

Kinematická analýza a programování robota typu SCARA

Radim Soukup

Bakalářská práce
2020



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav automatizace a řídicí techniky

Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE
(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Radim Soukup**
Osobní číslo: **A17043**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Inteligentní systémy s roboty**
Forma studia: **Prezenční**
Téma práce: **Kinematická analýza a programování robota typu SCARA**
Téma práce anglicky: **The Kinematic Analysis and Programming of a SCARA Robot**

Zásady pro vypracování

1. Nakreslete zadaný typ manipulátoru v 3D grafickém návrhovém systému SolidWorks.
2. Zaveďte souřadné systémy podle DH a proveďte úplný kinematický popis systému (transformační matice).
3. Pomocí systému Matlab vykreslete pohotovostní pracovní prostor uvedeného robota.
4. Vypočítejte inverzní kinematickou úlohu pro uvedený typ robota a ověřte ji pomocí kinematického modelu.
5. Na reálném zařízení naprogramujte typovou úlohu (např. vytvoření vlastního podpisu psacím písmem pomocí v efektoru pružně uchycené tužky).
6. Vypracujte úplnou metodiku reálného cvičení s programováním tohoto robota.

Rozsah bakalářské práce: **Do 60 stran**
Rozsah příloh: **Do 20 obr.**
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. **Úředníček, Z.:** Robotika, skriptá UTB ve Zlíně, Zlín 2012, 978780774547223-7
2. **Siciliano, B., Khatib, O.:** Springer Handbook of Robotics. Springer-Verlag Berlin Heidelberg 2008, ISBN: 978-3-540-23957-4, e-ISBN: 978-3-540-30301-5
3. **Jazar, R. N.:** Theory of Applied Robotics: Kinematics, Dynamics, and Control, Springer Science + Business Media, LLC, New York, 2007, ISBN-13:978-0-387-32475-3
4. **Critchlow, A. J.** Introduction to Robotics. New York: Macmillan, 1985. ISBN 0023255900
5. **Siciliano B, Sciavicco L, Villani L, Oriolo G (2009) Robotics: Modelling, planning and control.** Springer-Verlag, Berlin Heidelberg

Vedoucí bakalářské práce: **doc. RNDr. Ing. Zdeněk Úředníček, CSc.**
Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce: 20. prosince 2019
Termín odevzdání bakalářské práce: 15. května 2020



doc. Mgr. Milan Adámek, Ph.D.
děkan

prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

Ve Zlíně dne 20. prosince 2019

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 3. 8. 2020

Radim Soukup, v. r.
podpis diplomanta

ABSTRAKT

Bakalářská práce se zabývá průmyslovým robotem typu SCARA, konkrétně robotem Stäubli RS40b. V teoretické části jsou představeny pojem průmyslový robot, možnosti programování a obecná kinematická analýza průmyslového robota. Praktická část obsahuje vytvořený 3D model daného typu robota v systému SolidWorks, zavedení souřadných soustav podle Denavit-Hartenbergovy notace, kompletní kinematickou analýzu daného robota, vykreslení pracovního prostoru daného typu robota v systému Matlab a naprogramování reálného zařízení v Stäubli Robotic Suite tak, aby reprodukovalo podpis.

Klíčová slova: Stäubli, SCARA, průmyslový robot, programování, kinematická analýza

ABSTRACT

The Bachelor's thesis deals with an industrial robot of the SCARA type, specifically robot Stäubli RS40b. In the theoretical part is introduced the concept of an industrial robot, programming options and general kinematic analysis of the industrial robot. The practical part contains a created 3D model of given type of robot in the SolidWorks system, the introduction of coordinate systems according to Denavit-Hartenberg notation, complete kinematic analysis of the given robot, rendering of workspace of given type of robot in Matlab system and programming of real device in Stäubli Robotic Suite to reproduction signature.

Keywords: Stäubli, SCARA, industrial robot, programming, kinematic analysis

Tímto chci poděkovat vedoucímu této bakalářské práce doc. RNDr. Ing. Zdeňkovi Úředníčkovi, CSc. za jeho rady a neobyčejnou ochotu. Dále chci poděkovat rodině a své vzácné sestře Adree za jejich podporu.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	9
1 ROBOTIKA	10
1.1 PRŮMYSLOVÝ ROBOT	10
1.2 PROGRAMOVÁNÍ PRŮMYSLOVÝCH ROBOTŮ	11
1.3 UPLATNĚNÍ A RŮST PRŮMYSLOVÝCH ROBOTŮ	12
2 KINEMATIKA PRŮMYSLOVÝCH ROBOTŮ	13
2.1 KINEMATICKÉ VAZBY	13
2.2 STUPNĚ VOLNOSTI.....	14
2.3 KINEMATICKÝ ŘETĚZEC	15
2.4 KONFIGURACE SCARA	16
2.5 PŘÍMÁ KINEMATICKÁ ÚLOHA	17
2.5.1 Zavedení souřadných soustav podle Denavit-Hartenbergovy notace	17
2.5.2 Transformace mezi souřadnými soustavami	19
2.5.3 Pracovní prostor	21
2.6 INVERZNÍ KINEMATICKÁ ÚLOHA.....	21
II PRAKTICKÁ ČÁST	23
3 MODEL ROBOTY SCARA	24
4 KINEMATIKA	25
4.1 ZAVEDENÍ SOUŘADNÝCH SOUSTAV	25
4.2 VÝPOČET PŘÍMÉ KINEMATICKÉ ÚLOHY	26
4.3 POHOTOVOSTNÍ PRACOVNÍ PROSTOR	28
4.3.1 Stručný popis skriptu pro vykreslení pracovního prostoru	28
4.4 VÝPOČET INVERZNÍ KINEMATICKÉ ÚLOHY	31
4.4.1 Ověření inverzní kinematické úlohy na kinematickém modelu.....	34
5 TYPOVÁ ÚLOHA PODPIS	41
5.1 DIGITALIZACE PODPISU	41
5.2 PROGRAMOVÁNÍ PODPISU.....	43
ZÁVĚR	48
SEZNAM POUŽITÉ LITERATURY	49
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	51
SEZNAM OBRÁZKŮ	52
SEZNAM TABULEK	53
SEZNAM PŘÍLOH	54

ÚVOD

Nástup robotů pozitivně ovlivnil mnoho odvětví a zajistil vyšší komfort lidské populace. Robotizace je mladým, ale zároveň perspektivním oborem, který pomáhá dosahovat vyšší efektivity nejen výrobních a technologických firem, ale prostupuje i do dalších oblastí jako například do logistiky.

Jedním z důvodů, proč společnosti kladou důraz na robotizaci a automatizaci, je dosahování lepších výkonů za nižší ceny než při zapojení lidského faktoru. Nejen, že jsou v porovnání s lidským kapitálem efektivnější, ale také snižují míru chybovosti.

Existuje mnoho uspořádání robotů, například antropomorfní, sférické, kartézské a další. Mezi nejčastěji používané patří uspořádání SCARA. Tento robot může pomoci firmám svou rychlostí a přesností, která je na tak vysoké úrovni, že se robotem lze věrohodně podepsat. Využívá se především na typ úlohy „pick and place“ a rychlé montáže.

V roce 2019 bylo hlavním tématem Mezinárodního strojírenského veletrhu v Brně Průmysl 4.0 a Digitální továrna. V České Republice se jedná o aktuální téma a plně automatizovaná výroba je součástí podstaty čtvrté průmyslové revoluce. Hlavní podstatou Průmyslu 4.0 se stává digitalizace. Výhodou je minimalizace poruchovosti a maximalizace efektivity. Díky tomu by se mělo dosahovat lepších výsledků se zvládnutím měnící se poptávky. Člověk by pak v budoucnu mohl vykonávat pouze činnosti, kterých průmyslový robot není schopen. Mohlo by to mít za následek vyšší počet průmyslových robotů ve výrobě než pracovníků.

Cílem této práce je nakreslit robota Stäubli RS40b typu SCARA v návrhovém systému SolidWorks, zavést souřadné systémy podle Denavit-Hartenbergovy notace, provést úplný popis systému, pomocí systému Matlab vykreslit pohotovostní pracovní prostor uvedeného robota, vypočítat inverzní kinematickou úlohu pro uvedený typ robota, ověřit inverzní kinematickou úlohu pomocí kinematického modelu, na reálném zařízení naprogramovat typovou úlohu a vypracovat úplnou metodiku reálného cvičení s programováním tohoto robota.

I. TEORETICKÁ ČÁST

1 ROBOTIKA

Robotika je moderním multidisciplinárním oborem, který se zaměřuje na studium robotů a jim podobných zařízení. K tomu využívá znalosti mechaniky, elektrotechniky, informačních technologií až po computer science, umělé inteligence, měřicí techniky a další disciplíny. V současné době je nejvíce propojená s oborem automatického řízení.[3]

Tento vědní obor je velmi mladý. V průmyslové praxi se roboty začaly používat od 20. století. V průmyslu se roboty většinou používaly jako mechanické manipulátory.[8]

Manipulátor od latinského manus (ruka, rameno), je označení pro stroje napodobující funkci lidského ramene, které jsou programovatelné a říditelné.[15][10]

Roboty se používají v nejrůznějších oblastech. Tradičně se využívají ve strojírenství jako nástroje technologie pro zpracování, obrábění nebo manipulaci. V současné době se používají i ve zdravotnictví, strojírenství, vojenství a dalších oborech.[4]

K řešení problémů robotiky je využito více oborů. Základ tvoří matematika a geometrie k popisu pohybu a prostředí, v kterém se robot pohybuje, mechanika studující pohyb tělesa, počítačová technika a programování.[8]

1.1 Průmyslový robot

Robot je spojován s pokusy o vytvoření umělé napodobeniny člověka. Touto snahou vzniklo mnoho zařízení, která ulehčují člověku život. Z ekonomického hlediska patří největší zásluhy průmyslovým robotům.[3]

Slovo robot je odvozeno ze slova „robot“, která má v češtině význam fyzicky náročná práce. Toho využil český spisovatel Karel Čapek ve své hře R.U.R.. Od této doby si lidé pod pojmem robot představují jakékoliv automatické zařízení. Odborníci roboty chápou jako neživé stroje, a proto se skloňují podle neživotného rodu „hrad“.[5]

Robot je počítačem řízený integrovaný systém, který vykonává cílevědomý pohyb v reálném prostředí a čase. Je to stroj, který dokáže vykonávat podobné činnosti jako člověk nebo zvíře. Nejčastěji se jedná o pohybové a manipulační činnosti.[3]

Rozdíl mezi průmyslovým robotem a robotickým ramenem spočívá ve specifikaci robotu. Průmyslový robot se skládá z manipulátoru, koncového efektoru, řídicích orgánů a akčních členů.[15][24][17]

Roboty můžeme dělit dle uspořádání mechanického systému na:

1. Sériové: Tvoří jej série tuchách článků, kde každá dvojice tuhých článků je spojena rotačním nebo translačním kloubem. Taková kinematická struktura má otevřenou smyčku v řetězci.[1]
2. Paralelní: U takového mechanického systému jsou vytvořené kinematické smyčky.[1]
3. Kombinované: Kombinace sériových a paralelních robotů. Patří sem například chodící roboty. [8]

Průmyslové roboty se dále dělí do generací:

1. První generace jsou roboty vykonávající pevně naprogramované sekvenční operace. Typická činnost byla tzv. „pick and place“ neboli zvedni a polož. Pozdější roboty této generace, označované jako $1\frac{1}{2}$ generace, začaly používat senzory.
2. Druhá generace jsou roboty vybaveny větším počtem senzorů než předchozí $1\frac{1}{2}$ generace a s tím spojeným vnímáním. K řízení této generace byl potřeba řídicí počítač. Tyto roboty jsou vybaveny pro práci v systému oko-ruka. Pozdější roboty této generace označované jako $2\frac{1}{2}$ generace mohou vykonávat komplexní vnímatelně pohybové funkce.
3. Třetí generace jsou roboty vybaveny umělou inteligencí. Typická vlastnost této generace robotů je schopnost učit se a adaptovat při řešení úloh.

Průmyslové roboty mají různé konfigurace, proto se při výběru řídíme konstrukčními parametry jako tuhost, typ a rozměry. Dále se řídíme funkčními parametry jako pracovní prostor, přesnost a rychlost. V neposlední řadě provozními parametry jako doba programování, provozní náklady a spolehlivost.[15]

1.2 Programování průmyslových robotů

Roboty, které se mohou programovat, dělíme na roboty s pevným programem, proměnlivým programem, a kognitivní roboty.[17]

Metody programování jsou:

1. On-line- Také zvané jako „přímé“ a má více postupů. První postup „play-back“ je takový, že člověk pohybuje robotem tak, jak by se měl robot pohybovat. Robot poté

opakuje tento pohyb nebo pohyby. Další postup se nazývá „teach in“. Operátor přímo naprogramuje polohy dle technologických údajů.[17]

2. Off-line- Tato metoda spočívá v matematickém modelování. Chování robota je vizualizováno na obrazovce počítače. Tato metoda nevyžaduje fyzickou přítomnost robota. Výhoda této metody spočívá v možnosti modelování a simulací pracovního cyklu robota. Nevýhodou je, že v realitě se robot nechová úplně stejně jako model.
3. Hybridní- Je spojení obou výše zmíněných metod. Při takovémto programování se nejdříve řeší pohybové údaje robota off-line a pracovní část cyklu on-line, aby bylo dosaženo požadované přesnosti.[17]

1.3 Uplatnění a růst průmyslových robotů

Využívání robotů se zvyšuje. V roce 2014 byl průměrný počet víceúčelových průmyslových robotů 66 robotů na 10 tisíc pracovníků. Česká republika byla mírně nad průměrem. [23]

V roce 2017 byl podle Mezinárodní federace robotiky největší nárůst proti předcházejícímu roku nasazování robotů v elektrotechnice a elektronice o 27 %, automobilovém a kovozpracujícím průmyslu o 21 %.[18]

Podle prezidenta IFR Junji Tsudy mohou roboty až zdvojnásobit uspokojení trhu jak pro malé tak i střední firmy.[19]

Studie s názvem „Humans Wanted, Robots Need You“ se zabývala otázkou, zda roboty připraví lidi o práci. Výsledkem je, že více než 87 % zaměstnavatelů zaměstná více lidí vlivem automatizace. Studie zahrnuje 44 zemí. Podle ní si případní nahrazení zaměstnanci stále naleznou práci, jen je nutné, aby se rekvalifikovali. [20]

2 KINEMATIKA PRŮMYSLOVÝCH ROBOTŮ

Kinematika patří do mechaniky, která je jednou z technických a přírodních věd. Mechanika se zabývá geometrií pohybu těles a jejich vzájemnými souvislostmi. Kinematika je nauka o geometrii pohybu tuhého tělesa vzhledem k prostoru a času [9][8].

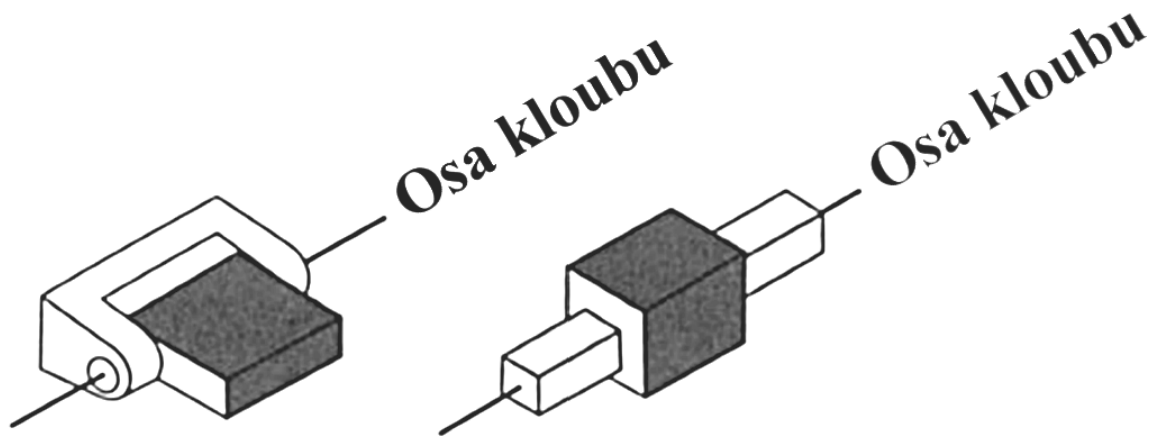
Přímá úloha kinematiky je základní úlohou, která je řešena v kinematice robotů. V této kinematické úloze se hledají vztahy mezi kartézskými souřadnými systémy jednotlivých článků robotu a kloubovými souřadnicemi robota. Tato úloha je formulována pomocí homogenní transformační matice mezi jednotlivými souřadnými systémy kinematických členů. Inverzní kinematická úloha řeší určení kloubových souřadnic. Tato úloha spočívá v řešení soustavy nelineárních algebraických rovnic. Oproti přímé kinematické úloze má inverzní kinematická úloha obtížnější řešení. Řešení obou úloh je nutné k plánování dráhy robota. [3]

2.1 Kinematické vazby

Geometrickou vazbu mezi dvěma členy nazýváme kinematickou dvojicí. Kinematické dvojice mohou být obecné, rotační, posuvné, valivé a jiné.[14][15]

Obecná kinematická dvojice z obrázku (2) váže těleso 2 jedním bodem A ke křivce k_1 pevného tělesa 1. Těleso 2 se může ve směru tečny t_A posouvat ke křivce k_1 a nezávisle na posunu se otáčet kolem tohoto bodu. Takto vázané těleso je určeno dvěma na sobě nezávislými souřadnicemi. Určit polohu tělesa lze, pokud je známa jedna souřadnice bodu A a úhel libovolné přímky p , nebo druhá souřadnice bodu A vázaná na rovnici křivky k_1 . [1]

Manipulátor tvoří články vzájemně spojeny klouby. Rotační dvojicí nazýváme články spojeny rotačním kloubem. Rotační kloub umožní relativní rotaci dvou spojených článků. Posuvnou dvojicí nazýváme články spojeny posuvným kloubem. Posuvný kloub umožňuje vzájemný posuvný pohyb dvou článků. Tyto klouby lze vidět na obrázku (1). [1][3]



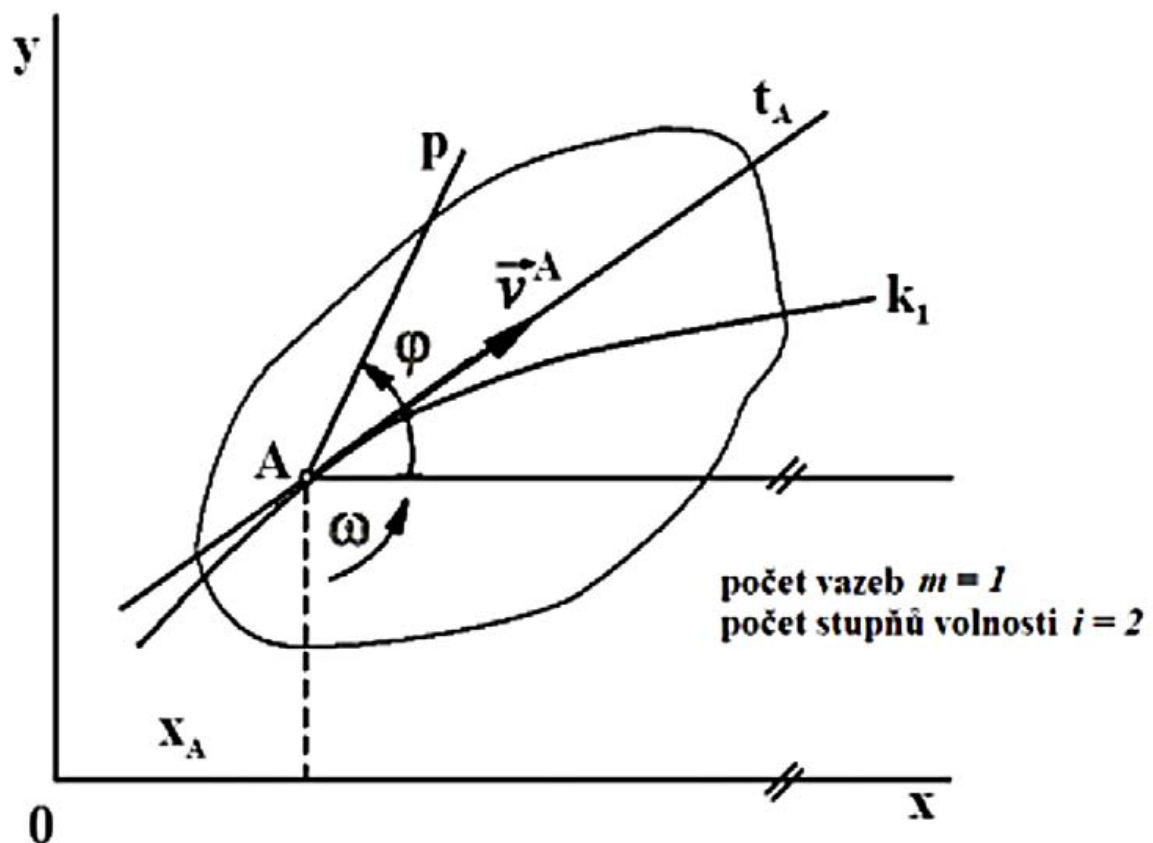
Obrázek 1 Rotační a translační kloub[1]

2.2 Stupně volnosti

Ze schématu kinematických řetězců neboli mechanismů lze určit pohyblivost rovinných kinematických řetězců. Počet stupňů volnosti tělesa vzhledem k rámu je dán počtem na sobě nezávislých souřadnic pro jednoznačné určení tělesa. Toto číslo udává i počet pohybů bodu nezávisle na sobě.[9,10]

V prostoru má volné těleso 6 stupně volnosti, které se značí jako i . Počet vazeb označme jako m . Závislost mezi stupni volnosti a počtem vazeb v prostoru je:[9]

$$i = 3 - m \quad (1)$$



Obrázek 2 Obecná kinematická dvojice[1]

2.3 Kinematický řetězec

Články mechanismu jsou spojeny kloubem, který tomuto systému dává omezení pohybu. Počet stupňů volnosti kloubu udává počet kloubových souřadnic, potřebných k popsání relativního pohybu dvou spojených tuhých těles.[11]

Kinematickým řetězcem se rozumí soustava n členů, kde je každý člen spojen minimálně s jedním navazujícím členem patřící do kinematické dvojice. Uzavřeným kinematickým řetězcem je ten, jehož každý člen je spojen alespoň s dvěma dalšími. V opačném případě se jedná o otevřený kinematický řetězec.[9]

Stupně volnosti kinematického řetězce s jedním členem pevně spojeným s rámem vypočítáme ze vztahu:[9]

$$i = 3(n - 1) - 2(r + p + v) - o \quad (2)$$

Kde n je počet členů,

r je počet rotačních kinematických dvojic,

p je počet translačních kinematických dvojic,

v je počet valivých kinematických dvojic,

o je počet obecných kinematických dvojic.

2.4 Konfigurace SCARA

Mechanická konfigurace SCARA byla vynalezena roku 1978 Hiroshimem Makino. Průlom způsobily nízké náklady na jeho sestavení a kinematická konfigurace, která umožnila rychlé pohyby. Zkratka SCARA znamená Selective Compliance Assembly Robot Arm.[2][12]

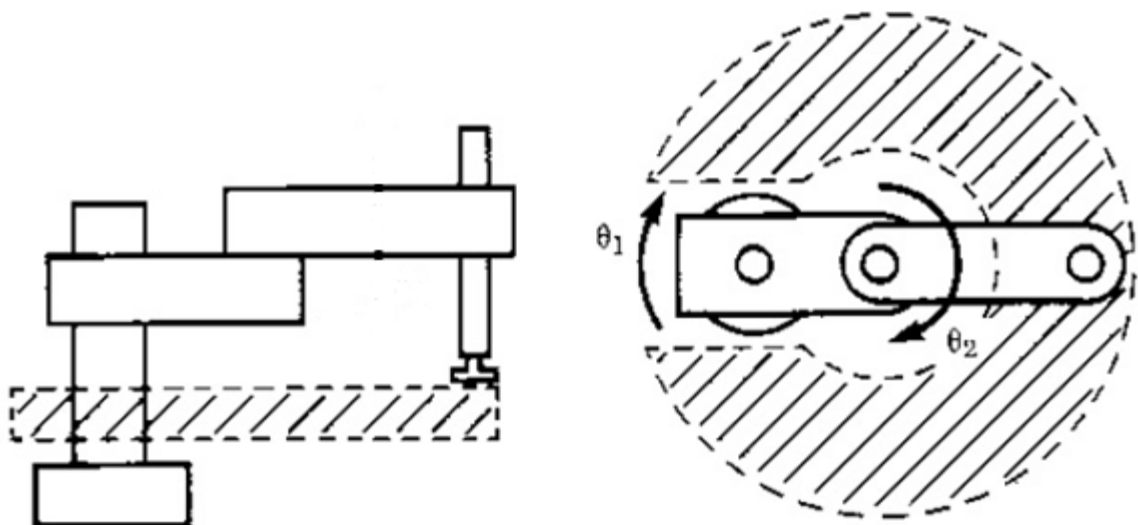
Konfigurací robota rozumíme uspořádání a typ jeho kloubů. [15]

Konfigurace robota SCARA má dva rotační klouby umožňující orientaci v rovině a třetí translační. Případně další rotační kloub, který umožňuje dodefinovat polohu efektoru v prostoru.[12][1]

Typický SCARA robot má 4 osy $R||R||P||R$. Výhodou této konfigurace je především přesnost, která je největší ze všech robotů, a tuhost.[15][11]

Translační kloub na reálném SCARA robotu je realizován jako rotační motor s kuličkovým šroubem.[11]

Roboti konfigurace SCARA se uplatňují hlavně v elektrotechnickém a strojírenském průmyslu, kde „stačí“ práce nad pracovní rovinou, a dále v potravinářském i farmaceutickém průmyslu.[22]



Obrázek 3 Robot SCARA a jeho pracovní prostor[13]

2.5 Přímá kinematická úloha

Cílem přímé kinematiky je určení globálních souřadnic libovolného bodu kinematické soustavy robota, většinou koncového efektoru. Homogenní transformační matice T je nástrojem pro výpočet přímé kinematické úlohy.[15]

Přímá kinematická úloha pro sériově spojenou kinematiku manipulátoru robota hledá polohu a orientaci např. koncového efektoru vzhledem k souřadné soustavě rámu udávající absolutní polohu všech bodů kinematické struktury.[2]

Jedná se o statický geometrický problém. Pro koncový efektor se vypočítá jeho poloha a orientace vzhledem k rámu. [13]

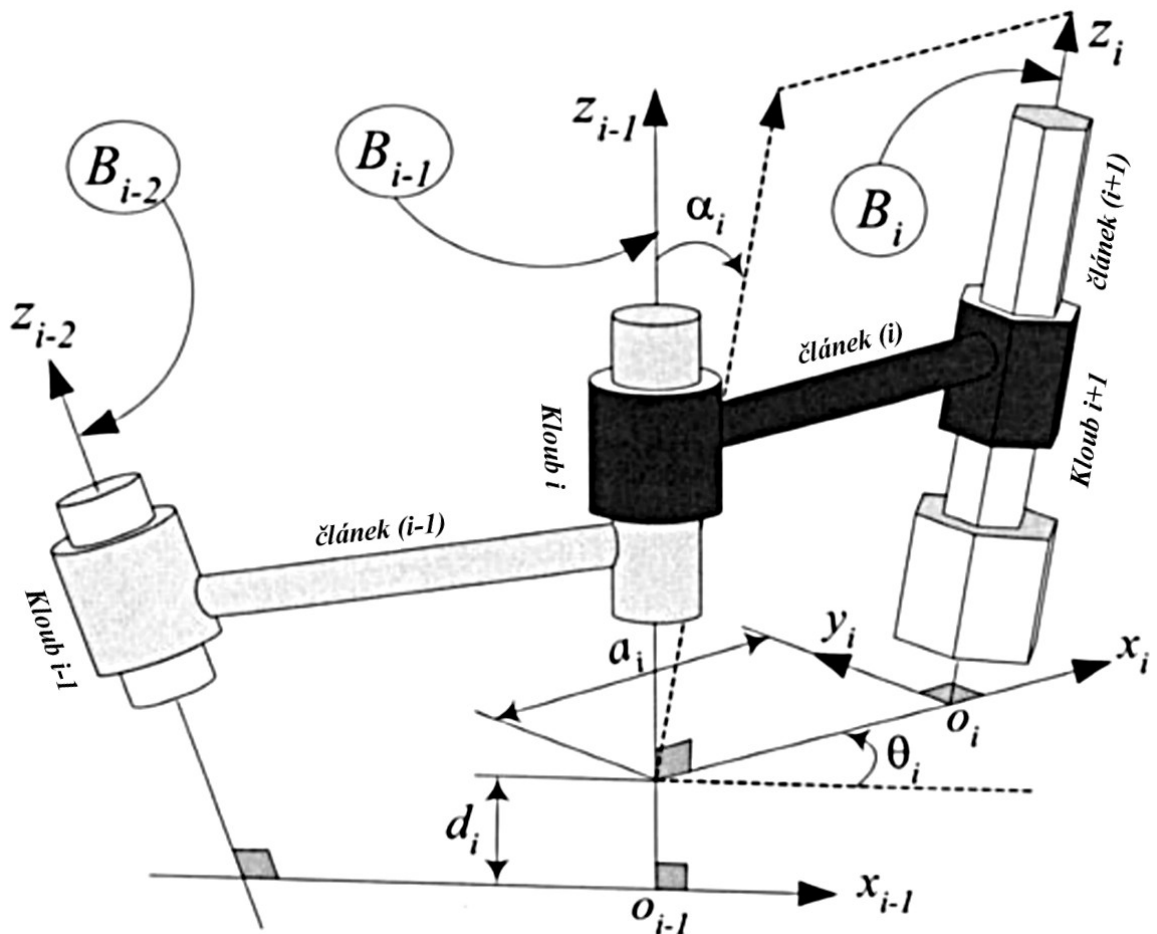
Jinými slovy, pro sériového robota je úkolem najít vzájemné polohy členů určené vzdálenostmi a úhly jejich kloubů, a tím nalézt polohu koncového efektoru. K vyjádření pohybu robota je zapotřebí zvolit jeho počáteční polohu. Počáteční poloha může být zvolena kteroukoliv konfigurací os, ale z praktického hlediska se volí počáteční poloha v konfiguraci os, která je pevně spojena s rámem mechanismu.[8]

2.5.1 Zavedení souřadných soustav podle Denavit-Hartenbergovy notace

Vztah mezi dvěma souřadnými soustavami může být kinematicky popsán pomocí čtyř parametrů. Dva parametry mají konstrukční význam a jsou dány konstrukcí manipulátoru. Další dva parametry určují kloubové souřadnice. Dohromady jednoznačně určují vzájemnou polohu dvou na sebe navazujících souřadných soustav. Pro zavedení souřadných soustav podle DH pravidel musí souřadné soustavy přesně splňovat definovaná pravidla. [13][1]

DH pravidla nemají jednoznačnou definici v případě, kdy:

1. Souřadný systém má specifikován pouze směr osy z
2. Dvě následující osy jsou paralelní a není-li jednoznačně definována volba jejich kolmice, tj. osa x [15]



Obrázek 4 Článek (i - 1), (i), a (i + 1) spolu se soustavou souřadnic B_i a B_{i-1} . Pro zavedení souřadných soustav podle DH pravidel musí souřadné soustavy přesně splňovat definovaná pravidla. [13][1]

Sériový robot má pro n kloubů $n+1$ článků. Články čísujeme od 0 od základního článku spojeného s rámem a postupně další, až po článek koncového efektoru. Klouby očísujeme rovněž postupně, ale od čísla 1 pro kloub spojující nepohyblivý článek s prvním pohyblivým a čísujeme postupně až po poslední n -tý kloub. [1]

1. Osa z_i je spojena s osami $i+1$ kloubu

Všechny klouby jsou reprezentovány osou z , kterou vždy začínáme identifikaci. Kladný směr osy z je volitelný. Osa z je totožná nebo rovnoběžná s osou kloubu. Pro translační kloub lze vybrat kteroukoliv osu, která je paralelní se směrem posunutí. [1]

2. Počátek O_i se nachází v průsečíku osy z_i a společné normály mezi osami z_{i-1} a z_i [15]
3. Osa x_i je definována podél společné normály mezi osami z_{i-1} a z_i , směřující od z_{i-1} k z_i

Obecně osy z mohou být mimoběžkami, avšak vždy existuje jedna přímka vzájemně kolmá k libovolným dvěma přímkám, zvaná společná normála. Společná normála je nejkratší vzdáleností mezi dvěma mimoběžkami.

Pokud jsou dvě a více os z rovnoběžných, pak existuje nekonečný počet společných normál. [1]

4. Osa y_i pro pravotočivou souřadnou soustavu je definována vektorovým součinem. [1]

$$\hat{y}_i = \frac{\vec{z}_i \times \vec{x}_i}{|\vec{z}_i \times \vec{x}_i|}$$

Tímto způsobem se zavádí souřadná soustava pro každý článek.

Úplná poloha souřadné soustavy je v předchozí soustavě definovaná čtyřmi parametry:

1. a_i je vzdálenost mezi osy z_{i-1} a z_i měřená podél osy x_i [1]
2. α_i je úhel od z_{i-1} do z_i měřený okolo osy x_{i-1} [1]
3. d_i je vzdálenost mezi osou x_{i-1} a x_i měřená podél osy z_{i-1} [1]
4. θ_i je úhel osy x_{i-1} k ose x_i měřený okolo z_{i-1} [1]

Dva parametry jsou vždy konstantní a závislé pouze na geometrii. [15]

Zmíněné veličiny se nazývají Denavit-Hartenbergovy parametry a jsou standardem pro určení vzájemné polohy v 3D prostoru dvou souřadných soustav spojených s články robota. Nicméně výklad použití těchto parametrů je v literaturách různý.[8]

2.5.2 Transformace mezi souřadnými soustavami

Obecně transformace souřadné soustavy B_i článku (i) vzhledem k souřadné soustavě B_{i-1} článku (i-1) je funkcí čtyř parametrů daného spojení. U každého reálného kloubu robota jsou tři parametry určeny konstrukcí a čtvrtý je tvořen kloubovou souřadnicí.[2]

Definováním souřadných soustav se každý vztah rozdělí na jednoduché dílčí problémy. Vztah transformace mezi i-tou a i-1-ní souřadnou soustavou se označuje jako ${}^{i-1}T_i$. Transformaci vektorů definovaných v souřadné soustavě B_i na popis v souřadné soustavě B_{i-1} lze zapsat zapsat: [2]

$${}^{i-1}T_i = D_{z_{i-1}, d_i} R_{z_{i-1}, \theta_i} D_{x_{i-1}, a_i} R_{x_{i-1}, \alpha_i} \quad (3)$$

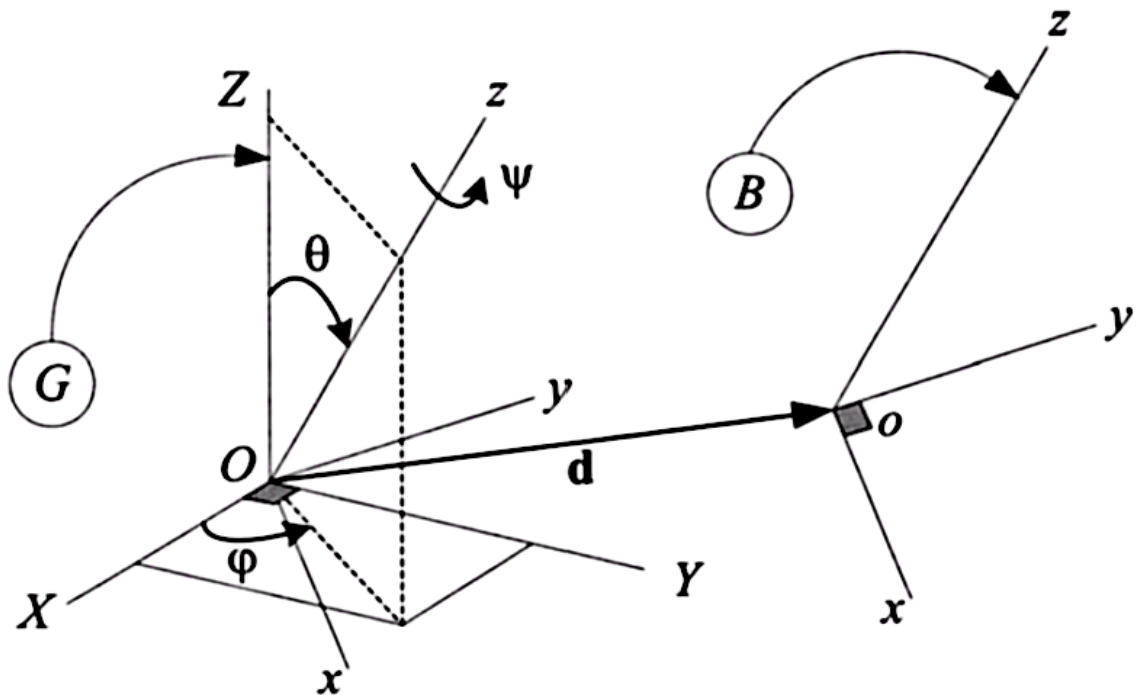
$${}^{i-1}T_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \cos(\alpha_i) & \sin(\theta_i) \sin(\alpha_i) & a_i \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \cos(\alpha_i) & -\cos(\theta_i) \sin(\alpha_i) & a_i \sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$= \begin{bmatrix} {}^{i-1}R_i & {}^{i-1}d_i \\ 0 & 1 \end{bmatrix}$$

$${}^{i-1}T_i \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} = \begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} \quad (5)$$

Správnost transformační matice lze ověřit prvky čtvrtého sloupce. Tyto prvky by měly být souřadnicemi počátku i -té souřadné soustavy v soustavě $i-1$ -ní. Po vypočítání transformačních matic pro jednotlivé souřadné soustavy se mezi sebou vynásobí zleva, aby byla nalezena jedna transformační matice, která bude funkcí všech kloubových souřadnic. [13]

Umístění tuhého tělesa v prostoru je dáno jeho polohou a orientací v globální souřadné soustavě, tedy polohou jeho tělesové souřadné soustavy v globální souřadné soustavě. Pozici tuhého tělesa lze vyjádřit i v jiné souřadné soustavě pomocí transformace souřadnic. Výsledkem transformace je tzv. homogenní transformační matice T . Jde o kompaktní formu vyjádření transformace souřadnic mezi dvěma souřadnými systémy. Transformační matice nejsou ortogonální (inverzní se nerovná transponované). [15]



Obrázek 5 Rotační a translační pohyb lokální soustavy tuhého tělesa B vzhledem ke globální soustavě souřadnic tuhého tělesa G

2.5.3 Pracovní prostor

Pracovní prostor je objem prostoru, kterého lze dosáhnout koncovým efektem. Tento prostor je omezen geometrickými vlastnostmi robotu.[1] Pracovní prostor zásadně ovlivňuje rozsah práce robotu svou velikostí i tvarem. Z toho důvodu je lepší si pro daný úkol pracovní prostor nejprve spočítat. To mimo jiné rozhodne i o výběru konfigurace robotu.[12]

Rozděluje se na:

1. Dosažitelný – Tento prostor je dosažitelný alespoň jednou orientací efektoru robotu. Právě tento pracovní prostor rozhoduje o volbě požadovaného robotu.[12]
2. Pohotovostní – Tohoto prostoru může dosáhnout koncový efektor s libovolnou orientací Je podmnožinou dosažitelného pracovního prostoru. [12]

2.6 Inverzní kinematická úloha

Inverzní kinematická úloha hledá požadované kinematické kloubové souřadnice při dané poloze koncového efektoru nebo globální polohu jiného bodu mechanické soustavy robotu.[1]

V obecném případě sériově spojeného manipulátoru s šesti stupni volnosti je známa transformační matice 0T_6 . Z definice transformace, kde se vyskytují trigonometrické funkce, je zřejmé, že inverzní kinematický problém pro sériového robota vede k řešení nelineárních algebraických rovnic. Tři z těchto rovnic se vztahují k vektoru polohy zvoleného bodu v rámci homogenní transformace. Další tři z těchto rovnic se vztahují k jeho orientaci. Tyto nelineární rovnice mohou vést k mnoha nebo žádnému řešení. V případě, že je poloha a orientace koncového efektoru mimo pracovní prostor manipulátoru, pak rovnice nemají řešení.[1]

Řešení, která existují, často nelze řešit analyticky. Z toho důvodu je nutné nalézt řešení numerickými metodami.[2][3]

Problémem inverzní kinematiky je vypočítat všechny možné úhly nebo posunutí kloubů tak, aby vyhovovaly dané poloze a orientaci koncového efektoru nebo globální poloze jiného bodu mechanické soustavy robota. Pokud se požadovaná poloha nachází mimo pracovní prostor, pak úloha nemá řešení.[13]

Při známé předpokládané poloze bodu koncového efektoru je úlohou nalézt kinematické kloubové souřadnice jednotlivých kloubů robota.[8]

II. PRAKTICKÁ ČÁST

3 MODEL ROBOTA SCARA

K vytvoření modelu byl použit systém SolidWorks. Pro analýzu bylo výhodné vytvořit zjednodušený model robota. Zjednodušení spočívá v jednoduchých tvarech jednotlivých článků manipulátoru a zanedbání detailů. Hmoty článků je pro tyto účely zanedbána. Články mají reálnou velikost.

Při vytváření dílů byl postup takový, že daný díl byl změřen na reálném zařízení a zjednodušen. Následně byl v systému SolidWorks vytvořen nový díl. Pro daný díl byla vytvořena skica a přidán materiál nástrojem „přidat vytažením“. Pro snazší vytváření dílů byla použita i referenční geometrie. Nakonec byly přidány díry.

Vytvořený model robota SCARA je sestava složená z 8 dílů. Tato sestava je na obrázku (6). Jednotlivé díly jsou vzájemně spojeny vazbami. Článek 0 a článek 1 jsou spojeny vazbou plochy k ploše a soustředěnou vazbou. Stejně tak je spojen článek 1 s článkem 2. Článek 2 a článek 3 jsou spojeny soustředěnou vazbou a další vazbou nastavující omezení pohybu. Koncový efektor je pevně spojen s článkem 3 a jediné pohyby, které může vykonat, jsou zavření nebo otevření.

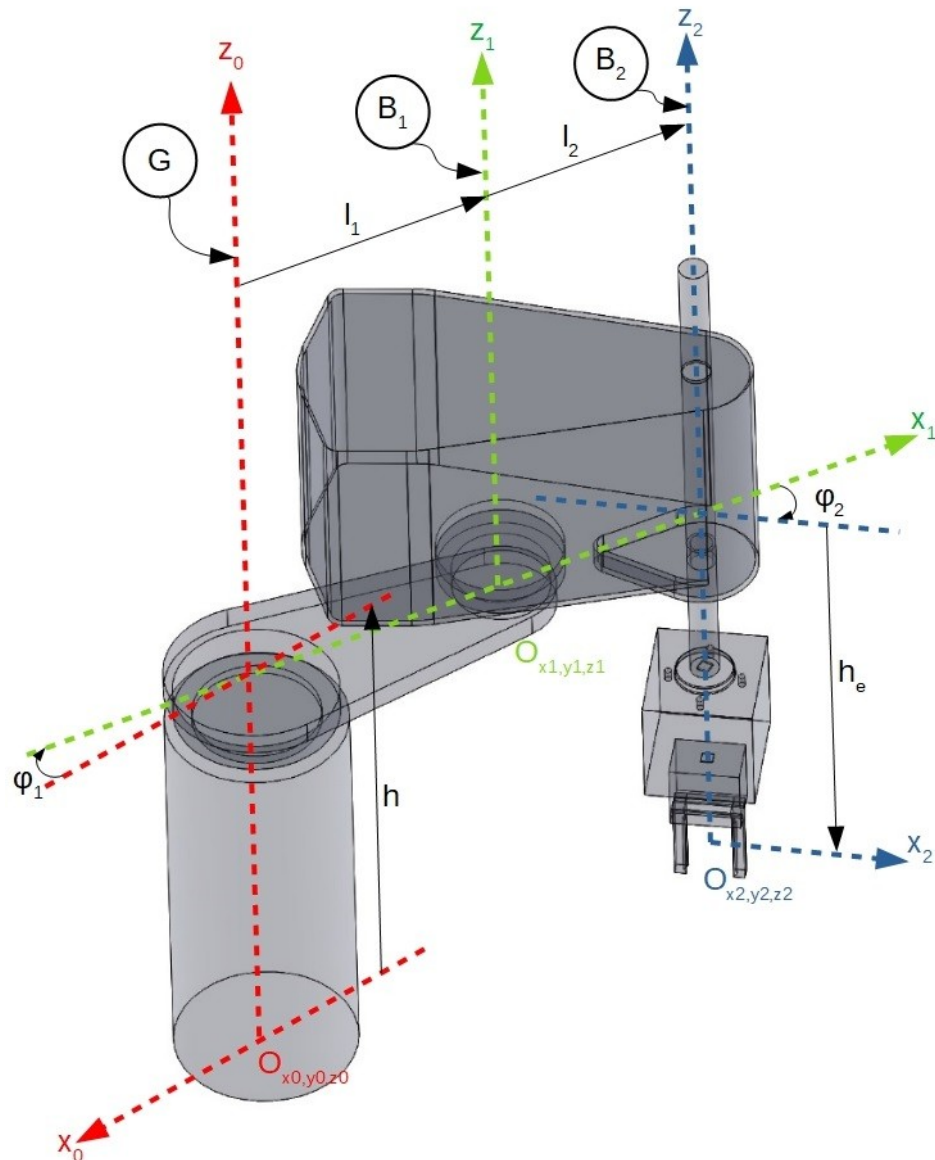
Jednotlivé konstrukční parametry potřebné ke kinematické analýze jsou v tabulce.

Název parametru	Hodnota
l_1	230
l_2	170
h	200

Tabulka 1 Konstrukční parametry robota Stäubli RS40b

4 KINEMATIKA

4.1 Zavedení souřadných soustav



Obrázek 6 Souřadné soustavy pro robota typu SCARA podle Denavit-Hartenbergovy notace

Obrázek (6) zobrazuje již stanovené souřadné soustavy podle Denavit-Hartenbergovy notace. Nejprve byly zavedeny osy z . Jejich kladný směr byl zvolen vzhůru od podstavy manipulátoru. Z takto zvolených os z se podle společné normály mezi osami z_{i-1} a z_i dokreslí osy x_i , orientované od osy z_{i-1} k ose z_i . Osy y nejsou nezbytně nutné k určení parametrů DH notace, protože jsou dány vektorovým součinem osy x a z . Tím jsou zavedeny souřadné soustavy a je možné určit parametry podle DH notace.

Parametr a_i prvního článku je vzdálenost mezi osou z_0 a z_1 podél osy x_1 . Tato vzdálenost je označena jako l_1 . U druhého článku je tento parametr vzdálenost mezi osou z_1 a z_2 podél osy x_2 . Tato vzdálenost je označena jako l_2 . Parametr α_i bude pro oba články 0° , protože všechny osy z jsou rovnoběžné. Parametr d_i prvního článku je označen jako h a je vzdáleností mezi osou x_0 a x_1 podél osy z_0 . Pro druhý článek je tento parametr vzdálenost mezi osou x_1 a x_2 podél osy z_1 . Z obrázku lze vidět směr osy z_1 , který vede opačným směrem k této vzdálenosti, proto bude tento parametr se záporným znaménkem a označen jako $-h_e$. Parametr Θ_i pro první článek je úhel osy x_0 k ose x_1 podél osy z_0 . Tento úhel je označen jako φ_1 . Osy x_0 a x_1 směřují každá na opačnou stranu, proto k úhlu φ_1 je přičteno 180° . Pro druhý článek je tento parametr úhel osy x_1 k ose x_2 podél osy z_1 . Tento úhel je označen jako φ_2 . Jeho znaménko bude záporné kvůli směru otáčení. Všechny DH parametry zapíšeme přehledně do tabulky.

Číslo článku	Parametry			
	a_i	α_i	d_i	Θ_i
1	l_1	0°	h	$180^\circ + \varphi_1$
2	l_2	0°	$-h_e$	$-\varphi_2$

Tabulka 2 Parametry podle Denavit-Hartenbergovy notace

4.2 Výpočet přímé kinematické úlohy

Pro získání bodu koncového efektoru v souřadnicích globální souřadné soustavy je nutné dosadit DH parametry do obecné homogenní transformační matice z rovnice (4).

V tomto případě má robot pouze dva články. Bude nutné sestavit dvě transformační matice pro transformaci přiléhajících soustav a jednu, která bude transformovat z globální souřadné soustavy do souřadné soustavy koncového efektoru.

První transformační matice pro přiléhající souřadné soustavy tuhých těles B_0 a B_1 bude:

$${}^0T_1 = \begin{bmatrix} \cos(\pi + \varphi_1) & -\cos(0) \sin(\pi + \varphi_1) & \sin(0) \sin(\pi + \varphi_1) & l_1 \cos(\pi + \varphi_1) \\ \sin(\pi + \varphi_1) & \cos(0) \cos(\pi + \varphi_1) & -\sin(0) \cos(\pi + \varphi_1) & l_1 \sin(\pi + \varphi_1) \\ 0 & \sin(0) & \cos(0) & h \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Po úpravě:

$${}^0T_1 = \begin{bmatrix} -\cos(\varphi_1) & \sin(\varphi_1) & 0 & -l_1 \cos(\varphi_1) \\ -\sin(\varphi_1) & -\cos(\varphi_1) & 0 & -l_1 \sin(\varphi_1) \\ 0 & 0 & 1 & h \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Druhá transformační matice přiléhajících souřadných soustav tuhých těles B₁ a B₂ bude:

$${}^1T_2 = \begin{bmatrix} \cos(\varphi_2) & \cos(0) \sin(\varphi_2) & -\sin(0) \sin(\varphi_2) & l_2 \cos(\varphi_2) \\ -\sin(\varphi_2) & \cos(0) \cos(\varphi_2) & -\sin(0) \cos(\varphi_2) & -l_2 \sin(\varphi_2) \\ 0 & \sin(0) & \cos(0) & -h_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

Po úpravě:

$${}^1T_2 = \begin{bmatrix} \cos(\varphi_2) & \sin(\varphi_2) & 0 & l_2 \cos(\varphi_2) \\ -\sin(\varphi_2) & \cos(\varphi_2) & 0 & -l_2 \sin(\varphi_2) \\ 0 & 0 & 1 & -h_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

Nyní lze vypočítat transformační matici pro určení polohy koncového efektoru v globální souřadné soustavě. Tuto transformační matici lze získat vynásobením transformační matice 0T_1 z rovnice (7) zleva transformační maticí 1T_2 z rovnice (9).

$${}^0T_2 = {}^0T_1 \cdot {}^1T_2 \quad (10)$$

$${}^0T_2 = \begin{bmatrix} -\cos(\varphi_1) & \sin(\varphi_1) & 0 & -l_1 \cos(\varphi_1) \\ -\sin(\varphi_1) & -\cos(\varphi_1) & 0 & -l_1 \sin(\varphi_1) \\ 0 & 0 & 0 & h \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\varphi_2) & \sin(\varphi_2) & 0 & l_2 \cos(\varphi_2) \\ -\sin(\varphi_2) & \cos(\varphi_2) & 0 & -l_2 \sin(\varphi_2) \\ 0 & 0 & 1 & -h_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (11)$$

$${}^0T_2 = \begin{bmatrix} -\cos(\varphi_2 - \varphi_1) & \sin(\varphi_2 - \varphi_1) & 0 & -l_2 \cos(\varphi_1 - \varphi_2) - l_1 \cos(\varphi_1) \\ -\sin(\varphi_2 - \varphi_1) & -\cos(\varphi_2 - \varphi_1) & 0 & -l_2 \sin(\varphi_1 - \varphi_2) - l_1 \sin(\varphi_1) \\ 0 & 0 & 1 & h - h_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12)$$

Poloha počátku ${}^B O_{x_2 y_2 z_2}$ souřadné soustavy tuhého tělesa B₂ je:

$${}^B O_{x_2 y_2 z_2} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (13)$$

Tento bod se transformuje z lokálních do globálních souřadnic.

$${}^G O_{x_2 y_2 z_2} = {}^0T_2 \cdot {}^B O_{x_2 y_2 z_2} = \begin{bmatrix} -\cos(\varphi_2 - \varphi_1) & \sin(\varphi_2 - \varphi_1) & 0 & -l_2 \cos(\varphi_1 - \varphi_2) - l_1 \cos(\varphi_1) \\ -\sin(\varphi_2 - \varphi_1) & -\cos(\varphi_2 - \varphi_1) & 0 & -l_2 \sin(\varphi_1 - \varphi_2) - l_1 \sin(\varphi_1) \\ 0 & 0 & 1 & h - h_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (14)$$

Po úpravě:

$${}^G O_{x_2 y_2 z_2} = \begin{bmatrix} -l_2 \cos(\varphi_1 - \varphi_2) - l_1 \cos(\varphi_1) \\ -l_2 \sin(\varphi_1 - \varphi_2) - l_1 \sin(\varphi_1) \\ h - h_e \\ 1 \end{bmatrix} = \begin{bmatrix} X_e \\ Y_e \\ Z_e \\ 1 \end{bmatrix} \quad (15)$$

Tato rovnice je výsledek přímé kinematické úlohy. Známe-li geometrii a kinematická data manipulátoru, pak lze určit souřadnice koncového bodu v globální souřadné soustavě.

4.3 Pohotovostní pracovní prostor

Pro vykreslení objemu pohotovostního pracovního prostoru v systému Matlab se vypočítají body koncového efektoru vztažené ke globální souřadné soustavě dosazením do vztahů z rovnice (15). Kloubové souřadnice ϕ_1 , ϕ_2 a h_e jsou konstrukčně omezeny intervalem. Tato omezení je možné zjistit z manuálního ovladače robota nebo v programu Stäubli Robotics Suite.

Pro omezení kloubových souřadnic ϕ_1 , ϕ_2 a h_e platí:

$$\varphi_1 \in \langle -105; 105 \rangle^\circ \quad (16)$$

$$\varphi_2 \in \langle -143; 143 \rangle \quad (17)$$

$$h_e \in \langle 0; 200 \rangle \text{ mm} \quad (18)$$

Pro DH parametry l_1 , l_2 a h platí:

$$l_1 = 230 \text{ mm} \quad (19)$$

$$l_2 = 170 \text{ mm} \quad (20)$$

$$h = 200 \text{ mm} \quad (21)$$

Výpočet a vykreslení bodů koncového efektoru v systému Matlab je provedeno pomocí skriptu pro vykreslení pracovního prostoru v příloze P I.

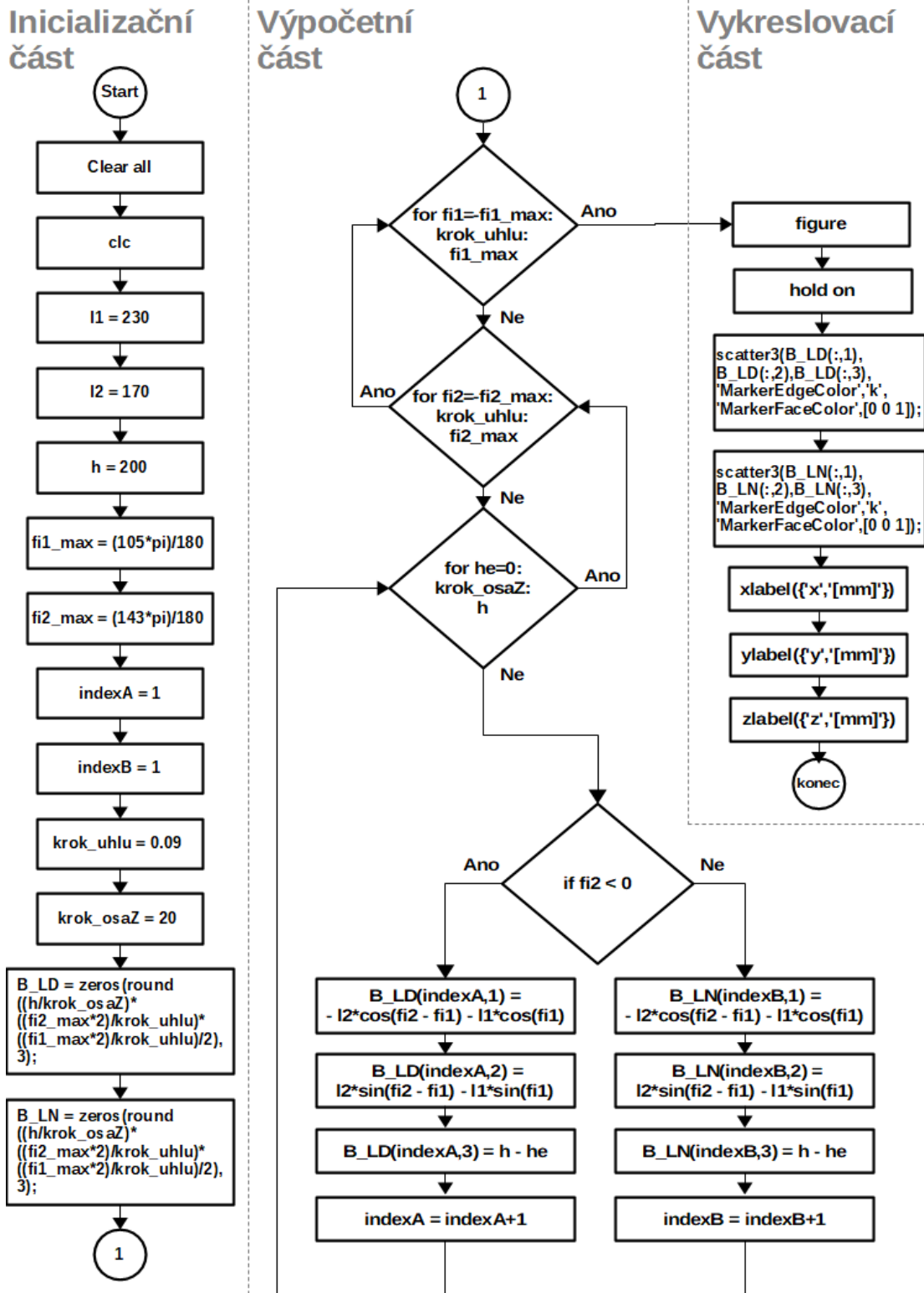
4.3.1 Stručný popis skriptu pro vykreslení pracovního prostoru

K lepší orientaci v skriptu a vývojovém diagramu slouží tabulka (3), která obsahuje výčet proměnných tohoto skriptu a jejich význam. Důležité jsou především proměnné s názvem B_LD a B_LN, které obsahují body pro dvě možné konfigurace robota, tedy konfigurace kloub nahoru a dolů.

Název proměnné	Význam
l1	DH parametr l_1
l2	DH parametr l_2
H	DH parametr h
fi1_max	Maximální hodnota omezení kloubové souřadnice φ_1
fi2_max	Maximální hodnota omezení kloubové souřadnice φ_2
indexA	Proměnná pro indexování vypočítaných bodů pro konfiguraci kloub dolů
indexB	Proměnná pro indexování vypočítaných bodů pro konfiguraci kloub nahoru
krok_uhlu	Hodnota, o kterou se mění kloubové souřadnice φ_1 a φ_2 v smyčkách for
krok_osaZ	Hodnota, o kterou se mění kloubová souřadnice h_c v smyčce for
B_LD	Body pracovního prostoru pro konfiguraci kloub dolů
B_LN	Body pracovního prostoru pro konfiguraci kloub nahoru

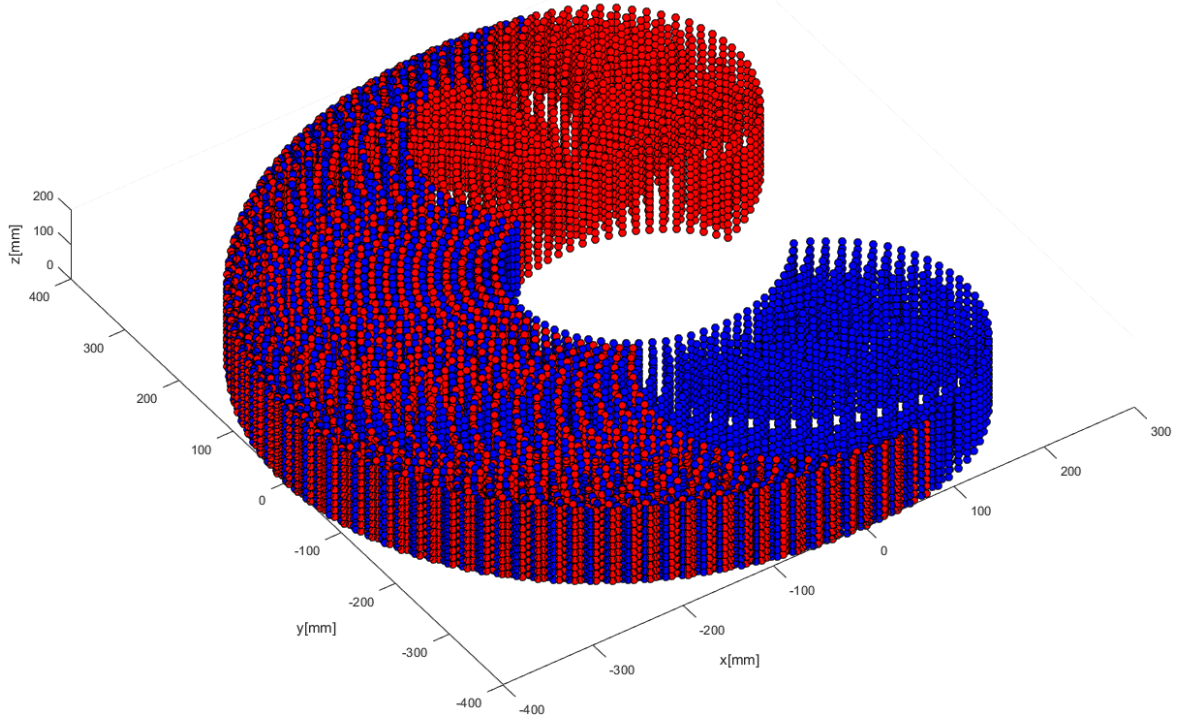
Tabulka 3 Proměnné skriptu pro vykreslení pohotovostního pracovního prostoru

Vývojový diagram tohoto skriptu se nachází na obrázku (7). Skript je rozdělen do tří částí. První inicializační část slouží k inicializaci proměnných z tabulky (3). Druhá výpočetní část dosazuje do vztahu z rovnice (15) všechny možné kombinace kloubových souřadnic v daném omezení z rovnic (16),(17) a (18). Před samotným výpočtem se výsledky rozdělí podle velikosti proměnné fi2 do dvou různých matic. Ty reprezentují dvě možné konfigurace manipulátoru. Třetí vykreslovací část vykresluje tyto dvě matice resp. proměnné B_LD a B_LN. Pro proměnnou B_LD byla zvolena modrá barva a červená barva pro proměnnou B_LN.



Obrázek 7 Vývojový diagram skriptu pro vykreslení pracovního prostoru

Po spuštění skriptu se zobrazí graf s výsledkem. Ten lze vidět na obrázku (8), kde se nachází dva objemy pro dvě možné konfigurace. Pohotovostní pracovní prostor se nachází v jejich průniku. Vykreslené body spolu dohromady tvoří dosažitelný pracovní prostor.



Obrázek 8 Vykreslený pohotovostní pracovní prostor pro robota SCARA RS40B pomocí systému Matlab

4.4 Výpočet inverzní kinematické úlohy

V této úloze budou analyticky nalezeny vztahy výpočtu jednotlivých kloubových souřadnic pro danou polohu koncového efektoru. K výpočtu je nutné znát vztah z rovnice (15).

Nejprve bude vypočítána kloubová souřadnice h_e , protože tato úloha se řeší od souřadné soustavy koncového efektoru po globální souřadnou soustavu. K výpočtu této kloubové souřadnice je použit vztah pro osu Z_e v rovnici (15). Úpravou této rovnice se získá vztah pro výpočet kloubové souřadnice h_e .

$$h - h_e = Z_e \Rightarrow h_e = h - Z_e \quad (22)$$

K nalezení dalších kloubových souřadnic φ_1 a φ_2 budou potřebné pouze osy X_e a Y_e , protože v ose Z_e se tyto kloubové souřadnice nevyskytují.

$$\begin{bmatrix} X_e \\ Y_e \end{bmatrix} = \begin{bmatrix} -l_2 \cos(\varphi_1 - \varphi_2) - l_1 \cos(\varphi_1) \\ -l_2 \sin(\varphi_1 - \varphi_2) - l_1 \sin(\varphi_1) \end{bmatrix} \quad (23)$$

Nyní bude hledán vztah pro kloubovou souřadnici φ_2 .

$$X_e^2 + Y_e^2 = [-l_1 \cos(\varphi_1) - l_2 \cos(\varphi_1 - \varphi_2)]^2 + [-l_1 \sin(\varphi_1) - l_2 \sin(\varphi_1 - \varphi_2)]^2 \quad (24)$$

$$X_e^2 + Y_e^2 = l_1^2 \cos^2(\varphi_1) + 2l_1 l_2 \cos(\varphi_1) \cos(\varphi_1 - \varphi_2) + l_2^2 \cos^2(\varphi_1 - \varphi_2) + l_1^2 \sin^2(\varphi_1) - 2l_1 l_2 \sin(\varphi_1) \sin(\varphi_1 - \varphi_2) + l_2^2 \sin^2(\varphi_1 - \varphi_2) \quad (25)$$

$$\begin{aligned} X_e^2 + Y_e^2 &= l_1^2 \cos^2(\varphi_1) + 2l_1 l_2 \cos(\varphi_1) \cdot (\cos(\varphi_2) \cos(\varphi_1) + \sin(\varphi_2) \sin(\varphi_1)) \\ &\quad + l_2^2 (\cos(\varphi_2) \cos(\varphi_1) + \sin(\varphi_2) \sin(\varphi_1))^2 + l_1^2 \sin^2(\varphi_1) \\ &\quad - 2l_1 l_2 \sin(\varphi_1) \cdot (\sin(\varphi_2) \cos(\varphi_1) - \cos(\varphi_2) \sin(\varphi_1)) \\ &\quad + l_2^2 (\sin(\varphi_2) \cos(\varphi_1) - \cos(\varphi_2) \sin(\varphi_1))^2 = \\ &= l_1^2 \cos^2(\varphi_1) + 2l_1 l_2 (\cos^2(\varphi_1) \cos(\varphi_2) + \cos(\varphi_1) \sin(\varphi_1) \sin(\varphi_2)) \\ &\quad + l_2^2 (\cos^2(\varphi_1) \cos^2(\varphi_2) + 2 \cos(\varphi_1) \cos(\varphi_2) \sin(\varphi_1) \sin(\varphi_2) \\ &\quad + \sin^2(\varphi_1) \sin^2(\varphi_2)) + l_1^2 \sin^2(\varphi_1) + 2l_1 l_2 (\sin^2(\varphi_1) \cos(\varphi_2) \\ &\quad - \cos(\varphi_1) \sin(\varphi_1) \sin(\varphi_2)) \\ &\quad + l_2^2 (\sin^2(\varphi_2) \cos^2(\varphi_1) - 2 \sin(\varphi_1) \cos(\varphi_2) \cos(\varphi_1) \sin(\varphi_2) \\ &\quad + \cos^2(\varphi_2) \sin^2(\varphi_1)) \end{aligned} \quad (26)$$

$$\begin{aligned} &l_1^2 + 2l_1 l_2 \cos(\varphi_2) \\ &\quad + l_2^2 [\cos^2(\varphi_1) (\cos^2(\varphi_2) + \sin^2(\varphi_2)) \\ &\quad + \sin^2(\varphi_1) (\sin^2(\varphi_2) + \cos^2(\varphi_2))] = \\ &= l_1^2 + l_2^2 + 2l_1 l_2 \cos(\varphi_2) \end{aligned} \quad (27)$$

Byl nalezen výsledek:

$$\cos(\varphi_2) = \frac{X_e^2 + Y_e^2 - l_1^2 - l_2^2}{2l_1 l_2} \quad (28)$$

Použitím vztahu pro goniometrickou funkci tangens se výsledek upraví:

$$\begin{aligned} \tan^2\left(\frac{\varphi_2}{2}\right) &= \frac{1 - \cos(\varphi_2)}{1 + \cos(\varphi_2)} = \frac{1 - \frac{X_e^2 + Y_e^2 - l_1^2 - l_2^2}{2l_1 l_2}}{1 + \frac{X_e^2 + Y_e^2 - l_1^2 - l_2^2}{2l_1 l_2}} = \\ &= \frac{2l_1 l_2 + l_1^2 + l_2^2 - X_e^2 - Y_e^2}{2l_1 l_2 - l_1^2 - l_2^2 + X_e^2 + Y_e^2} = \frac{(l_1 + l_2)^2 - (X_e^2 + Y_e^2)}{(X_e^2 + Y_e^2) - (l_1 - l_2)^2} \end{aligned} \quad (29)$$

Konečným výsledkem je:

$$\varphi_2 = \pm 2 \cdot \operatorname{atan} \left(\sqrt{\frac{(l_1 + l_2)^2 - (X_e^2 + Y_e^2)}{(X_e^2 + Y_e^2) - (l_1 - l_2)^2}} \right) \quad (30)$$

Znaménko \pm v rovnici (30) znamená dvě řešení, která se nazývají loket nahoru a loket dolů.

Zbývá dopočítat kloubovou souřadnici φ_1 , pro zmíněné dvě řešení. Tedy:

$$\begin{aligned} X_e \cdot (l_1 + l_2 \cos(\varphi_2)) - Y_e \cdot l_2 \sin(\varphi_2) &= [-l_2 \cos(\varphi_2 - \varphi_1) - l_1 \cos(\varphi_1)] \cdot (l_1 + l_2 \cos(\varphi_2)) \\ &\quad - [l_2 \sin(\varphi_2 - \varphi_1) - l_1 \sin(\varphi_1)] \cdot l_2 \sin(\varphi_2) \\ &= -l_1^2 \cos(\varphi_1) - l_1 l_2 (\cos(\varphi_2) \cos(\varphi_1) + \sin(\varphi_2) \sin(\varphi_1)) \\ &\quad - l_1 l_2 \cos(\varphi_1) \cos(\varphi_2) \\ &\quad - l_2^2 \cos(\varphi_2) (\cos(\varphi_2) \cos(\varphi_1) + \sin(\varphi_2) \sin(\varphi_1)) \\ &\quad + l_1 l_2 \sin(\varphi_1) \sin(\varphi_2) \\ &\quad - l_2^2 \sin(\varphi_2) (\sin(\varphi_2) \cos(\varphi_1) - \cos(\varphi_2) \sin(\varphi_1)) \\ &= -l_1^2 \cos(\varphi_1) - 2l_1 l_2 \cos(\varphi_1) \cos(\varphi_2) - l_2^2 (\cos^2(\varphi_2) \cos(\varphi_1) \\ &\quad + \cos(\varphi_2) \sin(\varphi_1) \sin(\varphi_2) - \sin(\varphi_1) \sin(\varphi_2) \cos(\varphi_2) \\ &\quad + \sin^2(\varphi_2) \cos(\varphi_1)) \\ &= -l_1^2 \cos(\varphi_1) - 2l_1 l_2 \cos(\varphi_1) \cos(\varphi_2) - l_2^2 \cos(\varphi_1) \\ &\quad \underbrace{}_{X_e^2 + Y_e^2} \\ &= -\cos(\varphi_1) \cdot (l_1^2 + 2l_1 l_2 \cos(\varphi_2) + l_2^2) \\ &= -(X_e^2 + Y_e^2) \cdot \cos(\varphi_1) \end{aligned} \quad (31)$$

První výsledek je:

$$\cos(\varphi_1) = -\frac{X_e(l_1 + l_2 \cos(\varphi_2)) - Y_e \cdot l_2 \sin(\varphi_2)}{(X_e^2 + Y_e^2)} \quad (32)$$

A podobně:

$$\begin{aligned} Y_e(l_1 + l_2 \cos(\varphi_2)) - X_e \cdot l_2 \sin(\varphi_2) &= [l_2 \sin(\varphi_2 - \varphi_1) - l_1 \sin(\varphi_1)] \cdot (l_1 + l_2 \cos(\varphi_2)) \\ &\quad - [l_2 \cos(\varphi_2 - \varphi_1) + l_1 \cos(\varphi_1)] \cdot l_2 \sin(\varphi_2) \\ &= -l_1^2 \cdot \sin(\varphi_1) + l_1 \cdot l_2 \\ &\quad \cdot (\sin(\varphi_2) \cdot \cos(\varphi_1) - \cos(\varphi_2) \cdot \sin(\varphi_1)) - l_1 \cdot l_2 \\ &\quad \cdot \sin(\varphi_1) \cdot \cos(\varphi_2) \\ &\quad + l_2^2 \cdot \cos(\varphi_2) \cdot (\sin(\varphi_2) \cdot \cos(\varphi_1) - \cos(\varphi_2) \cdot \sin(\varphi_1)) - l_1 \cdot l_2 \\ &\quad \cdot \cos(\varphi_1) \cdot \sin(\varphi_2) \\ &\quad - l_2^2 \cdot \sin(\varphi_2) (\cos(\varphi_2) \cdot \cos(\varphi_1) + \sin(\varphi_2) \cdot \sin(\varphi_1)) \\ &= -l_1^2 \cdot \sin(\varphi_1) - 2 \cdot l_1 \cdot l_2 \cdot \sin(\varphi_1) \cdot \cos(\varphi_2) \\ &\quad - l_2^2 (\sin(\varphi_2) \cdot \cos(\varphi_2) \cdot \cos(\varphi_1) + \sin^2(\varphi_2) \cdot \sin(\varphi_1)) \\ &\quad + \cos^2(\varphi_2) \cdot \sin(\varphi_1) - \sin(\varphi_2) \cdot \cos(\varphi_2) \cdot \cos(\varphi_1)) \\ &= -l_1^2 \cdot \sin(\varphi_1) - 2 \cdot l_1 \cdot l_2 \cdot \sin(\varphi_1) \cdot \cos(\varphi_2) - l_2^2 \cdot \sin(\varphi_1) \\ &\quad \underbrace{}_{X_e^2 + Y_e^2} \\ &= -\sin(\varphi_1) \cdot (l_1^2 + 2l_1 l_2 \cos(\varphi_2) + l_2^2) = -(X_e^2 + Y_e^2) \cdot \sin(\varphi_1) \end{aligned} \quad (33)$$

Druhý výsledek je:

$$\sin \varphi_1 = -\frac{Y_e(l_1 + l_2 \cos(\varphi_2)) + X_e \cdot l_2 \sin(\varphi_2)}{(X_e^2 + Y_e^2)} \quad (34)$$

Použitím vztahu pro goniometrickou funkci tangens se výsledek upraví:

$$\begin{aligned} \tan(\varphi_1) &= \frac{\sin(\varphi_1)}{\cos(\varphi_1)} = \frac{-\frac{Y_e(l_1 + l_2 \cos(\varphi_2)) + X_e \cdot l_2 \sin(\varphi_2)}{(X_e^2 + Y_e^2)}}{-\frac{X_e(l_1 + l_2 \cos(\varphi_2)) - Y_e \cdot l_2 \sin(\varphi_2)}{(X_e^2 + Y_e^2)}} = \\ &= \frac{-Y_e \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) - X_e \cdot l_2 \cdot \sin(\varphi_2)}{-X_e \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) + Y_e \cdot l_2 \cdot \sin(\varphi_2)} \end{aligned} \quad (35)$$

Konečným výsledkem je:

$$\varphi_1 = \operatorname{atan}\left(\frac{-Y_e \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) - X_e \cdot l_2 \cdot \sin(\varphi_2)}{-X_e \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) + Y_e \cdot l_2 \cdot \sin(\varphi_2)}\right) \quad (36)$$

4.4.1 Ověření inverzní kinematické úlohy na kinematickém modelu

K ověření inverzní kinematické úlohy je využita kružnice s poloměrem 50 mm a počátkem zvoleným tak, aby celá kružnice ležela v pracovním prostoru robotu. Počátek kružnice se tedy v globální souřadné soustavě nachází na souřadnicích:

$$\begin{bmatrix} x_{pk} \\ y_{pk} \\ z_{pk} \end{bmatrix} = \begin{bmatrix} -250 \\ 0 \\ 250 \end{bmatrix} \quad (37)$$

Nejprve se inverzní kinematická úloha ověří pro dva body dané kružnice. Tyto dva body se vypočítají pomocí parametrické rovnice kružnice.

První bod P_{Bk1} bude s úhlem 0 rad. Tedy:

$$x_{P_{Bk1}} = x_{pk} + r_k \cdot \cos(0) = -250 + 50 = -200 \text{ mm} \quad (38)$$

$$y_{P_{Bk1}} = y_{pk} + r_k \cdot \sin(0) = 0 + 0 = 0 \text{ mm} \quad (39)$$

$$z_{P_{Bk1}} = z_{pk} + r_k \cdot \cos(0) = 250 + 50 = 300 \text{ mm} \quad (40)$$

Druhý bod P_{Bk2} bude s úhlem 1,6336282 rad. Tedy:

$$\begin{aligned} x_{P_{Bk2}} &= x_{pk} + r_k \cdot \cos(1,6336282) = -250 - 3,1395260 \\ &= -253,13953 \text{ mm} \end{aligned} \quad (41)$$

$$y_{P_{Bk2}} = y_{pk} + r_k \cdot \sin(1,6336282) = 0 + 49,901336 = 49,901336 \text{ mm} \quad (42)$$

$$z_{P_{Bk2}} = z_{pk} + r_k \cdot \cos(1,6336282) = 250 - 3,1395260 = 246,86047 \text{ mm} \quad (43)$$

Pro body P_{Bk1} a P_{Bk2} se vypočítají jednotlivé kloubové souřadnice z rovnic (22), (30) a (36) získaných z inverzní kinematické úlohy.

Kloubová souřadnice φ_2 pro bod kružnice P_{Bk1} :

$$\begin{aligned} \varphi_{2,P_{Bk1}} &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{(l_1+l_2)^2 - (x_{P_{Bk1}}^2 + y_{P_{Bk1}}^2)}}{\sqrt{(x_{P_{Bk1}}^2 + y_{P_{Bk1}}^2) - (l_1-l_2)^2}} \right) = \\ &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{(230+170)^2 - (40000)}}{\sqrt{(40000) - (230-170)^2}} \right) \\ &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{160000 - 40000}}{\sqrt{40000 - 3600}} \right) = \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{120000}}{\sqrt{36400}} \right) \end{aligned} \quad (44)$$

$$\varphi_{2,P_{Bk1}} = \pm 2,1347441 \text{ rad} \quad (45)$$

Kloubová souřadnice φ_2 pro bod kružnice P_{Bk2} :

$$\begin{aligned} \varphi_{2,P_{Bk2}} &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{(l_1+l_2)^2 - (x_{P_{Bk2}}^2 + y_{P_{Bk2}}^2)}}{\sqrt{(x_{P_{Bk2}}^2 + y_{P_{Bk2}}^2) - (l_1-l_2)^2}} \right) \\ &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{(230+170)^2 - (64079,622 + 2490,1433)}}{\sqrt{(64079,622 + 2490,1433) - (230-170)^2}} \right) \\ &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{160000 - 66569,765}}{\sqrt{66569,765 - 3600}} \right) \\ &= \pm 2 \cdot \operatorname{atan2} \left(\frac{\sqrt{93430,235}}{\sqrt{62969,765}} \right) \end{aligned} \quad (46)$$

$$\varphi_{2,P_{Bk2}} = \pm 1,7668092 \text{ rad} \quad (47)$$

Kloubová souřadnice φ_1 pro bod kružnice P_{Bk1} :

$$\begin{aligned} \varphi_{1,P_{Bk1}} &= \text{atan2} \left(\frac{-y_{P_{Bk1}} \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) - x_{P_{Bk1}} \cdot l_2 \cdot \sin(\varphi_2)}{-x_{P_{Bk1}} \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) + y_{P_{Bk1}} \cdot l_2 \cdot \sin(\varphi_2)} \right) \\ &= \text{atan2} \left(\frac{200 \cdot 170 \cdot \sin(\pm 2,1347441)}{200 \cdot (230 + 170 \cdot \cos(\pm 2,1347441))} \right) \\ &= \text{atan2} \left(\frac{34000 \cdot \sin(\pm 2,1347441)}{46000 + 34000 \cdot \cos(\pm 2,1347441)} \right) \end{aligned} \quad (48)$$

$$\varphi_{1,P_{Bk1}} = \pm 0,8014690 \text{ rad} \quad (49)$$

Kloubová souřadnice φ_1 pro bod kružnice P_{Bk2} :

$$\begin{aligned} \varphi_{1,P_{Bk2}} &= \text{atan2} \left(\frac{-y_{P_{Bk2}} \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) - x_{P_{Bk2}} \cdot l_2 \cdot \sin(\varphi_2)}{-x_{P_{Bk2}} \cdot (l_1 + l_2 \cdot \cos(\varphi_2)) + y_{P_{Bk2}} \cdot l_2 \cdot \sin(\varphi_2)} \right) \\ &= \text{atan2} \left(\frac{-49,901336 \cdot (230 + 170 \cdot \cos(\pm 1,7668092)) + 253,13953 \cdot 170 \cdot \sin(\pm 1,7668092)}{253,13953 \cdot (230 + 170 \cdot \cos(\pm 1,7668092)) + 49,901336 \cdot 170 \cdot \sin(\pm 1,7668092)} \right) \\ &= \text{atan2} \left(\frac{-11477,307 - 8483,2271 \cdot \cos(\pm 1,7668092) + 43033,720 \cdot \sin(\pm 1,7668092)}{58222,092 + 43033,720 \cdot \cos(\pm 1,7668092) + 8483,2271 \cdot \sin(\pm 1,7668092)} \right) \end{aligned} \quad (50)$$

$$\varphi_{1,P_{Bk2}} = -0,8973194 \text{ rad} \quad (51)$$

$$\varphi_{1,P_{Bk2}} = 0,5080510 \text{ rad} \quad (52)$$

Kloubové souřadnice φ_1 a φ_2 mají dva výsledky pro dvě možné konfigurace manipulátoru daného typu robota. V případě zmíněné kružnice se bod koncového efektoru vždy nachází v pracovním prostoru, proto jsou všechny výsledky kloubových souřadnic pro oba body P_{Bk1} a P_{Bk2} platné.

Kloubová souřadnice h_e , pro bod kružnice P_{Bk1} se vypočítá:

$$h_{e,P_{Bk1}} = 300 - 300 = 0 \text{ mm} \quad (53)$$

Kloubová souřadnice h_e , pro bod kružnice P_{Bk2} se vypočítá:

$$h_{e,P_{Bk2}} = 300 - 246,86047 = 53,13953 \text{ mm} \quad (54)$$

Vypočítané kloubové souřadnice se dosadí do kinematického modelu z rovnice (15). Pro kloubové souřadnice φ_1 a φ_2 se vypočítají konfigurace loket nahoru a loket dolů.

Tedy:

Pro bod P_{Bk1} v konfiguraci loket nahoru:

$$\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} = \begin{bmatrix} -170 \cdot \cos(0,8014690 - 2,1347441) - 230 \cdot \cos(0,8014690) \\ -170 \cdot \sin(0,8014690 - 2,1347441) - 230 \cdot \sin(0,8014690) \\ 300 - 0 \end{bmatrix} = \begin{bmatrix} -200 \\ 0 \\ 300 \end{bmatrix} \quad (55)$$

Pro bod P_{Bk1} v konfiguraci loket dolů:

$$\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} = \begin{bmatrix} -170 \cdot \cos(-0,8014690 + 2,1347441) - 230 \cdot \cos(-0,8014690) \\ -170 \cdot \sin(-0,8014690 + 2,1347441) - 230 \cdot \sin(-0,8014690) \\ 300 - 0 \end{bmatrix} = \begin{bmatrix} -200 \\ 0 \\ 300 \end{bmatrix} \quad (56)$$

Pro bod P_{Bk2} v konfiguraci loket dolů:

$$\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} = \begin{bmatrix} -170 \cdot \cos(-0,8973194 + 1,7668092) - 230 \cdot \cos(-0,8973194) \\ -170 \cdot \sin(-0,8973194 + 1,7668092) - 230 \cdot \sin(-0,8973194) \\ 300 - 53,13953 \end{bmatrix} = \begin{bmatrix} -253,13953 \\ 49,901336 \\ 246,86047 \end{bmatrix} \quad (57)$$

Pro bod P_{Bk2} v konfiguraci loket nahoru:

$$\begin{bmatrix} X_e \\ Y_e \\ Z_e \end{bmatrix} = \begin{bmatrix} -170 \cdot \cos(0,5080510 - 1,7668092) - 230 \cdot \cos(0,5080510) \\ -170 \cdot \sin(0,5080510 - 1,7668092) - 230 \cdot \sin(0,5080510) \\ 300 - 53,13953 \end{bmatrix} = \begin{bmatrix} -253,13953 \\ 49,901336 \\ 246,86047 \end{bmatrix} \quad (58)$$

Z výsledků je zřejmé, že se shodují se zadanými body P_{Bk1} a P_{Bk2} . Tím je ověřeno, že inverzní kinematická úloha byla vypočítána správně. Z výpočtu je jasné, že ověření inverzní kinematické úlohy zabere hodně místa a času. Doposud ověření závisí na dvou bodech z celé kružnice. Pro úplnost ověření byl vytvořen skript pro ověření inverzní kinematické úlohy v příloze P II, který provede ověření pro všechny body kružnice. Výhoda tohoto skriptu je rychlost, s jakou se vypočítají všechny výsledky ověření a hlavně možnost jejich vizualizace.

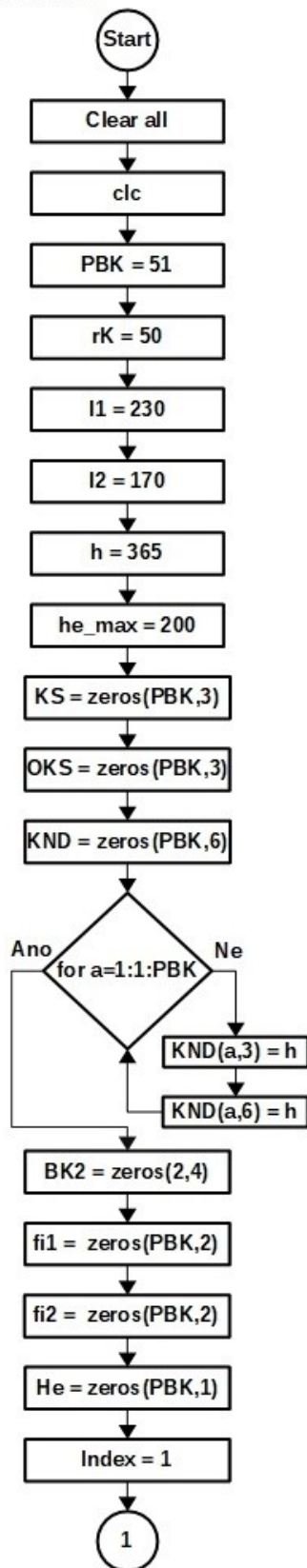
Pro orientaci v tomto skriptu je v tabulce (4) uveden výčet proměnných a jejich význam.

Název proměnné	Význam
a	Proměnné pro smyčky for
c	
c	
d	
e	
uhel	
index	Vkládání výsledků na daný index
pol	Index polohy bodu mimo začátek
h	DH parametr h
l1	DH parametr l_1
l2	DH parametr l_2
f1	Kloubová proměnná φ_1
f2	Kloubová proměnná φ_2
he	Kloubová proměnná h_e
he_max	Maximální vysunutí koncového efektoru
rK	Poloměr kruhu
BK2	Body druhého kloubu pro loket nahoru a dolů
KND	Body kloubu nahoru a dolů
KS	Kloubové souřadnice
OKS	Ověřené kloubové souřadnice
PK	Počátek kruhu
PBK	Počet bodů kruhu
DU	Délka úsečky
VB	Velikost vykreslovaných bodů

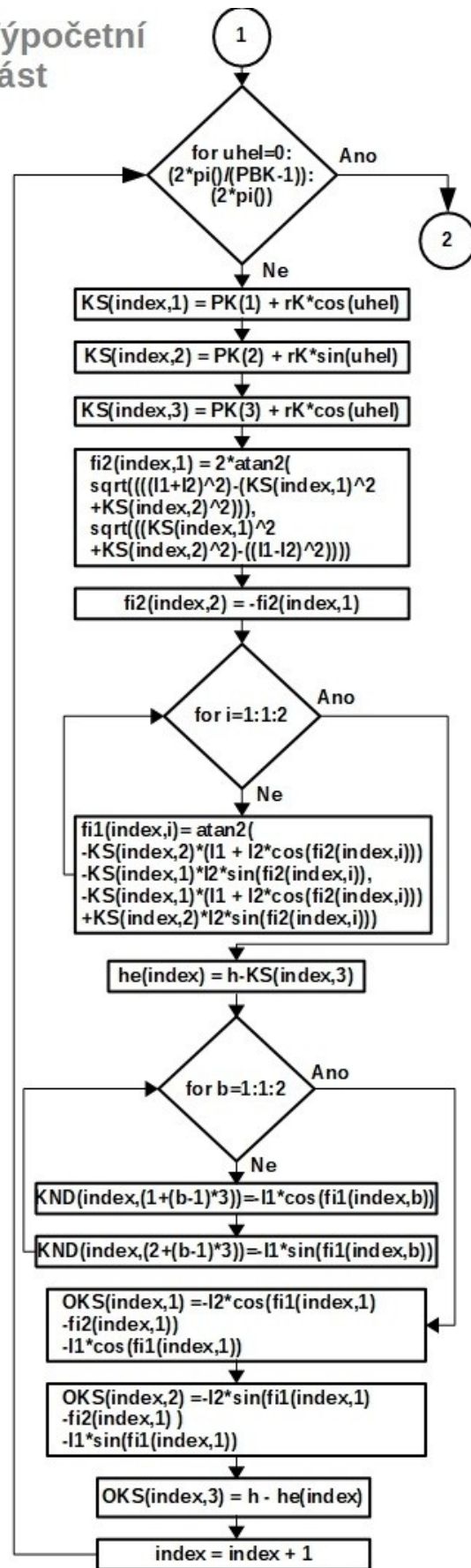
Tabulka 4 Proměnné skriptu BP_overeneInvKin

Na obrázku (9) a (10) je zobrazen algoritmus tohoto skriptu. Je rozdělen do pomyslných tří částí. První inicializační část slouží pouze k inicializaci proměnných. Druhá výpočetní část obsahuje vzorce nejen pro výpočet bodů kruhu, ale i kloubových souřadnic včetně jejich dosazení do kinematického modelu. Jinými slovy se ve výpočetní části programu odehrává stejný výpočet jako výše, ale pro všechny body dané kružnice. Třetí vykreslovací část skriptu slouží pouze k vykreslení výsledků do grafu.

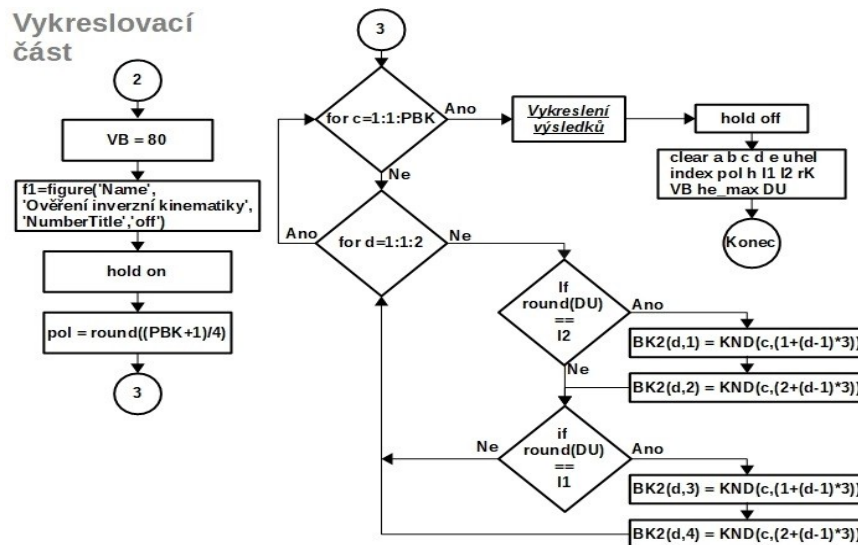
Inicializační část



Výpočetní část

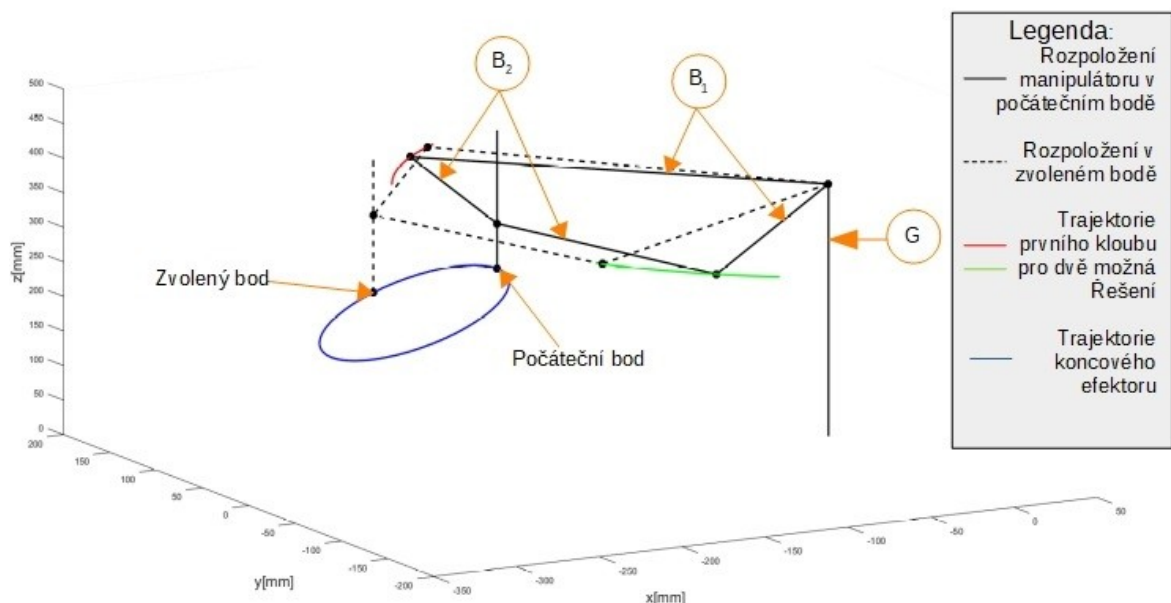


Obrázek 9 Vývojový diagram skriptu pro ověření inverzní kinematické úlohy 1.část



Obrázek 10 Vývojový diagram skriptu pro ověření inverzní kinematické úlohy 2.část

Výsledkem ověření inverzní kinematické úlohy pomocí skriptu z přílohy P II je obrázek (11). Černé úsečky značí články manipulátoru. Plné černé úsečky vyznačují dvě konfigurace článků manipulátoru v počátečním bodě, tedy bodě P_{BK1} . Stejná logika je aplikovaná na čárkované úsečky, ale pro jiný zvolený bod, tedy P_{BK2} . Červená a zelená čára značí trajektorii prvního kloubu pro dvě možná řešení. Modrá kružnice je stejná jako kružnice využitá výše při výpočtu bodů P_{BK1} a P_{BK2} , které jsou na obrázku (11) zakresleny jako bod počátku a zvolený bod. Modrá kružnice je vykreslena z bodů, které byly ověřeny pomocí kinematického modelu. Originální kružnice není vykreslena.



Obrázek 11 Ověření inverzní kinematické úlohy

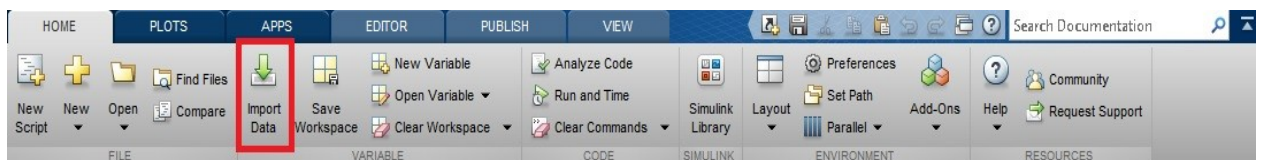
5 TYPOVÁ ÚLOHA PODPIS

V této úloze robot Stäubli RS40b typu SCARA reprodukuje podpis napsaný psacím písmem.

5.1 Digitalizace podpisu

Nejprve je nutné se podepsat na čistý papír psacím písmem a tenkým černým fixem. Jméno a příjmení se napíše jednotlivě na půlky bílého listu A4. Tento list s podpisem se naskenuje do počítače jako obrázek typu JPG.

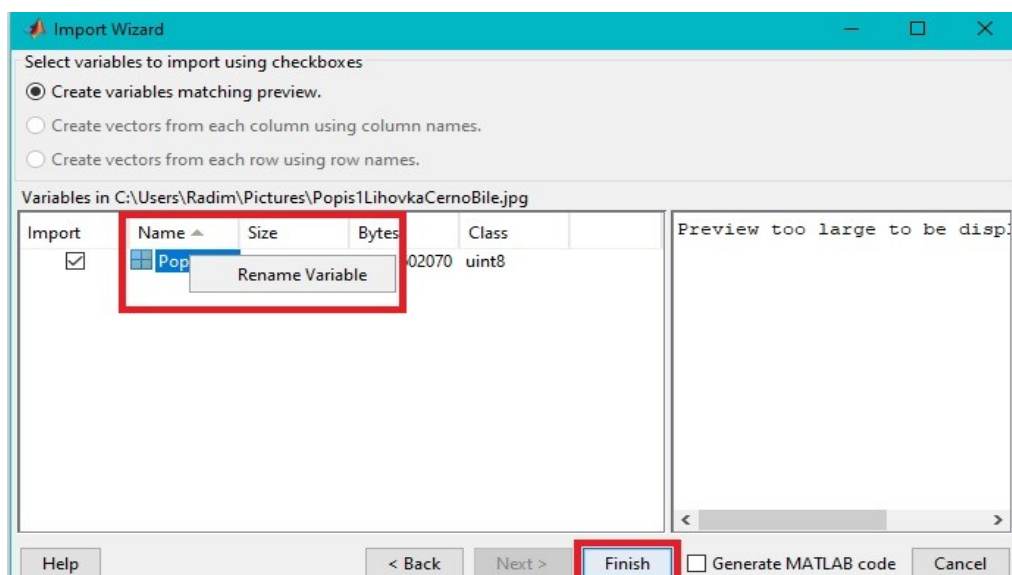
Obrázek s podpisem se importuje do systému Matlab. V záložce HOME se klikne na ikonu s nápisem Import Data, která je červeně označena na obrázku (12).



Obrázek 12 Import dat do programu Matlab

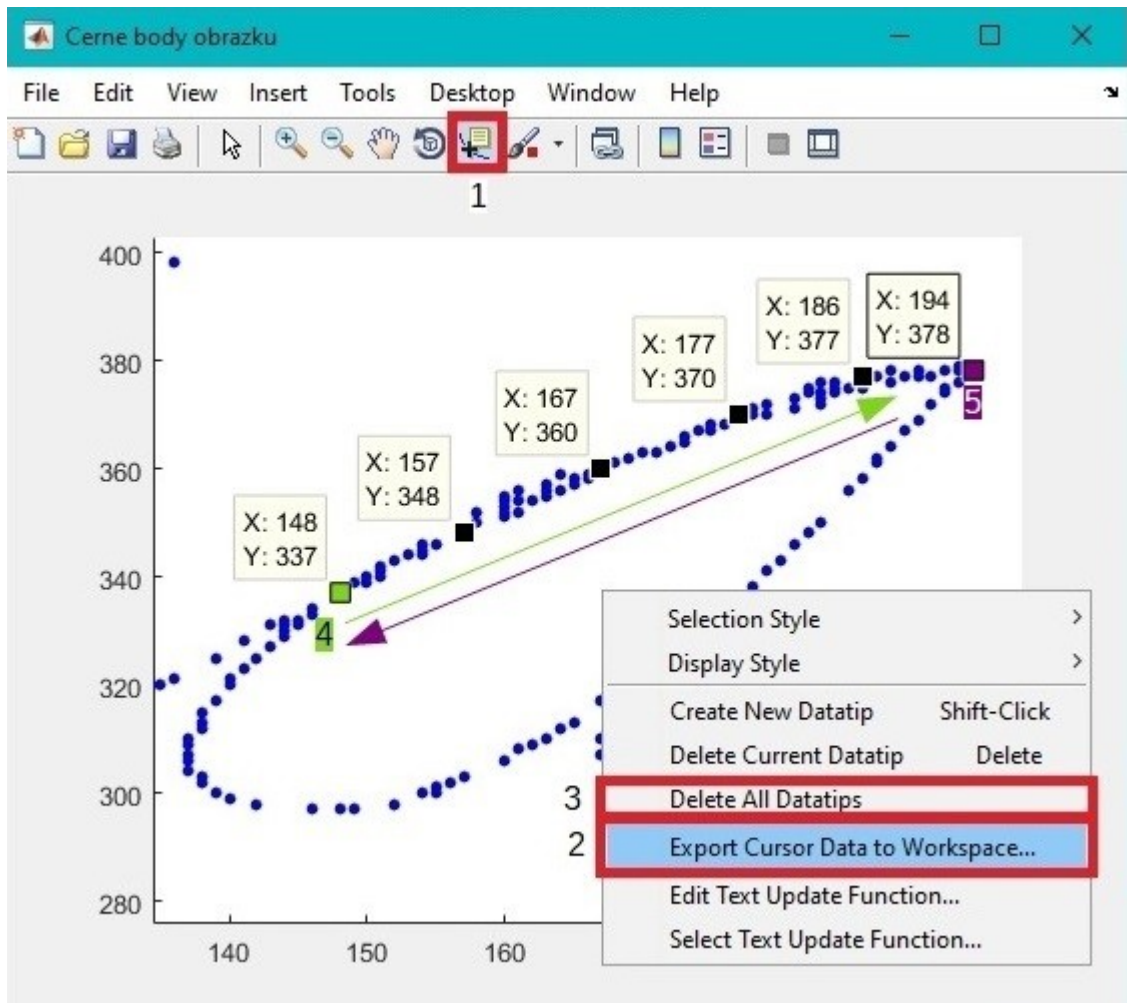
V okně, které se zobrazí, se vybere obrázek s podpisem z místa na disku, kde se nachází, a výběr se potvrdí tlačítkem otevřít.

Zobrazí se Import Wizard, viz. obrázek (13), a zde se přejmenuje název obrázku s podpisem v sloupci Name na Objekt. Kliknutím na tlačítko Finish se obrázek pod názvem Objekt zobrazí v proměnných v systému Matlab.



Obrázek 13 Import Wizard programu Matlab

Nyní se spustí skript z přílohy P III. Tento skript uloží všechny černé body z vloženého obrázku pod názvem Objekt do pole s názvem „*CerneBody*“ a následně tyto body zobrazí do nově vytvořeného okna s názvem „*Cerne body obrázku*“, viz obrázek (14).



Obrázek 14 Využití datového kurzoru v programu Matlab

Pořadí bodů je nutné určit pomocí nástroje „*Data Cursor*“. Tento nástroj se nachází v horní liště okna „*Cerne body obrázku*“. Na obrázku (14) je označen číslem jedna. Kliknutím na něj a držením klávesy shift je možné označit body v pořadí, ve kterém byly označeny.

K seřazení bodů pomocí nástroje „*Data Cursor*“ lze využít dva postupy:

1. Jméno nebo příjmení se bude skládat z malých skupin seřazených bodů. Například u písmen a, d, p atd., kde je nutné se vracet po již označených bodech. Příklad je uveden na obrázku (14), kde od zeleného bodu označeného číslem 4 jsou body značeny po směru zelené šipky k fialovému bodu číslo 5 a pro označení stejných bodů v opačném pořadí po směru fialové šipky je využit skript z přílohy P IV, jehož výstupem je matice „*BodyVystup*“. Ta obsahuje všechny body včetně jejich

překlopeného pořadí. Výhody této metody jsou malá šance ztráty všech označených bodů, snadné úpravy a přehlednost.

2. Jméno nebo příjmení označíme do jedné velké skupiny bodů. Tento postup je nevýhodný z důvodu nepřehlednosti. Konkrétně při označení pořadí bodů u písmen a, d, p atd. nebo zvolení špatného pořadí. Výhoda tohoto postupu spočívá v tom, že není nutné skládat podpis po malých skupinách bodů.

Pro uložení seřazených bodů se musí kliknout pravým tlačítkem myši kdekoli v okně a zvolit možnost „*Export Cursor Data to Workspace*“. Na obrázku (14) je tato volba označena číslem 2. Následně se zobrazí okno pro pojmenování matice, do které se body uloží. Uživatelsky se nabídne název „*cursor_info*“.

Před novou skupinou bodů je nutné uložit označené body pod jiným názvem a odstranit „*Datatips*“, nebo-li datové listy. Tato možnost je k nalezení při stlačení pravého tlačítka myši v okně „*Cerne body obrazku*“ viz číslo 3 na obrázku (14).

5.2 Programování podpisu

Vytvoření projektu v systému Stäubli Robotic Suite je vhodné z důvodu snadného vložení bodů podpisu a simulaci programu.

Nejprve je nutné si vytvořit novou buňku v záložce „*New*“ na levé straně okna pomocí nástroje „*New cell wizard*“. Po zvolení této možnosti bude v novém okně vyžádáno jméno buňky a její umístění. Po potvrzení se zobrazí další okno, kde je nutné zvolit kontrolér. Z nabízených možností se zvolí „*Add a local controller*“. Zobrazí se nabídka pro výběr robotických ramen. Zde se pod nabídkou „*Arm family*“ vybere „*RS*“ a pod nabídkou „*Arm model*“ „*rs40b*“ a zvolí se příslušný robot. Následuje okno s nastavením možností kontroléru k zvolenému robotickému rameni. Je důležité použít stejnou verzi, kterou má reálné zařízení. V tomto případě je to verze „*s6.6.4*“. Potvrzením nastavení kontroléru se otevře poslední okno se shrnutím.

Takto vytvořená buňka nemá potřebný simulační model ani aplikaci. Aplikace se vytvoří pravým kliknutím myši na kontrolér v „*Cell Explorer*“ a vybere se první možnost „*New Application*“ nebo se použije klávesová zkratka CTRL+N,A. Následně je vyžadován název aplikace, umístění a template. Potvrzením výběru vytvoří aplikaci, která se nachází v „*Cell Explorer*“ pod záložkou kontroléru.

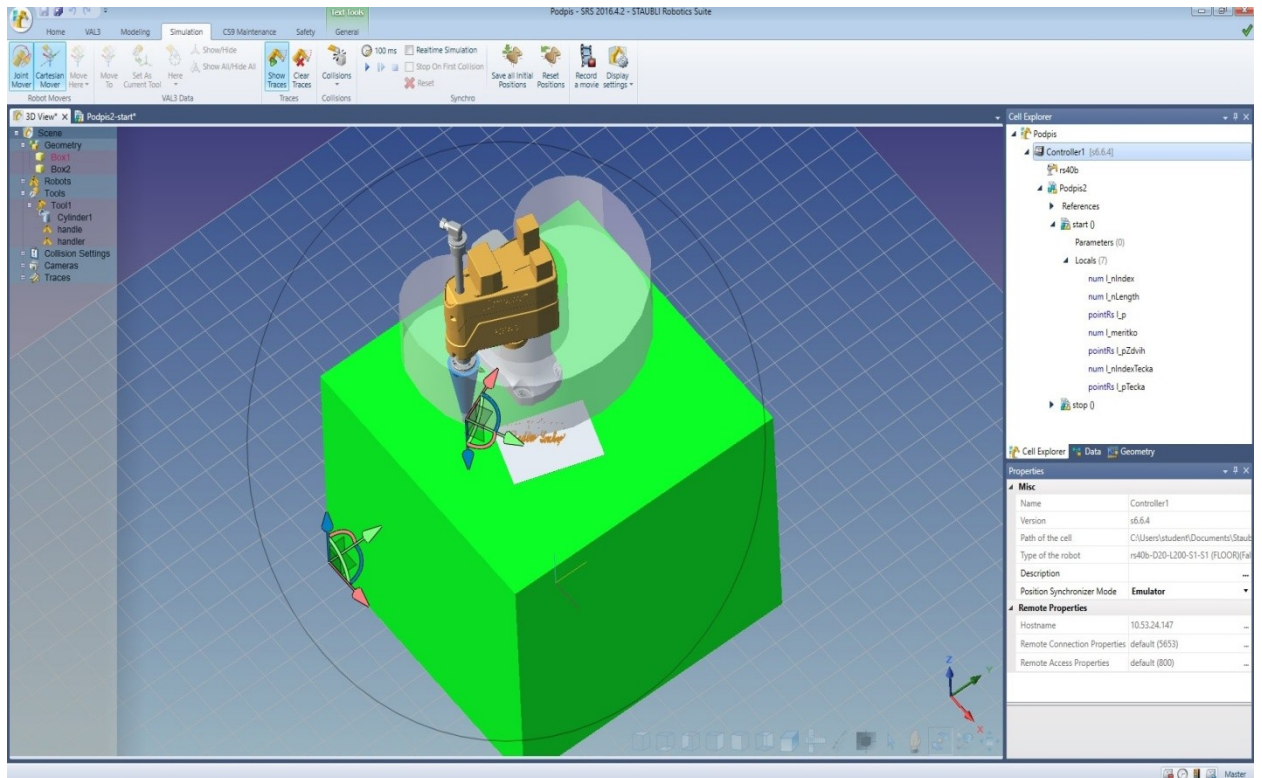
Do vytvořené aplikace lze vkládat proměnné pravým kliknutím myši na „Locals“ programu start(). Obdobně, ale v záložce „Data“ pravým kliknutím myši na aplikaci je možné vložit nová data. Pro vložení již vytvořených bodů je využit datový typ „pointRS“, který se příhodně přejmenuje a nastaví se jeho velikost. Velikost je rovna počtu bodů, které budou do dat vloženy. Otevřením dat se zobrazí tabulka prázdných hodnot. Do této tabulky se jednoduše přepokopí data bodů jména nebo příjmení.

Model pro simulaci bude tvořen stolem, robotem, papírem a nástrojem. V záložce „Home“ se kliknutím myši na „Show 3D view“ zobrazí simulační model.

Robot se v modelu nachází od vytvoření aplikace, ale v modelu chybí stůl, nástroj a papír. Stůl i papír se vytvoří z krabice. Krabice se do modelu vkládá ze záložky „Modeling“ kliknutím na výběr „Add“ a položku „Box“. Nástroj je vytvořen z cylindru, který se vkládá podobně jako krabice, ale jako položka se vybere „Cylinder“. Těmto objektům se musí nastavit vlastnosti. Při označení objektu se v okně vpravo dole zobrazí „Properties“. V „Properties“ lze nastavit rozměry a pojmenovat objekt. V této práci má stůl rozměry $x = 1000$ mm, $y = 1000$ mm a $z = 750$ mm. Papír má rozměry papíru A4 tedy $x = 210$ mm, $y = 297$ mm a $z = 1$ mm. Rozměry cylindru jsou výška = 230 mm, spodní rádius = 10 mm, horní rádius = 50 mm a základní vrcholy = 30 mm.

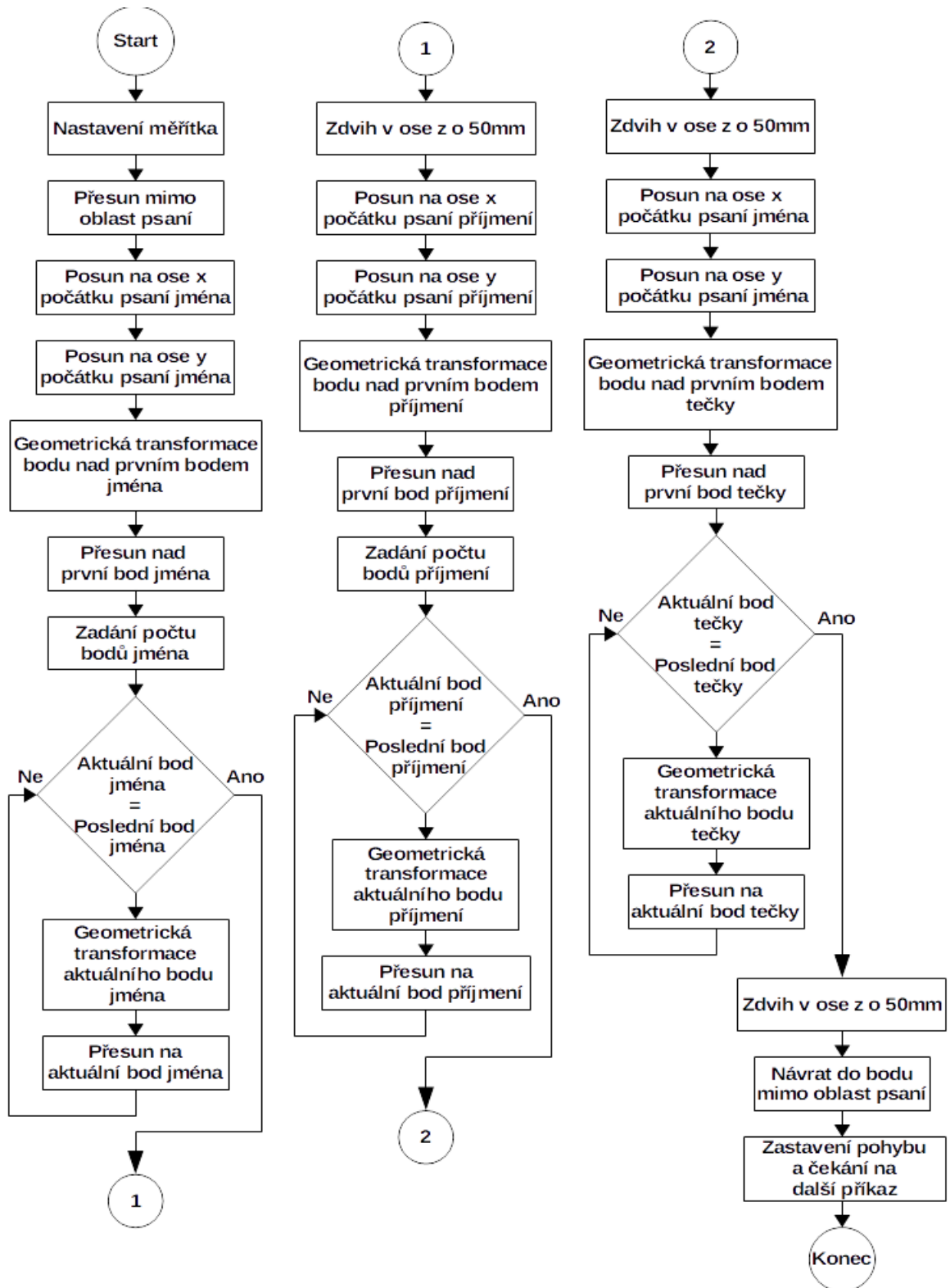
Cylindru bylo nutné upravit souřadnou soustavu, aby mohl být vhodně použit jako nástroj. Pro upravení souřadné soustavy cylindru do jeho spodní části bylo využito nástroje „Edit Reference Frame“ v záložce „Modeling“. Po této úpravě se z cylindru vytvořil nástroj v záložce „Modeling“ zvolením výběru „New“ a kliknutím na položku „Tool“. Nástroj se spojí s robotem přetažením zástupce nástroje na zástupce robota v levé postranní liště s názvem „Scene“.

Konečnou úpravou simulačního modelu bylo umístění robota s papírem na stůl. K tomu byly využity nástroje skupiny „Object Positionners“ v záložce „Modeling“. Výsledný simulační model lze vidět na obrázku (15).



Obrázek 15 Simulační model robota RS40b v systému Staubli Robotic Suite

Ve vytvořené aplikaci se vlastní kód píše do programu `start()` mezi příkazy `begin` a `end`. Tento program se nachází v záložce kontroléru. Vývojový diagram programu pro podepsání lze vidět na obrázku (16). Z tohoto vývojového diagramu je jasné, že program nerozlišuje mezi jménem, příjmením a tečkou. Vše považuje za pole bodů. Program nejprve uvede nástroj do bodu bezpečí. Z tohoto bodu se přesune nad počáteční bod pro psaní jména a spustí se do počátečního bodu, od kterého začne psát. Po dokončení jména se nástroj zvedne nad poslední bod jména a přesune se nad první bod příjmení. Spustí se do bodu příjmení a začne psát příjmení. Po dokončení příjmení se nástroj zvedne nad posledním bodem příjmení a přesune se nad první bod tečky. Tečka je v tomto případě kružnice, která na reálném zařízení vytvoří tečku, protože má dostatečně malý poloměr. Po dokončení tečky se nástroj opět zvedne a přesune se do bodu bezpečí. Tím program končí.

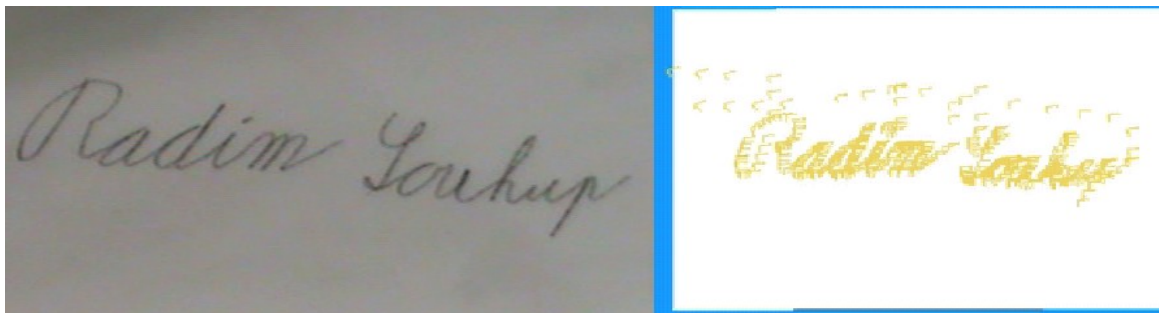


Obrázek 16 Vývojový diagram programu start()

Pro simulaci se program nejprve spustí v emulátoru. V Cell Exploreru při kliknutí pravého tlačítka myši na kontrolér se nabídne „debug on“ a levým tlačítkem myši se zvolí „Emulator“ nebo se použije klávesová zkratka CTRL+D,E. Emulátor funguje úplně stejně

jako ruční ovládání. Než se v emulátor zapne, je nutné zvolit v záložce „Simulation“ „Start synchro“, které má modrou ikonu play. V té samé záložce se zapne „Show Traces“, aby byl vidět podpis. Nyní se v emulátoru zapnou motory a vytvořená aplikace.

Po úspěšné simulaci se program nahraje do reálného zařízení, kde se pouze změní bod jBezpeci a souřadný systém papíru na ovládacím panelu robotu. Po této malé úpravě se program spustí a jeho výsledek je vidět na obrázku (17), kde je navíc porovnán s výsledkem simulace.



Obrázek 17 Porovnání podpisu z reálného zařízení (vlevo) a simulace (vpravo)

ZÁVĚR

Cílem bakalářské práce bylo naprogramovat robota Stäubli RS40b typu SCARA tak, aby napsal celé jméno psacím písmem. Podpis byl digitalizován v systému Matlab. Pro naprogramování a simulaci byl využit systém Stäubli Robotic Suite. Po úspěšné simulaci byla úloha provedena na reálném zařízení. Výsledkem byl, až na detaily, totožný podpis. Důvodem, proč podpis není totožný, je odpružení nástroje, který používá reálné zařízení.

V úvodu teoretické části bakalářské práce je vysvětleno, čím se zabývá robotika. Nachází se zde definice průmyslového robotu a jeho dělení do skupin podle uspořádání mechanického systému, manipulačního zařízení a generací. Jsou zde popsány programovací metody průmyslového robotu a robot typu SCARA.

Pokračování teoretické části tvoří kinematická analýza, která je základem pro praktickou část. Je zde vysvětlena kinematika průmyslových robotů, která se dělí do úloh na přímou a inverzní kinematickou úlohu. Dále je zde definována DH notace a postup pro zavedení souřadných soustav v jejím souladu. Rovněž se zde nachází popis transformace mezi souřadnými systémy a vysvětlení pojmu pracovní prostor.

V praktické části byl vytvořen 3D model robota Stäubli RS40b typu SCARA v systému SolidWorks. Tento 3D model byl základem k zavedení souřadných soustav v souladu s DH notací. Na základě zavedených souřadných systémů a DH parametrů byla vypočítána přímá kinematická úloha. S pomocí systému Matlab byl vykreslen pohotovostní pracovní prostor robota Stäubli RS40b. Dále byla vypočítána inverzní kinematická úloha a ověřena pomocí kinematického modelu. Ověření inverzní kinematické úlohy bylo graficky znázorněno pomocí systému Matlab. Praktická část pokračuje úplnou metodikou reálného cvičení pro naprogramování typové úlohy v programu Stäubli Robotic Suite pro reprodukci podpisu pomocí robota Stäubli RS40b.

Z výše uvedeného vyplývá, že všechny body této práce byly splněny. Při dodržení metodiky bude student schopen reprodukovat vlastní podpis pomocí robota Stäubli RS40b.

SEZNAM POUŽITÉ LITERATURY

- [1] Úředníček, Z.: Robotika, skripta UTB ve Zlíně, Zlín 2012, 978-80-7454-223-7
- [2] Jazar, R. N.: Theory of Applied Robotics: Kinematics, Dynamics, and Control, Springer Science + Business Media, LLC, New York, 2007, ISBN-13:978-0-387-32475-3
- [3] ŠOLC, František. Robotika, modelování a řízení robotů: Robotics, modelling and control of robots : teze přednášky ke jmenování profesorem v oboru "Technická kybernetika". Brno: VUTIUM, 2004. ISBN 80-214-2618-7.
- [4] ZÁDA, Václav. *Robotika: matematické aspekty analýzy a řízení*. Liberec: Technická univerzita v Liberci, 2012. ISBN 978-80-7372-882-3.
- [5] KOLÍBAL, Zdeněk. *Roboty a robotizované výrobní technologie*. Brno: Vysoké učení technické v Brně - nakladatelství VUTIUM, 2016. ISBN 978-80-214-4828-5.
- [6] Havel, Ivan M. Robotika: úvod do teorie kognitivních robotů. 1. vyd. Praha: SNTL, 1980. 279,
- [7] JULINA, Miloslav. *Mechanika: kinematika pro školu a praxi*. Praha: Scientia, 2000. ISBN 80-7183-219-7.
- [8] KARGER, Adolf a Marie KARGEROVÁ. *Základy robotiky a prostorové kinematiky*. Praha: České vysoké učení technické, 2000. ISBN 80-01-02183-1.
- [9] VALENDIN, Miroslav. *Mechanika I*. Vyd. 2. Zlín: Univerzita Tomáše Bati ve Zlíně, Fakulta technologická, 2003. Učební texty vysokých škol. ISBN 80-7318-121-5.
- [10] *Mechanika tuhých těles: Kinematika*. Ústí nad Labem, 2008. Skriptum. Univerzita J. E. Purkyně v Ústí nad Labem, Fakulta výrobních technologií a managementu. Vedoucí práce Prof. Ing. Milan Apetaur, DrCs., Ing. Tomáš Hrala.
- [11] GHOSAL, Ashitava. *Robotics: fundamental concepts and analysis*. New York: Oxford University Press, 2006. ISBN 978-0-19-567391-3.
- [12] SICILIANO, Bruno a Oussama KHATIB. *Springer handbook of robotics*. Berlin: Springer, c2008. ISBN 978-3-540-23957-4.
- [13] CRAIG, John J. *Introduction to robotics: mechanics and control*. 2nd ed. Reading, Mass.: Addison-Wesley, c1989. ISBN 0-201-09528-9.
- [14] JULINA, Miloslav. *Mechanika: kinematika pro školu a praxi*. Praha: Scientia, 2000. ISBN 80-7183-219-7.
- [15] PISKAČ, Luděk. *Průmyslové roboty*. 2. přeprac. vyd. Plzeň: Západočeská univerzita, 2004. ISBN 80-7043-278-0.
- [16] ŠÁMAL, Oldřich. *Technická mechanika: kinematika*. Praha: Informatorium, 2018. ISBN 978-80-7333-134-4.
- [17] APETAUR, Milan a František KLIMENDA. *Mechanika výrobních strojů: manipulátory a průmyslové roboty*. Ústí nad Labem: Univerzita J.E. Purkyně v Ústí nad Labem, Fakulta výrobních technologií a managementu, 2014. ISBN 978-80-7414-782-1.

- [18] Free Downloads. *International federation of robotics* [online]. Frankfurt: IFR International Federation of Robotics, 2020 [cit. 2020-02-28]. Dostupné z: <https://ifr.org/free-downloads/>
- [19] Akcelerátorem robotizace bude umělá inteligence. *Robotic journal*. 2018, 3(3), 2. ISSN 2533-4436.
- [20] Humans Wanted, Robots Need You: Record Number of Employers Globally Plan to Increase or Maintain Headcount as a Result of Automation. *ManpowerGroup* [online]. ManpowerGroup, 2018 [cit. 2020-02-28]. Dostupné z: <https://www.manpowergroup.com/media-center/news-releases/humans+wanted+robots+need+you+record+number+of+employers+globally+plan+to+increase+or+maintain+headcount+as+a+result+of+automation>
- [21] Robotika v chytrém průmyslu. *Robotic journal*. 2019, 4(1), 1. ISSN 2533-4436.
- [22] Roboty ve strojírenských aplikacích. *Robotic journal*. 2019, 4(3), 1. ISSN 2533-4436.
- [23] Trh průmyslových robotů zaznamenává rekord. *Robotic journal*. 2016, 1(1), 2. ISSN 2533-4425.
- [24] I kolegové do sebe občas šťouchnou. *Robotic journal*. 2017, 2(1), 3. ISSN 2533-4425.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

DH Denavit-Hartenberg

SCARA Selective Compliance Assembly Robot Arm[2]

JPG Joint Photographic Group

R Rotační

T Translační

SEZNAM OBRÁZKŮ

Obrázek 1 Rotační a translační kloub[1]	14
Obrázek 2 Obecná kinematická dvojice[1].....	15
Obrázek 3 Robot SCARA a jeho pracovní prostor[13]	16
Obrázek 4 Článek $(i - 1)$, (i) , a $(i + 1)$ spolu se soustavou souřadnic B_i a B_{i-1}	18
Obrázek 5 Rotační a translační pohyb lokální soustavy tuhého tělesa B vzhledem ke globální soustavě souřadnic tuhého tělesa G	21
Obrázek 6 Souřadné soustavy pro robota typu SCARA podle Denavit-Hartenbergovy notace	25
Obrázek 7 Vývojový diagram skriptu pro vykreslení pracovního prostoru	30
Obrázek 8 Vykreslený pohotovostní pracovní prostor pro robota SCARA RS40B pomocí systému Matlab	31
Obrázek 9 Vývojový diagram skriptu pro ověření inverzní kinematické úlohy 1.část	39
Obrázek 10 Vývojový diagram skriptu pro ověření inverzní kinematické úlohy 2.část	40
Obrázek 11 Ověření inverzní kinematické úlohy	40
Obrázek 12 Import dat do programu Matlab	41
Obrázek 13 Import Wizard programu Matlab	41
Obrázek 14 Využití datového kurzoru v programu Matlab	42
Obrázek 15 Simulační model robota RS40b v systému Stäubli Robotic Suite	45
Obrázek 16 Vývojový diagram programu start().....	46
Obrázek 17 Porovnání podpisu z reálného zařízení (vlevo) a simulace (vpravo)	47

SEZNAM TABULEK

Tabulka 1 konstrukční parametry robota Stäubli RS40b.....	24
Tabulka 2 Parametry podle Denavit-Hartenbergovy notace.....	26
Tabulka 3 Proměnné skriptu pro vykreslení pohotovostního pracovního prostoru.....	29
Tabulka 4 Proměnné skriptu BP_overeneInvKin	38

SEZNAM PŘÍLOH

Příloha P I: Skript pro vykreslení pohotovostního pracovního prostoru

Příloha P II: Skript pro ověření inverzní kinematické úlohy

Příloha P III: Skript pro vykreslení černých pixelů obrázku

Příloha P IV: Skript pro překlopení bodů

Příloha P V: Program start()

PŘÍLOHA P I: SKRIPT PRO VYKRESLENÍ POHOTOVOSTNÍHO PRACOVNÍHO PROSTORU

```
%Smazání všech proměnných
clear all
%Smazání příkazového okna
clc

% l1 - Proměnná reprezentující délku "l1"
l1 = 230;
% l2 - Proměnná reprezentující délku "l2"
l2 = 170;
%h - Proměnná reprezentující délku "h"
h = 200;

% Maximální hodnota z intervalu pro kloubovou proměnnou "fi1"
fi1_max = (105*pi)/180;
% Maximální hodnota z intervalu pro kloubovou proměnnou "fi2"
fi2_max = (143*pi)/180;

% index - Slouží k zápisu příslušné proměnné na stravný index.
%          Po ukončení skriptu se hodnota této proměnné rovná počtu bodů,
%          které
%          budou vykresleny.
indexA = 1;
indexB = 1;
% krok_uhlu - Hodnota této proměnné určuje o kolik [rad] se má úhel
%            "fi1" a "fi2" otočit v každém cyklu smyčky for
krok_uhlu = 0.09;
% krok_osaZ - Hodnota této proměnné určuje o kolik [mm] se má proměnná
%            "he"
%            posunout v každém cyklu smyčky for
krok_osaZ = 20;

% Inicializace bodů pro konfiguraci manipulátoru jako loket dolů
B_LD =
zeros(round((h/krok_osaZ)*((fi2_max*2)/krok_uhlu)*((fi1_max*2)/krok_uhlu)
/2),3);
% Inicializace bodů pro konfiguraci manipulátoru jako loket nahoru
B_LN =
zeros(round((h/krok_osaZ)*((fi2_max*2)/krok_uhlu)*((fi1_max*2)/krok_uhlu)
/2),3);

% Smyčka "for fi1=0:krok_uhlu:fi1_int" mění pouze hodnotu úhlu "fi1".
for fi1=-fi1_max:krok_uhlu:fi1_max
    %Smyčka "fi2=0:krok_uhlu:fi2_int" mění pouze hodnotu úhlu "fi2".
    for fi2=-fi2_max:krok_uhlu:fi2_max
        %Smyčka "he=0:krok_osaZ:200" mění pouze hodnotu úhlu "he".
        for he=0:krok_osaZ:h
            % Zde se nachází rovnice vypočítané z přímé kinematické
            %úlohy.
            if fi2 < 0
                B_LD(indexA,1) = - l2*cos(fi1 - fi2) - l1*cos(fi1);
                B_LD(indexA,2) = - l2*sin(fi1 - fi2) - l1*sin(fi1);
                B_LD(indexA,3) = h - he;
                indexA = indexA+1;
            elseif fi2 > 0
                B_LN(indexB,1) = - l2*cos(fi1 - fi2) - l1*cos(fi1);
                B_LN(indexB,2) = - l2*sin(fi1 - fi2) - l1*sin(fi1);
```

```
        B_LN(indexB,3) = h - he;
        indexB = indexB+1;
    end
end
end
end
```

```
figure
hold on
scatter3(B_LD(:,1),B_LD(:,2),B_LD(:,3), 'MarkerEdgeColor', 'k',
'MarkerFaceColor', [0 0 1]);
scatter3(B_LN(:,1),B_LN(:,2),B_LN(:,3), 'MarkerEdgeColor', 'k',
'MarkerFaceColor', [1 0 0]);
xlabel('x[mm]')
ylabel('y[mm]')
zlabel('z[mm]')
```


PŘÍLOHA P II: SKRIPT PRO OVĚŘENÍ INVERZNÍ KINEMATICKÉ ÚLOHY

```
clear all
clc

%PBK - Počet bodů kruhu
PBK = 51;

%rK - Poloměr kruhu
rK = 50;

%PK - Počátek kruhu [x y z]
PK = [-250 0 250];

%l1 - DH parametr
l1 = 230;
%l2 - DH parametr
l2 = 170;
%h - DH parametr
h=365;

%he_max- Maximální vysunutí koncového efektoru
he_max = 200;

%KS - Kloubové souřadnice [Xe,Ye,Ze]
KS = zeros(PBK,3);

%OKS - Ověřené kloubové souřadnice [Xe,Ye,Ze]
OKS = zeros(PBK,3);

%KND - Kloub nahoru (kloub) dolů
KND = zeros(PBK,6);
for a=1:1:PBK
    KND(a,3)=h;
    KND(a,6)=h;
end

%BK2 - Bod kloubu 2, obsahuje souřadnice pro kloub nahoru a kloub dolů [x
y;x y]
BK2 = zeros(2,4);

fi1 = zeros(PBK,2);
fi2 = zeros(PBK,2);
he = zeros(PBK,1);

index=1;

for uhel=0:(2*pi()/ (PBK-1)):(2*pi())
    %Vytvoření bodu kruhu
    KS(index,1) = PK(1) + rK*cos(uhel); %Xe
    KS(index,2) = PK(2) + rK*sin(uhel); %Ye
    KS(index,3) = PK(3) + rK*cos(uhel); %Ze

    %Výpočet kloubových proměnných fí
    fi2(index,1) = 2*atan2(
        sqrt((((l1+l2)^2)-(KS(index,1)^2 +KS(index,2)^2))), ...
```

```

        sqrt(((KS(index,1)^2 + KS(index,2)^2)-((l1-l2)^2)))...
    );
    fi2(index,2) = -fi2(index,1);

    for i=1:1:2
        fil(index,i)= atan2(
            -KS(index,2)*(l1+l2*cos(fi2(index,i)))-KS(index,1)*l2*sin(fi2(index,i)), ...
            -KS(index,1)*(l1 + l2*cos(fi2(index,i))) + KS(index,2)*l2*sin(fi2(index,i))...
        );
    end

    %Výpočet kloubové proměnné he
    he(index) = h-KS(index,3);

    %Výpočet bodů polohy (nahoru/dolů) druhého kloubu
    for b=1:1:2
        KND(index, (1+(b-1)*3))=-l1*cos(fil(index,b));
        KND(index, (2+(b-1)*3))=-l1*sin(fil(index,b));
    end

    %Ověření vypočítaných kloubových proměnných na kinematickém modelu
    OKS(index,1) = -l2*cos(fil(index,1)-
        fi2(index,1))-l1*cos(fil(index,1));
    OKS(index,2) = -l2*sin(fil(index,1)-
        fi2(index,1))-l1*sin(fil(index,1));
    OKS(index,3) = h - he(index);
    % Zde jsou vzorce, pro výpočet s druhým výsledkem úhlu "fi2"
    % OKS(index,1) = -l2*cos( fil(index,2) - fi2(index,2) ) -
    l1*cos(fil(index,2));
    % OKS(index,2) = -l2*sin( fil(index,2) - fi2(index,2) ) -
    l1*sin(fil(index,2));
    % OKS(index,3) = h - he(index);

    index = index + 1;
end

%VelikostBodu
VB = 80;

%Vytvoření okna pro vykreslení výsledku
f1 = figure('Name','Ověření inverzní kinematiky','NumberTitle','off');
hold on

% Index výběru dat z proměnných, pro vykreslení uspořádání v jiné poloze
než počáteční
pol = round((PBK+1)/4);

%Hledání bodu druhého kloubu pro hledaný bod kruhu v místě jiném než je
začátek kruhu
for c=1:1:PBK
    for d=1:1:2
        DU = sqrt((OKS(pol,1) - KND(c, (1+(d-1)*3)))^2 + (OKS(pol,2) -
        KND(c, (2+(d-1)*3)))^2);
        if round(DU) == l2
            BK2(d,1) = KND(c, (1+(d-1)*3));
            BK2(d,2) = KND(c, (2+(d-1)*3));
        end
        DU = sqrt(KND(c, (1+(d-1)*3))^2 + KND(c, (2+(d-1)*3))^2);
        if round(DU) == l1
            BK2(d,3) = KND(c, (1+(d-1)*3));
        end
    end
end

```

```

        BK2(d,4) = KND(c, (2+(d-1)*3));
    end
end
end

% Vykreslení výsledků
for e=1:1:2
plot3([0 BK2(e,3)], [0 BK2(e,4)], [h h], 'k', 'LineWidth',2)
plot3([0 BK2(e,1)], [0 BK2(e,2)], [h h], 'k--', 'LineWidth',2)
plot3([BK2(e,3) OKS(1,1) ], [BK2(e,4) OKS(1,2)] , [h h], 'k',
'LineWidth',2)
plot3([BK2(e,1) OKS(pol,1)], [BK2(e,2) OKS(pol,2)], [h h], 'k--
', 'LineWidth',2)
plot3(KND(:, (1+(e-1)*3)), KND(:, (2+(e-1)*3)), KND(:, (3+(e-
1)*3)), 'Color', [abs(1-e) abs(e-2) 0], 'LineWidth',2)
scatter3(BK2(e,3), BK2(e,4), h, VB, 'k', 'filled')
scatter3(BK2(e,1), BK2(e,2), h, VB, 'k', 'filled')
end

plot3([0,0], [0,0], [0,h], 'k', 'LineWidth',2)
plot3(OKS(:,1), OKS(:,2), OKS(:,3), 'b', 'LineWidth',2)
scatter3(0,0,h,VB, 'k', 'filled')

scatter3(OKS(1,1), OKS(1,2), h, VB, 'k', 'filled')
scatter3(OKS(pol,1), OKS(pol,2), h, VB, 'k', 'filled')
plot3( [OKS(1,1) OKS(1,1) ], [OKS(1,2) OKS(1,2) ], [OKS(1,3)
OKS(1,3)+he_max], 'k' , 'LineWidth',2)
plot3( [OKS(pol,1) OKS(pol,1)], [OKS(pol,2) OKS(pol,2)], [OKS(pol,3)
OKS(pol,3)+he_max], 'k--', 'LineWidth',2)
scatter3(OKS(1,1) , OKS(1,2) , OKS(1,3) , VB, 'k', 'filled')
scatter3(OKS(pol,1), OKS(pol,2), OKS(pol,3), VB, 'k', 'filled')

xlabel('x[mm]', 'FontSize',16)
ylabel('y[mm]', 'FontSize',16)
zlabel('z[mm]', 'FontSize',16)

axis([-350 50 -200 200 0 500])

hold off

clear a b c d e uhel index pol h l1 l2 rK VB he_max DU

```

PŘÍLOHA P III: SKRIPT PRO VYKRESLENÍ ČERNÝCH PIXELŮ OBRÁZKU

```
%Vstupem musí být obrázek s názvem "Objekt" a formátu JPG
clc
% Parametry obrázku
Xpixelu = size(Objekt,2);
Ypixelu = size(Objekt,1);

% Počet odstínů černé zahrnuté do výsledků
k=2;

index=1;
for faze=1:1:2
    for x=1:1:Xpixelu%Procházení řady x pixelů celého obrázku
        for y=1:1:Ypixelu%Procházení řady y pixelů celého obrázku

            if faze==2
                if (x == 2) && (y==2)
                    CerneBody = zeros(index,2);
                    index=1;
                end
            end

            if (Objekt(y,x,1)<k) && (Objekt(y,x,2)<k) &&
(Objekt(y,x,3)<k)%Vyhodnocení černé barvy pixelu
                if faze==2
                    CerneBody(index,1) = x;
                    CerneBody(index,2) = Ypixelu - y;
                    index = index+1;
                else
                    index = index+1;
                end
            end
        end
    end
end
figure('Name','Cerne body obrazku','NumberTitle','off')
scatter(CerneBody(:,1),CerneBody(:,2),20,'b','filled')
```

PŘÍLOHA P IV: SKRIPT PRO PŘEKLOPENÍ BODŮ

```
BodyVstup = zeros(size(cursor_info,2),2);
BodyInv = zeros(size(cursor_info,2),2);

BodyVystup = zeros((2*size(cursor_info,2)-1),2);

for a=1:1:size(cursor_info,2)
    b = size(cursor_info,2)+1-a;

    BodyVstup(b,1) = cursor_info(a).Position(1);
    BodyVstup(b,2) = cursor_info(a).Position(2);

    BodyInv(a,1) = cursor_info(a).Position(1);
    BodyInv(a,2) = cursor_info(a).Position(2);
end

index = 1;

for c=1:1:(2*size(cursor_info,2))

    if c > size(cursor_info,2)+1
        BodyVystup(index,1)=BodyInv(c-5,1);
        BodyVystup(index,2)=BodyInv(c-5,2);
        index = index+1;
    end

    if c < size(cursor_info,2)+1
        BodyVystup(index,1)=BodyVstup(c,1);
        BodyVystup(index,2)=BodyVstup(c,2);
        index = index+1;
    end
end
```

PŘÍLOHA P V: PROGRAM START()

```
begin
  // nastavení měřítka
  l_meritko=2
  // přesun do bodu mimo oblast psaní
  movej(jBezpeci,tPero,mNomSpeed)
  // posun na ose x a y počátku psaní jména
  pPocatek.trsf.x=50
  pPocatek.trsf.y=-50
  // geometrická transformace - otočení osy z směrem dolů

  l_pZdvih=compose(pPocatek,fPapir,{pJmeno[0].trsf.x/l_meritko,pJmeno[0].trsf.y/l_meritko,0,180,0,0})
  // přesun nad první bod jména
  movel(appro(l_pZdvih,{0,0,-50,0,0,0}),tPero,mNomSpeed)
  // počet bodů jména
  l_nLength=319
  for l_nIndex=0 to l_nLength
    // geometrická transformace - otočení osy z směrem dolů

  l_p=compose(pPocatek,fPapir,{pJmeno[l_nIndex].trsf.x/l_meritko,pJmeno[l_nIndex].trsf.y/l_meritko,0,180,0,0})
    // přesun do bodu s nástrojem tPero
    movel(l_p,tPero,mNomSpeed)
  endFor
  // po posledním bodu - zdvihnout v ose z o 50mm
  movel(appro(l_p,{0,0,-50,0,0,0}),tPero,mNomSpeed)
  // posun na ose x a y počátku psaní příjmení
  pPocatek.trsf.x=((200)/l_meritko)+50
  pPocatek.trsf.y=-25
  // geometrická transformace - otočení osy z směrem dolů

  l_pZdvih=compose(pPocatek,fPapir,{pPrijmeni[0].trsf.x/l_meritko,pPrijmeni[0].trsf.y/l_meritko,0,180,0,0})
  // přesun nad první bod příjmení
  movel(appro(l_pZdvih,{0,0,-50,0,0,0}),tPero,mNomSpeed)
  l_nLength=273
  for l_nIndex=0 to l_nLength
    // geometrická transformace - otočení osy z směrem dolů

  l_p=compose(pPocatek,fPapir,{pPrijmeni[l_nIndex].trsf.x/l_meritko,pPrijmeni[l_nIndex].trsf.y/l_meritko,0,180,0,0})
    // přesun do bodu s nástrojem tPero
    movel(l_p,tPero,mNomSpeed)
  endFor
  // po posledním bodu - zdvihnout v ose z o 50mm
  movel(appro(l_p,{0,0,-50,0,0,0}),tPero,mNomSpeed)
  pPocatek.trsf.x=50
  pPocatek.trsf.y=-50

  l_pZdvih=compose(pPocatek,fPapir,{pTecka[0].trsf.x/l_meritko,pTecka[0].trsf.y/l_meritko,0,180,0,0})
  movel(appro(l_pZdvih,{0,0,-50,0,0,0}),tPero,mNomSpeed)
  // movel(appro(l_pZdvih,{0,0,50,0,0,0}),tPero,mNomSpeed)
```

```
for l_nIndexTecka=0 to 10
  // geometrická transformace - otočení osy z směrem dolů

l_pTecka=compose(pPocatek,fPapir,{pTecka[l_nIndexTecka].trsf.x/l_m
eritko,pTecka[l_nIndexTecka].trsf.y/l_meritko,0,180,0,0})
  // přesun do bodu s nástrojem tPero
  movej(l_pTecka,tPero,mNomSpeed)
endFor
move1(appro(l_pTecka,{0,0,-50,0,0,0}),tPero,mNomSpeed)
// návrat do bodu mimo oblast psaní
movej(jBezpeci,tPero,mNomSpeed)
waitEndMove()
end
```