.

Při snaze o přesné polohování koncového členu robotu (end efektoru) se obvykle vychází z kinematického modelu robotu, jehož parametry se snažíme pomocí tradičních kalibračních metod určit (identifikovat). Tyto metody ovšem obvykle v sobě přímo nezahrnují vlivy, jako je poddajnost a jiné skryté vlivy (například teplotní roztažnost atd.) a z nich vycházející chybu závislou na konkrétní konfiguraci robotu v prostoru. Odtud vychází myšlenka vyvinutí kalibračních metod, které by kinematický model rozšířily o korekci polohy zahrnující výše zmíněné vlivy. Výhodou takových metod by bylo, že by v synergii kombinovaly již poměrně přesný kinematický model, využívající znalosti geometrie robotu, s poměrně nepřesnou aproximací chybové funkce závislé na konkrétní konfiguraci robotu.

Vyvstává otázka, jakým vhodným způsobem aproximovat chybovou funkci nad kinematickým modelem. Velmi diskutovanými tématy poslední doby jsou strojové učení, umělé neuronové sítě a neuro-fuzzy sítě. Z této množiny témat mě během magisterského studia na Fakultě strojní ČVUT nejvíce zaujala problematika neuro-fuzzy sítí a generování lokálních lineárních modelových stromů, která se navíc jevila jako vhodná volba k aproximaci. Tudíž jsem se rozhodl tuto diplomovou práci zaměřit na posouzení, zda je opravdu možné těmito algoritmy generovat modely, které by byly schopné aproximovat chybovou funkci, a na následné porovnání tradiční kinematické kalibrační metody a metody kombinující kinematický model s aproximací chybové funkce za použití neuro-fuzzy modelů, popř. lokálních lineárních modelových stromů.

2. Cíle práce

Cílem této práce je prozkoumat využitelnost neuro-fuzzy modelů v rámci kinematické kalibrace. Konkrétně to znamená zvolit vhodný algoritmus generující neuro-fuzzy modely a realizovat je v programovacím jazyku Matlab, dále provést porovnání tradičně užívaných kalibračních metod s metodami využívajícími neuro-fuzzy modely, ověřit funkčnost kalibrační metody využívající aproximaci chyby za pomoci neuro-fuzzy modelů, a to jak na simulovaném mechanismu, tak na datech z reálného měření na robotickém ramenu.

Cíle předmětné práce jsou shrnuty v následujících bodech:

1. Seznámit se s problematikou kalibrace robotických ramen.
2. Seznámit se s postupy vytváření neuro-fuzzy modelů.
3. Vybrat algoritmus vhodný ke generování neuro-fuzzy modelů chyby pro účely kalibrace.
4. Realizovat zvolený algoritmus v programovacím jazyce Matlab.
5. Zvolit vhodný mechanismus, na kterém bude provedeno porovnání různých přístupů kalibrace, a vytvořit jeho simulační model.
6. Provést porovnání základní kalibrace a přístupů využívajících neuro-fuzzy modely na simulačním modelu mechanismu.
7. Ověřit funkčnost a správnost použití neuro-fuzzy modelů k aproximaci chyby na simulačním modelu mechanismu.
8. Ověřit funkčnost a správnost použití neuro-fuzzy modelů k aproximaci chyby na datech získaných z měření reálného robotického ramena.

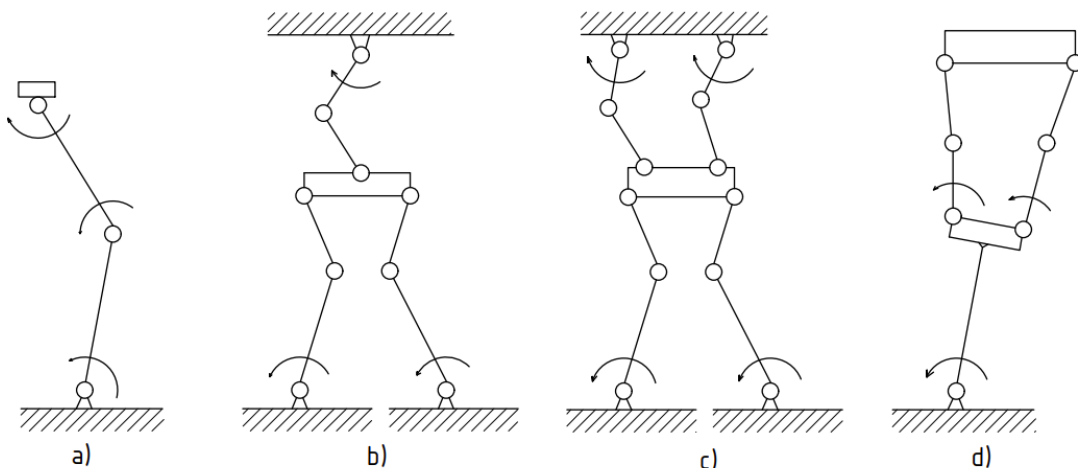
3. Přehled zkoumané problematiky

3.1. Průmyslové roboty

Nelze na úvod nepřipomenout, že téměř celosvětově užívaný pojem „robot“ (ve smyslu nahrazení lidské činnosti) má svůj základ v českém jazyku. Toto slovo bylo poprvé užito v divadelní hře Karla Čapka R.U.R. z roku 1920. Sám Karel Čapek ovšem nebyl autorem tohoto slova, tím byl jeho mladší bratr Josef Čapek. Toto slovo se rychle rozšířilo a začalo být běžně užíváno laickou i odbornou veřejností k pojmenování automatických a mechanických strojů. V současné době z pohledu průmyslu toto slovo označuje automatizované stroje nahrazující manuální práci lidí. V oboru robotika ovšem přesnou všeobecně uznávanou definici toho, co to robot je, nenajdeme. Podle normy ISO 8373:2012 [1] je průmyslový robot definován jako (přeloženo z angličtiny): „Průmyslový robot je automaticky řízený, opakovaně programovatelný, víceúčelový manipulátor pro použití v aplikaci průmyslové automatizace, programovatelný ve třech či více osách, které mohou být pevné či pohyblivé v prostoru. Průmyslový robot zahrnuje samotný manipulátor včetně svých pohonů, řídicí systém včetně programovacího panelu a komunikačního rozhraní (hardware a software).“

3.1.1. Klasifikace dle struktury

Možností, jak klasifikovat průmyslové roboty a manipulátory, je vícero. Zde bude uvedeno rozdělení dle kinematické struktury (uspořádání kinematických dvojic pohonů v mechanismu), jelikož toto rozdělení je nejdůležitější z pohledu kalibrace. Dle [2] je možné mechanismy robotů rozdělit do tří kategorií: *sériové mechanismy*, *neredundantní paralelní mechanismy*, *redundantní paralelní mechanismy*. Navíc je vhodné zmínit i kategorii kombinující sériovou strukturu s paralelní (*sérioparalelní mechanismy*).



Obr. 1 Struktura a) sériová b) neredundantní paralelní c) redundantní paralelní d) sérioparalelní

Sériové mechanismy (viz Obr. 1a) mají otevřenou kinematickou smyčku. Důsledkem sériové struktury (řazením kinematických dvojic za sebe) je postupné nasčítávání chyb jednotlivých kinematických dvojic a průhybů těles (mechanismus je poddajný jako jeho nejpoddajnější člen),

kteře vedou k nepřesnému polohování koncového členu. Při snaze dosáhnout lepší tuhosti tyto mechanismy nabývají na rozměrech a hmotnosti, což pak negativním způsobem ovlivňuje jejich dynamické schopnosti. Naopak výhody mechanismů využívajících tyto struktury jsou: relativní jednoduchost, snadná ovladatelnost a dobrý poměr zástavbového ku pracovnímu prostoru. U těchto mechanismů obvykle počet stupňů volnosti odpovídá počtu pohonů a senzorů mechanismu. Nejčastěji užívanou variantou je šestiosý angulární robot (robotické rameno). Na tento typ robotu je předmětná práce primárně zaměřena.

Neredundantní paralelní mechanismy (viz Obr. 1b) se skládají z více kinematických smyček, které procházejí jednotlivými částmi mechanismu a spojují rám s platformou. Paralelní mechanismy mají obecně lepší poměr tuhosti ku hmotnosti, z čehož vyplívá, že mají výrazně lepší dynamické vlastnosti než sériové mechanismy. Také díky paralelní struktuře dosahují lepší přesnosti polohování. Nevýhody těchto mechanismů jsou horší poměr zástavbového ku pracovnímu prostoru, složitější ovládání (nelineární přenosové vztahy akčních členů), možný vznik kolizí vzpěr, vznik singulárních poloh (zabraňující hladkému pohybu mechanismu). Neredundantní paralelní mechanismy mají stejně stupňů volnosti jako pohonů a senzorů.

Redundantní paralelní mechanismy (viz Obr. 1c) vznikají rozšířením neredundantních paralelních mechanismů minimálně o jednu další kinematickou smyčku spojující platformu s rámem. Redundantní paralelní mechanismy mají méně stupňů volnosti než pohonů a senzorů. Tato přeúčtená struktura má za nevýhodu ještě složitější řízení oproti neredundantní paralelní struktuře, při kterém musí být ošetřené riziko „přetahování“ pohonů. Na druhou stranu přejímají pozitivní vlastnosti paralelní struktury, potlačují problém singulárních poloh a umožňují řídit napětí ve struktuře k vymezení vůlí v kloubech a pohonech. Další výhodou je, že tyto mechanismy jsou schopny využít senzoru v nadbytečných pohonech k redundantní kalibraci (kalibrační metoda uzavřené smyčky).

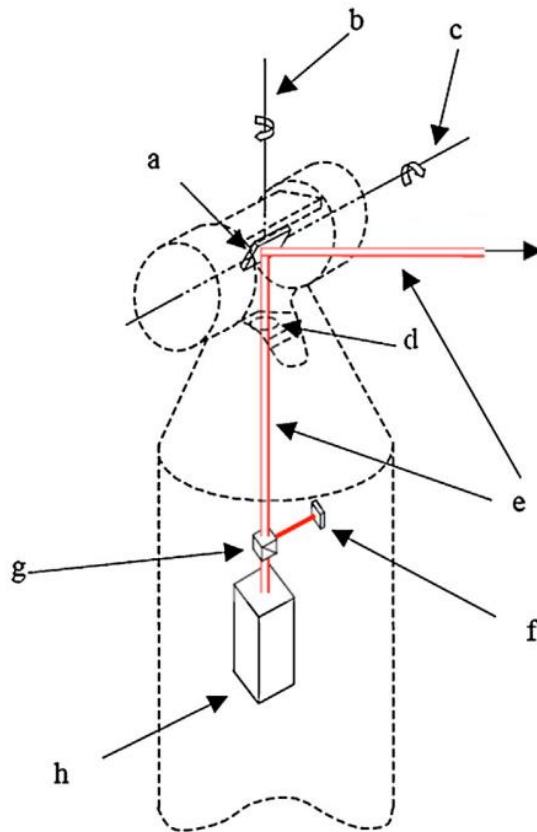
Sérioparalelní mechanismy (viz Obr. 1d) se snaží vhodným způsobem kombinovat sériovou a paralelní (nejčastěji neredundantní) strukturu a tím dosáhnout pozitivních vlastností obou struktur.

3.2. Měřicí zařízení

Pro potřeby co nejpřesnějšího polohování mechanismu je nezbytné jeho přesné řízení a kalibrace. Aby bylo možné kalibrovat mechanismy se sériovou strukturou (myšleno metodami otevřené smyčky), je třeba mechanismus propojit s přesným externím měřicím zařízením, které poskytne informaci o poloze koncového bodu (end efektoru) a tím umožní uzavřít kinematickou smyčku. Toto měřicí zařízení musí být schopné poskytnout informaci o poloze koncového bodu minimálně o řád přesněji, než má být výsledná přesnost mechanismu po kalibraci. Jednou z možností je najíždět robotem na dotyková čidla přesně umístěná v prostoru, nebo ke koncovému bodu robotu připojit dobře známou kalibrovanou sériovou strukturu se senzory umístěnými v kloubech, přes kterou je uzavřena kinematická smyčka. Tyto metody jsou ale často poměrně nepřesné či nerobustní. Jako velmi vhodnou volbou se jeví použití měřidel založených na optické bázi.

3.2.1. Laser tracker

Zařízení laser tracker bylo vynalezeno v polovině 80. let - Dr. Lau a kolektiv, Národní institut standardů a technologií (NIST, USA). Laser trackery jsou přenosné souřadnicové měřicí systémy, které měří za pomoci laserového paprsku polohu kooperujícího objektu (nejčastěji sféricky uchyceného koutového odražeče) v prostoru. Přístroj samotný zaznamenává vzdálenost k objektu společně s dvěma úhly, které poskytují informaci o poloze ve sférickém souřadnicovém systému spjatém s měřicím zařízením.



Obr. 2 Schéma laser trackeru převzato z [3]

Laser tracker se skládá z mechanických a optických částí schematicky zobrazených na Obr. 2. Zařízení má dvě říditelné rotační osy (b – vertikální, c – horizontální), které jsou ideálně kolmé na sebe a protínají se v jednom bodě, který definuje počátek sférického souřadnicového systému spjatého s měřicím zařízením. V tomto bodě se také odráží laserový paprsek (e) od zrcátka (a) otočného kolem osy c. Osy b a c jsou opatřeny enkodéry, které slouží k odečítání úhlů (na obrázku nejsou zobrazeny). Prvek h reprezentuje interferometr a zdroj laserového paprsku, který dále prochází do dělícího členu g. Část paprsku vycházející ze zdroje je v systému zachována jako referenční paprsek pro interferometrický výpočet. Měřicí část paprsku se odráží od zrcátka (a), zasáhne koutový odražeč a navrátí se do laser trackeru. Část tohoto zpětného paprsku je odkloněna do PSD detektoru (f), kde je vyhodnocena odchylka vystředění paprsku, ze které je vypočteno natočení os b, c tak, aby další paprsek byl vystředěný. Zbývá část zpětného paprsku odchází do interferometru, kde je systémem vyhodnocena vzdálenost k objektu.

Přesnost měření polohy laser trackeru je obvykle v řádu 0,025 mm. Největší nepřesnost do měření vnáší chyba z odečítání úhlů. Proto je pro zvýšení přesnosti někdy využíváno dvou a více laser trackerů rozmístěných v prostoru společně mířících na jeden koutový odražeč. Rozbor zdrojů nepřesností měření a metody využití více laser trackerů v kooperaci jsou velmi detailně popsány v odkaze [3].

3.3. Kalibrace

Problematika kalibrace by se obecně dala zahrnout jako podoblast identifikace systémů. Hlavním cílem kalibrace je zpřesnění (optimalizace) parametrů matematického modelu, aby co nejlépe popisoval chování systému reálného světa. Což poté přirozeně vede i k případnému zpřesnění řízení daného systému.

Kalibrační metody se obecně skládají z následujících kroků (viz [4]):

1. Sestavení kalibračního modelu
2. Měření v různých polohách mechanismu
3. Identifikace chyb parametrů modelu na základě naměřených dat
4. Kompenzace parametrů pomocí identifikovaných chyb
5. Použití zpřesněných parametrů pro řízení systému, zlepšení přesnosti

U mechanických systémů, jako jsou robotická ramena, je základním přístupem kalibrace (metody s otevřenou smyčkou) polohování koncového end-efektoru, odečítání jeho polohy za pomoci externího měřicího zařízení (např. Laser tracker) a současné zaznamenávání polohy pohonů skrze jejich vnitřní čidla. Pro takové systémy platí, že jejich počet stupňů volnosti odpovídá počtu pohonů, potažmo počtu vnitřních čidel.

Kalibrační metody mechanismů lze podle [4] rozdělit do následujících tří úrovní. Kalibrační metody prvního stupně se snaží najít vztah mezi reálnou polohou kinematické vazby (nejčastěji kloubu) a signálem, který je generovaný čidlem odpovídající vazby. Kalibrační metody druhého stupně jsou definovány jako celková kinematická kalibrace mechanismu. Tím rozumíme, že jsou kalibrovány některé nebo všechny rozměry mechanismu. Polohové a úhlové nepřesnosti jsou zahrnuty jako chybové parametry návrhových rozměrů členů mechanismu. Kinematické dvojce jsou uvažovány jako nižší kinematické dvojce s neznámou polohou v prostoru, která je určena během kalibrace. Většina dnes používaných postupů spadá právě do této kategorie. Poslední skupinou jsou kalibrační metody třetího stupně, které berou v úvahu i negeometrické zdroje nepřesností, jako jsou poddajnosti jednotlivých členů, vlivy tření a dynamických parametrů. Tyto metody bývají obvykle kombinovány s předešlými úrovněmi.

Další možné rozdělení kalibračních metod je na metody s otevřenou a uzavřenou smyčkou. K tradičnějšímu přístupu patří používání metod s otevřenou smyčkou, kdy je poloha pracovního bodu určena za pomoci externího měřicího zařízení. Méně časté jsou metody s uzavřenou smyčkou, které nepracují s externím měřicím zařízením, ale využívají přímo informace z nadbytečného množství senzorů umístěných v kalibrovaném systému (označováno, jako samo-kalibrace), či u redundantních paralelních mechanismů, kde není potřeba využívat nadbytečné senzory, ale stačí informace ze senzorů v pohonech mechanismu (označováno, jako redundantní kalibrace).

3.3.1. Základní algoritmus kalibrace

Pojem základní kalibrace a informace o ní je zpracován podle [5], kde se pod tímto pojmem rozumí algoritmus umožňující kalibraci parametrů kinematické struktury bez uvažování chyb čidel. Zkoumané kinematické struktury v sobě obsahují kinematické smyčky (uzavřené cesty v přidruženém grafu kinematického modelu). Tyto smyčky jsou popsány vazbovými podmínkami pro danou kalibrační pozici mechanismu:

$$f(\mathbf{d}, \mathbf{s}, \mathbf{v}) = \mathbf{0} \quad (3.1)$$

kde \mathbf{d} reprezentuje skutečné rozměry mechanismu, \mathbf{s} měřené hodnoty na vnitřních čidlech mechanismu a \mathbf{v} polohy end-efektoru získané z externího měřicího zařízení (tyto hodnoty považujeme za přesné). Pro n naměřených vzorků pak můžeme rovnice sdružit do tvaru:

$$F(\mathbf{d}, \mathbf{S}, \mathbf{V}) = \mathbf{0} \quad (3.2)$$

$$\mathbf{F} = [f_1, f_2, \dots, f_n]^T; \quad \mathbf{S} = [s_1, s_2, \dots, s_n]^T; \quad \mathbf{V} = [v_1, v_2, \dots, v_n]^T \quad (3.3)$$

Samotný algoritmus kalibrace je založen na modifikované Newtonově metodě pro přeúřčený systém nelineárních algebraických rovnic, kde rozměry mechanismu \mathbf{d} jsou stejné pro všechny kalibrační pozice mechanismu. Rozměry z výkresové dokumentace neodpovídající skutečným rozměrům označme jako $\bar{\mathbf{d}}$ a použijí se jako výchozí odhad pro nultou iteraci. Z Taylorova rozvoje:

$$F(\bar{\mathbf{d}}, \mathbf{S}, \mathbf{V}) + J_d \partial \mathbf{d} + \dots = \mathbf{0} \quad (3.4)$$

kde J_d je Jacobiho matice parciálních derivací kinematických vazeb (3.2) podle kalibrovaných rozměrů \mathbf{d} . Poté rovnici (3.4) můžeme přepsat do tvaru:

$$-F(\bar{\mathbf{d}}, \mathbf{S}, \mathbf{V}) = J_d \partial \mathbf{d} = \partial \mathbf{r} \quad (3.5)$$

a pro i -tý krok Newtonovy metody platí:

$$\partial \mathbf{d}_i = (J_{di}^T J_{di})^{-1} J_{di}^T \partial \mathbf{r}_i \quad (3.6)$$

kde $\partial \mathbf{r}_i$ je vektor odchylek vypočtený z měřených a kalibrovaných veličin z předchozího kroku. Nové hodnoty kalibrovaných veličin se vyjádří jako:

$$\mathbf{d}_{i+1} = \mathbf{d}_i + \partial \mathbf{d}_i \quad (3.7)$$

V tomto iteračním procesu se dále pokračuje, dokud není dosažena požadovaná přesnost. Tento postup nám poskytuje pro zadaná data jedinečné řešení, které je typicky stejné pro širokou oblast odhadů počátečních parametrů.

3.3.2. Kalibrace s uvažováním chyby čidel

Při kalibraci reálného robotu se do celého procesu kromě chyb vzniklých v mechanických částích projevují také chyby generované nepřesným měřením veličin na čidlech. Ve snaze zvýšit přesnost robota je poté vhodné zvolit metodu, která zachybování kalibračních dat bere v potaz. Jednou z takových metod je například metoda popsaná v [6], která předpokládá, že je možné kinematický model robota zapsat, jako:

$$\mathbf{f}(\mathbf{x}, \mathbf{p}) = \mathbf{0} \quad \mathbf{f}: R^k \times R^n \rightarrow R^m, \quad (3.8)$$

kde $\mathbf{x} \in R^k$ je vektor proměnných spjatých s pohybem a $\mathbf{p} \in R^n$ je vektor kalibrovaných parametrů. Vektor \mathbf{x} obsahuje chyby způsobené odchylkami kloubů, end-efektoru, vůlemi a dalšími neznámými vlivy. Aby bylo možné zaručit, že kinematické smyčky budou vždy uzavřené, vyžaduje se splnění podmínek: $k \geq m$ a $h(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}) = m$. Pod zápisem $h()$ se rozumí hodnota matice.

V i -té kalibrační poloze odečteme naměřený vektor \mathbf{x}_i , který uvažujeme, že se skládá z přesných hodnot $\bar{\mathbf{x}}_i$ a chyb měření $\hat{\mathbf{x}}_i$: $\mathbf{x}_i = \bar{\mathbf{x}}_i + \hat{\mathbf{x}}_i$. Hodnoty vektoru \mathbf{p} jsou v různých konfiguracích robota konstantní a můžeme je rozepsat jako původní odhad $\bar{\mathbf{p}}$ a chybu $\hat{\mathbf{p}}$: $\mathbf{p} = \bar{\mathbf{p}} + \hat{\mathbf{p}}$. Poté se rovnice (3.8) dá zapsat pro N kalibračních poloh jako:

$$\mathbf{f}(\mathbf{x}_i, \mathbf{p}) = \mathbf{f}(\bar{\mathbf{x}}_i + \hat{\mathbf{x}}_i, \bar{\mathbf{p}} + \hat{\mathbf{p}}) = \mathbf{0} \quad i = 1, \dots, N \quad (3.9)$$

Cílem kalibrace je najít hodnoty \mathbf{p} , které zlepší přesnost kalibrovaného kinematického modelu pro všechny kombinace chyby parametrů $\hat{\mathbf{p}}$ a naměřeného šumu $\hat{\mathbf{x}}_i$ splňující (3.9).

Vektor $\boldsymbol{\eta}$ Gaussova šumu se střední hodnotou nula a kovariancí $\boldsymbol{\Sigma}$ má hustotu pravděpodobnosti úměrnou $e^{-\boldsymbol{\eta}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta}}$ tudíž maximalizace pravděpodobnosti odpovídá minimalizaci funkce $\boldsymbol{\eta}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta}$. Pokud zavedeme předpoklad, že chyby $\hat{\mathbf{x}}_i$ a $\hat{\mathbf{p}}$ jsou vzájemně nezávislé, pak lze psát, že rozptyl($\hat{\mathbf{x}}_i$) = $E(\hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T) = \boldsymbol{\Sigma}_x$ a rozptyl($\hat{\mathbf{p}}$) = $\boldsymbol{\Sigma}_p$. Poté odhad s nejvyšší pravděpodobností splňující rovnici (3.9) minimalizuje:

$$\chi^2 = \sum_{i=1}^N \hat{\mathbf{x}}_i^T \boldsymbol{\Sigma}_x^{-1} \hat{\mathbf{x}}_i + \hat{\mathbf{p}}^T \boldsymbol{\Sigma}_p^{-1} \hat{\mathbf{p}} \quad (3.10)$$

Poté k řešení minimalizace je nutné vyjádřit Lagrangian:

$$\mathcal{L} = \chi^2 + \sum_{i=1}^N \boldsymbol{\lambda}_i^T \mathbf{f}(\bar{\mathbf{x}}_i + \hat{\mathbf{x}}_i, \bar{\mathbf{p}} + \hat{\mathbf{p}}), \quad (3.11)$$

kde $\boldsymbol{\lambda}_i$ jsou Lagrangeovy multiplikátory. Z první derivace (3.11) obdržíme extrémální rovnice. Kdybychom tyto rovnice chtěli řešit za pomoci Newtonovy metody s očekávanou kvadratickou konvergencí, potřebovali bychom druhé parciální derivace \mathbf{f} . Pro zjednodušení lze použít iterativní postup zanedbávající druhé derivace na úkor pomalejší (lineární) konvergence.

Dále zavedme normalizované chybové proměnné $\mathbf{y}_i \in R^k$ a $\mathbf{q} \in R^n$ s jednotkovými kovariancemi splňující:

$$\hat{\mathbf{x}}_i = \Sigma_x^{1/2} \mathbf{y}_i, \quad \hat{\mathbf{p}} = \Sigma_p^{1/2} \mathbf{q} \quad (3.12)$$

Kde symbol $^{1/2}$ značí druhou odmocninu matice. Ve většině případů jsou prvky \mathbf{y}_i a \mathbf{q} navzájem nezávislé a $\Sigma_x^{1/2}$, $\Sigma_p^{1/2}$ jsou diagonální matice s prvky odpovídajícími směrodatným odchylkám. Aplikací substituce (3.12) na rovnici (3.10) lze psát:

$$\chi^2 = \sum_{i=1}^N \mathbf{y}_i^T \mathbf{y}_i + \mathbf{q}^T \mathbf{q}. \quad (3.13)$$

Zde si lze povšimnout, že nově zavedené proměnné \mathbf{y}_i a \mathbf{q} jsou bezrozměrné.

Iterační proces započneme s nulovým počátečním odhadem a v každé iteraci budeme hledat korekci $\Delta \mathbf{y}_i$ a $\Delta \mathbf{q}$ minimalizující:

$$\sum_{i=1}^N (\mathbf{y}_i + \Delta \mathbf{y}_i)^T (\mathbf{y}_i + \Delta \mathbf{y}_i) + (\mathbf{q} + \Delta \mathbf{q})^T (\mathbf{q} + \Delta \mathbf{q}) \quad (3.14)$$

a zároveň splňující linearizované podmínky:

$$\mathbf{J}_{y_i} \Delta \mathbf{y}_i + \mathbf{J}_{q_i} \Delta \mathbf{q} = -\mathbf{f}(\bar{\mathbf{x}}_i + \Sigma_x^{1/2} \mathbf{y}_i, \bar{\mathbf{p}} + \Sigma_p^{1/2} \mathbf{q}) \quad i=1, \dots, N, \quad (3.15)$$

kde \mathbf{J}_{y_i} a \mathbf{J}_{q_i} jsou Jacobiho matice parciálních derivací získané užitím řetězkového pravidla:

$$\begin{aligned} \mathbf{J}_{y_i} &= \frac{\partial \mathbf{f}}{\partial \mathbf{x}} (\bar{\mathbf{x}}_i + \Sigma_x^{1/2} \mathbf{y}_i, \bar{\mathbf{p}} + \Sigma_p^{1/2} \mathbf{q}) \Sigma_x^{1/2} \\ \mathbf{J}_{q_i} &= \frac{\partial \mathbf{f}}{\partial \mathbf{p}} (\bar{\mathbf{x}}_i + \Sigma_x^{1/2} \mathbf{y}_i, \bar{\mathbf{p}} + \Sigma_p^{1/2} \mathbf{q}) \Sigma_p^{1/2} \end{aligned} \quad (3.16)$$

Dále seskupíme normalizované chybové proměnné \mathbf{y}_i skrze polohy měření do jednoho vektoru $\mathbf{y} = (\mathbf{y}_1^T, \dots, \mathbf{y}_N^T)^T$. Obdobně i $\mathbf{f}(\mathbf{x}_i, \mathbf{p})$ do vektoru \mathbf{F} a Jacobiány \mathbf{J}_{y_i} , \mathbf{J}_{q_i} do blokové struktury \mathbf{J}_y , \mathbf{J}_q . Po tomto seskupení můžeme rovnice (3.14) a (3.15) přepsat do tvaru:

$$(\mathbf{y} + \Delta \mathbf{y})^T (\mathbf{y} + \Delta \mathbf{y}) + (\mathbf{q} + \Delta \mathbf{q})^T (\mathbf{q} + \Delta \mathbf{q}) \quad (3.17)$$

$$\mathbf{J}_y \Delta \mathbf{y} + \mathbf{J}_q \Delta \mathbf{q} = -\mathbf{F} \quad (3.18)$$

Použitím QR rozkladu $\mathbf{QR} = \mathbf{J}_y^T$ a vynásobením obou stran rovnice (3.18) nesingulární maticí \mathbf{R}^{-T} dostaneme:

$$\mathbf{Q}^T \Delta \mathbf{y} + \mathbf{D} \Delta \mathbf{q} = -\mathbf{R}^{-T} \mathbf{F}, \quad (3.19)$$

kde matice $\mathbf{D} = \mathbf{R}^{-T} \mathbf{J}_q$ a matice \mathbf{Q} je matice s ortogonálními sloupci a rozměru $Nk \times Nm$ ($k \geq m$), kterou lze doplnit maticí \mathbf{Q}' rozměru $Nk \times N(k - m)$ tak, že $(\mathbf{Q}\mathbf{Q}')$ je ortogonální. Následně se dá minimalizační kritérium (3.17) zapsat jako:

$$\|Q^T(\mathbf{y} + \Delta\mathbf{y})\|^2 + \|Q'^T(\mathbf{y} + \Delta\mathbf{y})\|^2 + \|\mathbf{q} + \Delta\mathbf{q}\|^2. \quad (3.20)$$

Díky tomu, že se $Q'^T(\mathbf{y} + \Delta\mathbf{y})$ nevyskytuje ve vazbových rovnicích platí, že:

$$Q'^T(\mathbf{y} + \Delta\mathbf{y}) = 0. \quad (3.21)$$

Přeuspořádáním a vynásobením obou stran rovnice (3.19) $Q^T \mathbf{y}$ dostaneme:

$$Q^T(\mathbf{y} + \Delta\mathbf{y}) = Q^T \mathbf{y} - R^{-T} \mathbf{F} - D\Delta\mathbf{q}. \quad (3.22)$$

Dosazením z rovnic (3.21), (3.22) do rovnice (3.20) získáme:

$$\|Q^T \mathbf{y} - D\Delta\mathbf{q} - R^{-T} \mathbf{F}\|^2 + \|\mathbf{q} + \Delta\mathbf{q}\|^2. \quad (3.23)$$

Řešení minimalizace odpovídá řešení soustavy lineárních rovnic:

$$\begin{pmatrix} D \\ I \end{pmatrix} \Delta\mathbf{q} = \begin{pmatrix} Q^T \mathbf{y} - R^{-T} \mathbf{F} \\ -\mathbf{q} \end{pmatrix} \quad (3.24)$$

Odtud lze vyjádřit opravu odhadnutých kalibrovaných hodnot $\Delta\mathbf{q}$. Pro výpočet oprav $\Delta\mathbf{y}$ vyjdeme z rovnic (3.21), (3.22) a lze poté psát:

$$\mathbf{y} + \Delta\mathbf{y} = Q(Q^T \mathbf{y} - R^{-T} \mathbf{F}) \quad (3.25)$$

V každém kroku iterace se poté, viz předchozí postup, napočtou korekce $\Delta\mathbf{q}$, $\Delta\mathbf{y}$, ze kterých spolu s počátečním odhadem lze vyjádřit \mathbf{q} a \mathbf{y} , které se stávají odhadem pro další krok iterace. Iterační proces se ukončí, dosáhne-li se dostatečné přesnosti \mathbf{q} a \mathbf{y} (dostatečně malé změny mezi iteračními kroky).

3.3.3. Složitost a přesnost kalibrace

Provádí-li se kalibrace konkrétního mechanismu, je vhodné se také pozastavit nad tím, jak přesný výsledný kalibrační model může být. Je zřejmé, že výsledná přesnost bude záviset na parametrech jako jsou: přesnost měření polohy koncového bodu, počet a rozmístění proměřených poloh, počet a přesnost použitých senzorů. To, co na první pohled nemusí být zcela zřejmé je, že velký vliv má i konkrétní sestavení kalibračního modelu, volba a počet kalibrovaných proměnných. Aby bylo možné porovnat různé kalibrační modely, či provést jejich optimalizaci, je třeba zavést nějaké hodnotící kritérium – cílovou funkci (cost function). V praxi se osvědčilo použití kritérií na bázi podmíněnosti Jacobiho matice.

Řeší-li se problém zapsatelný jako soustava rovnic ve tvaru:

$$A\vec{x} = \vec{b} \quad (3.26)$$

a zahrnou-li se do soustavy nepřesnosti a zaokrouhlení, lze zkoumat řešení této soustavy ve tvaru:

$$(\mathbf{A} + \Delta\mathbf{A})(\vec{x} + \Delta\vec{x}) = \vec{b} + \Delta\vec{b}. \quad (3.27)$$

Za předpokladu, že $\Delta\mathbf{A} = 0$ lze vyjádřit:

$$\Delta\vec{x} = \mathbf{A}^{-1}\Delta\vec{b} \rightarrow \|\Delta\vec{x}\| \leq \|\mathbf{A}^{-1}\| \cdot \|\Delta\vec{b}\| \quad (3.28)$$

$$\mathbf{A}\vec{x} = \vec{b} \rightarrow \|\vec{x}\| \geq \frac{\|\vec{b}\|}{\|\mathbf{A}\|} \quad (3.29)$$

Pro relativní chybu řešení tedy platí:

$$\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|} \leq \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\| \frac{\|\Delta\vec{b}\|}{\|\vec{b}\|} \quad (3.30)$$

$$C_p = \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|. \quad (3.31)$$

Číslo C_p se nazývá podmíněnost matice. Pokud je matice dobře podmíněná, znamená to, že malé změny v koeficientech mají za důsledek malé změny v řešení. Pro $C_p \gg 1$ mluvíme o špatně podmíněných maticích. Příklad, kdy $\Delta\mathbf{A} \neq 0$ popisuje šíření chyb způsobených například numerickým zaokrouhlováním:

$$\frac{\|\Delta\vec{x}\|}{\|\vec{x}\|} \leq C_p \frac{\frac{\|\Delta\vec{A}\|}{\|\mathbf{A}\|} + \frac{\|\Delta\vec{b}\|}{\|\vec{b}\|}}{1 - C_p \frac{\|\Delta\vec{A}\|}{\|\mathbf{A}\|}}. \quad (3.32)$$

Při kinematické kalibraci je řešená úloha popsána rovnicemi (3.5), resp. (3.6), odkud je pro řešení soustavy nezbytné, aby Jacobiho matice soustavy \mathbf{J} nebyla singulární. K singulárnosti matice dochází, je-li kalibrační model nevhodně sestaven. Možné příčiny jsou: závislé zavedení kalibračních parametrů, či nevhodná volba kalibračních poloh, aj. Pokud matice \mathbf{J} není singulární, ale nemá k tomuto stavu daleko, mluvíme o špatně podmíněnosti Jacobiho matice (viz výše) a důsledkem je, že i malé chyby vstupní veličiny vedou k značným chybám veličiny výstupní. Pro určení podmíněnosti \mathbf{J} lze využít vztah (3.31) a následně vyjádřit: $C_J = \|\mathbf{J}\| \cdot \|\mathbf{J}^{-1}\|$. Je-li $C_J \gg 1$, pak je Jacobiho matice \mathbf{J} špatně podmíněná a důsledkem je, že parametry, získané kalibrací, nemusejí odpovídat skutečnosti a mohou být zatíženy značnými chybami. [4]

Dalším důležitým pojmem je kalibrovatelnost C . Tato vlastnost kalibračního iteračního kroku i je zavedena jako:

$$C = \|\mathbf{J}_{ai}^T \mathbf{J}_{ai}\| \cdot \|(\mathbf{J}_{ai}^T \mathbf{J}_{ai})^{-1}\|. \quad (3.33)$$

V každém kroku iterace kalibrovatelnost poskytuje bezrozměrné číslo, které umožňuje lepší vhled do závislosti parametrů a výsledků kalibrace. Dle [2] kalibrovatelnost uvádí míru schopnosti daného mechanismu získat přesné kinematické rozměry postupem kinematické kalibrace a

následné přesnosti polohování výstupního členu. Kalibrovatelnost lze také použít jako kritérium cílové funkce například při optimalizaci výběru konfigurací mechanismu pro kalibraci.

3.4. Neuro-fuzzy modelování

Neuro-fuzzy modelování systémů je velmi univerzální nástroj sloužící obecně k aproximaci především nelineárních funkcí více proměnných. Jeho hlavní výhodou je, že v sobě snoubí schopnost učení umělých neuronových sítí a pro lidský mozek snadno představitelnou fuzzy strukturu. Z těchto vlastností jasně vyplývá, že se dá využít ve velkém množství aplikací napříč různými vědeckými a inženýrskými obory. Toto téma bývá často skloňováno v souvislosti se strojovým učením, identifikací systému, potažmo i teorií řízení. Tato práce se primárně zaměřuje na použití z hlediska identifikace mechanických systémů – především na kinematickou kalibraci robotických ramen, kde je hlavním cílem porozumět fungování systému v reálném světě a vytvořit jeho matematický model.

Neuro-fuzzy modely lze vhodně použít u složitých nelineárních systémů, kde model systému vytvořený na základě fyzikálních vlastností nabývá na složitosti a stává se pro potřeby řízení systému v reálném čase nepoužitelný. U neuro-fuzzy modelů je mnohem větší volnost volby komplexnosti a k ní vztahované přesnosti modelu. Druhou významnou doménou použití jsou případy, kdy není možné získat o procesech v systému potřebné informace pro vytvoření přesného fyzikálního modelu. Tento přístup bývá označován jako černá skříň (black box). Pracuje se pouze s informacemi o chování výstupů v závislosti na chování vstupů systému.

Spojité nelineární model dynamického chování mechanického systému se dá obecně zapsat za použití stavového popisu ve tvaru:

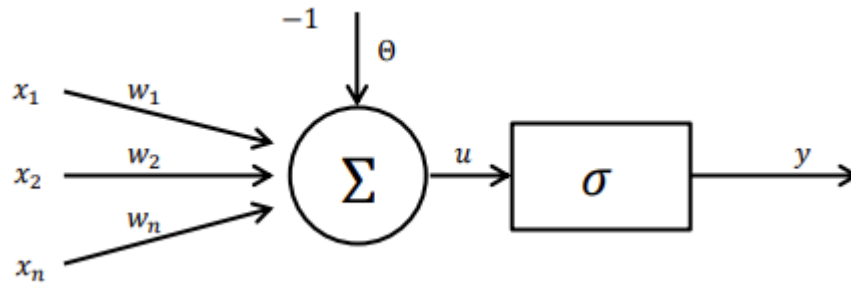
$$\dot{\mathbf{z}} = f(\mathbf{z}, \mathbf{u}) \quad (3.34)$$

kde \mathbf{z} je n rozměrný stavový vektor, $\dot{\mathbf{z}}$ je časová derivace stavového vektoru \mathbf{z} , \mathbf{u} je m dimenzionální vektor vstupu a f je časově invariantní funkcí stavového vektoru a vektoru vstupu. Právě tuto funkci f se obvykle snažíme identifikovat či modelovat.

Tato práce se zabývá problematikou vytváření kinematických modelů systémů, které jsou uvažovány v ustáleném stavu. Snaha identifikovat či modelovat funkci f , ale zůstává z podstaty stejná, pouze v popisu nevystupuje časová derivace.

3.4.1. Umělé neuronové sítě

Konstrukce umělých neuronových sítí vychází z inspirace biologickými neuronovými sítěmi. Snahou je vytvořit univerzální nástroj schopný učení. Celá síť se skládá z pospojované struktury neuronů. Každý neuron (Obr. 3) sčítá vstupní podněty x_i vynásobené váhami w_i a přesáhne-li součet jistou aktivační mez θ , vloží jej do aktivační funkce σ .



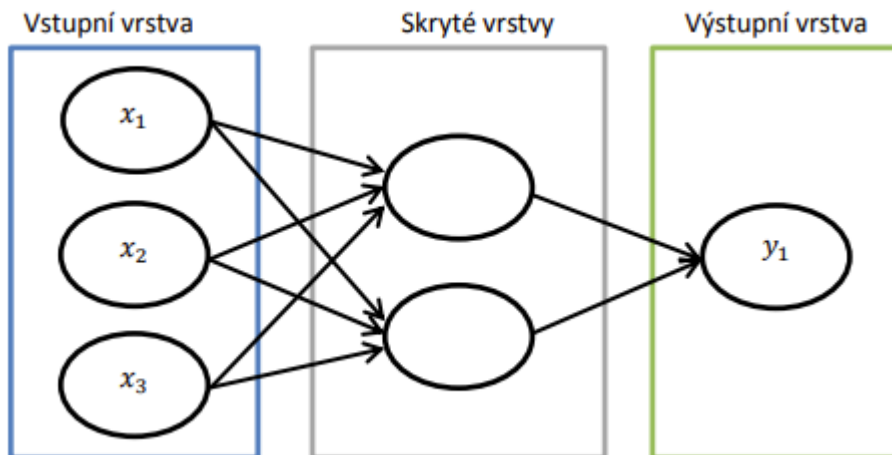
Obr. 3 Model neuronu převzato z [4]

Výstup y z neuronu je popsán rovnicí:

$$y = \sigma \left(\sum_{i=1}^N x_i w_i - \theta \right). \quad (3.35)$$

Chce-li se, aby neuronová síť byla schopna aproximace nelineárních funkcí je nutné volit aktivační funkce nelineární (nejčastěji funkce Sigmoid či hyperbolický tangens). V případě použití lineární aktivační funkce by celá síť přešla do lineárního systému kombinujícího lineární funkce, který by nebyl schopen nelineární aproximace.

V praxi se používá celá řada různých struktur propojení neuronů. Základním uskupením je vícevrstvá síť, která se skládá ze vstupní vrstvy, zvoleného množství skrytých vrstev a vrstvy výstupní. Pojmy: vstupní vrstva, skrytá vrstva a výstupní vrstva jsou snadno pochopitelné z Obr. 4.



Obr. 4 Struktura vícevrstvé neuronové sítě převzato z [4]

Klasické neuronové sítě nejsou podle [4] vhodným nástrojem k modelování přímé kinematické úlohy. Důvodem je jejich náročné trénování a velký počet neuronů (rozsáhlý výsledný model).

3.4.2. Fuzzy logika

Pojmy fuzzy logika (mlhová logika) a fuzzy množiny jsou pojmy zavedené matematikem Lotfi Zadehem. Fuzzy logika využívá stejných prvků jako lidské uvažování. Logické výroky v ní ohodnocují míru pravděpodobnosti. Zavádí lingvistické proměnné, jejichž hodnoty jsou věty v přirozeném či umělém jazyce. Rozdílem oproti tradiční booleovské logice je, že tyto proměnné nenabývají pouze dvou stavů ($\{\text{true}, \text{false}\} / \{1, 0\}$), ale hodnot z intervalu $\langle 0, 1 \rangle$.

V klasické teorii množin je možno množinu A popsat její charakteristickou funkcí $\mu_A: X \rightarrow \{0, 1\}$,

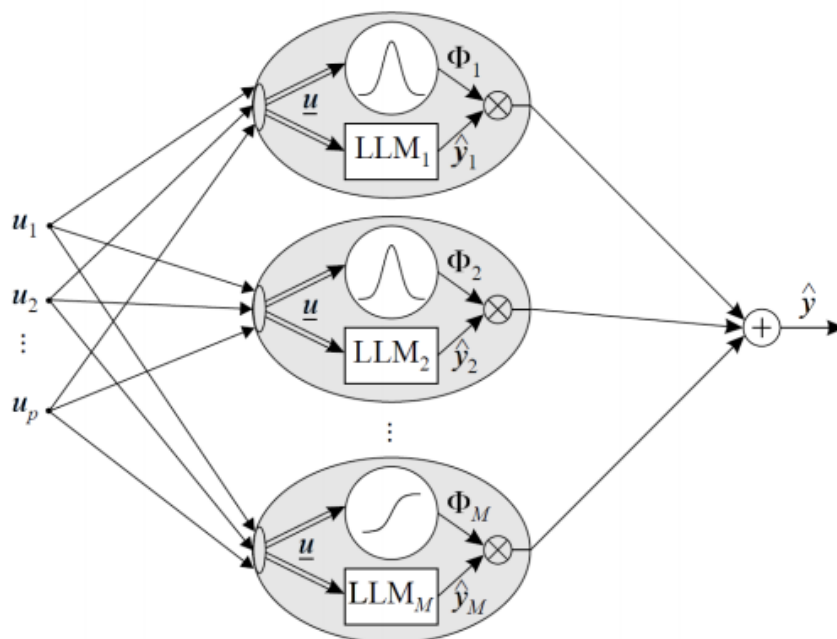
$$\mu_A(x) = \begin{cases} 1 & \text{Pro } x \in A \\ 0 & \text{Pro } x \notin A \end{cases} \quad (3.36)$$

Pro zavedení fuzzy podmnožiny A univerza X je třeba zobecnit charakteristickou funkci μ_A tak, aby mohla nabývat pravdivostních hodnot v intervalu reálných čísel $\langle 0, 1 \rangle$: $\mu_A: X \rightarrow \langle 0, 1 \rangle$. Pro každý prvek $x \in X$ hodnota $\mu_A(x)$ říká do jaké míry je x prvkem fuzzy množiny A .

Charakteristické funkce μ_A , často označované též jako funkce platnosti/příslušnosti, mohou nabývat nejrůznějších tvarů. Obecně se dají rozdělit na lineární (tvořeny lineárními funkcemi) a křivkové (často používanými zástupci jsou Gaussiány). [4]

3.4.3. Neuro-fuzzy sítě a modely

Základním přístupem lokálních neuro-fuzzy modelů je rozdělení vstupního prostoru na lineární podprostory (LLM) s fuzzy funkcemi platnosti. Každý z těchto podprostorů s příslušnou funkcí platnosti reprezentuje jeden fuzzy neuron. Celkový model se tedy dá chápat jako neuro-fuzzy síť s jednou skrytou vrstvou a jedním lineárním neuronem ve výstupní vrstvě viz Obr. 5.



Obr. 5 Struktura neuro-fuzzy sítě přejato z [7]

Výstup \hat{y} takové sítě se poté dá zapsat v tvaru:

$$\hat{y} = \sum_{i=1}^M (w_{i0} + w_{i1}u_1 + \dots + w_{ip}u_p) \Phi_i(\mathbf{u}), \quad (3.37)$$

kde $\mathbf{u} = [u_1, u_2, \dots, u_p]^T$ je vektor vstupů modelu, M je počet neuronů ve skryté vrstvě a w_{ij} jsou lineární parametry i -tého neuronu. $\Phi_i(\mathbf{u})$ je normalizovaná funkce platnosti i -tého lokálního lineárního modelu, která je funkcí vektoru vstupů \mathbf{u} . Normalizovaná funkce platnosti je typicky volena jako normalizovaný Gausián. Jsou-li tyto Gausiány osově ortogonální, dají se funkce platnosti vyjádřit jako:

$$\Phi_i(\mathbf{u}) = \frac{\mu_i(\mathbf{u})}{\sum_{j=1}^M \mu_j(\mathbf{u})}, \quad (3.38)$$

kde

$$\mu_i(\mathbf{u}) = \exp\left(-\frac{1}{2}\left(\frac{(u_1 - c_{i1})^2}{\sigma_{i1}^2} + \frac{(u_2 - c_{i2})^2}{\sigma_{i2}^2} + \dots + \frac{(u_p - c_{ip})^2}{\sigma_{ip}^2}\right)\right). \quad (3.39)$$

Z rovnice (3.39) je zřejmé, že $\Phi_i(\mathbf{u})$ je závislé na souřadnicích středů c_{ij} jednotlivých lineárních modelů a na standardních směrodatných odchylkách σ_{ij} . Tyto odchylky se dají vyjádřit jako:

$$\sigma_{ij} = k_\sigma \Delta_{ij}, \quad (3.40)$$

kde Δ_{ij} reprezentuje velikost i -té podoblasti v j -té dimenzi a parametr k_σ faktor proporcionality mezi velikostí podoblasti a standardní směrodatnou odchylkou. Určení těchto parametrů je samostatná úloha nelineární optimalizace. Často se tato úloha ale obchází heuristickou volbou parametru $k_\sigma = \frac{1}{3}$, která ve většině případů nabízí uspokojivé řešení.

Na tomto místě je vhodné také zmínit velmi důležitou vlastnost funkcí platnosti vycházející z rovnice (3.38) a to, že součet všech hodnot funkcí platnosti jednotlivých LLM je pro daný vstup vždy roven jedné viz rovnice (3.41).

$$\sum_{i=1}^M \Phi_i(\mathbf{u}) = 1 \quad (3.41)$$

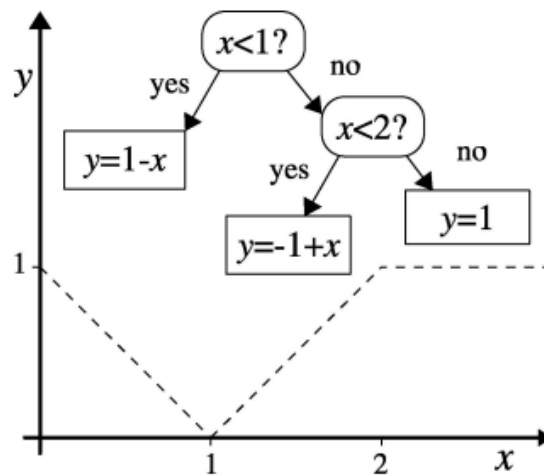
[7], [4]

3.4.4. Lineární modelové stromy

Obecně se úloha učení n dimenzionální funkce např. (3.34) dá rozdělit po jednotlivých dimenzích na n samostatných úloh učení. Každá z nich se dá poté zapsat jako regresní problém (3.42). Pozorované hodnoty y jsou zatíženy šumem ε s neznámým rozptylem σ^2 . Cílem regresní analýzy je nalezení aproximační funkce \hat{f} k funkci f , která minimalizuje zvolenu cílovou funkci (cost function), např. sumu čtverců chyb, na učícím souboru vzorků.

$$y = f(\mathbf{z}, \mathbf{u}) + \varepsilon \quad (3.42)$$

Zde se nabízí možnost funkci \hat{f} vyjádřit za pomoci lineárního modelového stromu. Přesněji binárního rozhodovacího stromu s lineárním modelem v každém listu, kde každý vnitřní bod stromové struktury obsahuje dělicí podmínku závislou na regresních proměnných, která rozděluje data do dvou podoblastí (označováno, jako levý a pravý podstrom či „dítě“) viz Obr. 6.



Obr. 6 Znáznornění stromové struktury převzato z [8]

Nejjednodušší a zároveň nejčastěji používaný přístup uvažuje dělicí podmínky osově-paralelní (každá podmínka pracuje pouze s jedním regresorem). Rozhodovací strom dělí vstupní prostor do vícedimenzionálních obdélníků. Každý z těchto obdélníků obsahuje právě jeden lineární model:

$$\hat{f}(\mathbf{z}, \mathbf{u}) = \hat{f}(\mathbf{X}) = \mathbf{X}^T \hat{\boldsymbol{\theta}} \quad (3.43)$$

kde \mathbf{X} je d rozměrný sloupcový vektor složený z $d-1$ numerických složek a jedné konstanty (poslední člen vektoru je 1, pro zjednodušení zápisu) a $\hat{\boldsymbol{\theta}}$ je d -rozměrný sloupcový vektor parametrů. Dále je zavedena chyba i -tého vzorku e_i a suma čtverců chyb J pro N vzorků:

$$e_i = y_i - \hat{f}(X_i) \quad (3.44)$$

$$J = \sum_{i=1}^N e_i^2 \quad (3.45)$$

Poté lze provést seskupení pro N vzorků v jenom listu (vícedimenzionálním obdélníku):

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}; \mathbf{X} = \begin{bmatrix} \mathbf{X}_1^T \\ \vdots \\ \mathbf{X}_N^T \end{bmatrix}; \mathbf{e} = \begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix} \quad (3.46)$$

a za pomoci metody nejmenších čtverců minimalizovat kvadratickou odchylku $J = \mathbf{e}^T \mathbf{e}$:

$$\hat{\boldsymbol{\theta}}_{LS} = [\mathbf{X}^T \mathbf{X}]^{-1} \mathbf{X}^T \mathbf{y}. \quad (3.47)$$

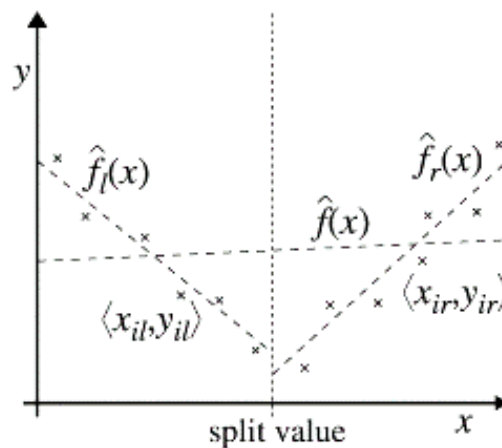
Čtverec chyb aproximace (residual sum of squares – RSS) lze vyjádřit, jako sumu čtverců odchylek:

$$RSS = \sum_{i=1}^N (y_i - \mathbf{X}_i^T \hat{\boldsymbol{\theta}}_L)^2 \quad (3.48)$$

Rozsáhleji pojednáno a rozšířeno o inkrementální schéma výpočtu čtverce chyb aproximace v [8].

4. Dávkový indukční algoritmus – Batch-RD

Nejběžnějším přístupem dávkových (batch) algoritmů k vytváření lokálních modelových stromů je začít shora od takzvaného kořene (celé sady vzorků) a postupovat indukčně směrem dolů. V každém uzlu je trénovací sada vzorků rekurzivně dělena za pomoci dělicích pravidel (splitting rules), dokud není strom dostatečně přesný nebo nejsou splněny jiné podmínky zastavující dělení. Přeznačí-li se N vzorků v jednom bodu jako $\langle x_i, y_i \rangle$, poté každý potenciální bod dělení rozdělí tyto vzorky do dvou disjunktních podmnožin (označovány jako levá a pravá podmnožina). N_l vzorků v levé podmnožině můžeme zapsat jako $\langle x_{il}, y_{il} \rangle$ obdobně N_r vzorků v pravé podmnožině jako $\langle x_{ir}, y_{ir} \rangle$.



Obr. 7 Dělení vzorků na podmnožiny převzato z [8]

Pro aproximaci funkce jedné proměnné je dělení prostoru na podmnožiny zobrazeno na Obr. 7. Zde je také každému z podprostorů přidělena aproximace lineární funkcí levému $\hat{f}_l(x)$, pravému $\hat{f}_r(x)$ zkonstruovaných na základě vzorků na příslušné straně od dělicího bodu. Otázkou je, jak tyto aproximace získat. Jako funkční řešení se osvědčilo provést minimalizaci cílové funkce:

$$RSS_l + RSS_r = \sum_{i=1}^{N_l} (y_{il} - \hat{f}_l(x_{il}))^2 + \sum_{i=1}^{N_r} (y_{ir} - \hat{f}_r(x_{ir}))^2. \quad (4.1)$$

Toto řešení vyžaduje spočtení lineárních odhadů nejmenších čtverců $\hat{\theta}_{LS}$ za pomoci rovnice (3.47) pro obě podmnožiny vzorků na každé straně od potenciálního bodu dělení. Je zřejmé, že počet potenciálních bodů dělení se zvyšuje s velikostí (počtem vzorků) a dimenzí řešené úlohy. Úloha se stává velmi rychle výpočetně náročná, až neřešitelná, pro otestování všech potenciálních bodů dělení. Z tohoto důvodu je u těchto indukčních algoritmů většinou voleno menší množství potenciálních bodů dělení (podmnožina ze všech možných), a to i přesto, že následkem je pravděpodobná suboptimální realizace dělení prostoru. Obvykle je voleno rovnoměrné dělení pozorovaného rozsahu na κ kandidátů dělení pro každý z $d - 1$ regresorů. Tudíž v každém uzlu stromu je $\kappa (d - 1)$ potenciálních bodů dělení. Parametr κ je jedním ze dvou vstupních parametrů řídících stavbu stromu. Experimentálně bylo zjištěno, že rozumnou volbou pro rozsáhlé data sety je $\kappa \in \{3; 5\}$. Menší hodnoty mohou lehce snižovat přesnost generovaných stromů, zatímco větší vyžadují značně více výpočtů a nepřinášejí významné zvýšení přesnosti. (U data setů s méně

vzorky, např. v řádech stovek, nehraje výpočetní čas významnou roli a není důvod nezkusit použít vyšší hodnoty κ .)

4.1. RD dělicí pravidlo

Dělicí pravidlo (splitting rule) má za cíl posoudit, zda dva lineární modely na každé straně potenciálního bodu dělení dávají lepší výsledek než aproximace lineárním modelem přes celou oblast. Jedním z používaných dělicích pravidel je RD dělicí pravidlo, které je založeno na rozdílech součtu čtverců odchylek od lineárních modelů a jejich statistickém zpracování.

Otázku, zda dva lineární modely na každé straně potenciálního bodu dělení aproximují vzorky lépe než jeden lineární model, lze testovat jako hypotézu. Nulová hypotéza je, že aproximovaná funkce je lineární v celém uzlu ($H_0: f(\mathbf{X}) = \mathbf{X}^T \hat{\boldsymbol{\theta}}$), zatím co alternativní hypotéza je, že tomu tak není. Tři lineární modely jsou vytvořeny v uzlu (viz Obr. 7): $\hat{f}(x)$ na základě všech N vzorků se sumou čtverců chyb RSS , $\hat{f}_l(x)$ na základě N_l vzorků ležících vlevo od potenciálního bodu dělení se sumou čtverců chyb RSS_l a $\hat{f}_r(x)$ na základě N_r vzorků ležících vpravo od potenciálního bodu dělení se sumou čtverců chyb RSS_r . Je zřejmé, že dva modely vždy budou extrapolovat data stejně, nebo lépe než jediný model, což se dá zapsat jako $RSS_l + RSS_r \leq RSS$. Když je alternativní hypotéza splněna, poté platí $RSS_l + RSS_r \ll RSS$. Tato alternativní hypotéza může být zkoumána Chow testem [9], standardním statistickým testem zkoumajícím, zda jsou koeficienty dvou lineárních regresí na různých souborech dat stejné. Poté může být pro nulovou hypotézu ukázáno, že statistika

$$F_{batch} = \frac{(RSS - RSS_l - RSS_r)(N - 2d)}{(RSS_l + RSS_r)d} \quad (4.2)$$

je rozložena dle Fischerova rozdělení s d a $N - 2d$ stupni volnosti (d viz kapitola 3.4.4). Potenciální bod dělení s nejmenší pravděpodobností hypotézy H_0 by měl být nejlepší volbou pro dělení, což odpovídá statistice F_{batch} s nejmenší přidruženou p -hodnotou. Zde si lze povšimnout, že jelikož je N , RSS , d konstantní pro všechny kandidáty dělení, p -hodnota je minimalizována, je-li minimalizováno $RSS_l + RSS_r$.

Tato statistika kromě nalezení nejlepšího místa dělení, také navíc může určit, zda je skutečně vhodné dělení realizovat. Označíme-li nejmenší p -hodnotu jako α , poté by dělení mělo být realizováno jen v případě, že je α dostatečně malé, aby vyloučilo hypotézu H_0 s požadovanou pravděpodobností. Zde je potřeba ještě aplikovat Bonferroniho korekci pro testování stejné hypotézy mnohokrát současně. Koeficient α_{split} je určen z Fischerova rozdělení (s d a $N - 2d$ stupni volnosti) tak, aby byla hypotéza H_0 vyloučena s pravděpodobností 95 % (toto je vlastně další „skrytý“ parametr algoritmu), poté může být podmínka dělení vyjádřena jako:

$$\alpha < \frac{\alpha_{split}}{\kappa(d-1)} \quad (4.3)$$

4.2. Pravidlo pro zastavení dělení

Obzvláště u rozsáhlých datových sad je žádoucí omezit růst stromu. Pro zastavení dělení stromu (omezení jeho dalšího růstu) je třeba zavést pravidlo, podle kterého algoritmus pozná, že další dělení už není výhodné. Jednou možností je prosté omezení, do jaké vrstvy může algoritmus indukovat. Toto řešení je velmi jednoduché na implementaci, ale také velmi nerobustní a zanášející do stromové struktury zbytečné nepřesnosti. (Některé oblasti potřebují být aproximovány více modely než jiné.)

Univerzálnějšího řešení tohoto problému je dosaženo odhadnutím příspěvku k celkové přesnosti stromu daným dělením v porovnání k přesnosti celého stromu. Následné dělení je poté provedeno jen v případě, že je tento příspěvek dostatečně velký. Parametr δ , který toto popisuje je zaveden jako

$$\delta = \frac{1}{s_y^2} \left(\frac{RSS}{N-d} - \frac{RSS_l + RSS_r}{N_l + N_r - 2d} \right) \quad (4.4)$$

kde s_y je rozptyl hodnot y (viz rovnice (3.43)), který zajišťuje, že tento parametr je neměnný pro změnu měřítka v hodnotách y a poskytuje odhad snížení odchylky, pokud dojde k dělení uzlu. Jak strom roste (aproximuje vzorky přesněji) parametr δ klesá.

Finální podmínka zastavení dělení tedy zní: „Pokud δ poklesne pod prahovou hodnotu δ_0 je dělení zastaveno.“ Tato podmínka dělení je ještě doplněná podmínkou zaručující, že je možné sestavit lineární modely po dělení: „K dělení dochází pouze je-li v každém nově vytvořeném uzlu alespoň trojnásobek počtu vzorků N_l, N_r , než je parametrů lineárních modelů d .“ Prahová hodnota δ_0 je druhým vstupním parametrem řídícím stavbu stromu.

4.3. Algoritmus

V této kapitole je nejdříve popsána algoritmicizace dávkového algoritmu Batch-RD. Jedná se o neinkrementální (dávkovou) formu algoritmu označovaného, jako IMTI – *Incremental model tree induction* s RD dělicím pravidlem vycházející z algoritmu *RETIS* [10]. Dále je popsána autorem provedená realizace tohoto algoritmu v programovacím jazyku Matlab a testování funkcionality na konkrétních testovacích funkcích.

Tento algoritmus se vyznačuje tím, že vstupní prostor prokládá nespojitými (na sebe ne navazujícími) lineárními modely. V některých aplikacích nespojitost výsledného modelu nemusí být překážkou, v jiných, kde je spojitost vyžadována, je možné celkový model vyhladit za pomoci funkcí platnosti diskutovaných v kapitole 3.4.3. Toto vyhlazení má ovšem za následek poměrně znatelné zhoršení přesnosti modelu a díky tomu dle [8] není příliš často užíváno. Jak je také diskutováno v 3.4.3, k dosažení vyšší přesnosti by bylo třeba provést optimalizace použitých funkcí platnosti k vyhlazení, ale tento problém není v této práci adresován. Je vhodné zmínit, že u inkrementální formy zde diskutovaného algoritmu je po vypočtení modelů užívána metodika snižující nadměrnou komplexnost modelu (Pruning). Při dávkové formě algoritmu pozbývá její použití význam.

4.3.1. Obecná struktura algoritmu

Hlavní částí algoritmu je funkce provádějící hledání a testování volby nevhodnějšího místa dělení, následné případné dělení data setu a prokládání oblastí lineárními modely (viz Obr. 8). Tato funkce volá v sobě sebe samu (je tzv. funkcí rekurzivní), čímž vzniká rekurzivní struktura algoritmu. Byť je struktura algoritmu na první pohled velmi jednoduchá, díky vnitřní rekurzi vyvstává poměrně složitý problém zavedení datových struktur v algoritmu. Toto je obecná vlastnost algoritmizace stromových struktur.

```
function DávkováIndukce (soubor dat ds):  
  
1   provedení statistiky na ds k nalezení místa dělení  
2   if dělicí pravidlo == True && Pravidlo zastavení dělení == False  
3       rozdělení dat na dvě části (dsleft dsright)  
4       leftChild = DávkováIndukce(dsleft)  
5       rightChild = DávkováIndukce(dsright)  
6       return vnitřní uzel stromu s leftChild a rightChild  
7   else  
8       return listový uzel s lineárním modelem napočteným z ds dle (3.47)  
9   end  
  
end function
```

Obr. 8 Struktura indukční funkce (pseudokód)

4.3.2. Konkrétní realizace algoritmu

Tato forma dávkového indukčního algoritmu s RD dělicím pravidlem byla realizována v programovacím jazyku Matlab. Celý kód včetně testovacích datových sad je umístěn v příloze. Komentovat zde celý kód by nebylo přínosné, proto se v této části zaměřím především na mnou zavedené datové struktury a okomentuji použité funkce.

Celý kód je spouštěn skriptem Call_batch_induction.m, kde jsou nejdříve načteny ze souboru vstupní data potřebná pro vytvoření stromové struktury. Vstupní data jsou poté uložena v matici (poli) ds o rozměru $N \times (d + 1)$, která dodržuje strukturu $ds = [X \ y]$, kde X , y je zavedeno viz kapitola 3.4.4. Následuje definice parametrů algoritmu: proměnná k definuje v kolika bodech bude testováno dělení podle každé osy (regresoru), proměnná $delta$ definuje prahovou hodnotu δ_0 definovanou v kapitole 4.2. Následně je do matice $limits$ (rozměru $2 \times (d - 1)$) uložen rozsah jednotlivých regresorů viz:

$$limits = \begin{bmatrix} x_{1 \min} & \dots & x_{(d-1) \min} \\ x_{1 \max} & \dots & x_{(d-1) \max} \end{bmatrix}. \quad (4.5)$$

Dále je do proměnné S_y načten kvadrát variace vzorků y aproximované funkce. Poté následuje výpočet nulté vrstvy stromu (kořene), což znamená vytvoření lineárního modelu přes celý vstupní data set ds . K tomu slouží funkce $LLSE(ds)$, která má jako vstup data set ds a vrací parametry lineárního modelu napočtené z rovnice (3.47), které jsou uloženy do vektoru O_Is (o rozměru $d \times 1$). Následuje výpočet sumy čtverců chyby této aproximace Rss (viz rovnice (3.48)) za pomoci funkce $RS(ds, O_Is)$. V tuto chvíli je již vypočteno vše potřebné k spuštění samotného výpočtu.

Základem výpočtu je funkce diskutovaná v kapitole 4.3.1. Ve vytvořeném programu se ekvivalent této funkce jmenuje `Batch_induction`. Vstupem této funkce jsou: ds , O_Is , Rss , $limits$, $vrstva$, k , $vrstva_max$, $delta_stop$, S_y . Výstupem z této funkce je matice $Model$. Parametr $vrstva$ se funkcí předává, aby bylo možné trasovat, ve které vrstvě vznikl který lineární model a parametr $vrstva_max$ slouží k možnému omezení maximální vrstvy, do které algoritmus může indukovat. (Pokud nechci tuto funkci využít na počátku výpočtu dříve, než je volána indukční funkce, nastavím tento parametr na neúměrně velkou hodnotu např. 100.) Na začátku funkce se provede statistické ohodnocení potenciálních míst dělení. Aby se snížila výpočetní náročnost, předávají se funkci `Batch_induction` hodnoty O_Is , Rss , $limits$ z předešlé vrstvy. Poté je nalezeno optimální místo dělení a jsou-li splněny podmínky dělení a zároveň nesplněny podmínky zastavení, je provedeno skutečné rozdělení data setu ds na ds_l a ds_r , najdou se v těchto data setech nové limity regresorů ($limits_l$ a $limits_r$) a napočtou se nové lineární modely, přesněji řečeno z důvodu úspory výpočetní náročnosti tyto parametry jsou ukládány již při výpočtu statistického ohodnocení potenciálních míst dělení a v této fázi jsou pouze přiřazeny ty správné. Následně je rekurzivně volána funkce `Batch_induction` na tyto nové datové sady (ds_l a ds_r) a s nimi spjaté proměnné. Funkce zde končí návratem datové struktury $Model$ složené z pod sebou řazených datových struktur $Model$ ($leftChild$ a $rightChild$) navracených z funkce `Batch_induction`. Nemá-li podmínka dělení splněna, nebo již v jedné z oblastí není dostatečný počet vzorků k sestavení lineárních modelů, či je platná podmínka zastavení dělení funkce (bylo dosaženo požadované přesnosti), pouze se sestaví datová struktura $Model$ ze vstupních parametrů, která se navrátí jako výstup funkce. Datová struktura $Model$ navracená z nejnižších vrstev má následující tvar:

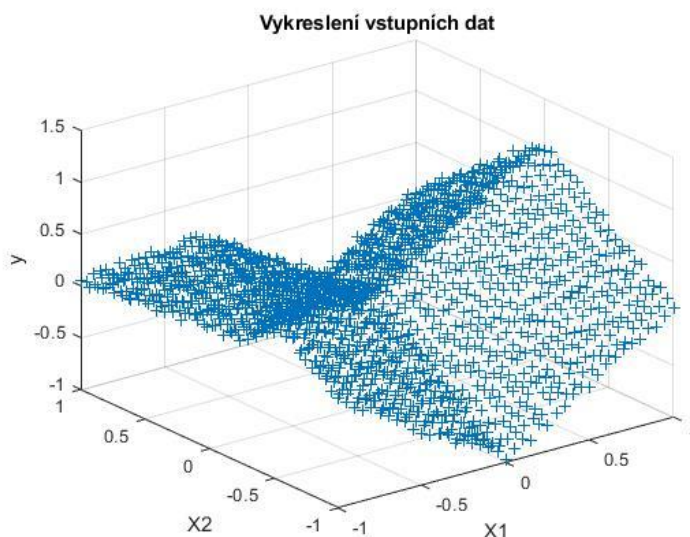
$$Model = \begin{bmatrix} x_{1 \min} & \dots & x_{(d-1) \min} & vrstva \\ x_{1 \max} & \dots & x_{(d-1) \max} & Rss \\ \theta_1 & & \theta_2 \dots & \theta_d \end{bmatrix}. \quad (4.6)$$

Odtud je zřejmé, že po dokončení celé indukce je navracen celkový model, ve kterém každé tři řádky reprezentují: jednu oblast proloženou jedním lineárním modelem, vrstvu, ve které byl model napočten a odchylku daného lineárního modelu od testovaných dat. Celkový navracený model je tedy matice $Model$ o rozměru $(3 \cdot \text{počet modelů}) \times d$.

Z celkového stromového modelu lze poté vypočítat aproximaci libovolného bodu ležícího v mezích vstupních dat. K tomu slouží funkce $tree_calc(Model, X_i^T)$, která navrací hodnotu aproximace v bodě X_i^T . Pro vypočtení aproximace z vyhlazeného neuro-fuzzy modelu je možné použít funkci $tree_smoothing_calc(Model, X_i^T)$, která navrátí aproximovanou hodnotu v bodu X_i^T s využitím funkcí platnosti viz kapitola 3.4.3.

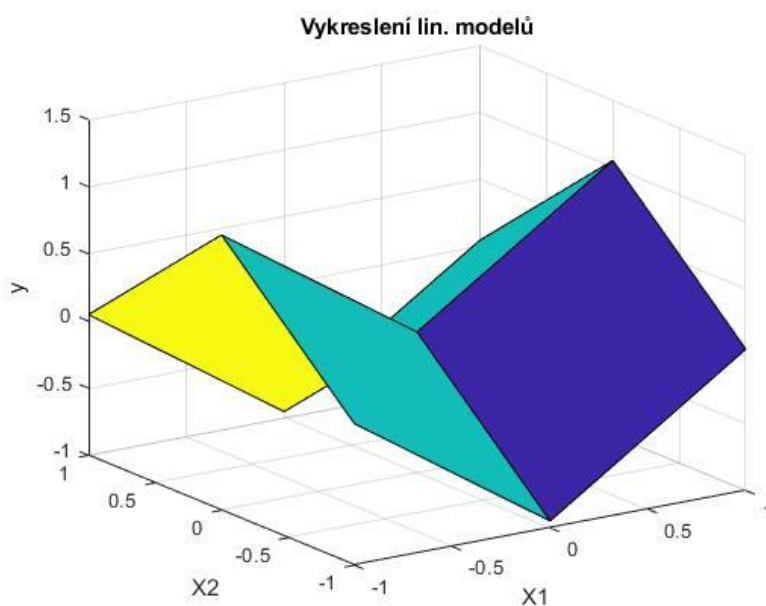
4.3.3. Testování algoritmu

Pro otestování funkcionality algoritmu jsem zvolil jednoduché testovací funkce dvou proměnných, jelikož se jejich výstup dá snadno vykreslit v závislosti na vstupu a tím snadno prokázat, že algoritmus pracuje, jak má. Nejdříve jsem si vygeneroval učící datovou sadu 1681 vzorků na funkci složenou ze čtyř rovin v prostoru zatížených šumem ($y = |x_1| - |x_2| + rand() \cdot 0,1$) viz Obr. 9.



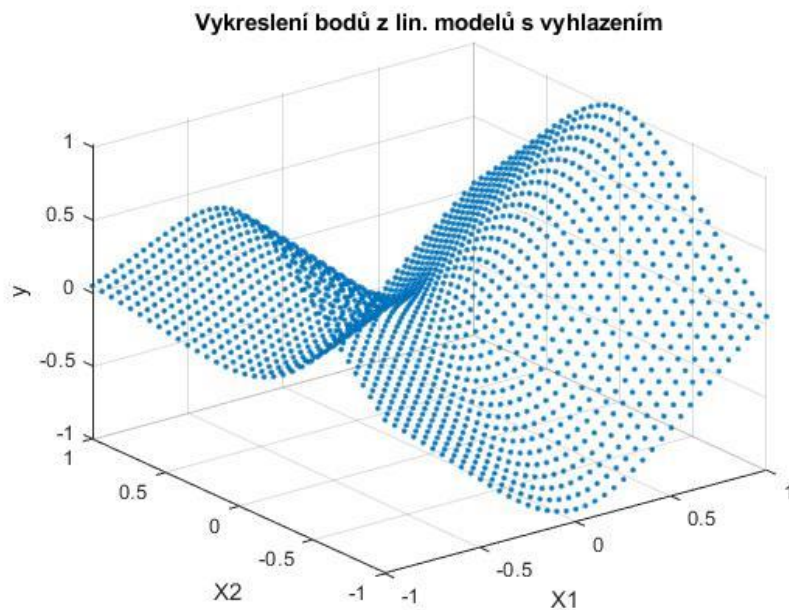
Obr. 9 Vstupní data testovací funkce 1

Dále jsem tato data použil jako vstup do algoritmu, který vytvořil stromový model. Algoritmus správně rozpoznal, že se jedná o čtyři oblasti, aproximovatelné lineárním modelem s dostatečnou přesností a po rozdělení prostoru na čtyři oblasti přestal dále dělit a navrátil model skládající se z čtyř podoblastí s lineárními modely viz Obr. 10.



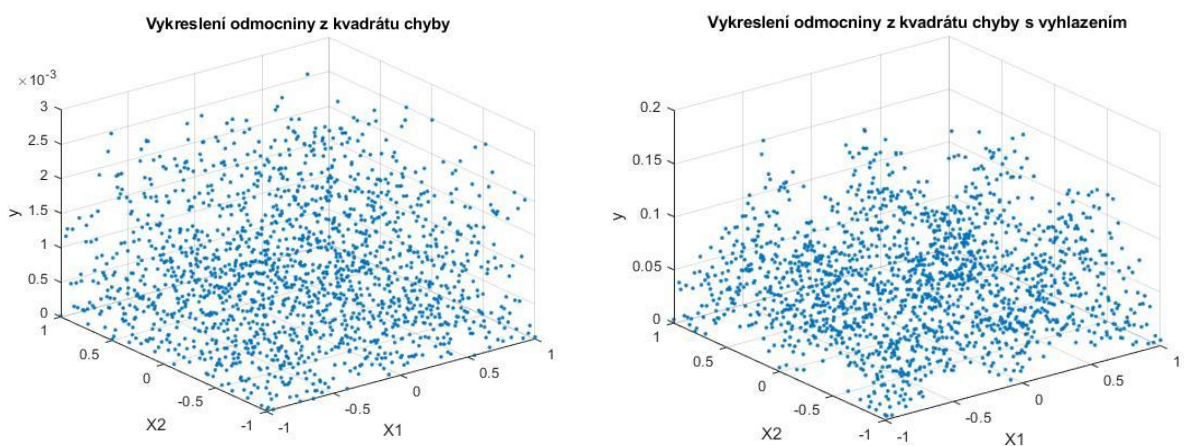
Obr. 10 Vykreslení výstupu algoritmu aproximace funkce 1

Pro porovnání jsou níže vykresleny i body aproximace z modelu s vyhlazením viz Obr. 11, kde je na první pohled vidět, že v místech, kde funkce není hladká vznikají chyby aproximace.



Obr. 11 Vykreslení bodů aproximace funkce 1 s vyhlazením

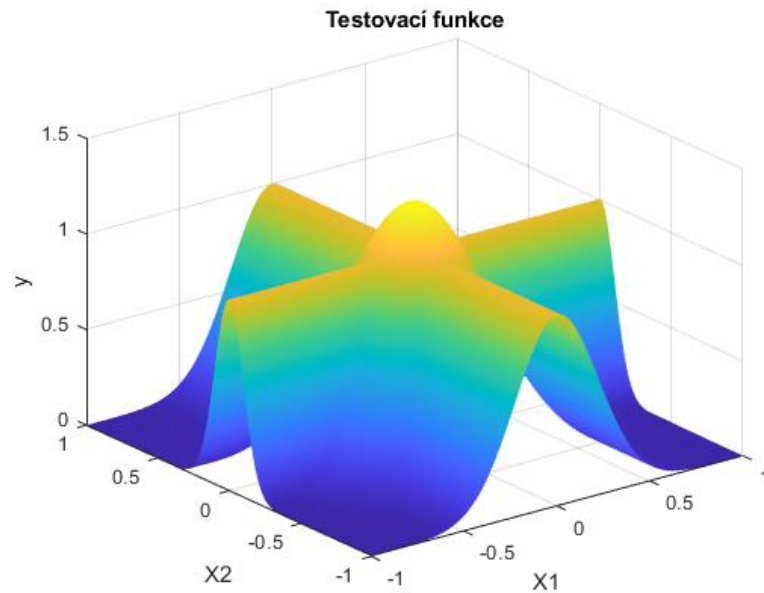
Na Obr. 12 je zobrazeno porovnání odmocnin kvadrátu chyb aproximace. Je zřejmé, že pro tuto testovací funkci vychází aproximace jen za použití lineárních modelových stromů minimálně o řád lépe než neoptimalizované vyhlazení za pomoci funkcí platnosti.



Obr. 12 Porovnání odmocniny kvadrátu chyby modelu bez a s vyhlazením

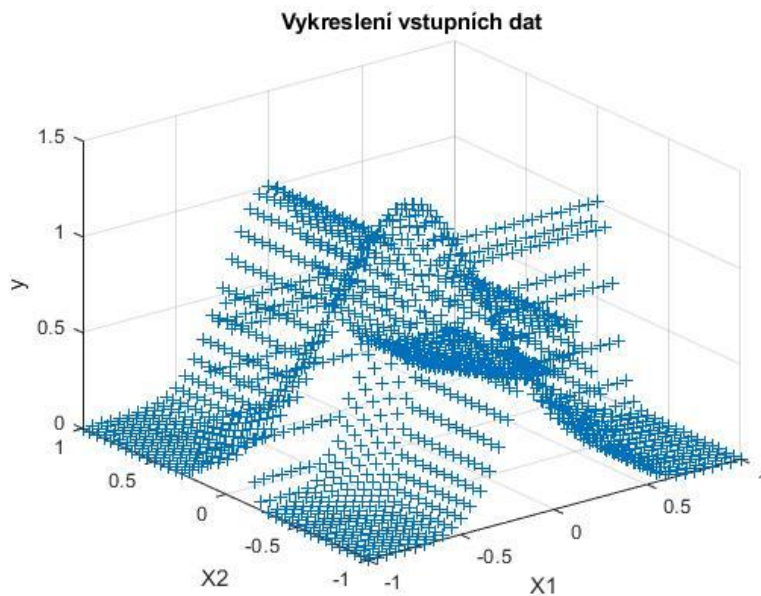
Z pohledu použití diskutovaného algoritmu kompenzace chyby nad kinematickým modelem je nejdůležitější kvalita aproximace nelineárních funkcí. Proto jsem jako druhou testovací funkci algoritmu zvolil nelineární funkci převzatou z [8]:

$$y = \max \left\{ e^{-10x_1^2}, e^{-50x_2^2}, 1,25e^{-5(x_1^2+x_2^2)} \right\}. \quad (4.7)$$



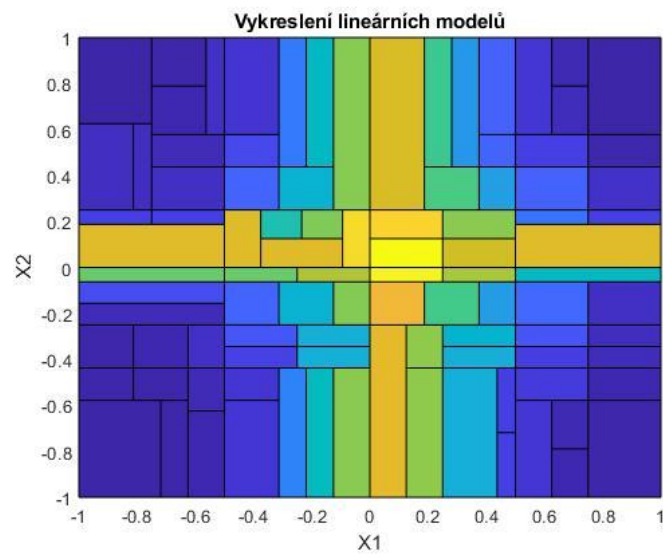
Obr. 13 Testovací funkce 2

Poté jsem vygeneroval sadu testovacích dat (1681 vzorků viz Obr. 14), které jsem poté použil jako vstup do algoritmu.



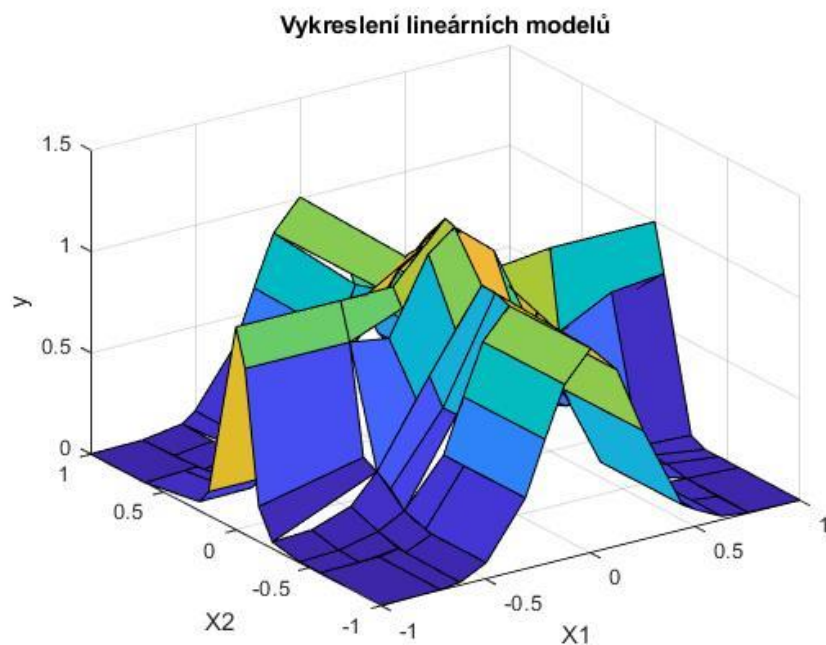
Obr. 14 Vstupní data testovací funkce 2

Na Obr. 15 je ukázáno, jak algoritmus rozdělil prostor na 97 podoblastí. Vstupní parametry algoritmu byly pro tento výpočet nastaveny: $k=3$ a $\text{delta_stop}=0.0001$.



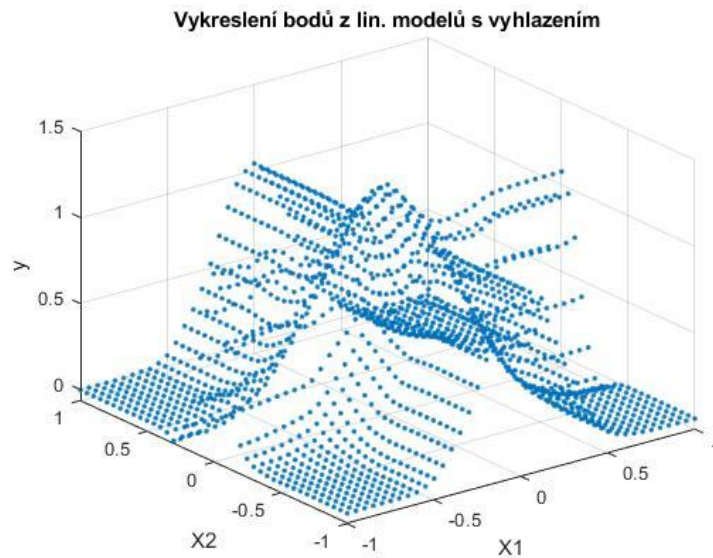
Obr. 15 Ukázka rozdělení prostoru algoritmem pro testovací funkci 2

Vykreslení lineárních modelů je zobrazeno na Obr. 16 (automatická barevná škála hodnot pro Obr. 15 a Obr. 16 téměř odpovídá škále použité na Obr. 13).

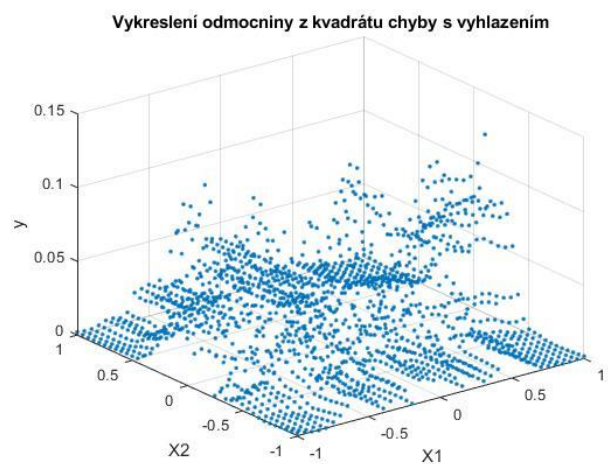
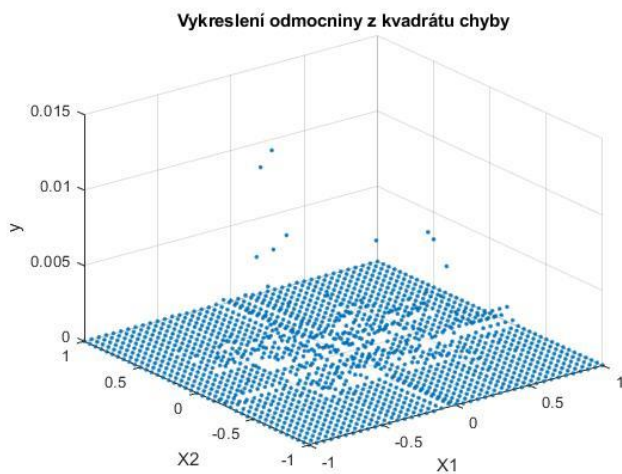


Obr. 16 Vykreslení výstupu algoritmu aproximace funkce 2

Pro účel srovnání zde ještě uvedu vykreslení bodů z vyhlazeného modelu zobrazené na Obr. 17 a grafy odmocniny kvadrátu chyby modelu bez a s vyhlazením zobrazené na Obr. 18.



Obr. 17 Vykreslení bodů aproximace funkce 2 s vyhlazením



Obr. 18 Porovnání odmocniny kvadrátu chyby modelu bez a s vyhlazením

Stejně jako u první testovací funkce i zde je zřejmé, že aproximace pouze za pomoci lineárních modelů danou funkci aproximuje lépe než aproximace s vyhlazením za pomoci funkcí platnosti.

Z výsledků této kapitoly plyne, že diskutovaná realizace dávkového algoritmu Batch-RD je schopna vytvořit uspokojivou aproximaci obecné nelineární funkce více proměnných, tudíž se jeví jako vhodný kandidát pro použití za účelem kalibrace.

5. Modely

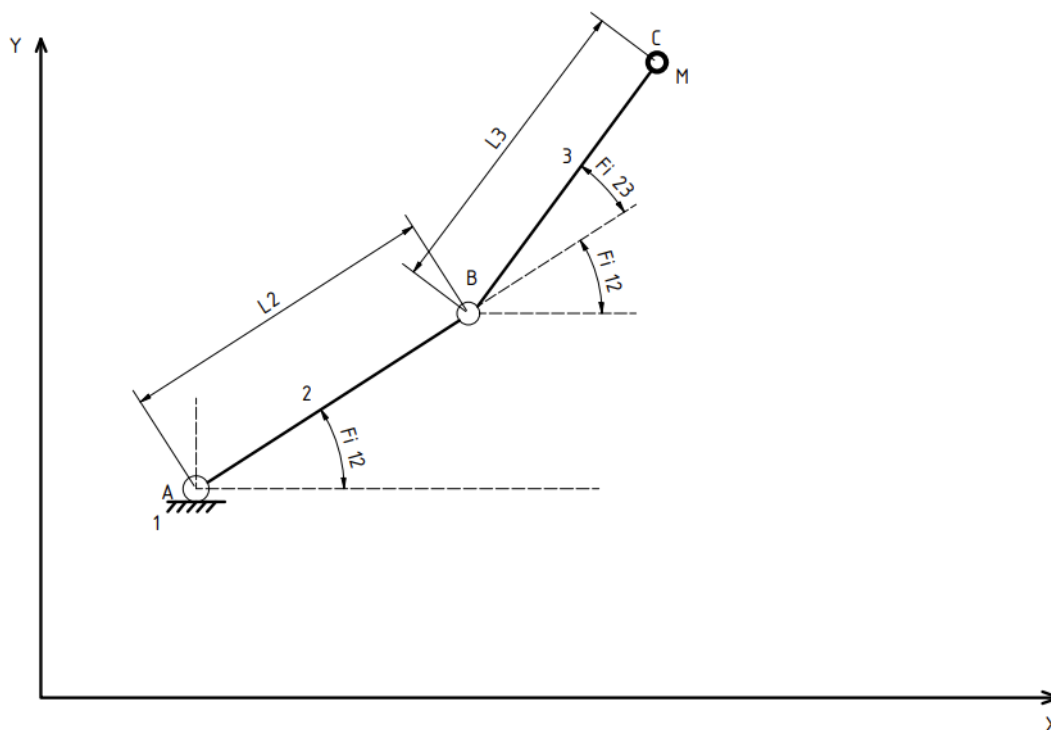
5.1. Rovinné modely

S cílem porovnat různé přístupy kinematické kalibrace bylo potřeba zvolit mechanismus, který namodeluji a otestuji na něm různé postupy. Jelikož se tato práce primárně zabývá kalibrací robotických ramen, bylo na místě zvolit mechanismus se sériovou strukturou. S přihlédnutím k jednoduchosti jsem z této množiny mechanismů zvolil rovinný dvouramenný mechanismus se dvěma stupni volnosti viz schéma na Obr. 19.

Pro následné porovnání budu pracovat se stejnými vstupy modelů ($\varphi_{12} = \left\{ -\frac{\pi}{2} : 0.1 : \frac{\pi}{2} \right\}$, $\varphi_{23} = \left\{ -\frac{3\pi}{4} : 0.1 : \frac{3\pi}{4} \right\}$, tedy s 1536 různými polohami mechanismu). Při vykreslení v grafech je pro jednoduchost φ_{12} značeno jako φ_1 a φ_{23} jako φ_2 .

5.1.1. Nepoddajný kinematický model

Jako první přiblížení jsem vytvořil kinematický model mechanismu zobrazeného na Obr. 19, u kterého jsem uvažoval všechny členy nepoddajné. Jako vstupy modelu jsou uvažovány úhly φ_{12} (natočení tělesa 2 vůči rámu) a φ_{23} (natočení tělesa 3 vůči tělesu 2). Parametry modelu jsou souřadnice bodu A [X_A, Y_A] a délky ramen (L_2, L_3). Výstupem rozumíme souřadnice bodu C [X_C, Y_C].



Obr. 19 Schéma rovinného mechanismu

Pro mechanismus lze zapsat vazbové rovnice (5.1) a (5.2).

$$X_A + L_2 \cdot \cos(\varphi_{12}) + L_3 \cdot \cos(\varphi_{12} + \varphi_{23}) - X_C = 0 \quad (5.1)$$

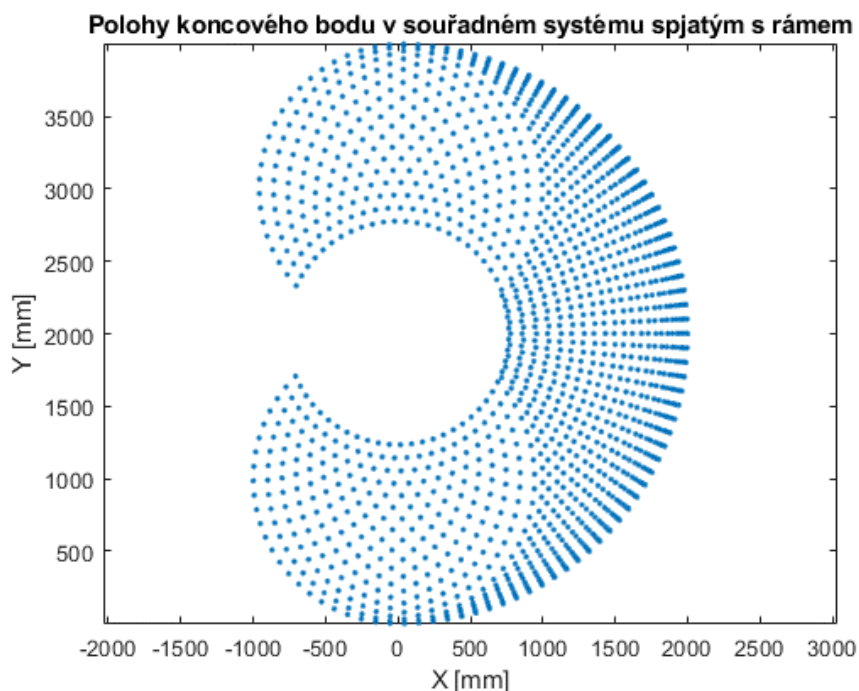
$$Y_A + L_2 \cdot \sin(\varphi_{12}) + L_3 \cdot \sin(\varphi_{12} + \varphi_{23}) - Y_C = 0 \quad (5.2)$$

Po následné úpravě (vyjádření výstupních souřadnic bodu C) platí:

$$X_C = X_A + L_2 \cdot \cos(\varphi_{12}) + L_3 \cdot \cos(\varphi_{12} + \varphi_{23}) \quad (5.3)$$

$$Y_C = Y_A + L_2 \cdot \sin(\varphi_{12}) + L_3 \cdot \sin(\varphi_{12} + \varphi_{23}) \quad (5.4)$$

Pokud se umístí počátek mechanismu (bod A) v souřadném systému spjatým s rámem na souřadnice: $X_A = 2000 \text{ mm}$, $X_B = 0 \text{ mm}$ a délku obou ramen se zvolí: $L_2 = L_3 = 1000 \text{ mm}$, poté lze vykreslit polohy koncového bodu jak je zobrazeno na Obr. 20.

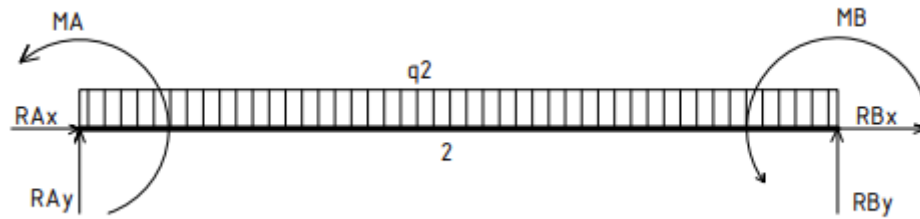


Obr. 20 Polohy koncového bodu z nepoddajného modelu

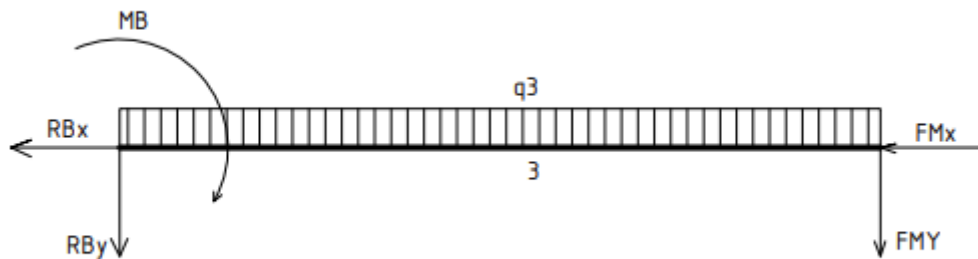
5.1.2. Poddajný kinematický model

V druhém přiblížení jsem potřeboval získat referenční model, který by přesněji popisoval reálné chování robotu. Pro zavedení nelinearit jsem zvolil namodelování mechanismus z Obr. 19 pomocí dvou poddajných nosníků v tíhovém poli (tíhové zrychlení g orientované opačně vůči směru osy Y) s uvažováním hmotností obou členů a přidané hmotnosti M v bodu C. Výstupy z tohoto modelu následně budu chápat pro účel porovnání jako ekvivalent provedeného měření za pomoci externího měřicího přístroje (např. Laser trackeru) viz kapitola 3.1.

Nejdříve je třeba provést uvolnění nosníků viz Obr. 21 a Obr. 22.



Obr. 21 Uvolnění členu 2



Obr. 22 Uvolnění členu 3

Následně je zavedena délková hustota μ_i a spojité zatížení q_i , kde ρ je hustota, A průřez nosníku a α_i natočení i -tého nosníku vůči ose X souřadného systému spjatého s rámem:

$$\mu_i = \rho_i \cdot A_i \quad (5.5)$$

$$q_i = \mu_i \cdot g \cdot \sin(\alpha_i). \quad (5.6)$$

Poté je třeba sestavit rovnice rovnováhy:

$$X: R_{Ax} + R_{Bx} - \sin(\varphi_{12}) \cdot g \cdot \rho_2 \cdot A_2 \cdot L_2 = 0 \quad (5.7)$$

$$Y: R_{Ay} + R_{By} - \cos(\varphi_{12}) \cdot g \cdot \rho_2 \cdot A_2 \cdot L_2 = 0 \quad (5.8)$$

$$M_A: M_A + M_B - \frac{L_2}{2} \cdot \cos(\varphi_{12}) \cdot g \cdot \rho_2 \cdot A_2 \cdot L_2 + L_2 \cdot R_{By} = 0 \quad (5.9)$$

$$X: -R_{Bx} - \sin(\varphi_{12} + \varphi_{23}) \cdot g \cdot \rho_3 \cdot A_3 \cdot L_3 - M \cdot g \cdot \sin(\varphi_{12} + \varphi_{23}) = 0 \quad (5.10)$$

$$Y: -R_{By} - \cos(\varphi_{12} + \varphi_{23}) \cdot g \cdot \rho_2 \cdot A_3 \cdot L_3 - M \cdot g \cdot \cos(\varphi_{12} + \varphi_{23}) = 0 \quad (5.11)$$

$$M_C: -M_B - \frac{L_2}{2} \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot g \cdot \rho_3 \cdot A_3 \cdot L_3 - L_3 \cdot g \cdot M \cdot \cos(\varphi_{12} + \varphi_{23}) = 0. \quad (5.12)$$

Z rovnic rovnováhy se vyjádří reakční účinky:

$$R_{Ax} = M \cdot g \cdot \sin(\varphi_{12} + \varphi_{23}) + g \cdot L_2 \cdot \mu_2 \cdot \sin(\varphi_{12}) + g \cdot L_3 \cdot \mu_3 \cdot \sin(\varphi_{12} + \varphi_{23}) \quad (5.13)$$

$$R_{Ay} = M \cdot g \cdot \cos(\varphi_{12} + \varphi_{23}) + g \cdot L_2 \cdot \mu_2 \cdot \cos(\varphi_{12}) + g \cdot L_3 \cdot \mu_3 \cdot \cos(\varphi_{12} + \varphi_{23}) \quad (5.14)$$

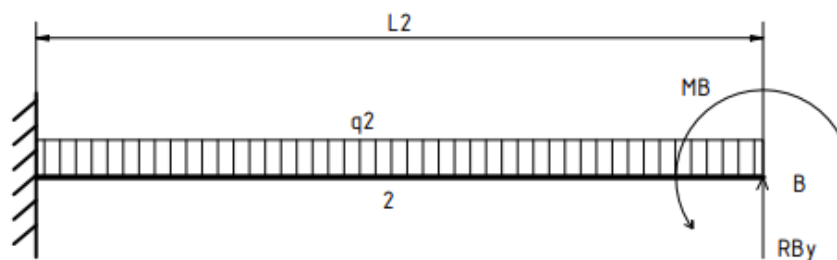
$$\begin{aligned} M_A = & \frac{1}{2} (\cos(\varphi_{12}) \cdot L_2^2 \cdot \mu_2 \cdot g) + g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot \mu_3 \cdot L_2 \cdot L_3 \\ & + M \cdot g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot L_2 + \frac{1}{2} \cdot (\cos(\varphi_{12} + \varphi_{23}) \cdot L_3^2 \cdot \mu_3 \cdot g) \\ & + M \cdot g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot L_3 \end{aligned} \quad (5.15)$$

$$R_{Bx} = -M \cdot g \cdot \sin(\varphi_{12} + \varphi_{23}) - g \cdot \sin(\varphi_{12} + \varphi_{23}) \cdot \mu_3 \cdot L_3 \quad (5.16)$$

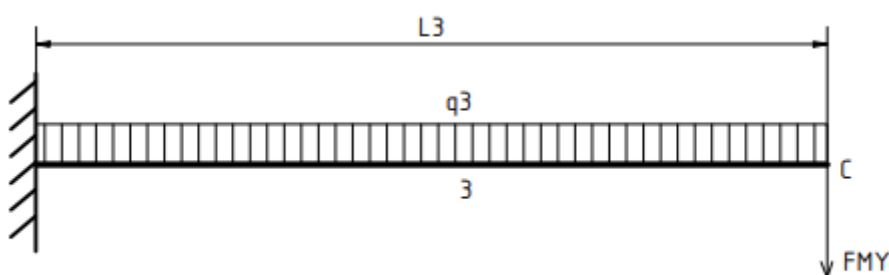
$$R_{By} = -M \cdot g \cdot \cos(\varphi_{12} + \varphi_{23}) - g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot \mu_3 \cdot L_3 \quad (5.17)$$

$$M_B = -M \cdot g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot L_2 - \frac{1}{2} \cdot (\cos(\varphi_{12} + \varphi_{23}) \cdot L_3^2 \cdot \mu_3 \cdot g) \quad (5.18)$$

V tuto chvíli je dáno zatížení nosníků a lze přistoupit k výpočtu deformací. Jelikož je nosník výrazně tužší v axiálním směru, je zde zavedeno zjednodušení a dále ve výpočtu uvažováno pouze ohybové namáhání viz Obr. 23 a Obr. 24.



Obr. 23 Namáhání vetknutého nosníku 2



Obr. 24 Namáhání vetknutého nosníku 3

Pro výpočet je třeba dále zavést modul pružnosti v tahu E a kvadratický moment průřezu k neutrální ose J_z . Poté je možné za pomoci Mohrových integrálů vyjádřit průhyb v a natočení φ členu 2 v bodě B a členu 3 v bodě C .

$$v_B = \frac{1}{E \cdot J_z} \cdot \left[\frac{1}{8} \cdot g \cdot \mu_2 \cdot \cos(\varphi_{12}) \cdot L_2^4 - \frac{1}{3} \cdot R_{By} \cdot L_2^3 - \frac{1}{2} \cdot M_B \cdot L_2^2 \right] \quad (5.19)$$

$$v_C = \frac{1}{E \cdot J_z} \cdot \frac{1}{24} \cdot \left[(g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot L_3^3) \cdot (8M + 3 \cdot L_3 \cdot \mu_3) \right] \quad (5.20)$$

$$\varphi_B = \frac{1}{E \cdot J_z} \cdot \left[-\frac{1}{6} \cdot g \cdot \mu_2 \cdot \cos(\varphi_{12}) \cdot L_2^3 + \frac{1}{2} \cdot R_{By} \cdot L_2^2 + M_B \cdot L_2 \right] \quad (5.21)$$

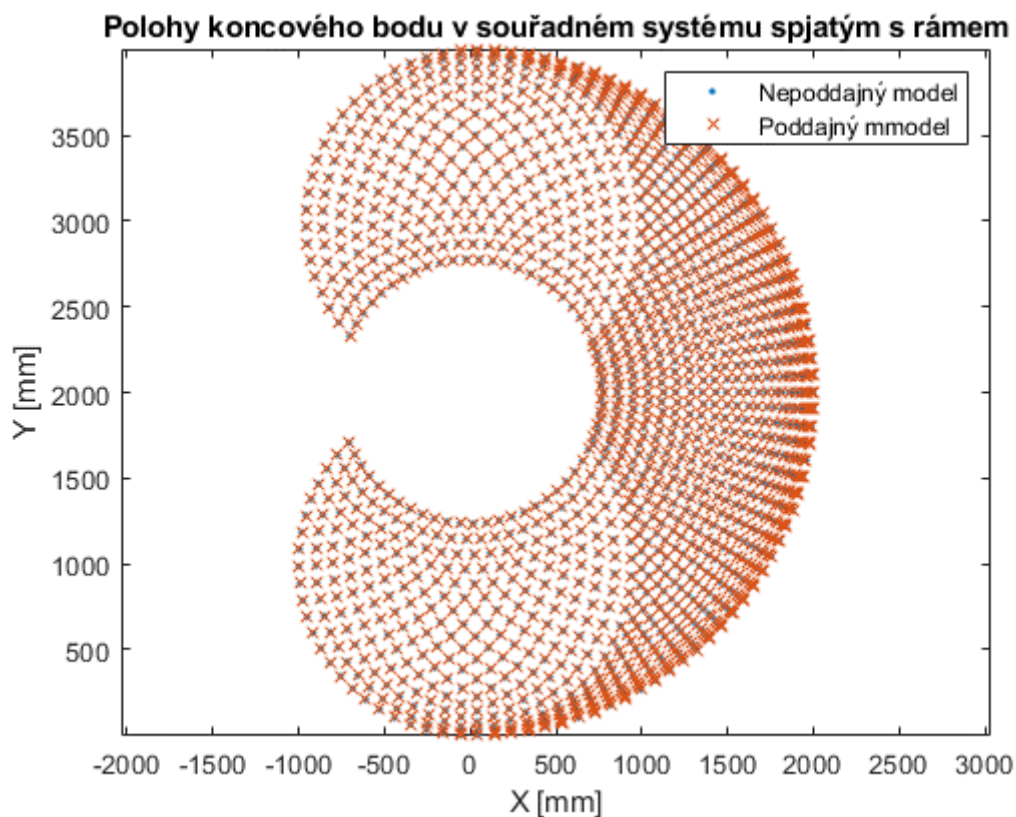
$$\varphi_C = \frac{1}{E \cdot J_z} \cdot \frac{1}{6} \cdot \left[-(g \cdot \cos(\varphi_{12} + \varphi_{23}) \cdot L_3^2) \cdot (3M + L_3 \cdot \mu_3) \right] \quad (5.22)$$

Nyní lze vyjádřit souřadnice koncového bodu C, označené jako \tilde{X}_C a \tilde{Y}_C :

$$\begin{aligned} \tilde{X}_C = X_A + L_2 \cdot \cos(\varphi_{12}) + v_B \cdot \cos\left(\varphi_{12} + \frac{3\pi}{2}\right) + L_3 \cdot \cos(\varphi_{12} + \varphi_{23} + \varphi_B) \\ + v_C \cdot \cos\left(\varphi_{12} + \varphi_{23} + \varphi_B + \frac{3\pi}{2}\right) \end{aligned} \quad (5.23)$$

$$\begin{aligned} \tilde{Y}_C = Y_A + L_2 \cdot \sin(\varphi_{12}) + v_B \cdot \sin\left(\varphi_{12} + \frac{3\pi}{2}\right) + L_3 \cdot \sin(\varphi_{12} + \varphi_{23} + \varphi_B) \\ + v_C \cdot \sin\left(\varphi_{12} + \varphi_{23} + \varphi_B + \frac{3\pi}{2}\right) \end{aligned} \quad (5.24)$$

Dále budu uvažovat, že oba nosníky jsou kruhového průřezu o poloměru $d=100$ mm vyrobené z oceli ($\rho = 7850 \frac{kg}{m^3}$, $E = 210 GPa$), gravitační zrychlení $g = 9,81 \frac{m}{s}$ a hmotnost v koncovém bodě $M = 10 kg$. Poloha koncového bodu pro takto definovaný model je spočtena a zobrazena na Obr. 25, kde je výstup porovnán s nepoddajným modelem.

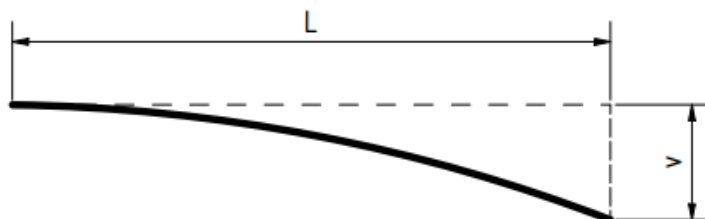


Obr. 25 Polohy koncového bodu z poddajného a nepoddajného modelu

5.1.3. Kalibrovaný nepoddajný kinematický model

Třetí model rovinného mechanismu vychází z kinematického modelu z kapitoly 5.1.1, který je rozšířen o kalibrační parametry. Za pomoci algoritmu základní kalibrace diskutovaném v kapitole 3.3.1 je kalibrační model následně zkalibrován.

Předpokládám-li, že oba členy mechanismu nejsou dokonale rovné, pro potřebu kalibrace můžu nahradit každý z nich strukturou na Obr. 26.



Obr. 26 Nahrazení ramene

Polohu koncového bodu C po kalibraci lze vyjádřit rovnicemi (5.25), (5.26). Tyto rovnice reprezentují kalibrovaný model, ve kterém vystupují kalibrační parametry (**d**): $\widehat{L}_2, \widehat{L}_3, v_2, v_3$ a měřené hodnoty na vnitřních čidlech mechanismu (**s**): $\varphi_{12}, \varphi_{23}$.

$$X_{C\text{ kal}} = X_A + \widehat{L}_2 \cdot \cos(\varphi_{12}) + v_2 \cdot \cos\left(\varphi_{12} + \frac{3\pi}{2}\right) + \widehat{L}_3 \cdot \cos(\varphi_{12} + \varphi_{23}) + v_3 \cdot \cos\left(\varphi_{12} + \varphi_{23} + \frac{3\pi}{2}\right) \quad (5.25)$$

$$Y_{C\text{ kal}} = Y_A + \widehat{L}_2 \cdot \sin(\varphi_{12}) + v_2 \cdot \sin\left(\varphi_{12} + \frac{3\pi}{2}\right) + \widehat{L}_3 \cdot \sin(\varphi_{12} + \varphi_{23}) + v_3 \cdot \sin\left(\varphi_{12} + \varphi_{23} + \frac{3\pi}{2}\right) \quad (5.26)$$

Celý postup kalibrace je podrobně popsán v kapitole 3.3.1, ale je vhodné zde zmínit vazbové rovnice, ze kterých jsem při kalibraci vycházel:

$$f_1 = X_A + \widehat{L}_2 \cdot \cos(\varphi_{12}) + v_2 \cdot \cos\left(\varphi_{12} + \frac{3\pi}{2}\right) + \widehat{L}_3 \cdot \cos(\varphi_{12} + \varphi_{23}) + v_3 \cdot \cos\left(\varphi_{12} + \varphi_{23} + \frac{3\pi}{2}\right) - \tilde{X}_C = 0 \quad (5.27)$$

$$f_2 = Y_A + \widehat{L}_2 \cdot \sin(\varphi_{12}) + v_2 \cdot \sin(\varphi_{12} + \frac{3\pi}{2}) + \widehat{L}_3 \cdot \sin(\varphi_{12} + \varphi_{23})$$

$$+ v_3 \cdot \sin(\varphi_{12} + \varphi_{23} + \frac{3\pi}{2}) - \check{Y}_C = 0 .$$
(5.28)

Hodnoty \check{X}_C a \check{Y}_C (viz kapitola 5.1.2) odpovídají z pohledu kalibrace hodnotám naměřeným externím měřicím přístrojem (parametrům v viz kapitola 3.3.1). Kalibrovatelnost (viz kapitola 3.3.3) pro tuto úlohu vyšla 1,78, což svědčí o velmi dobré podmíněnosti a kalibrovatelnosti úlohy. To se prokázalo i tím, že iterační algoritmus konvergoval k řešení již v druhé iteraci. Kalibrované parametry vyšli následovně: $\widehat{L}_2 = 1000 \text{ mm}$, $\widehat{L}_3 = 999,998 \text{ mm}$, $v_2 = 0,41 \text{ mm}$, $v_3 = 0,36 \text{ mm}$.

5.1.4. Nepoddajný kinematický model s korekcí realizovanou fuzzy modely

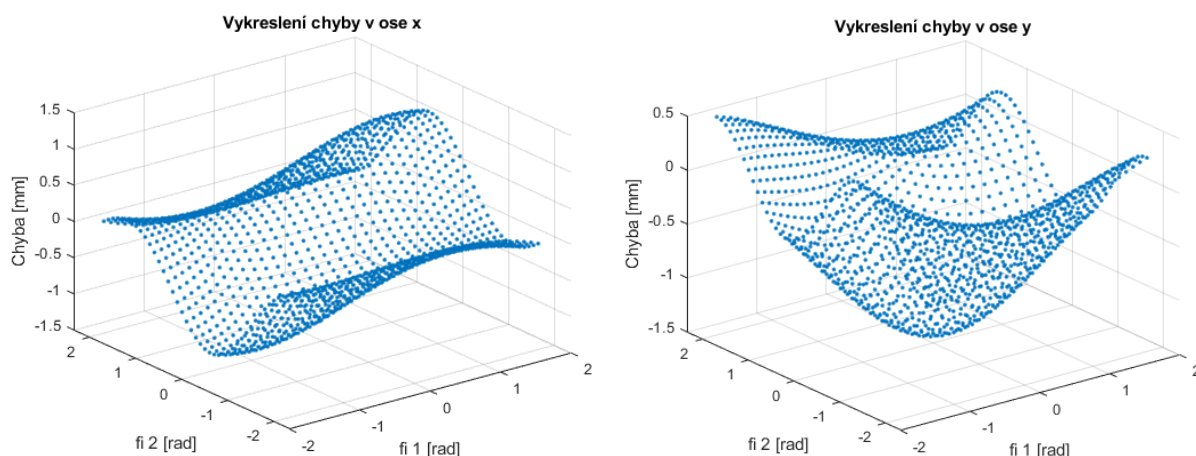
Myšlenkou čtvrtého modelu je využít znalosti o zkoumaném mechanismu v podobě použití nepoddajného modelu z kapitoly 5.1.1 a nad ním modelovat neuro-fuzzy modely chybu vůči poddajnému modelu z kapitoly 5.1.2.

Nejdříve jsem vyjádřil chybové funkce ve směru osy X a Y:

$$e_x(\varphi_{12}, \varphi_{23}) = \check{X}_C - X_C, \tag{5.29}$$

$$e_y(\varphi_{12}, \varphi_{23}) = \check{Y}_C - Y_C. \tag{5.30}$$

Poté jsem pro vstupní hodnoty φ_{12} a φ_{23} vygeneroval hodnoty chyby (viz Obr. 27), které jsem poté použil k vytvoření aproximačních modelů za použití lineárních modelových stromů.



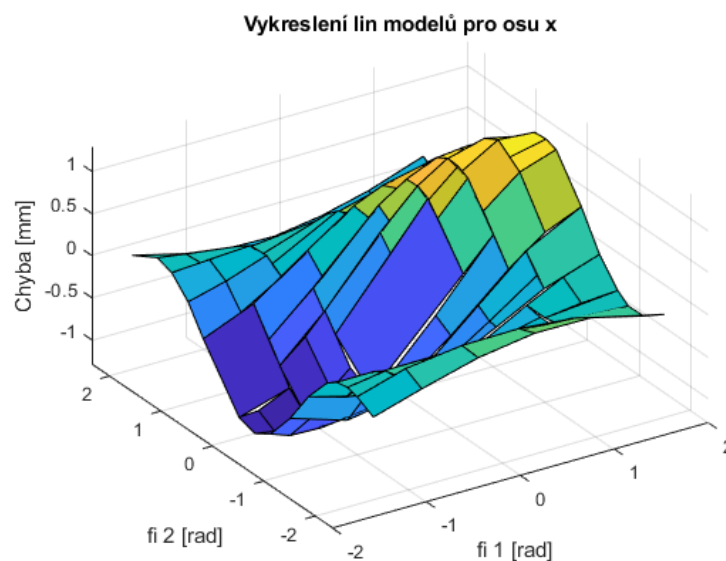
Obr. 27 Vykreslení chyby v ose X a Y

Označím-li výstup aproximace chyby jednotlivých os, napočtených z lineárních modelových stromů, jako $\widetilde{e}_x(\varphi_{12}, \varphi_{23})$ a $\widetilde{e}_y(\varphi_{12}, \varphi_{23})$ mohu poté výstupní polohy bodu C zapsat jako:

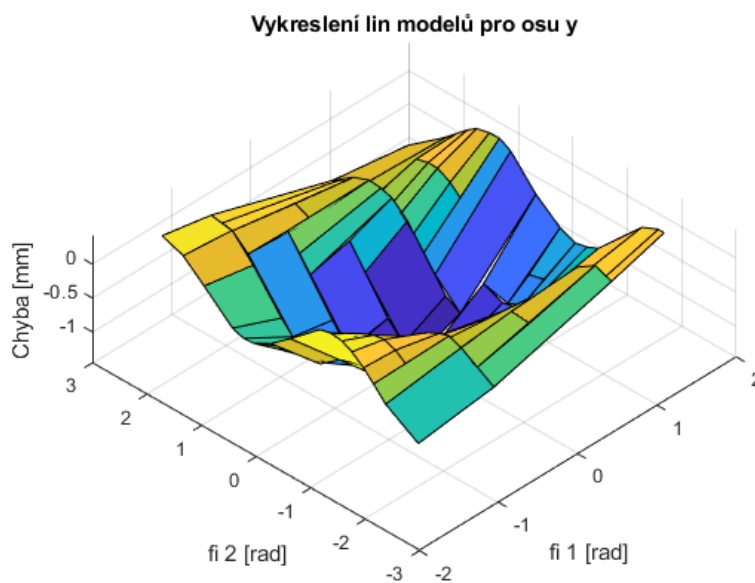
$$X_{C\text{ komb.}}(\varphi_{12}, \varphi_{23}) = X_C(\varphi_{12}, \varphi_{23}) + \widetilde{e}_x(\varphi_{12}, \varphi_{23}), \quad (5.31)$$

$$Y_{C\text{ komb.}}(\varphi_{12}, \varphi_{23}) = Y_C(\varphi_{12}, \varphi_{23}) + \widetilde{e}_y(\varphi_{12}, \varphi_{23}). \quad (5.32)$$

Algoritmus napočítal aproximaci funkce chyby e_x pomocí 93 modelů a aproximaci funkce chyby e_y pomocí 75 modelů (viz Obr. 28 a Obr. 29).



Obr. 28 Vykreslení lineárních modelů aproximujících chybu v ose X (LLMT model)

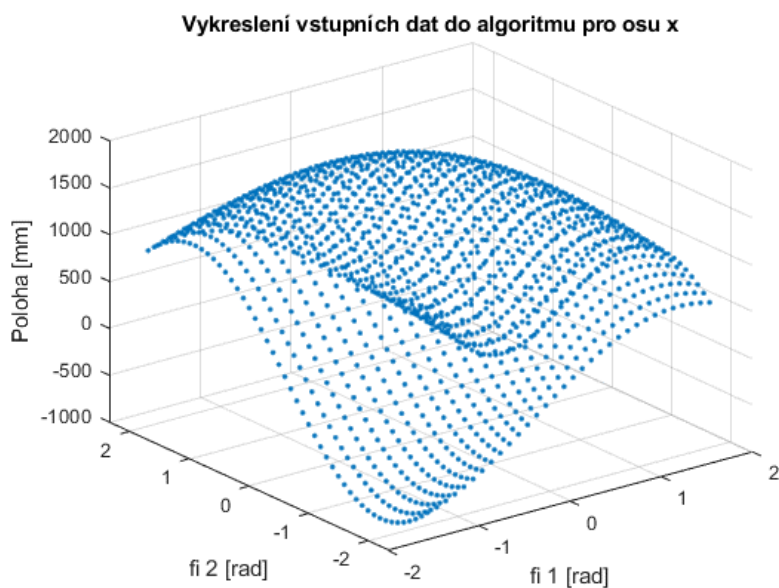


Obr. 29 Vykreslení lineárních modelů aproximujících chybu v ose Y (LLMT model)

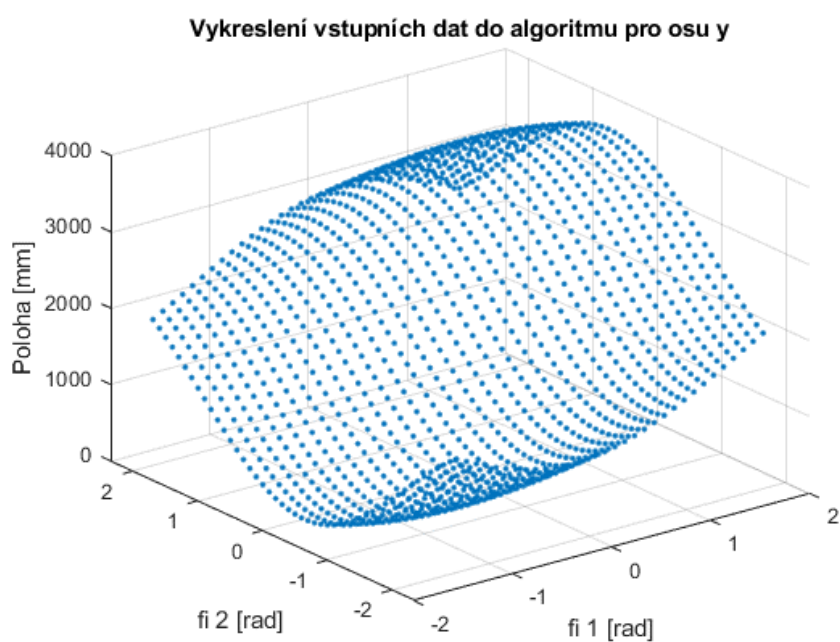
5.1.5. Poddajný kinematický model realizovaný pouze fuzzy modely

Poslední z modelů rovinného mechanismu využívá „black box“ přístupu, kdy se nepracuje s žádnými znalostmi o modelované struktuře. Snahou je aproximovat celý poddajný model z kapitoly 5.1.2 jen ze znalosti jeho vstupů a výstupů.

Nejdříve jsem vygeneroval pro zadané polohy pohonů mechanismu ($\varphi_{12}, \varphi_{23}$) podle rovnic (3.23) a (3.24) polohy koncového bodu C. Napočtené polohy jsou zobrazeny na Obr. 30 a Obr. 31.

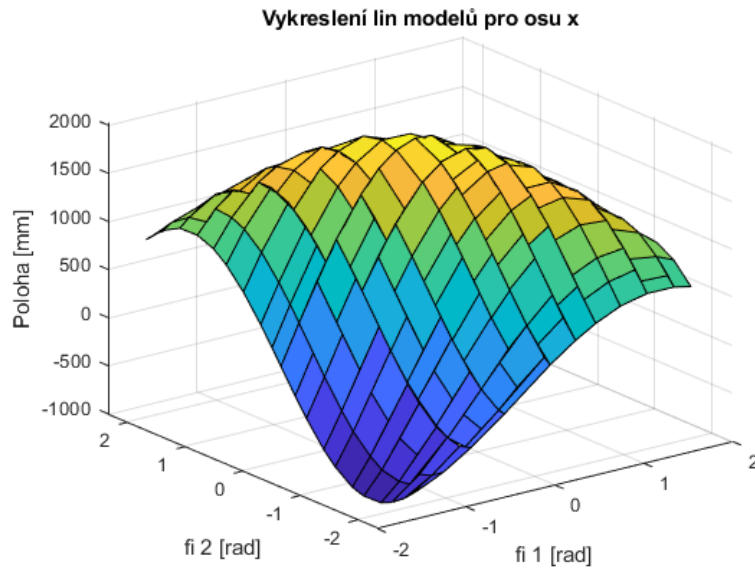


Obr. 30 Polohy koncového bodu v ose X

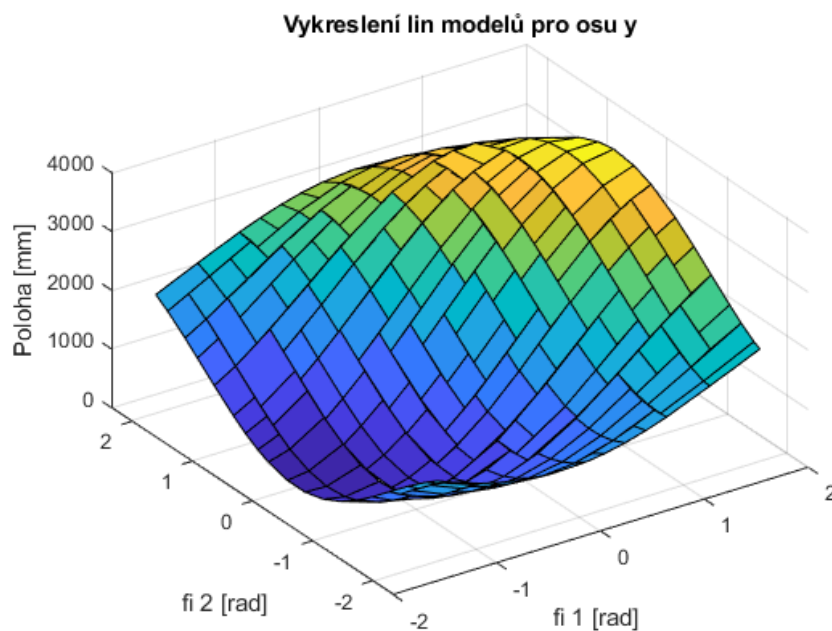


Obr. 31 Polohy koncového bodu v ose Y

Tyto polohy jsem dále využil jako vstup (učící datovou sadu) do algoritmu, který vygeneroval aproximace za použití lineárních modelových stromů. Polohu bodu C v ose X algoritmus approximoval 211 lineárními modely a polohu v ose Y approximoval 209 lineárními modely. Celkové modely jsou vyobrazeny na Obr. 32. a Obr. 33.



Obr. 32 Aproximace pomocí LLMT modelu v ose X

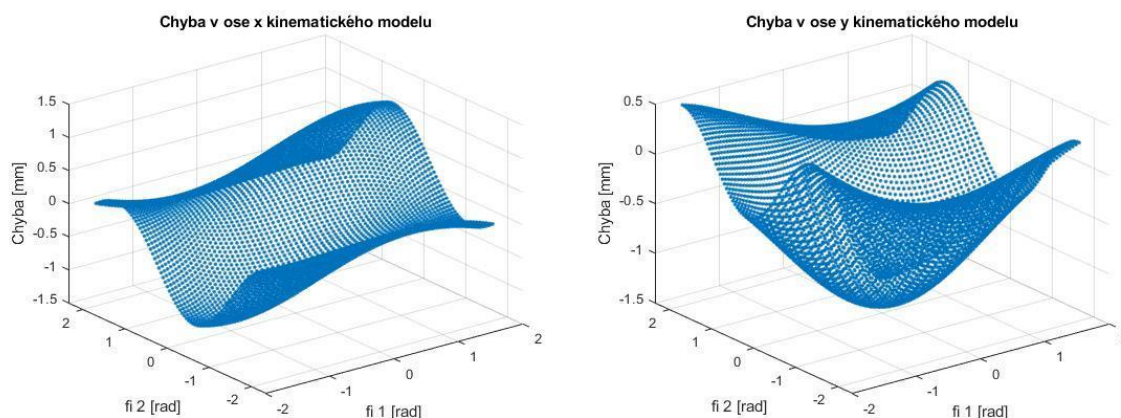


Obr. 33 Aproximace pomocí LLMT modelu v ose Y

5.1.6. Porovnání modelů

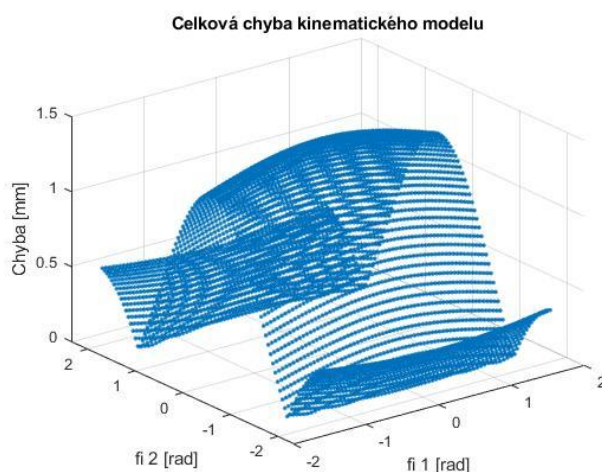
Závěrem kapitoly 5.1 je porovnání jednotlivých přístupů ke kalibraci (chyby jednotlivých modelů vůči poddajnému modelu, jehož výstupy uvažuji za přesné). K vytvoření modelů byla použita jedna datová sada (učící sada 1536 vzorků). K porovnání přesnosti modelů je vhodné použít jinou datovou sadu (testovací sadu). Kdyby se provedlo porovnání na učící datové sadě, získala by se jen informace o úspěšnosti aproximace v daných bodech. Já jsem jako testovací datovou sadu zvolil $\varphi_{12} = \left\{ -\frac{\pi}{2} : 0.05 : \frac{\pi}{2} \right\}$, $\varphi_{23} = \left\{ -\frac{3\pi}{4} : 0.05 : \frac{3\pi}{4} \right\}$, tedy 5985 vzorků odpovídajících stejnému počtu poloh mechanismu. Učící datová sada je tedy podmnožinou datové sady testovací. Pro zjednodušení zápisu je zaveden pojem „celková chyba“ e_{celk} . Pokud se v této kapitole hovoří o celkové chybě, jedná se o odmocninu ze součtu kvadrátu vzdáleností: $e_{celk} = \sqrt{(\tilde{X}_c - X_c)^2 + (\tilde{Y}_c - Y_c)^2}$.

Nejdříve je diskutováno porovnání nepoddajného kinematického modelu (kapitola 5.1.1) vůči modelu poddajnému (kapitola 5.1.2). Vyjádření chyby v jednotlivých osách je zobrazeno na Obr. 34.



Obr. 34 Chyba nepoddajného kinematického modelu vůči poddajnému modelu

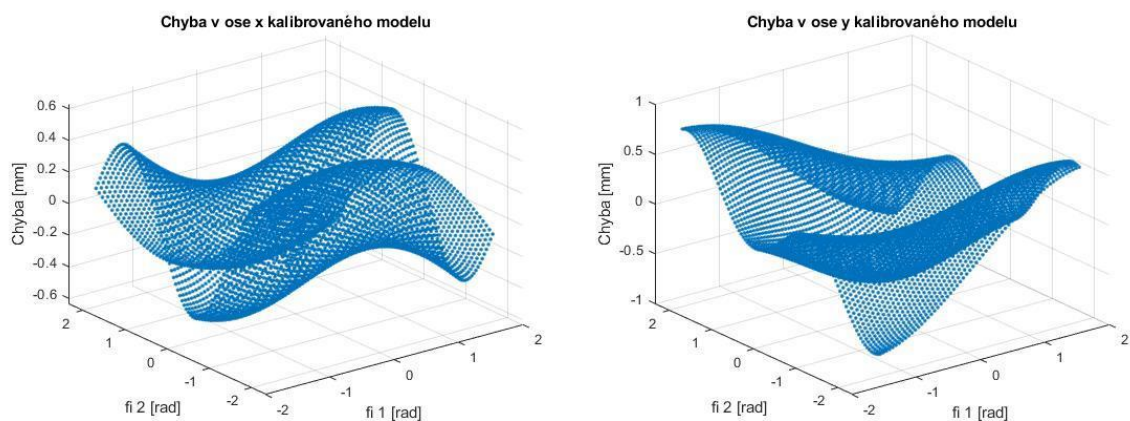
Celková chyba je vykreslena na Obr. 35.



Obr. 35 Celková chyba kinematického modelu vůči poddajnému modelu

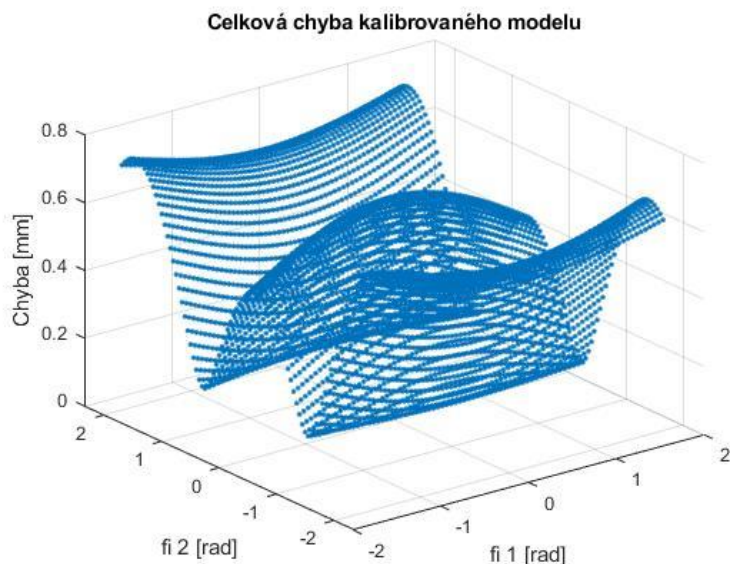
Průměrná celková chyba nepoddajného modelu na celé oblasti je $\overline{e_{celk\ kin}} = 0,643\ mm$ a maximální hodnota chyby na oblasti je $1,427\ mm$.

Druhé porovnání zkoumá chybu kalibrovaného modelu (kapitola 5.1.3) vůči modelu poddajnému (kapitola 5.1.2). Vykreslení chyby v jednotlivých osách je zobrazeno na Obr. 36.



Obr. 36 Chyba kalibrovaného kinematického modelu vůči poddajnému modelu

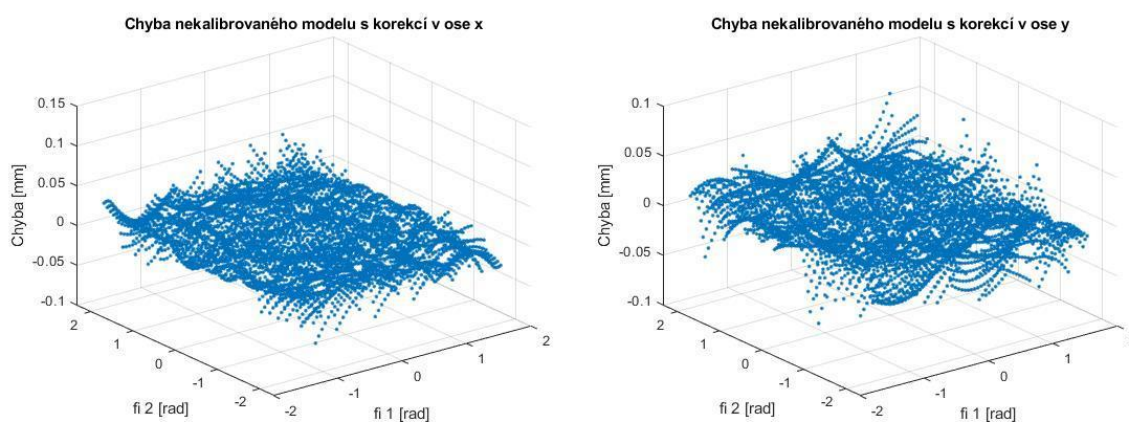
Celková chyba modelu je vykreslena níže na Obr. 37.



Obr. 37 Celková chyba kalibrovaného kinematického modelu vůči poddajnému modelu

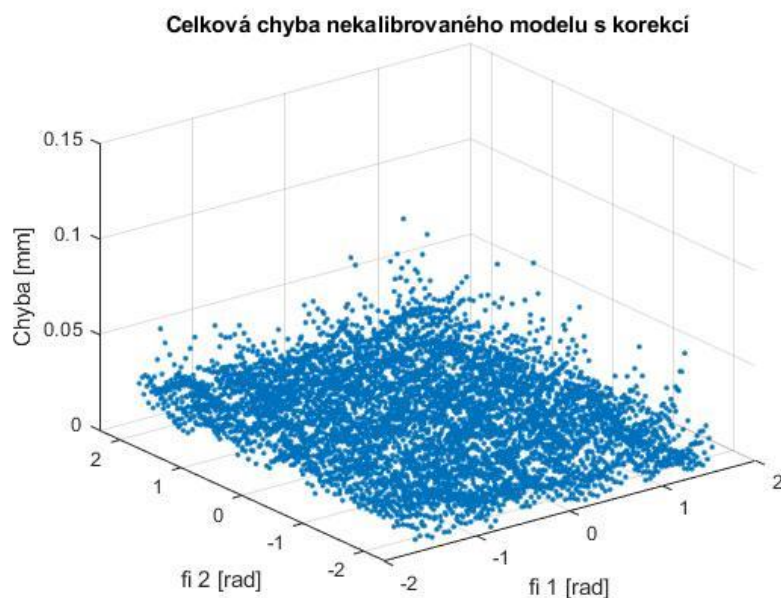
Průměrná celková chyba kalibrovaného modelu na celé oblasti je $\overline{e_{celk\ kal}} = 0,447\ mm$ a maximální hodnota chyby na oblasti je $0,716\ mm$.

Třetí porovnání zkoumá chybu nepoddajného modelu s korekcí realizovanou LLMT a fuzzy modely (kapitola 5.1.4) vůči modelu poddajnému (kapitola 5.1.2). Vykreslení chyby v jednotlivých osách je zobrazeno na Obr. 38.



Obr. 38 Chyba nepoddajného modelu s korekcí realizovanou fuzzy modely vůči poddajnému modelu

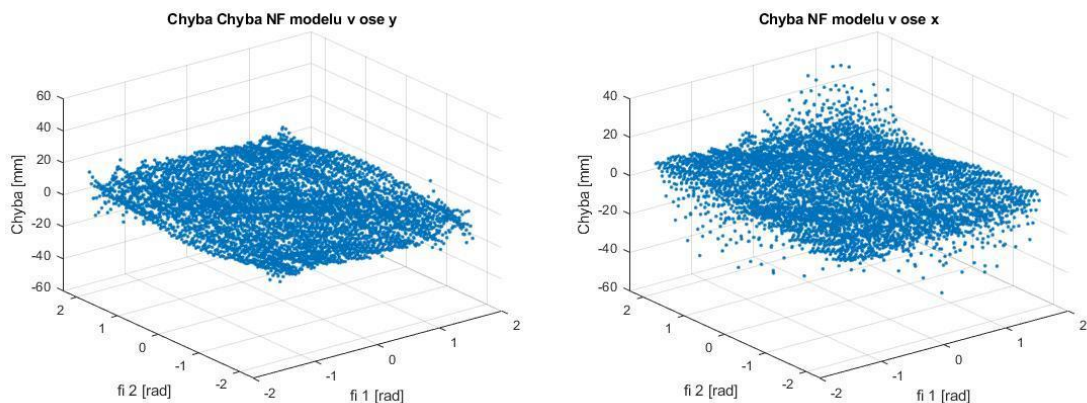
Celková chyba modelu je vykreslena níže na Obr. 39.



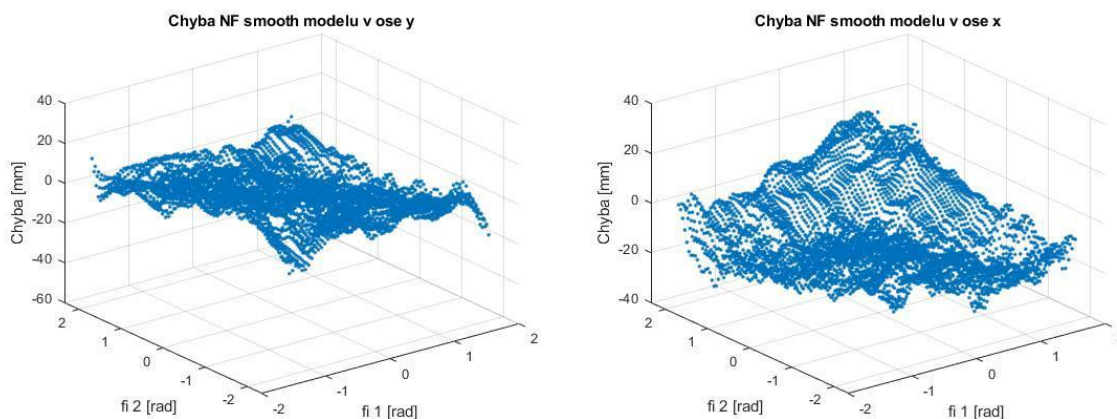
Obr. 39 Celková chyba nepoddajného modelu s korekcí realizovanou fuzzy modely vůči poddajnému modelu

Průměrná celková chyba tohoto modelu na celé oblasti je $\overline{e_{celk\ komb}} = 0,0157\ mm$ a maximální hodnota chyby na oblasti je $0,131\ mm$.

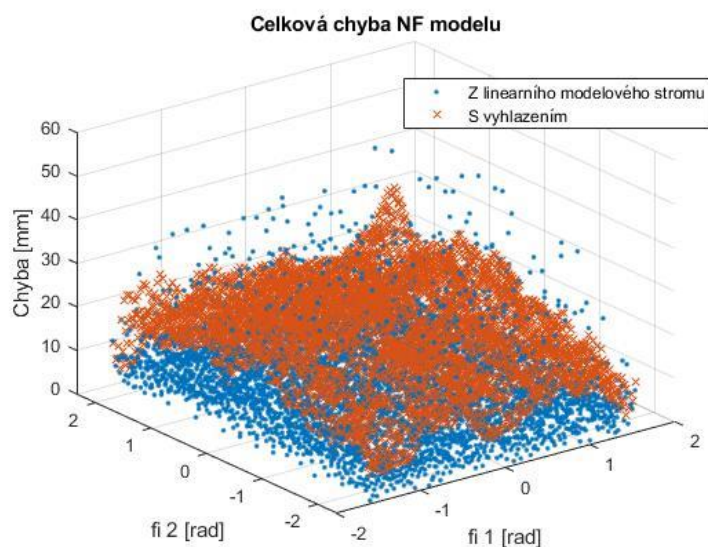
Čtvrté a poslední porovnání se zaměřuje na chybu poddajného modelu realizovaného fuzzy modely (kapitola 5.1.5) vůči poddajnému modelu (kapitola 5.1.2). Zobrazení chyby v jednotlivých osách za použití lineárních modelových stromů (LLMT) je na Obr. 40 a za použití lineárních modelových stromů s vyhlazením na Obr. 41. Celková chyba modelů je zobrazena na Obr. 42.



Obr. 41 Chyba modelu LLMT vůči poddajnému modelu



Obr. 40 Chyba modelu LLMT s vyhlazením vůči poddajnému modelu



Obr. 42 Celková chyba modelu LLMT s a bez vyhlazení vůči poddajnému modelu

Průměrná celková chyba LLMT modelu na celé oblasti je $\overline{e_{celk\ LLMT}} = 7,8766\text{ mm}$ a maximální hodnota chyby na oblasti je $42,8\text{ mm}$. Průměrná celková chyba na celé oblasti LLMT modelu s vyhlazením je $\overline{e_{celk\ LLMT\ vyhl}} = 19,1\text{ mm}$ a maximální hodnota chyby na oblasti je $52,09\text{ mm}$.

Tabulka 1 Srovnání chyby modelů vůči poddajnému modelu

Model	Průměrná celková chyba [mm]	Maximální chyba [mm]
Kinematický nepoddajný	0,643	1,427
Kinematický kalibrovaný	0,547	0,716
Kinematický + fuzzy korekce (LLMT)	0,0157	0,131
Fuzzy model – LLMT bez vyhlazení	7,877	42,8
Fuzzy model – LLMT s vyhlazením	19,1	52,09

Výsledky porovnání chyby modelů jsou shrnuty v Tabulka 1. Na chybu jednotlivých modelů je nutné pohlížet v širším kontextu. Na první pohled se zdá, že s přehledem nejlepší výsledky, vykazuje model, který kombinuje nepoddajný kinematický model s fuzzy korekcí za pomoci LLMT modelů. Toto je pravda, pokud uvažujeme dopřednou kinematickou úlohu. V tomto případě se tento model jeví jako velmi silný nástroj, vhodně využitelný například pro simulování poloh mechanismu, kde dosáhl více jak o řád vyšší přesnosti než kalibrovaný kinematický model. Na druhou stranu je ale otázkou, do jaké míry bychom byli schopni využít informace zahrnuté v tomto modelu při řešení inverzní kinematické úlohy, případně při snaze využít tento model v rámci řízení. Jako druhý nejlepší se jeví kalibrovaný kinematický model, který je hojně užíván a je prokázáno, že jej lze vhodně použít i v rámci řešení inverzní kinematické úlohy a řízení. Jen o trochu hůře dopadl nekalibrovaný kinematický model. Jako nejméně vhodná se jeví snaha modelovat celý mechanismus za pomoci neuro-fuzzy modelu (LLMT), jelikož dosáhl přesnosti, která je pro reálnou aplikaci nedostatečná, a navíc u něj vyvstává stejná otázka ohledně použitelnosti k řešení inverzní kinematické úlohy či řízení jako u nepoddajného kinematického modelu s aproximací chyby za pomoci LLMT.

Dále je třeba zmínit, že přesnost modelů je závislá i na počtu vzorků, na kterých byl model trénován či kalibrován. K sestavení kinematického modelu není třeba znát žádnou polohu v prostoru a stačí pouze znalost geometrie mechanismu z výkresové dokumentace. Ke kalibraci mechanismu je třeba znát geometrii mechanismu, a navíc i mít znalost o poloze koncového bodu minimálně v tolika konfiguracích mechanismu, kolik je kalibrovaných rozměrů (spíše v jejich násobku). Tyto body však nemusí být nutně rozloženy rovnoměrně v celém pracovním prostoru mechanismu. K vytvoření přesného neuro-fuzzy modelu (myšleno modelu celé kinematiky zahrnující deformace atd.) je třeba řádově většího množství vzorků rozložených rovnoměrně v celém pracovním prostoru, které by bylo velmi náročné a nákladné získat. Z takového rozsáhlého data setu, pro dosažení kžžené přesnosti, by se poté musely generovat mnohem komplexnější modely, které by byly výpočetně náročnější a nepoužitelné pro výpočet v reálném čase.

V předmětné práci jsem pro potřebu porovnání použil pro všechny modely stejné množství učicích vzorků. Toto množství bylo primárně voleno pro rozumnou aproximaci chyby nad kinematickým modelem, čímž jsou zde uvedená porovnání také zatížena. Kdyby se použilo řádově více učicích vzorků pro sestavení modelu diskutovaného v kapitole 5.1.5, vygenerovaly by se výrazně větší lineární modelové stromy (obsahující více modelů), a tím by bylo dosaženo i vyšší přesnosti. Například pokud jsem ve stejném prostoru vytvořil data set 148 680 učicích vzorků, kompenzace polohy v ose X byla aproximována 3283 modely a v ose Y 3331 modely. Pak průměrná celková chyba LLMT modelu na celé oblasti byla $\overline{e_{celk\ LLMT}} = 0,406\text{ mm}$ a maximální hodnota chyby na oblasti byla $3,299\text{ mm}$. Průměrná celková chyba na celé oblasti LLMT modelu s vyhlazením byla $\overline{e_{celk\ LLMT\ vyhl}} = 0,989\text{ mm}$ a maximální hodnota chyby na oblasti byla $2,413\text{ mm}$. Tyto výsledky se nejvíce podobají výstupům kinematického nepoddajného modelu, ale výpočetní náročnost je pro takto rozsáhlý LLMT model znatelně vyšší.

Pro účel řešení dopředné úlohy se aproximace chyby LLMT modely jeví jako vhodný nástroj zvyšující přesnost celkového modelu bez výrazného zvýšení výpočetní náročnosti. Při použití LLMT modelů v rámci řešení inverzní úlohy narážíme na problémy zmíněné výše. Pokud však není v použité aplikaci vyžadován výpočet inverzní úlohy v reálném čase, například při plánování trajektorie, je možné spočítat trajektorii pomocí kinematického modelu a pomocí LLMT modelu spočítat chyby na této trajektorii. Tyto chyby poté mohou být použity v řídicím systému robotu při numerickém iteračním řešení inverzní kinematické úlohy. V reálné aplikaci není inverzní kinematická úloha obecně řešitelná analyticky, tudíž je v řídicím systému robotu realizována numerickým iteračním výpočtem bez ohledu na to, zda je použita korekce chyby pomocí LLMT modelu.

5.2. Prostorové modely

V této kapitole je popsáno, jak se vytváří kinematické modely dopředné kinematiky prostorových robotů. Obecným cílem analýzy dopředné kinematické úlohy je nalezení vztahů vyjadřujících polohu a orientaci nástroje případně koncového bodu robota (end efektoru) jako funkci proměnných poloh a orientací pohonů. Robotické rameno se skládá z členů propojených kinematickými vazbami. Obvykle se jedná o vazby s jedním stupněm volnosti jako jsou rotační či posuvné vazby. Budeme-li dále uvažovat pouze kinematické dvojice s jedním stupněm volnosti, neznamená to, že dochází ke ztrátě obecnosti popisu, jelikož kinematické dvojice s více stupni volnosti jsou nahraditelné složením více kinematických dvojic s jedním stupněm volnosti. Například sférická kinematická dvojice je nahraditelná třemi rotačními kinematickými dvojicemi propojenými „virtuálními“ členy délky nula. Po této úvaze lze dojít k závěru, že účinek každé vazby se dá popsat právě jedním parametrem, reálným číslem (úhlem natočení pro rotační vazbu či posunutím pro posuvnou vazbu). Cílem analýzy dopředné kinematiky je určit kumulativní účinek těchto parametrů na polohu koncového bodu robota.

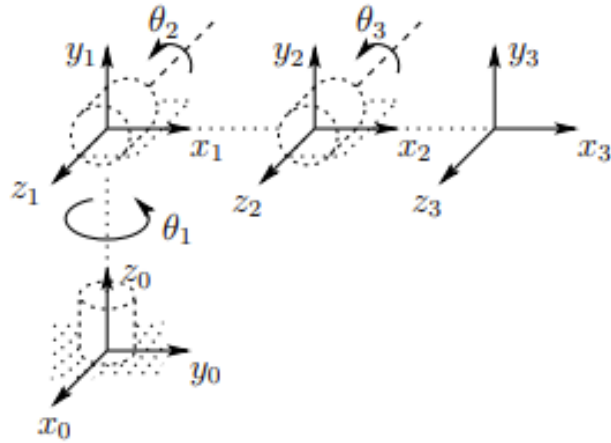
Dále jsou popsány konvence, které vedou k systematickému postupu této analýzy. Je možné kinematickou analýzu provést i bez těchto konvencí, jak je vidět například v kapitole 5.1.1 u rovinného mechanismu. U prostorových mechanismů však úloha s počtem členů rychle nabývá na složitosti a bez systematického přístupu se může stát téměř neřešitelnou. Další výhodou užívání těchto konvencí je vznik jistého univerzálního „jazyka“ či popisu, kterému je snazší porozumět.

Robotický manipulátor s $n+1$ členy (links) a n vazbami (joints). Předpokládá se, že každá vazba propojuje právě dva členy. Vazby jsou očíslovány 1 až n a členy 0 až n , kde člen 0 značí rám, který se uvažuje nepohyblivý. Z této konvence vyplývá, že vazba i propojuje členy $i-1$ a i . S každou i -tou vazbou je poté spojen vazbový parametr q_i viz:

$$q_i = \begin{cases} \theta_i & : \text{rotační } i - \text{tá vazba} \\ d_i & : \text{posuvná } i - \text{tá vazba} \end{cases} \quad (5.33)$$

5.2.1. Kinematický popis sledující strukturu robotu

K provedení kinematické analýzy se nejdříve ke každému členu mechanismu pevně přiřadí jeho souřadnicový systém. Ve výsledku dostaneme pro každý člen i souřadnicový systém O_i, x_i, y_i, z_i , což znamená, že poloha libovolného bodu ležícího na členu i a vyjádřená v i -tém souřadnicovém systému je konstantní pro libovolnou konfiguraci robota. Souřadnicový systém O_0, x_0, y_0, z_0 , ke kterému je robot připojen je zároveň pevně spjatý s rámem. Tato myšlenka přiřazení souřadnicových systémů a zde zmiňovaného popisu je ilustrována na Obr. 43.



Obr. 43 Ukázka připojení souřadnicových systémů ke kloubovému manipulátoru (převzato z [12])

Dále je třeba zavést homogenní transformační matice A_i (pro $i = 1, \dots, n$), které reprezentují pozici a natočení souřadnicového systému O_i, x_i, y_i, z_i vůči souřadnicovému systému $O_{i-1}, x_{i-1}, y_{i-1}, z_{i-1}$. Tyto transformační matice A_i budou obecně závislé na konkrétní konfiguraci robotu, ale díky předpokladu, že uvažujeme pouze vazby s jedním stupněm volnosti, budou matice A_i ve skutečnosti vždy funkcí pouze jedné proměnné:

$$A_i = A_i(q_i) \quad (5.34)$$

Obecně bývá zvykem homogenní transformační matice vyjadřující pozici a natočení souřadnicového O_j, x_j, y_j, z_j vůči souřadnicovému systému O_i, x_i, y_i, z_i označovat jako *transformační matice* T_j^i , které lze vyjádřit následovně:

$$T_j^i = A_{i+1}A_{i+2} \cdots A_{j-1}A_j \text{ pro } i < j \quad (5.35)$$

$$T_j^i = I \text{ pro } i = j \quad (5.36)$$

$$T_j^i = (T_j^i)^{-1} \text{ pro } i > j \quad (5.37)$$

Ze způsobu, jakým jsme zvolili připojení souřadnicových systémů k jednotlivým členům vyplývá, že poloha libovolného bodu ležícího na end efektoru vyjádřená v n -tém souřadnicovém systému je konstantní pro libovolnou konfiguraci robotu. Označí-li se poloha počátku tohoto n -tého souřadnicového systému v souřadnicovém systému spjatým s rámem jako vektor O_n^0 (3×1) a obdobně matice popisující natočení těchto systémů R_n^0 (3×3), lze definovat homogenní transformační matici H ($\mathbf{0}$ značí vektor nul (1×3)):

$$H = \begin{bmatrix} R_n^0 & O_n^0 \\ \mathbf{0} & 1 \end{bmatrix}. \quad (5.38)$$

Poté je poloha a natočení end efektoru vůči rámu dáno jako:

$$H = T_n^0 = A_1(q_1)A_2(q_2) \cdots A_n(q_n). \quad (5.39)$$

Každá z matic A_i má strukturu:

$$A_i = \begin{bmatrix} R_i^{i-1} & O_i^{i-1} \\ \mathbf{0} & 1 \end{bmatrix}. \quad (5.40)$$

Tudíž lze podle rovnice (5.35) psát:

$$T_j^i = A_{i+1}A_{i+2} \cdots A_{j-1}A_j = \begin{bmatrix} R_j^i & O_j^i \\ \mathbf{0} & 1 \end{bmatrix}, \quad (5.41)$$

kde matice R_j^i vyjadřuje relativní orientaci souřadnicového O_j, x_j, y_j, z_j vůči souřadnicovému systému O_i, x_i, y_i, z_i a lze ji vyjádřit pomocí částí matic A_i jako:

$$R_j^i = R_{i+1}^i R_{i+2}^{i+1} \cdots R_j^{j-1}. \quad (5.42)$$

Obdobně i vektor O_j^i lze rekurzivně vyjádřit ze vzorce:

$$O_j^i = O_{j-1}^i + R_{j-1}^i O_j^{j-1}. \quad (5.43)$$

Chceme-li tedy popsat pozici koncového bodu robotu B v souřadnicovém systému spjatým s rámem ($r_B^0 = [x_B^0 \ y_B^0 \ z_B^0 \ 1]^T$), který je v n -tém souřadnicovém systému (spjatým s nástrojem) vyjádřen vektorem $r_B^n = [x_B^n \ y_B^n \ z_B^n \ 1]^T$ lze psát:

$$r_B^0 = T_n^0 r_B^n. \quad (5.44)$$

Toto je v podstatě veškerý potřebný aparát pro řešení úlohy dopředné kinematiky. Stačí tedy vhodně umístit souřadné systémy, vyjádřit matice $A_i(q_i)$ a podle potřeby je mezi sebou pronásobit a získat transformační matici T_n^0 . Tento postup je vhodný pro vytvoření kinematického kalibračního modelu, jelikož umožňuje dobrý náhled do procesu díky tomu, že jsou souřadnicové systémy umístěny na členech robotu. Obecně platí, že tímto způsobem vytvořené kalibrační modely mají ve většině případů dobrou podmíněnost (viz kapitola 3.3.3).

5.2.2. Denavit – Hartenbergova konvence

Ke kinematickému popisu robotů postačuje popis sledující strukturu robotu (viz kapitola 5.2.1), nicméně existují i přístupy, kterými se dá dosáhnout značného zefektivnění a zjednodušení. Jedním z často používaných přístupů je tzv. Denavit – Hartenbergova konvence (D-H konvence či notace), pojmenovaná podle svých autorů (Jacques Denavit, Richard Hartenberg), kteří ji poprvé popsali v článku [11] publikovaném roku 1955.

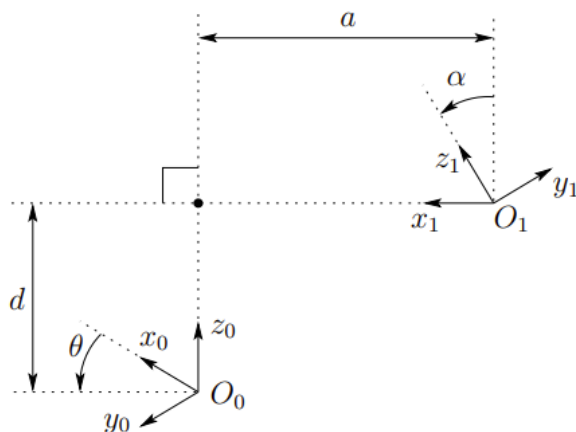
Snahou je systematizovat volbu umístění souřadnicových systémů. Dle této konvence je každá homogenní transformace A_i reprezentována složením čtyř základních transformací [viz rovnice (5.45)].

$$\begin{aligned}
 A_i &= Rotace_{z,\theta_i} Translace_{z,d_i} Translace_{x,a_i} Rotace_{x,\alpha_i} = & (5.45) \\
 &= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

kde čtyři parametry θ_i , a_i , d_i , α_i (tzv. Denavit – Hartenbergovy parametry) jsou svázány s členem i a vazbou i . Matice A_i je funkcí jedné proměnné, a tak pro každou vazbu jsou tři z těchto parametrů konstantní.

Může se zdát zvláštní, že k popisu jsou použity pouze čtyři parametry, když obecná homogenní transformace v prostoru by měla být určena parametry šesti. To je zapříčiněno tím, že není vyžadováno, aby počátky souřadných systémů ležely na členech mechanismu, ale mohou ležet volně v prostoru jen za splnění podmínky, že jsou členy propojeny. Právě vhodnou volbou počátků souřadných systémů se dá snížit počet parametrů z šesti na čtyři. Obecně toto není možné pro libovolnou homogenní transformaci. Otázkou tedy je, které transformace je možné zapsat ve tvaru (5.45).

Mějme dva souřadné systémy 0 a 1, poté existuje homogenní transformace A mezi nimi. Předpokládejme, další dvě podmínky: (DH 1) osa x_1 je kolmá na osu z_0 a (DH 2) Osa x_1 protíná osu z_0 . V [12] je dokázáno, že za splnění podmínek DH 1 a DH 2 existuje jednoznačné a , d , θ , α (viz Obr. 44) dostačující k určení libovolné homogenní transformace zapsatelné ve tvaru (5.45).



Obr. 44 Souřadnicové systémy splňující podmínky DH 1 a DH 2 (převzato z [12])

Poté je možné vyjádřit fyzickou interpretaci DH parametrů v matici A_i : a_i reprezentuje vzdálenost mezi osami z_{i-1} a z_i měřené podle osy x_i , úhel α_i mezi osami z_{i-1} a z_i kladný směr je uvažován od osy z_{i-1} k z_i , d_i je vzdálenost mezi počátkem O_{i-1} a průsečíkem os x_i , z_{i-1} měřená v tečné rovině k z_{i-1} a θ_i je úhel mezi osami x_{i-1} , x_i měřený v tečné rovině k ose z_{i-1} . Tato fyzická interpretace je základem při umísťování souřadnicových systémů, tak aby byly splněny podmínky DH 1 a DH2.

U libovolného robotického ramene je možné umístit souřadnicové systémy $0, 1, \dots, n$ takovým způsobem, aby byly podmínky uvedené výše splněny. (Někdy to může znamenat umístění některých z počátků souřadnicových systémů do neintuitivních poloh.) Důležité je si uvědomit, že splnění podmínek DH1 a DH2 při popisu konkrétního robotického ramene může vést na více variant rozmístění souřadnicových systémů. (Výsledné transformační matice T_n^0 však budou totožné.)

Rozmístění souřadnicových systémů, začneme umístěním os $z_0 \dots z_{n-1}$ intuitivním způsobem (volbou a_i a α_i lze získat libovolný směr osy z_i). Konkrétně přiřadíme osu z_i tak, aby byla osou $i+1$ vazby. Mohou nastat dva případy: a) pokud je $i+1$ vazba rotační, pak je osa z_i osou rotace této vazby; b) je-li $i+1$ vazba posuvná, pak osa z_i je umístěna ve směru posunutí. (Asociace $i+1$ vazby s i -tým souřadnicovým systémem je obdobná, jako v kapitole 5.2.1.). Následně umístíme souřadnicový systém spjatý s rámem tak, aby jeho počátek O_0 ležel na ose z_0 a aby osy x_0, y_0 byly umístěny libovolným způsobem při zachování pravotočivosti systému. Následně už další souřadnicové systémy umísťujeme iterativním způsobem (i -tý souřadnicový systém vychází z umístění $i-1$ souřadnicového systému). Při umísťování i -tého souřadnicového systému mohou nastat tři případy: a) osy z_i a z_{i-1} jsou mimoběžné, b) osy z_i a z_{i-1} jsou rovnoběžné, c) osy z_i a z_{i-1} jsou různoběžné. Nyní probereme tyto případy jednotlivě.

- a) **Osy z_i a z_{i-1} jsou mimoběžné:** poté existuje právě jedna příčka, úsečka kolmá na z_i i z_{i-1} , která poté definuje polohu osy x_i . Bod, kde tato osa protíná osu z_i , je poté počátkem O_i souřadnicového systému, který je dokončen volbou osy y_i tak, aby byl tento souřadnicový systém pravotočivý.
- b) **Osy z_i a z_{i-1} jsou rovnoběžné:** Pokud jsou osy z_i a z_{i-1} rovnoběžné, pak je mezi nimi nekonečné množství společných normálových rovin, tudíž podmínka DH 1 není schopna určit polohu osy x_i . V tomto případě můžeme zvolit počátek souřadnicového systému O_i libovolně na ose z_i . Toto umístění bývá voleno tak, aby pokud možno co nejvíce zjednodušilo výsledné vztahy. Takovou vhodnou volbou O_i je případ, kdy osa x_i je normálou z_i procházející skrze O_{i-1} (v tomto případě je $d_i = 0$). Když je zvolena osa x_i , zbývá umístit osu y_i tak, aby byl tento souřadnicový systém pravotočivý. Jelikož jsou osy z_i a z_{i-1} rovnoběžné, parametr α_i je roven nule.
- c) **Osy z_i a z_{i-1} jsou různoběžné:** V tomto případě je poloha osy x_i volena jako normála k rovině definované osami z_i a z_{i-1} . (Kladný směr volíme libovolně.) Počátek O_i je přirozeně umístit v průsečíku os z_i a z_{i-1} (v tomto případě bude parametr a_i roven nule). Libovolný bod ležící na ose z_i by ovšem také vyhovoval. Když je zvolena osa x_i a počátek O_i , zbývá jen umístit osu y_i tak, aby byl tento souřadnicový systém pravotočivý.

Zde popsaný postup funguje pro souřadnicové systémy $1, \dots, n - 1$ (0 je volen). Zbývá dodefinovat souřadnicový systém n , spjatý s end efektořem či nástrojem. U v současné době používaných robotických ramen je obvykle poslední pohyb θ_n vykonáván rotační vazbou end efektoru kolem osy z_{n-1} . V tomto případě je transformace mezi posledními dvěma souřadnicovými systémy ($n - 1$ a n) rotace kolem osy z_{n-1} o úhel θ_n následována posunutím podél osy z_{n-1} o vzdálenost d_n (může být i v opačném pořadí).

Z popsaného postupu jsme schopni vyjádřit u všech vazeb i parametry $\theta_i, a_i, d_i, \alpha_i$. Parametry a_i a α_i jsou vždy konstantní, jsou-li všechny vazby rotační, pak je tomu tak i u parametrů d_i a jedinou proměnou jsou úhly θ_i . Poté jsme schopni z rovnice (5.45) vyjádřit jednotlivé homogenní transformační matice A_i . Následně je možné dle rovnice (5.41) vyjádřit T_n^0 a zapsat celkový kinematický model rovnicí (5.44) stejným způsobem jako v předešlé kapitole 5.2.1.

[12]

Hlavní výhodou DH popisu je jeho větší efektivita, proto je hojně užíván k výpočtu inverzní kinematické úlohy v řídicích systémech robotů. Nespornými výhodami také jsou systematizace při vytváření kinematického popisu a univerzálnost. Značnou nevýhodou takto vytvořeného kinematického modelu je jeho špatná kalibrovatelnost, viz [13]. (Obzvláště problematické je, když jsou některé z os z_i a z_{i-1} téměř rovnoběžné, což zapříčiní posouvání příčky mezi osami.). Z tohoto důvodu je pro potřebu kalibrace vhodné vytvořit kalibrační model dle kapitoly 5.2.1, provést kalibraci a až poté, je-li to třeba, zkalibrovaný model transformovat do DH notace. Alternativou je pro účel kalibrace použití modifikovanou DH notaci.

6. Měření na reálném šestiosém robotu

Díky kontaktům Odboru mechaniky a mechatroniky (FS ČVUT, Ústav mechaniky, biomechaniky a mechatroniky) mně byla poskytnuta data z měření, provedených za účelem kalibrace šestiosého robotu Stäubli TX200.



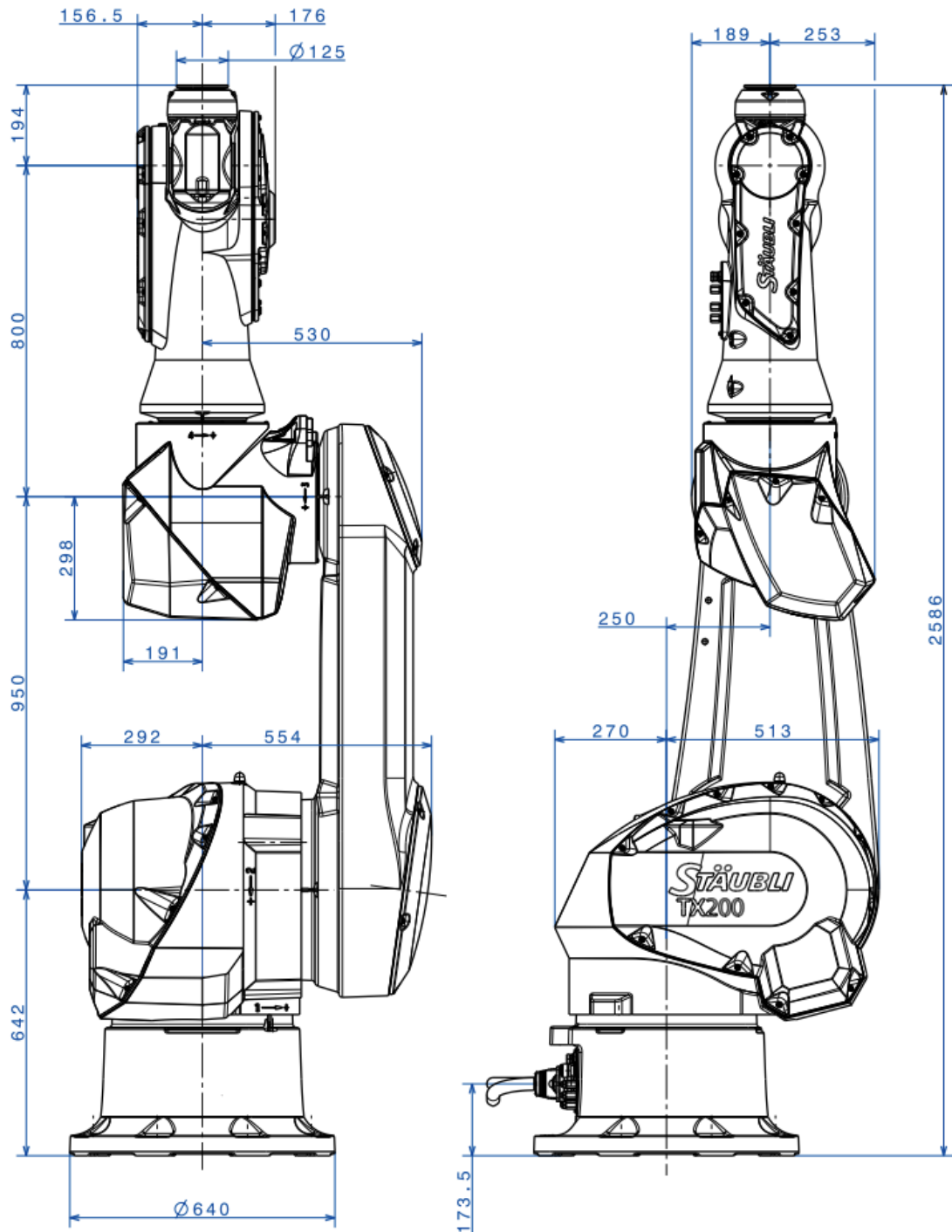
Obr. 45 Ukázka robotu ze série Stäubli TX200 [14]

6.1. Průmyslový robot Stäubli TX200

Průmyslový robot Stäubli TX200 je plně zakrytovaný šestiosý angulární robot s maximálním užitným zatížením 130 kg a maximálním dosahem 2194 mm vážící 980 kg. Výrobce uvádí, že by měl dosahovat opakovatelnosti dle ISO 9283 $\pm 0,06$ mm. Okótované schéma tohoto robotu je zobrazeno na Obr. 46.

Tabulka 2 DH parametry robotu TX200

Název	α [°]	a [mm]	d [mm]	θ [°]
1	0	0	642	0
2	-89,9955	249,2886	450,9658	-90,0587
3	0,0192	950,007	-450	89,9319
4	90,0423	0,6907	800,94	0,0045
5	-90,0598	-0,3995	0	-0,0045
6	90,0352	0,2086	194	0



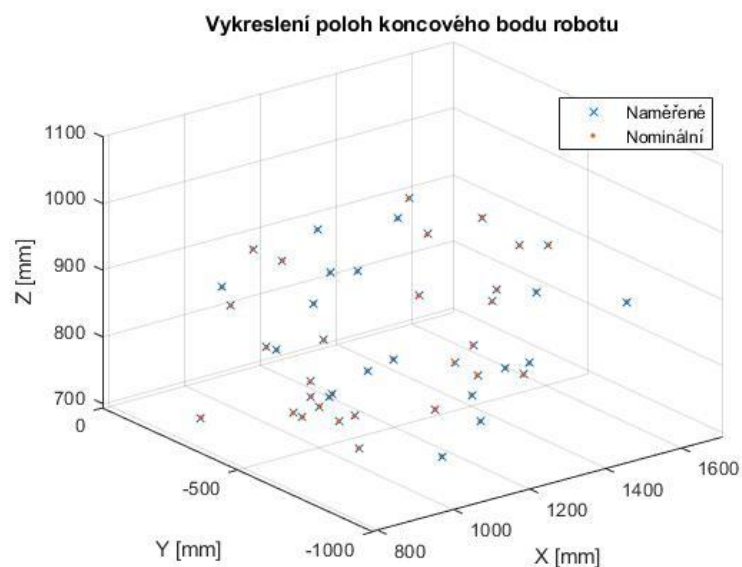
Obr. 46 Okótované schéma Staubli TX200 [14]

Inverzní kinematická úloha je v řídicím systému tohoto robotu realizována pomocí modelu vytvořeném dle DH konvence. Konkrétní DH parametry z řídicího systému jsou uvedeny v Tabulka 2. Pro upřesnění je nutné uvést, že pod parametrem θ je zde uveden lehce odlišný údaj, než v kapitole 5.2.2. V kapitole 5.2.2 je pod symbolem θ_i chápán celý úhel mezi osami x_{i-1} , x_i měřený v tečné rovině k ose z_{i-1} , zatímco zde by se tento údaj dal popsat jako offset počátku senzoru snímajícího úhel natočení. Parametr θ dle kapitoly 5.2.2 je součtem parametru uvedeného v této

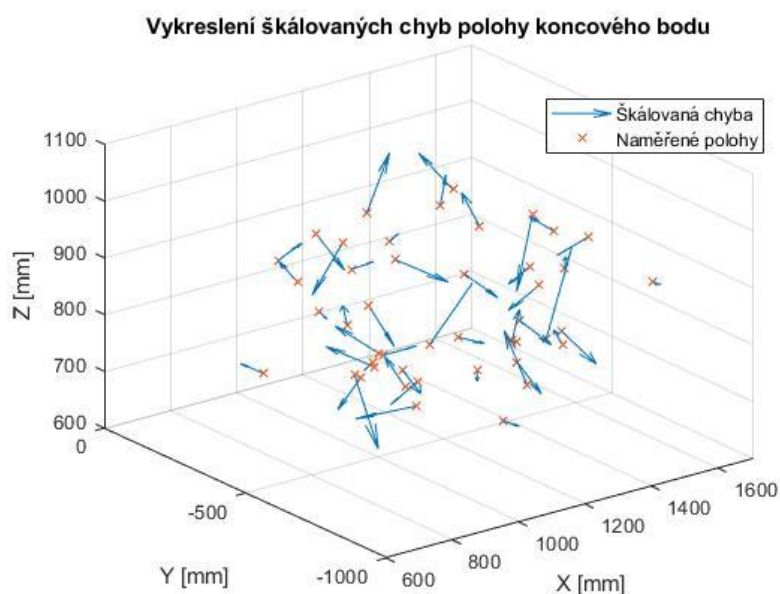
kapitole a úhlu odměřeného senzorem. Při bližším pohledu na Obr. 46 si lze povšimnout, že DH parametry a, d (viz Tabulka 2) sledují rozměry mechanismu.

6.2. Naměřená data

Měření robotu Stäubli TX200 bylo provedeno externím měřicím zařízením laser tracker (viz kapitola 3.2.1) v 45 různých konfiguracích. Jako výstup tohoto měření jsem pro každou konfiguraci obdržel: pozici koncového bodu naměřenou laser trakarem, úhly natočení vnitřních pohonů (odečtenou na vnitřních senzorech) a nominální polohu koncového bodu (poloha napočtená řídicím systémem robotu z úhlů natočení pohonů). Polohy koncového bodu jsou zobrazeny na Obr. 47. Pro lepší představu jsou na Obr. 48 vykresleny škálované vektory chyb mezi naměřenými a nominálními polohami koncového bodu.



Obr. 47 Polohy koncového bodu robotu (robot Stäubli TX200)



Obr. 48 Škálované chyby koncového bodu (robot Stäubli TX200)

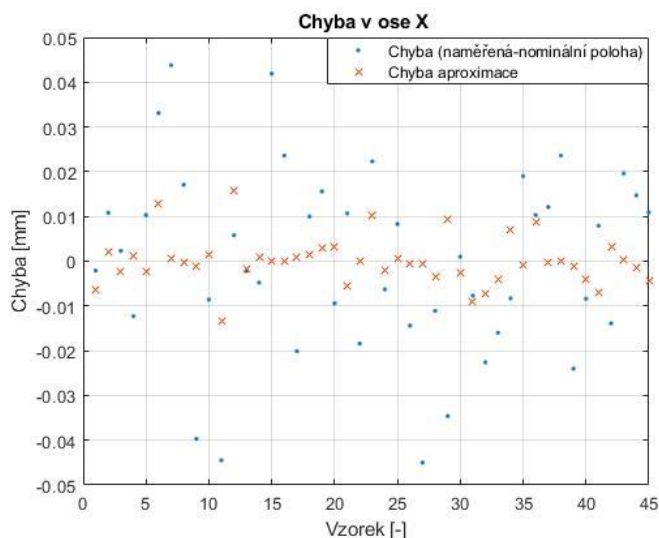
Rozložení chyb v prostoru se může na první pohled jevit velmi chaotické, ale po krátké úvaze je zřejmé, že rozložení chyb není ani tak funkcí polohy koncového bodu v prostoru, jako funkcí konkrétní konfigurace (natočení pohonů).

6.3. Aproximace chyby

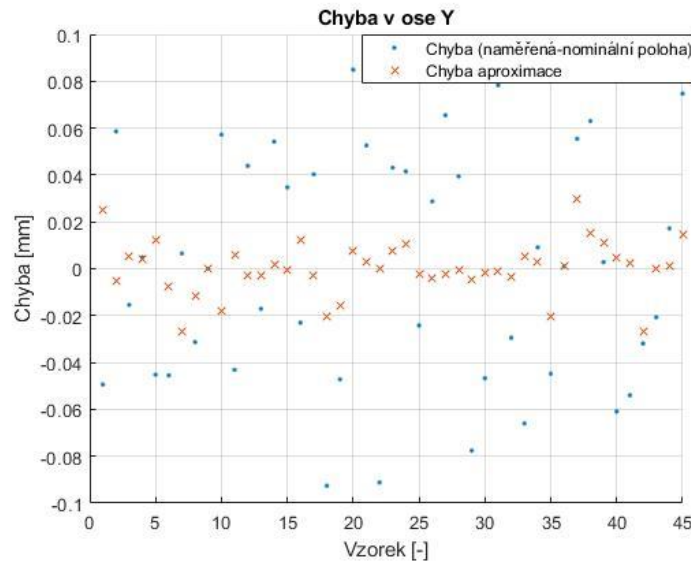
Vezmeme-li tedy naměřené polohy koncového bodu robotu jako přesné (nebudeme-li uvažovat chybu měření), poté lze chybu brát jako rozdíl naměřené a nominální polohy. Dá se tedy uvažovat, že tato chyba je v prostoru popsitelná neznámou funkcí natočení pohonů.

Proč se tedy tuto funkci nepokusit aproximovat neuro-fuzzy modely vytvořenými za pomoci algoritmu popsaném v kapitole 4? Takže stejně jako v kapitole 5.1.4 je nutné pro každou osu (X , Y , Z) vytvořit model aproximující chybu v daném směru. Jedinou drobnou komplikací je, že algoritmus je primárně určen pro práci s datovými sety obsahující velké množství vzorků. Výstup z měření robotu Stäubli TX200 poskytuje ale pouze 45 učicích vzorků. Z tohoto důvodu je nutné provést drobné úpravy týkající se pravidla zastavení (viz kapitola 4.1) a dělicího pravidla (viz kapitola 4.2). Jak již bylo zmíněno v kapitole 4 pro vytvoření přesného modelu z malé učicí sady se osvědčilo použít testování většího počtu potenciálních bodů dělení κ (parametr k). Bonferroniho korekce společně s pravidlem dělení zajišťují, aby nevznikaly zbytečná dělení. Pro malou učicí sadu může toto být nežádoucí. Tudíž je potřeba při výpočtu α_{split} snížit mezní pravděpodobnost pro vyloučení hypotézy H_0 (např. z 95 % na 70 %). Pravidlo zastavení dělení je složeno ze dvou podmínek: jedné týkající se dosažení přesnosti aproximace a druhé hovořící o počtu vzorků v podoblasti. Právě druhá podmínka: „K dělení dochází pouze je-li v každém nově vytvořeném uzlu alespoň trojnásobek počtu vzorků (N_l a N_r), než je parametrů lineárních modelů d .“ je v tomto případě velmi omezující (konkrétně $d=7$, tudíž v každé podoblasti by muselo být 21 vzorků). Zde se osvědčilo tuto podmínku přeformulovat na: „K dělení dochází pouze, je-li v každém nově vytvořeném uzlu alespoň $d+1$ vzorků ($N_l > d+1 \wedge N_r > d+1$).“

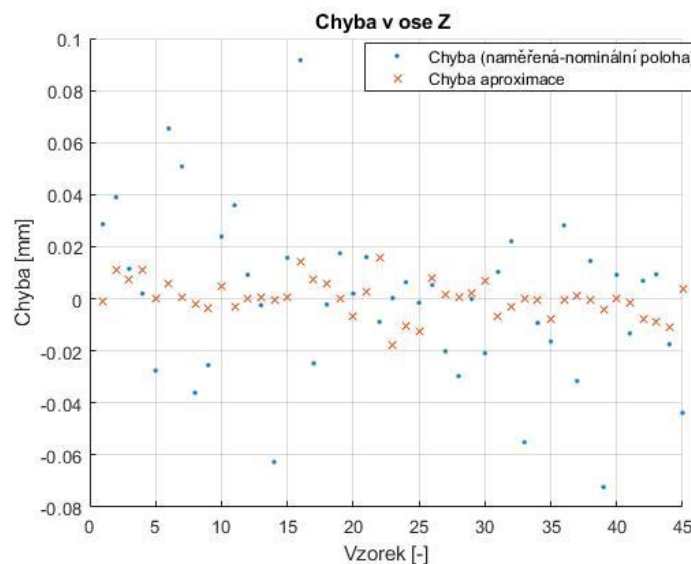
Po těchto úpravách jsem byl schopen vytvořit tři modely (modely aproximující chybu ve směru os X , Y , Z), každý skládající se z pěti podoblastí a lineárních modelů. Chyby aproximace těchto modelů jsou zobrazeny na Obr. 49, Obr. 50 a Obr. 51.



Obr. 49 Aproximace chyby v ose X (robot Stäubli TX200)

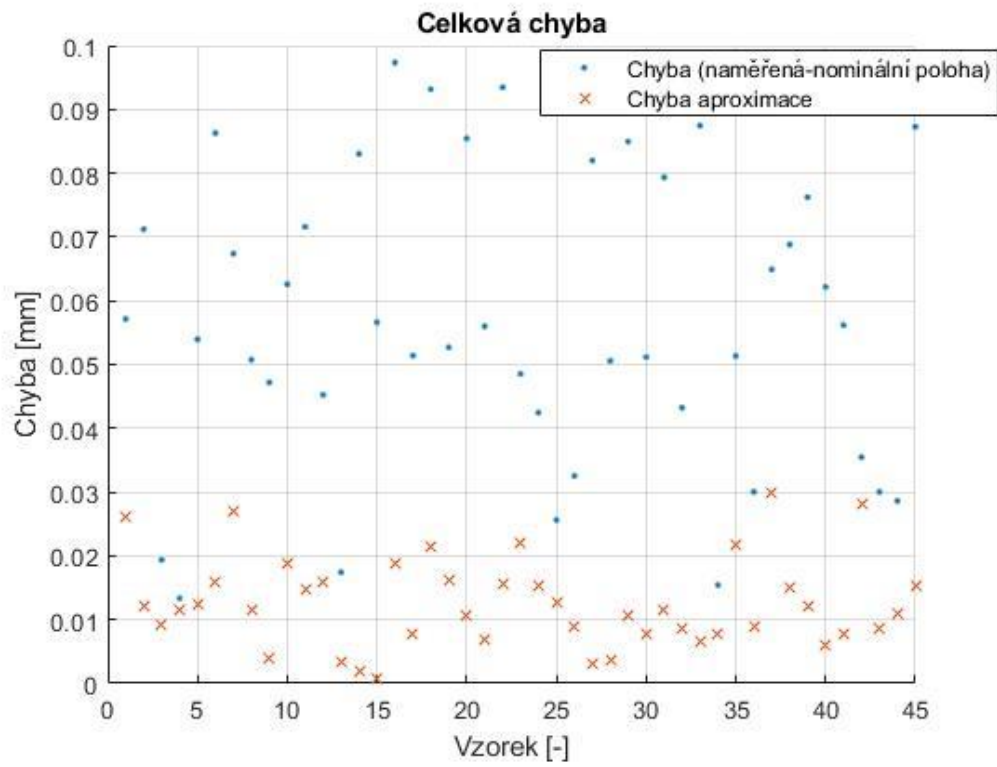


Obr. 50 Aproximace chyby v ose Y (robot Stäubli TX200)



Obr. 51 Aproximace chyby v ose Z (robot Stäubli TX200)

Porovnání celkové chyby nominálních poloh (myšleno ve smyslu vzdálenosti naměřené a nominální polohy) s celkovou chybou s korekcí neuro-fuzzy modely poté vypadá následovně: průměrná celková chyba nominálních poloh je 0,057 mm, průměrná celková chyba s korekcí LLMT modely je 0,0126 mm a průměrná celková chyba s korekcí LLMT modely s vyhlazením 0,0214 mm. Vykreslení celkové chyby nominálních poloh a chyby poloh s korekcí LLMT modely (celkové chyby aproximace) je na Obr. 52. V tomto obrázku si lze povšimnout, že užitím korekce LLMT modely došlo ke zlepšení přesnosti ve všech 45 měřených bodech. Z údajů uvedených v tomto odstavci tedy vyplývá, že užitím korekcí LLMT modely je možné v tomto případě dosáhnout zlepšení modelu dopředné kinematiky téměř o půl řádu. Tento údaj je zkrácený, jelikož je použitý stejný datový set jak učit i testovací (z důvodu malého množství naměřených vzorků). I tak lze konstatovat, že použití LLMT k aproximaci chyby funguje uspokojivě.



Obr. 52 Srovnání celkové chyby s a bez korekce LLMT modely (robot Staubli TX200)

7. Závěr

Tato diplomová práce je zaměřena na kalibraci robotických ramen. Zabývá se jednak tradiční kalibrací robotických ramen, jednak metodou kombinující kinematický model s kompenzací chyby za využití neuro-fuzzy modelů (NF) a lokálních lineárních modelových stromů (LLMT). Primárně jsou adresovány metody vytváření modelů řešících dopřednou kinematickou úlohu.

První část práce obsahuje teoretický úvod do problematiky: struktury robotických ramen, tradiční kalibrační metody, měřicí zařízení využívaná ke kalibraci, základy neuro-fuzzy modelování a rozbor dávkového indukčního algoritmu využívajícího RD dělicí pravidlo. Následuje rozbor konkrétní realizace dávkového indukčního algoritmu v programovacím jazyku Matlab a testování jeho funkcionality na vybraných testovacích funkcích. Tento algoritmus je v prvním přiblížení sledován jako použitelný k potřebám kalibrace za použití aproximace chybové funkce.

Druhá část práce se zabývá vytvořením a porovnáním modelů rovinného robotického ramene. V této části jsou vytvořeny a popsány modely: 1) nepoddajný kinematický model, 2) referenční kinematický model uvažující poddajnost mechanismu, 3) kalibrovaný kinematický model, 4) model kombinující nepoddajný kinematický model s kompenzací chyby pomocí NF modelu s použitím LLMT a 5) model aproximující celý referenční model pouze za použití LLMT a NF. Je provedeno podrobné srovnání a diskuse o výhodách a nevýhodách a také případech vhodného použití těchto čtyř modelů (model č. 2 je používán pouze jako referenční). Z pohledu výsledků řešení testovací dopředné kinematické úlohy dvouramenného rovinného robotu se nejlépe jeví model č. 4 kombinující nepoddajný kinematický model s kompenzací chyby pomocí NF modelu s použitím LLMT, který dosahoval více jak o řád větší přesnosti než kalibrovaný model č. 3, naopak nejhorší výsledky vykazoval model č. 5. Nakonec je rozebrána problematika vytváření prostorových kinematických modelů a jsou diskutovány případy jejich vhodného použití.

Poslední část práce je zaměřena na zpracování výstupů z měření průmyslového robotu Stäubli TX200 externím měřicím zařízením laser tracker za účelem kalibrace. Následně je za pomoci těchto naměřených hodnot vytvořen model aproximující chybu koncového bodu užitím LLMT modelu (obdoba modelu č. 4). V tomto konkrétním případě bylo teoreticky dosaženo zlepšení přesnosti o půl řádu.

Nejvýznamnějším závěrem této práce je prokázání, že pro potřeby kalibrace dopředné kinematické úlohy, se kalibrační metoda pracující s nekalibrovaným kinematickým modelem a kompenzací chyby realizovanou LLMT a NF modely jeví jako velmi vhodný nástroj.

Lze konstatovat, že stanovené cíle práce byly naplněny.

Náměty pro další výzkum, které plynou z náplně a výsledků předmětné práce, lze shrnout do následujících bodů: použitelnost LLMT a NF modelů z hlediska kalibrace a inverzní kinematické úlohy; optimalizace algoritmu (především z pohledu vyhlazení a zavedení datových struktur); provedení srovnání použitého algoritmu s podobnými algoritmy (např. LOLIMOT, IMTI, ...); optimalizace velikosti modelových stromů vzhledem k požadované přesnosti; realizace a zhodnocení výsledků kompenzace chyby LLMT a NF modely na řádově větším datovém setu, než je 45 vzorků z měření.

8. Literatura

- [1] ISO 8373:2012 (en), *Robots and robotic devices — Vocabulary*, ICS 25.040.30, 01.040.25, 2012.
- [2] V. Hamrle, „Kalibrovatelnost a její použití pro návrh paralelních kinematických struktur,“ disertační práce, České vysoké učení technické, Fakulta strojní, Ústav mechaniky biomechaniky a mechatroniky, Praha, 2009.
- [3] B. Muralikrishnan, S. Phillips a D. Sawyer, „Laser trackers for large-scale dimensional metrology,“ *Precision Engineering*, sv. 44, pp. 13-28, 2016.
- [4] T. Skopec, „KALIBRACE PARALELNÍCH MECHANISMŮ S ADAPTIVNÍ SLOŽITOSTÍ MODELU,“ disertační práce, České vysoké učení technické, Fakulta strojní, Ústav mechaniky biomechaniky a mechatroniky, Praha, 2012.
- [5] Z. Šika, V. Hamrle, M. Valášek a P. Beneš, „Calibrability as additional design criterion of parallel kinematic machines,“ *Mechanism and Machine Theory*, sv. 50, pp. 48-63, 2012.
- [6] C. Wampler, J. M. Hollerbach a T. Arai, „Implicit loop method for kinematic calibration and its application to closed-chain mechanisms,“ *IEEE Transactions on Robotics and Automation*, sv. 11, č. 5, pp. 710 - 724, 1995.
- [7] O. Nelles, A. Fink a R. Iserman, „LOCAL LINEAR MODEL TREES (LOLIMOT) TOOLBOX,“ v *IFAC System Identification*, Santa Barbara, California, 2000.
- [8] D. a. C. S. POTTS, „Incremental Learning of Linear Model Trees,“ *Machine Learning*, sv. 61, č. 1, pp. 5-48, 2005.
- [9] G. Chow, „Tests of equality between sets of coefficients in two linear regressions,“ *Econometrica*, č. 28, 1960.
- [10] A. Karlic, „Linear Regression in Regression Tree Leaves,“ *In Proceedings of ECAI-92*, pp. 440--441, 1992.
- [11] J. Denavit a R. Harteberg, „A kinematic notation for lower-pair mechanisms based on matrices,“ *trans. asme j. appl. mech*, sv. 23, pp. 215-221, 1955.
- [12] D. R. Bruce, „FORWARD KINEMATICS THE DENAVIT-HARTENBERG CONVENTION,“ 19 09 2003. [Online]. Available: <https://users.cs.duke.edu/~brd/Teaching/Bio/asmb/current/Papers/chap3-forward-kinematics.pdf>. [Přístup získán 22 07 2021].

[13] B. Petr, V. Petřík, Z. Šika a V. Michael, „Technical Report Calibration using two Laser Trackers LEICA AT901 MR,“ 2014 1 2014.

[14] Stäubli International AG, „TX200 Range 6 axis industrial robot,“ [Online]. Available: <https://www.staubli.com/en/file/5258.show>. [Přístup získán 2021 07 2021].