

**ČESKÉ VYSOKÉ
UČENÍ TECHNICKÉ
V PRAZE**

**FAKULTA
STROJNÍ**



BAKALÁŘSKÁ PRÁCE

**MANIPULÁTOR PRO
AUTOMATICKÝ SKLAD
DESKOVÉHO MATERIÁLU**

2022

**JAKUB
ŠVADLENA**

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

**Ústav mechaniky,
biomechaniky a
mechatroniky**



BAKALÁŘSKÁ PRÁCE

**Manipulátor pro automatický
sklad deskového materiálu**

**Board material manipulator for
automatic storage system**

**JAKUB
ŠVADLENA**

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: Švadlena Jméno: Jakub Osobní číslo: 492685
Fakulta/ústav: Fakulta strojní
Zadávací katedra/ústav: Ústav mechaniky, biomechaniky a mechatroniky
Studijní program: Teoretický základ strojního inženýrství
Studijní obor: bez oboru

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Manipulátor pro automatický sklad deskového materiálu

Název bakalářské práce anglicky:

Board material manipulator for automatic storage system

Pokyny pro vypracování:

- 1) Seznamte se s průmyslovými manipulátory v automatizovaném výrobním procesu
- 2) Seznamte se s kartézským průmyslovým manipulátorem
- 3) Sestavte parametrický simulační model kinematiky kartézského manipulátoru pro automatický sklad deskového materiálu
- 4) Navrhněte trajektorový plánovač pohybu skladového manipulátoru

Seznam doporučené literatury:

Tobola P., Automatický sklad deskového materiálu - diplomová práce, ČVUT v Praze - Fakulta strojní, 2021
Berscheid L., Kröger T., Jerk-limited Real-time Trajectory Generation with Arbitrary Target States, Karlsruhe Institut of Technology, Germany

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Jan Pelikán, Ph.D. odbor mechaniky a mechatroniky FS

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

Datum zadání bakalářské práce: 22.04.2022 Termín odevzdání bakalářské práce: 06.06.2022

Platnost zadání bakalářské práce: _____

Ing. Jan Pelikán, Ph.D.
podpis vedoucí(ho) práce

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

doc. Ing. Miroslav Španiel, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

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

_____ Datum převzetí zadání

_____ Podpis studenta

Čestné prohlášení

Prohlašuji, že jsem bakalářskou práci na téma **Manipulátor pro automatický sklad deskového materiálu** zpracoval samostatně, za použití uvedené literatury a za odborného vedení Ing. Jana Pelikána, Ph.D., vedoucího bakalářské práce.

V Praze dne 6. 6. 2022

Jakub Švadlena

Poděkování

Děkuji především vedoucímu mé bakalářské práce panu Ing. Janu Pelikánovi, Ph.D. za skvělý přístup, za vynaložený čas během konzultací a s tím spojené přínosné rady a připomínky, díky kterým mohla být tato práce zdárně dokončena.

Abstrakt

Práce řeší simulační model kartézského manipulátoru pro automatický skladový systém deskového materiálu v nábytkářské výrobě. Jádro simulačního modelu je zpracováno pomocí Simscape Multibody a pohyb manipulátoru je určen programem pro plánování trajektorie. Vyvinutý trajektorový plánovač řeší bezkolizní manipulaci materiálu ve skladovém prostoru a vlastní trajektorie pohybu je vygenerována v souladu S-křivkou rychlostního profilu. Výsledný pohyb směřuje k časově optimálnímu řešení, které po vhodném naladění může eliminovat nežádoucí dynamické chování reálné mechanické struktury manipulátoru. Simulační model a trajektorový plánovač jsou parametrické.

Klíčová slova

Kartézský manipulátor, automatický sklad, automatické plánování trajektorie, S-křivka rychlostního profilu

Abstract

This thesis deals with the simulation model of the cartesian manipulator in an automatic storage system for board material. The core of the simulation model is made in Simscape Multibody software and the movement of the manipulator is determined by the program for path planning. Designed path planner deals with non-collision manipulation of material in the storage system workspace and the trajectory is generated under conditions of S-curve shaped velocity profile. The resulting movement leads to a time-optimal solution that can eliminate unwanted dynamical behavior of the actual mechanical structure of the manipulator after proper tuning. The simulation model and path planner are parametrical.

Keywords

Cartesian manipulator, automatic storage system, automatic path planning, S-curve shaped velocity profile

Obsah

1	Úvod.....	1
2	Robotické systémy.....	2
2.1	Základní struktura robotických systémů.....	2
3	Robotické manipulátory	4
3.1	Struktury robotických manipulátorů	4
3.2	Klasifikace robotických manipulátorů.....	6
3.2.1	Pohybové charakteristiky.....	6
3.2.2	Počet stupňů volnosti.....	6
3.2.3	Druhy pohonů	6
3.2.4	Kinematická struktura.....	7
3.2.5	Tvar pracovního prostoru	8
3.3	Příklady robotických manipulátorů.....	9
3.3.1	Kartézský robotický manipulátor.....	10
3.3.2	Cylindrický robotický manipulátor.....	10
3.3.3	Sférický robotický manipulátor	11
3.3.4	SCARA robotický manipulátor	12
4	Simulační model.....	13
4.1	3D drátové zobrazení.....	13
4.1.1	Manipulátor.....	13
4.1.2	Generátor stohů desek.....	13
4.1.3	Animace pohybu	14
4.2	Simulační model v Simscape Multibody	15
4.2.1	Rám manipulátoru.....	15
4.2.2	Stohy.....	16
4.2.3	Vazby na příčném nosníku	16

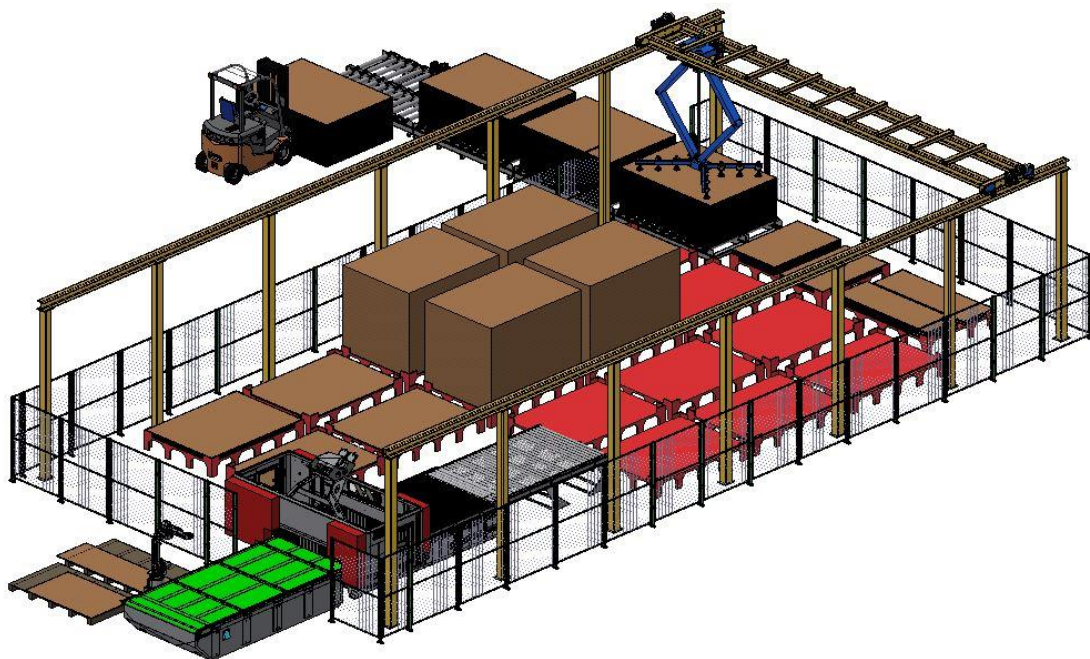
4.2.4	Pohyb v ose z	17
4.2.5	Pohyb dle naplánované trajektorie	17
5	Automatické plánování trajektorie	18
5.1	Vstupní data	18
5.2	Aproximačních polynomy	19
5.3	Tvorba bezkolizních bodů	19
5.4	Úprava bezkolizních bodů	20
5.4.1	Vymazání bezkolizních bodů	20
5.4.2	Úprava výšky bezkolizních bodů	21
5.4.3	Úprava počáteční (koncové) polohy	22
5.5	Vytvoření polynomů	23
5.6	Základní svislý pohyb	24
5.7	Rychlostní profil	25
5.7.1	Ideální S-křivka	25
5.7.1.1	Konvexní část	26
5.7.1.2	Konkávní část	27
5.7.1.3	Maximální hodnota jerk	28
5.8	Aplikace S-křivky	28
5.8.1	Osa x	29
5.8.2	Osa y	30
5.8.3	Osa z	31
5.8.4	Počáteční podmínky jednotlivých úseků	33
5.8.5	Dráha úseku s konstantní rychlostí	33
5.8.6	S-křivka pro svislý pohyb	34
5.9	Nedosažení maximální rychlosti	35
5.9.1	Svislý pohyb	36

5.10	Velikost maximální dosažitelné rychlosti	36
5.11	Volba výsledné dráhy	37
5.11.1	Časově optimální řešení.....	37
5.11.2	Dynamické účinky na soustavu.....	38
5.11.3	Dosažení maximální výšky.....	39
5.11.4	Vyhodnocení	39
5.12	Odeslání trajektorie.....	40
5.13	Další vývoj.....	41
5.13.1	Error dráha nevykreslena	42
5.13.2	Energeticky nejméně náročná cesta.....	43
5.13.3	Maximální absolutní zrychlení.....	43
6	Závěr	45
7	Seznam elektronických příloh	46
8	Bibliografie	47



1 ÚVOD

Práce je součástí projektu, kterým je návrh manipulátoru pro automatický sklad v nábytkářské výrobě. Tento projekt vznikl na základě potřeb průmyslu. Plánované rozměry automatického skladu jsou následující: délka 25 m, šířka 12 m, výška 3 m. Vzhledem k těmto proporcím je nejvhodnějším typem manipulátoru kartézský manipulátor, který bude pro tuto konkrétní aplikaci vybaven nůžkovým mechanismem zajišťujícím pohyb ve svislé ose z. Skladovaným materiálem budou dřevěné desky a jejich úchop bude zajištěn rámem s přísavkami, aby nedošlo k jejich poškození.



Obr. 1.1: Model skladu – zdroj: výzkumná zpráva projektu, 2021



2 ROBOTICKÉ SYSTÉMY

Robotické systémy se již historicky vyskytovaly u umělců, řemeslníků, inženýrů a vědců ve snaze vytvořit stroj, připomínající člověka v jeho jednání a uvažování. Moderní představa robotického systému vznikla při snaze společnosti vytvořit „náhradu“, která zastoupí člověka v činnostech, které jsou fyzicky náročné či dokonce nebezpečné. Slovo „robot“ bylo vytvořeno Karlem Čapkem v roce 1920 v díle Rossumovi univerzální roboti (R.U.R.). Dílo ukazuje morální otázku robotů a použití digitálních programovatelných přístrojů. Toto se stalo opakovaným námětem v románech, divadelních hrách a filmech. [1]

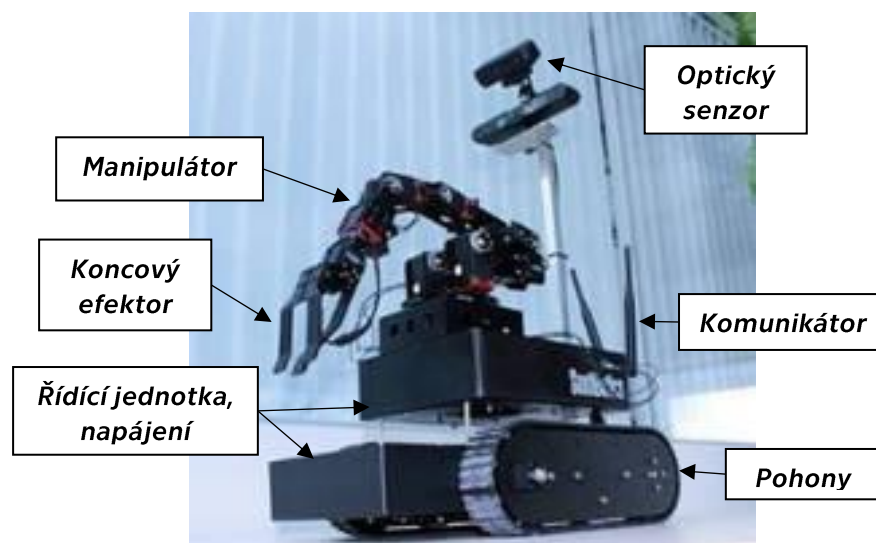
Roboty se rozšířily jako prostředek nahrazující lidskou činnost v nepříznivém pracovním prostředí. První digitálně programovatelný robot byl vytvořen v roce 1954. Tento robot, nazývaný *Unimate*, byl průmyslový manipulátor se sférickým pracovním prostorem a byl používán v továrně pro zvedání a přesun těžkých součástí z výroby. V roce 1960 byl zakoupen společností General Motors a stal se předchůdcem průmyslových robotů, které jsou dnes běžné u moderních výrobních a montážních linek. Při uplatnění robotiky v průmyslu byly hybnou silou zejména požadavky na výkon. Nosnost, opakovatelnost, přesnost a rychlost moderních robotů vysoce převyšuje schopnosti člověka. [1]

2.1 Základní struktura robotických systémů

V průběhu let bylo vytvořeno velké množství druhů robotů, od robotických manipulátorů až po mobilní roboty. Tyto roboty napodobují lidskou populaci a zvířata nebo mají speciální nové struktury pro dosažení požadovaných cílů. I přes tyto odlišnosti je zde však několik základních vlastností, které má mnoho robotů společných. [1]



Obr. 2.1 zobrazuje několik komponent typického robotického systému. Téměř všechny robotické systémy obsahují *pohony*, které slouží jako svařovací soustava systému, díky které má robotický systém možnost konat pohyb. Obvykle jsou poháněny elektřinou, pneumatickým systémem nebo hydraulickým systémem. Od doby, co jsou roboty dálkově ovládané, obsahuje velké množství robotů *komunikátor*. Komunikátor je součástí, která vysílá informaci základně a přijímá instrukce od vzdáleného počítače. Nezbytná vlastnost každého robota je vykazování určitého stupně autonomie nebo inteligence. Důležitou komponentou téměř všech robotických systémů je *řídící jednotka*. Může se skládat ze samotného procesoru nebo ní může být centrální počítač, který vyhodnocuje aktivitu několika procesorů. *Koncový efektor* obsahující *uchopovací zařízení* je také ve výbavě většiny robotů. Aby mohl robot pracovat v prostředí, o kterém nemá dostatek informací (například o předmětech, které jej obklopují), obsahuje sadu *senzorů*, která se skládá z mnoha druhů snímacích zařízení. Každý senzor je obvykle převodník, jehož vstupem jsou fyzikální veličiny a výstupem je elektrický signál. Pohyblivost, snímání a ovládání spotřebovávají energii, a proto musí mít robot vlastní zdroj *napájení*. Nejčastějším typem napájení je bateriový systém, v některých případech, zejména v průmyslu, je robot připojen do elektrické sítě. [1]



Obr. 2.1: Typické komponenty mobilního robotického systému [9]



3 ROBOTICKÉ MANIPULÁTORY

Důležitý typ robotického systému, kterému je věnována hlavní část této bakalářské práce, je *robotický manipulátor*. Robotické systémy tohoto druhu byly první, které dosáhly širokého spektra použití v průmyslu. Jak již bylo zmíněno v předchozí kapitole, robotické manipulátory se staly standardní položkou moderních výrobních linek. Mohou vykonávat velké množství úkolů, mezi které patří svařování, lakování, „pick and place“ operace, vrátání, stříhání a zvedání břemen. Robotické manipulátory jsou jedny z nejjednodušších příkladů reálného robotického systému. Jejich studie pomáhá klasifikovat základní principy a problémy při studii komplexnějších systémů. Ačkoliv například autonomní vozidlo nemusí připomínat robota ve výrobní lince, základní forma matematického problému, který musí být vyřešen pro řízení těchto dvou typů systému, může být podobná. Toto platí i pro modelování a řízení autonomního pozemního vozidla a autonomního letounu. Základní metody použitelné na jeden systém, mohou být občas odrazovým můstkem pro vývoj modelování a řízení dalšího systému. Dokonce je časté, že podsystém autonomního robotického systému může být modelovaný nebo řízený za použití technik vyvíjených pro robotický manipulátor. Takovým příkladem je ruka nebo noha humanoidního robota. [1] [2] [3]

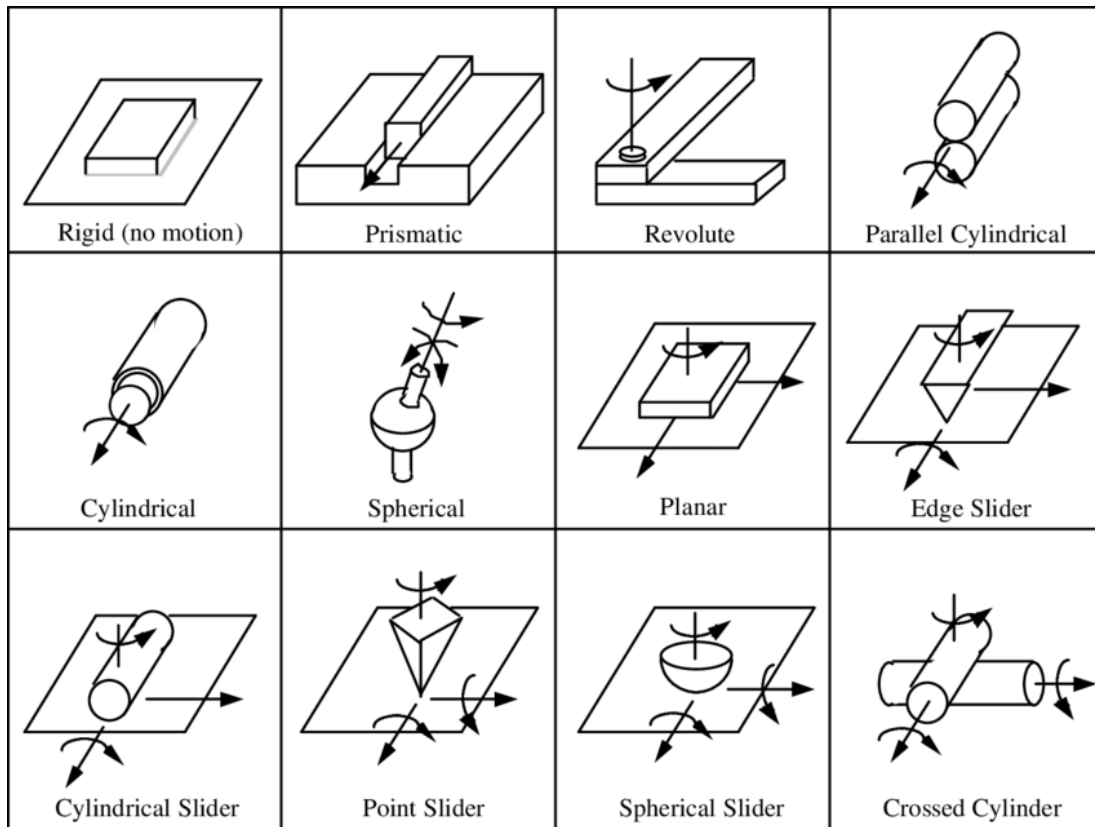
3.1 Struktury robotických manipulátorů

Mnoho robotů se skládá z jednotlivých těles, která jsou propojena vazbami. Jednotlivá tělesa jsou často považována za tuhá tělesa, nicméně pro vysoké rychlosti a velká zatížení mechanismů by měly být uvažovány jejich elastické deformace. Vazby, které spojují jednotlivá tělesa robota mohou být poměrně složité a může se u nich projevovat vůle nebo tření. Ideální vazba je spojení mezi tuhými tělesy robotického systému, které umožňuje pouze předurčené relativní pohyby, jako je posuv nebo rotace. Matematicky ideální vazba vytváří kinematické omezení pohybu mezi tuhými tělesy,



keré je dáno geometrií vazby. Běžné typy ideálních vazeb zahrnují *posuvnou, rotační, obecnou, sférickou nebo šroubovou vazbu*. Obr. 3.1 ukazuje základní typy ideálních vazeb a jejich stupně volnosti. [1] [2] [3]

Dva nezákladnější typy jsou posuvná vazba a rotační vazba. Téměř každý robotický manipulátor obsahuje právě tyto dva typy vazeb. Mnoho dalších typů ideálních vazeb může být pomocí těchto dvou následně vytvořeno. Například Kardanův kloub je složen ze dvou rotačních vazeb se vzájemně kolmými osami rotace. Posuvná vazba umožňuje pouze relativní posuv mezi dvěma tělesy po dané ose posuvu, zatímco rotační vazba umožňuje pouze relativní rotační pohyb kolem osy rotace. [1] [2] [3]



Obr. 3.1: Souhrn ideálních kinematických vazeb [10]

Nezávislé proměnné, které jsou použity k popisu pohybu robota nebo relativních pohybů jeho částí jsou nejčastěji nazývané *stupně volnosti* (anglicky *Degree of Freedom – DOF*). Počet stupňů volnosti ideální vazby je počet nezávislých proměnných nutných k popisu relativního pohybu, který daná vazba umožňuje. [1]



3.2 Klasifikace robotických manipulátorů

Nyní jsme si zavedli základní definice těles, vazeb a stupňů volnosti pro typický robotický manipulátor. Dále budeme pokračovat rozdělením robotických manipulátorů podle konkrétních charakteristik.

3.2.1 Pohybové charakteristiky

Jeden z nejběžnějších způsobů rozlišení mezi různými strukturami robotů jsou pohybové charakteristiky. *Rovinný manipulátor* je takový manipulátor, ve kterém konají všechna tělesa rovinné pohyby. U *prostorového manipulátoru* koná alespoň jedno těleso obecný prostorový pohyb, má tedy 6 stupňů volnosti. V některých případech je manipulátor konstruován tak, že je možný pouze určitý druh pohybu. *Sférický manipulátor* je navržen tak, že pohybující se části konají sférické pohyby kolem společného stacionárního bodu. *Cylindrický manipulátor* je konstruován tak, že koncový efektor koná pohyb kopírující povrch válce. [1] [2] [3]

3.2.2 Počet stupňů volnosti

Další způsob rozdělení robotů je na základě počtu a typů stupňů volnosti. *Univerzální robot* má 3 stupně volnosti, pokud se jedná o rovinného robota, případně 6 stupňů volnosti, pokud se jedná o prostorového robota. Robot je *redundantní*, pokud má více než 3 (resp. 6) stupňů volnosti. Redundantní robot může být použit pro pohyb kolem překážek a provádění operací v těsných prostorech. [1] [2] [3]

3.2.3 Druhy pohonů

Roboty jsou také často rozdělovány na základě typu pohonu. *Elektrický robot* je poháněn stejnosměrným servomotorem nebo krokovým motorem. Tento typ robota má výhodu čistoty a relativně jednoduchého řízení. *Hydraulický robot* je preferován pro aplikace, kde je vyžadována vysoká nosnost. Nevýhodou je nutnost údržby pro zamezení úniků a kontroly tlaku



hydraulické kapaliny. Pro vysoké rychlosti je pak preferován *pneumatický robot*. Z důvodu stlačitelnosti vzduchu může u těchto robotů být obtížné jejich řízení. [1] [2]

Robotický manipulátor s přímými pohony je takový, ve kterém je každá vazba řízena motorem přímo, tedy bez jakýchkoliv mechanismů pro přenos krouticího momentu. Tyto pohony bývají objemné a těžké, avšak eliminují vůli nebo flexibilitu pohonných mechanismů, které mohou ztížit řízení celého manipulátoru. U *konvenčního manipulátoru* generuje pohon krouticí moment, který je následně zvětšen přes převodový mechanismus. Tento typ umožňuje použití menších pohonů, ale díky převodovému mechanismu se zde vyskytují vůle. [1] [2]

3.2.4 Kinematická struktura

Kinematická struktura je další způsob, podle kterého mohou být rozděleny různé typy robotů. Obecně se obory robotiky zabývají problematikou přímé kinematiky, inverzní kinematiky nebo řízení syntézy mechanismů. Obory dynamiky více těles se zabývají studii numerických metod pro aproximaci řešení přímé dynamiky. V oblasti dynamiky více těles má struktura systému dramatický vliv na složitost simulace nebo odvození strategie řízení pro daný systém. [1] [2]

Robotický systém má topologii *kinematického řetězce*, pokud existuje jediná cesta, která spojuje první článek s posledním. Takový robot je v literatuře často označován také jako *sériový manipulátor* nebo *manipulátor s otevřenou smyčkou*. Ruka nebo noha humanoidního robota je ideální příklad kinematického řetězce. [1] [2]

Robotický systém má *stromovou topologii*, pokud je složen ze sestavy kinematických řetězců a netvoří žádnou uzavřenou smyčku. Celotělový humanoidní robot nebo vesmírná stanice na orbitu jsou dva známé příklady tohoto typu systému. [1] [2]



Posledním typem je robotický systém nazývaný se *systém uzavřené smyčky*, u něhož je kdykoliv možné zkonstruovat plynulou dráhu, která začíná v jednom tělese, pokračuje přes několik dalších těles a končí opět v počátečním tělese. Systém více těles autonomního pozemního vozidla je příkladem systému, který má topologii uzavřené smyčky. Dva robotické manipulátory, které spolupracují na společném úkolu zvedání břemen o vysokých hmotnostech, také tvoří systém s uzavřenou smyčkou. Stewartova platforma zobrazená na obr. 3.2 je běžná robotická platforma, která má konektivitu uzavřené smyčky. [1] [2]

Systémy robotických manipulátorů s uzavřenou smyčkou jsou často označovány jako *paralelní manipulátory*.

Samozřejmě jsou některé systémy tvořeny z podsystémů, které jsou tvořeny řetězci otevřené i uzavřené smyčky. Takové systémy jsou známé jako *hybridní manipulátory*. [1] [2]

V souhrnu jsou robotické systémy, které jsou tvořeny kinematickým řetězcem těmi nejzákladnějšími typy robotů, z nichž mohou být dále složeny mnohem komplikovanější systémy. Metody pro analýzu, simulaci nebo syntézu řízení pro kinematické řetězce mohou být použity na podsystémy s komplexnější strukturou. [1] [2]



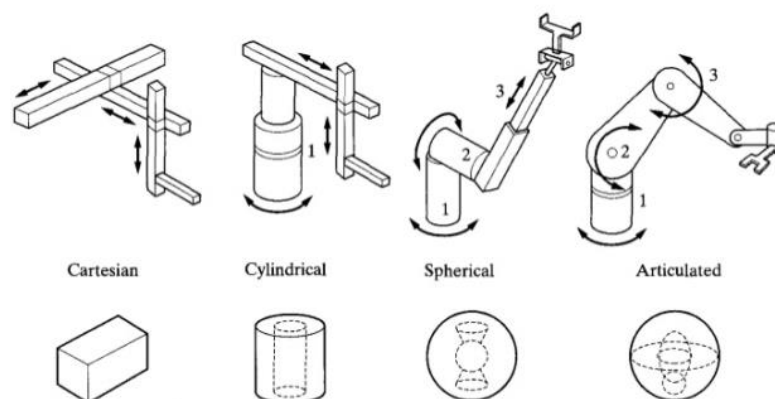
Obr. 3.2: Stewartova platforma [8]

3.2.5 Tvar pracovního prostoru

Další metoda pro klasifikaci robotů se zaměřuje na tvar jejich *pracovního prostoru*. Pracovní prostor je takový prostor, do kterého dosáhne koncový efektor manipulátoru. Množství bodů, které mohou být dosaženy koncovým efektořem alespoň v jedné orientaci nebo poloze se nazývá



dosažitelný pracovní prostor. Množství bodů, které mohou být dosaženy ve všech možných orientacích a polohách se nazývá *obratný pracovní prostor*. Z definice vyplývá, že obratný pracovní prostor je podmnožina dosažitelného pracovního prostoru. Je vhodné zmínit, že nejvíce průmyslových sériových manipulátorů je konstruováno tak, že první tři pohybuující se tělesa manipulátoru jsou delší než ta zbývající. Tato vnitřní tělesa jsou primárně určena pro řízení pozice koncového efektoru. Zbývající přívěsná tělesa jsou typicky použita pro řízení orientace koncového efektoru. Často se podstava spojená s prvními třemi tělesy nazývá rameno a zbývající tělesa tvoří zápěstí. Na obr. 3.3 jsou zobrazeny základní typy pracovních prostorů. [1] [2]



Obr. 3.3: Pracovní prostory robotických manipulátorů [11]

3.3 Příklady robotických manipulátorů

V následujících podkapitolách je popsáno několik běžných typů robotických manipulátorů. Všechny tyto manipulátory jsou složeny z několika těles spojených posuvnými nebo rotačními vazbami. Tyto vazby mohou být buď *poháněné* nebo *pasivní*. [1] [2]

Poháněná vazba je taková, u které pohon vytváří pohyb přímo, a to buď posuvný nebo rotační. Pokud k vazbě není přímo připojený pohon, jedná se o vazbu pasivní. [1] [2]

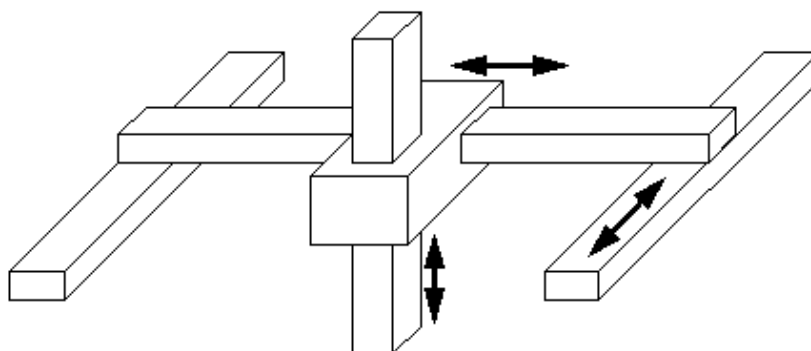
Robotické manipulátory jsou obvykle konstruovány v sekvencích PPP nebo RPP. Toto označení indikuje typ a uspořádání posuvných (P) a rotačních



(R) vazeb, ze kterých je manipulátor složen. Například PPP manipulátor je složen ze tří posuvných vazeb, zatímco RPP manipulátor je složen z rotační vazby, po které následují dvě vazby posuvné. Tato označení jsou často dobrým prvotním indikátorem základní geometrie pracovního prostoru a funkce robotického manipulátoru. [1] [2]

3.3.1 Kartézský robotický manipulátor

Kartézský robotický manipulátor je manipulátor typu PPP, který je definován třemi vzájemně kolnými posuvnými vazbami, jejichž uspořádání je zobrazeno na obr. 3.4. Jedná se o nejjednodušší typ robotického manipulátoru a je občas také nazýván „portálový robotický manipulátor“. PPP manipulátor má díky své jednoduché geometrii několik výhod, jako je například jednoduché řízení a díky absenci rotační vazby vysoká tuhost. Manipulátory tohoto typu tedy mohou být použity pro vysoká zatížení, a přesto dosahovat vysokých přesností při polohování. Nevýhodou těchto manipulátorů je velká zástavbová plocha a jejich pracovní prostor je menší než samotný manipulátor. Další nevýhodou je nutnost utěsnění posuvných vazeb, čímž se může zkomplikovat údržba. [1] [2] [4]



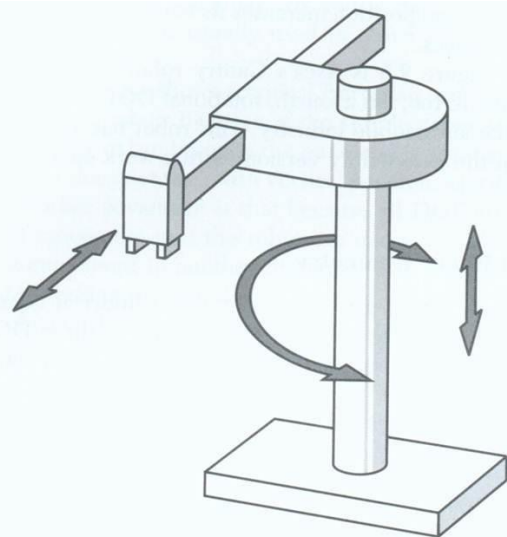
Obr. 3.4: Kartézský manipulátor [4]

3.3.2 Cylindrický robotický manipulátor

Předpokládejme záměnu první posuvné vazby u kartézského robotického manipulátoru za vazbu rotační. Po rozvážné volbě směru osy rotace získáme RPP robota, který je příkladem právě cylindrického manipulátoru,



zobrazeném na obr. 3.5. Je zřejmé, že pozici konce vodorovného ramene lze vyjádřit cylindrickými souřadnicemi, po nichž manipulátor přejímá svůj název. Jeho pracovní prostor se vyznačuje tvarem dutého válce. I přesto, že kinematické i dynamické modely cylindrických manipulátorů jsou komplexnější než u kartézských manipulátorů, lze je stále snadno odvodit. Související principy řízení lze také určit poměrně snadno. Díky topologii manipulátoru RPP se dobře hodí pro sahání do obrobků, které mají dutiny nebo jiné podobné komplexní geometrie. Jeho výhodou je dosažení vysoké přesnosti v „pick and place“ operacích na výrobní lince. Nevýhoda tohoto typu manipulátoru je, že jeho zadní část může v některých konfiguracích vyčnívat do pracovního prostoru a může způsobit střet s objekty v pracovním prostoru a komplikovat tak plánování a kontrolu trasy. Stejně jako u každého robota, který používá posuvné vazby s externím vedením, musí být vodící plochy bez nečistot. [1] [2] [5]



Obr. 3.5: Cylindrický manipulátor [10]

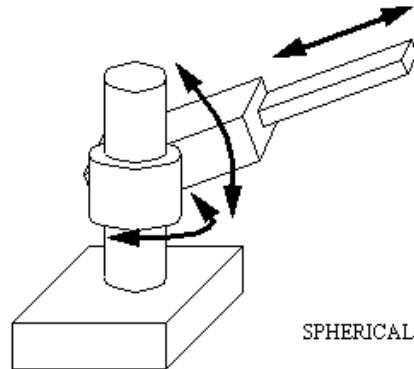
3.3.3 Sférický robotický manipulátor

Sférický robotický manipulátor RRP je sestaven ze dvou vzájemně kolmých rotačních vazeb a posuvné vazby. Pohyb koncového efektoru manipulátoru je vyjádřen pomocí sférických souřadnic. [1] [5]

Hlavní výhodou konstrukce tohoto typu robota je jeho vhodnost pro široký rozsah operací, které musí být prováděny na složitých geometriích. Dosažitelný sférický pracovní prostor je mnohem větší než robot samotný. Bohužel je tato výhoda spojena s mnohem komplikovanějším kinematickým i dynamickým modelem a se složitějšími principy řízení. Dvě vzájemně kolmé rotační vazby snižují tuhost a zvětšují vůle a výsledkem toho je pak



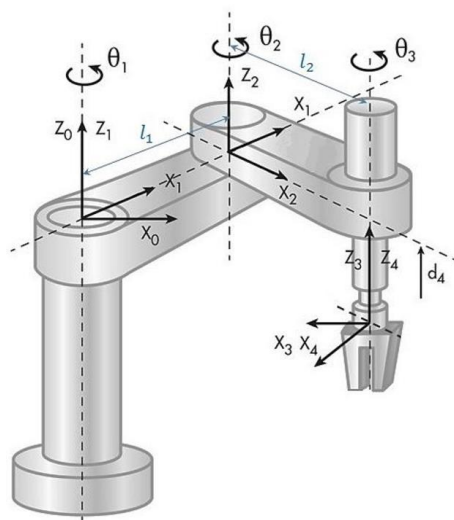
menší přesnost polohování. Sférický manipulátor je tedy vhodný pro operace, jako je svařování nebo lakování, které vyžadují menší přesnost než „pick and place“ operace, ale zároveň požadují dosah ve velkém a komplexním pracovním prostoru. [1] [5]



Obr. 3.6: Sférický manipulátor [5]

3.3.4 SCARA robotický manipulátor

SCARA robot byl představen jako kompromis mezi vysoce tuhými roboty (kartézské manipulátory) a roboty, které mají geometricky komplexní pracovní prostory (sférické manipulátory). Jeden z příkladů tohoto typu robota je na obr. 3.7. Díky rovnoběžnosti dvou os rotace rotačních vazeb je tento robot relativně poddajný při pohybu v horizontální rovině a relativně tuhý při pohybu kolmo k této rovině. Manipulátor SCARA je například vhodný pro „pick and place“ operace. [1] [5]



Obr. 3.7: Manipulátor SCARA [10]



4 SIMULAČNÍ MODEL

Simulační modely jsou vhodné pro digitalizaci manipulátoru a jeho pracovního prostoru. Dále slouží jako prostředek pro ověření bezpečnosti vytvořené trajektorie a získání představy o pohybu manipulátoru.

4.1 3D drátové zobrazení

Nejjednodušším zobrazením je 3D drátový model. Ten je vytvořen pomocí 3D grafu v prostředí Matlab. Výhodou tohoto modelu jsou menší nároky na hardware při jeho vykreslování a jeho jednoduchá editace. Udává nám však pouze zjednodušené zobrazení celého skladu s manipulátorem. Velmi praktické je ale vykreslení naplánované trajektorie, které tento simulační model umožňuje.

4.1.1 Manipulátor

Pro zobrazení manipulátoru je vytvořena funkce se vstupními veličinami, kterými jsou šířka, délka a výška celého manipulátoru. Dále poloha a natočení (v rovině xy) koncového efektoru manipulátoru, tedy středová poloha koncového efektoru s přísavkami pro uchopení dřevěných desek. Je zde také možnost upravit délku nůžek u nůžkového mechanismu.

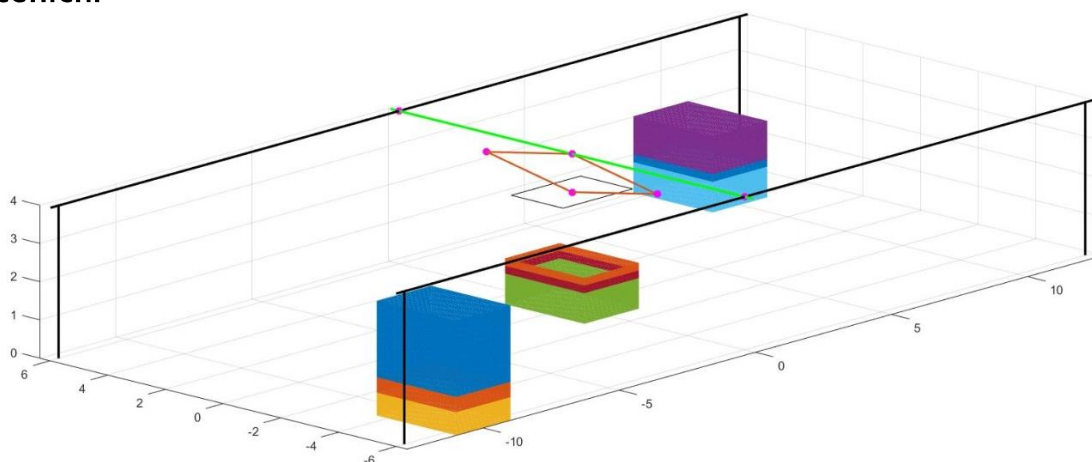
4.1.2 Generátor stohů desek

Po vytvoření vizualizace manipulátoru je vhodné zobrazit také stohy s deskami, které se ve skladu nacházejí. Vstupními daty pro generátor desek jsou matice stohů, obsahující informaci o poloze jednotlivých stohů, počtu desek a jejich typů v každém stohu a také natočení každého stohu. Výstupem této funkce jsou kromě zobrazení stohů ve 3D grafu také vypočtené hodnoty důležité pro další plánování, jako jsou například výšky, šířky a délky jednotlivých stohů desek odvíjejících se od jejich typu a počtu v každém



stohu. Každý typ desky je ve vizualizaci označen odlišnou barvou a rozměry všech desek jsou parametrické.

V této práci se omezíme na stohování desek pouze v jednom smyslu orientace a budeme předpokládat, že rozměry desek jsou stejné. Základní kód je však předpřipraven na rozšíření o možnost stohování v různých natočeních.



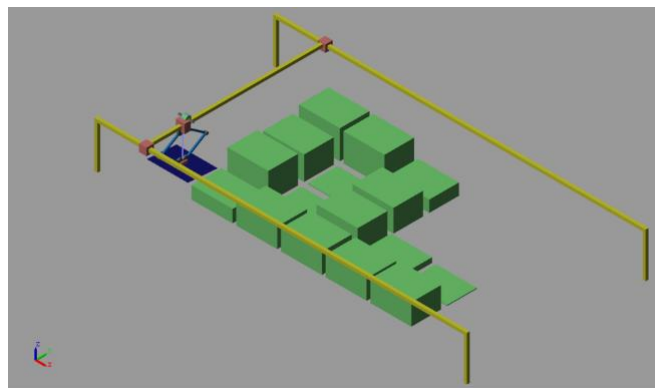
Obr. 4.1: Simulační model – drátové zobrazení

4.1.3 Animace pohybu

Pomocí tohoto modelu lze také animovat pohyb manipulátoru s dřevěnou deskou. To je vhodné využít zejména při prezentaci projektu zákazníkovi. Tato animace obsahuje výsledný pohyb manipulátoru, jehož koncový efektor s dřevěnou deskou za sebou zanechává zobrazenou výslednou trajektorii.

4.2 Simulační model v Simscape Multibody

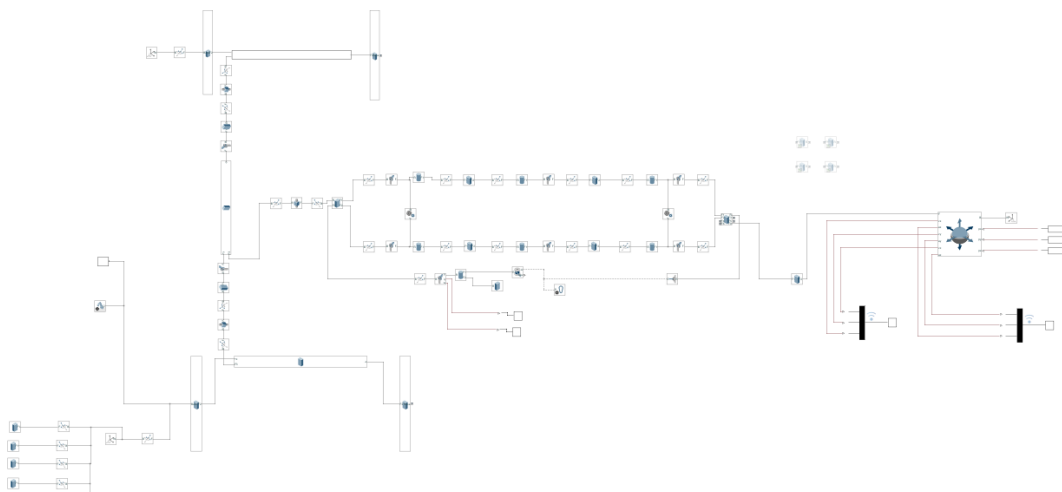
Další simulační model je vytvořený v prostředí Simscape Multibody. Má vyšší nároky na hardware, avšak získaná simulace je mnohem názornější. Je zde možné nakonfigurováním hmotových veličin provádět dynamické analýzy a dále celý manipulátor v tomto prostředí řídit. Tento simulační model bude sloužit jako digitální dvojče celého manipulátoru.



Obr. 4.2: Simulační model – Simscape Multibody

4.2.1 Rám manipulátoru

Veškeré součásti manipulátoru jsou vytvořeny pomocí tuhých těles, jejichž rozměry jsou přebrány z již zadaných parametrů skladu. Tato tělesa jsou spojena ideálními vazbami, jejichž kombinací můžeme docílit chování manipulátoru podobající se spojení těles vazbami reálnými.



Obr. 4.3: Simulační model v prostředí Simscape Multibody – blokové schéma



4.2.2 Stohy

Pro zřetelnou vizualizaci jsou zde vytvořené i jednotlivé stohy desek. Ty jsou znázorněny jednotnou barvou, a protože zde není možné vytvořit cyklus *if*, kterým by se rozhodovalo, zda nějaký stoh bude či nebude zobrazen, jsou zde všechny pozice stohů obsazené alespoň deskou o zanedbatelné tloušťce (0,0001 mm), resp. zobrazené stohy jsou o 0,0001 mm vyšší než ve skutečnosti. V praxi však budou odchylky výšek stohů od ideálních rozměrů větší, protože lze předpokládat, že desky budou vyráběné s horší přesností než 0,0001 mm.



Obr. 4.4: Blokové schéma – stohy s deskami

4.2.3 Vazby na příčném nosníku

Příčný nosník s jeřábovou kočkou bude poháněn dvěma elektromotory (každý na jednom konci nosníku). Tento nosník bude 12 m dlouhý a je tedy pravděpodobné, že i přes velice dobrou regulaci při řízení elektromotorů dojde ke vzpříčení. Reálný manipulátor bude díky vůlím v jednotlivých vazbách, poddajnosti těles a velkém krouticím momentu elektromotorů schopný dále konat pohyb, ale v simulačním modelu by díky nekonečné tuhosti a ideálním vazbám mohlo dojít k singulární poloze a simulace selže.

Pro zamezení těchto účinků se příčný nosník v modelu připojuje k hlavním nosníkům kromě posuvných vazeb také vazbami rotačními, které do modelu promítnou vůle v reálných vazbách.

Další modifikací by bylo možné simulovat poddajnost příčného nosníku, a to připojením další posuvné vazby (kolmo k hlavní posuvné vazbě). Tuto modifikaci však v kinematickém modelu pro účely plánování trajektorie zatím nebudeme uvažovat. Je to ovšem možné rozšíření.

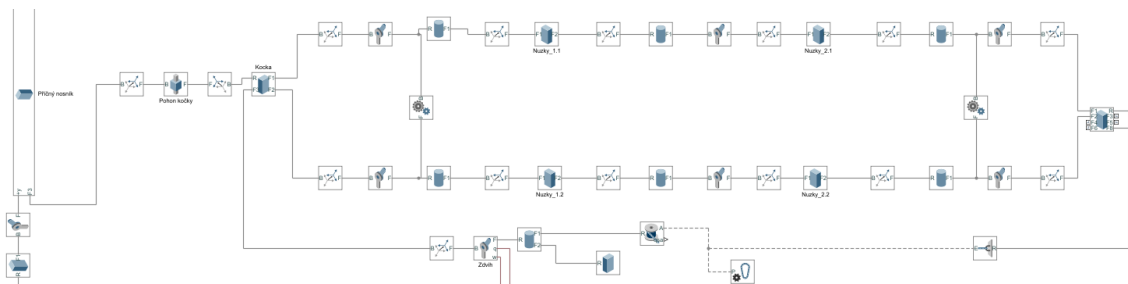


Obr. 4.5: Blokové schéma – vazby na příčném nosníku



4.2.4 Pohyb v ose z

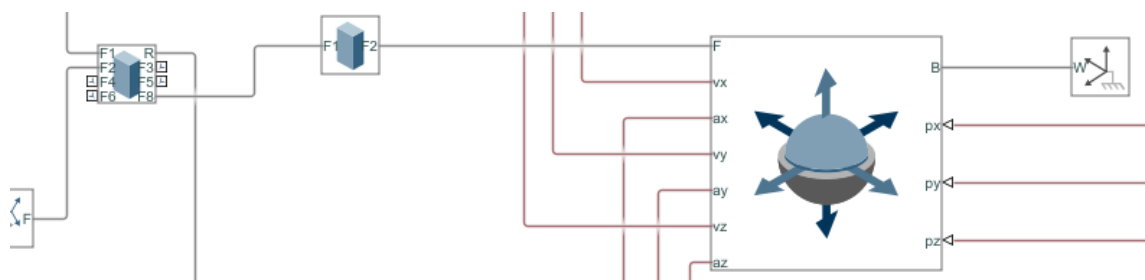
Pohybové ústrojí ve směru osy z je nejsložitější část konstrukce celého manipulátoru. Manipulátor je vybaven nůžkovým mechanismem, jehož pohyb je zajištěn navíjením (resp. odvíjením) ocelového lana na buben (resp. z bubnu). Nůžkový mechanismus je k jeřábové kočce připojen pomocí dvou rotačních vazeb (pro každé rameno jedna vazba). Pro zajištění jednoznačné polohy a odebrání nadbytečných stupňů volnosti je pohyb jednotlivých ramen nůžek synchronizován ozubenými koly. Stejné připojení je i na spodní straně nůžkového mechanismu, kde se k němu připojuje rám s přísavkami pro uchopení dřevěných desek. Bez této synchronizace ozubenými koly by se z nůžkového mechanismu stal čtyřkloubový mechanismus.



Obr. 4.6: Blokové schéma – nůžkový mechanismus

4.2.5 Pohyb dle naplánované trajektorie

Předpokládejme, že máme již naplánovanou trajektorii a chceme ji implementovat do tohoto simulačního modelu. To zajistíme pomocí vazby 6-DOF Joint, která nám umožňuje (jak již z názvu vyplývá) pohyb ve všech třech osách souřadného systému a také rotaci kolem nich (má 6 stupňů volnosti). Vazba je umístěna ve středovém bodu koncového efektoru (rám s přísavkami) a její vstupní data jsou polohy v jednotlivých osách x, y a z závislé na čase.



Obr. 4.7: Blokové schéma – vazba se 6 stupni volnosti



5 AUTOMATICKÉ PLÁNOVÁNÍ TRAJEKTORIE

Plánování trajektorie je možné provádět buď pomocí interpolace bezkolizních bodů (s určitou přesností) nebo jejich aproximace polynomy. Tato práce se zabývá plánováním trajektorie pomocí aproximačních polynomů vytvořených metodou nejmenších čtverců.

5.1 Vstupní data

Předpokládejme, že již máme vytvořený simulační model v drátovém zobrazení a jsou již zadána vstupní data:

- rozměry automatického skladu,
- poloha a natočení manipulátoru,
- délka ramen nůžkového mechanismu,
- poloha jednotlivých stohů a počet desek v každém z nich,
- rozměry dřevěných desek.

Dále je potřeba určit koncovou polohu (v budoucím systému bude generována automaticky na základě vlastního algoritmu stohování desek) a konkretizovat pohybové parametry, jako je maximální zrychlení a maximální rychlost v jednotlivých směrech.

Na obr. 5.1 je část programu pro plánování trajektorie, kde jsou zadávány vstupní hodnoty. Řádky zvýrazněné zelenou barvou jsou buď vysvětlivky k daným příkazům nebo příkazy, díky kterým bylo plánování trajektorie testováno pro specifické situace, zejména ty, při kterých na počátku vývoje plánovač selhával.



```
% Parametry stohu
stoh.poloha = [-6 -3.6; -6 -1.2; -6 1.2; -6 3.6;
              -3 -3.6; -3 -1.2; -3 1.2; -3 3.6;
              0 -3.6; 0 -1.2; 0 1.2; 0 3.6;
              3 -3.6; 3 -1.2;
              6 -3.6; 6 -1.2]; % polohy [x y]

stoh.pocet = numel(stoh.poloha(:,1)); % pocet stohu...PRO RANDOM POČTY

% stoh.typ = [1; 2; 3; 4; 5; 6; 7; 2; 5; 6; 4; 1; 7; 5; 3; 2]; % typy c
stoh.typ = randi([1 7],stoh.pocet, 1);

% stoh.pocetdesek = [10;10;10;10;10;10;10;10;10;100;10;10;10;10;10];
stoh.pocetdesek = randi([0 100],stoh.pocet,1); %random pocet desek
% stoh.pocetdesek = zeros(stoh.pocet);

% Bod A a B
poc = [-9; -3.6; 0.1];
set = [9; 1.2; 0.02];

% Bod A a B
poc = [-9; -3.6; 0.1];
set = [9; 1.2; 0.02];

% Maximální rychlosti a zrychlen:
vmax = 3; % [m/s] pro x, zbytek :
amax = 1; % [m/s^2] pro x, zbytek

% Počáteční podmínky
s0 = [poc(1); poc(2); poc(3)];
v0 = [0; 0; 0];
a0 = [0; 0; 0];

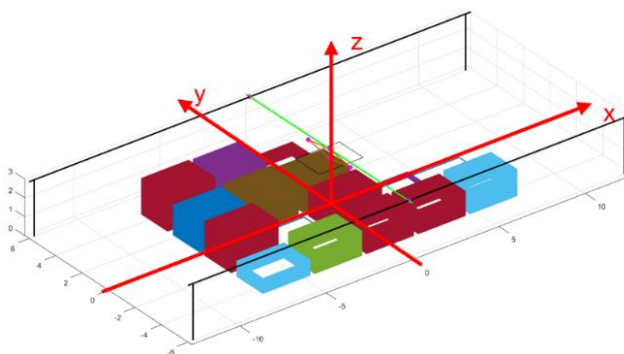
% Diskretizační krok
h = 0.01; % [s]...krok

maxvyska = robot.vyska_z - 0.2;
```

Obr. 5.1: Matlab – vstupní hodnoty

5.2 Aproximačních polynomy

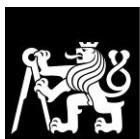
V rovině xy , neboli v rovině rovnoběžnou se základnou, budeme uvažovat pohyb po dráze polynomu 1. stupně, tedy po přímce. Pohyb v ose z , respektive v rovině xz , bude realizován polynomy 2. až 5. stupně, protože pohybem v této ose je řízeno manévrování mezi překážkami uvnitř skladu. Mohou však nastat výjimečné stavy, například prázdný sklad, kdy nám v rovině xz nic nebrání v použití polynomu 1. stupně.



Obr. 5.2: Souřadný systém manipulátoru

5.3 Tvorba bezkolizních bodů

Předpokládejme, že máme zadané vstupní veličiny dle kapitoly 5.1 a vytvořený polynom prvního stupně v rovině xy . Pak každému stohu ve skladu přiřadíme dva body, které se nacházejí na jejich okrajích a dva body, které se nacházejí již mimo tento stoh tak, aby vzdálenost mezi těmito body byla délka dřevěné desky (polovina délky desky ve stohu a polovina desky uchopené koncovým efektořem). Pokud je vytvořený polynom některému z těchto bodů blíže (v rovině xy), než je šířka dřevěné desky (předpokládáme stohování desek pouze v jednom smyslu orientace) nebo jej



dokonce protíná, označíme jej za bod kolize. Pak vezmeme bod nacházející se na přímce mezi počáteční a koncovou polohou se souřadnicí x stejnou, jako má bod kolize a přeneseme jej do matice bezkolizních bodů. Výšku bezkolizních bodů, neboli jejich souřadnici z , získáme jako výstupní veličinu při použití generátoru stohu desek odvíjející se od výšky stohů. Pro zaručení bezkolizní polohy daného bodu přičteme k jeho souřadnici z hodnotu bezpečnosti, kterou si také určíme jako vstupní veličinu.

Vytvořené body jsou však zatím pouhý základní náhled možných bezkolizních bodů, a ne všechny jsou v dalším plánování použitelné.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%% BEZKOLIZNÍ BODY %%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
bezpecnost = 0.3; % [m]  
for k = 1:stoh.pocet  
    if abs(stoh.poloha(k,2) - fyt(stoh.kraj_x(k,1))) < stoh.sirka(k)*1.3 %  
        bezkoliznibody(4*k-3,1) = stoh.kraj_x(k,1); % násobeno celou deskou  
        bezkoliznibody(4*k-3,2) = fyt(bezkoliznibody(4*k-3,1));  
        bezkoliznibody(4*k-3,3) = stoh.vyska(k) + bezpecnost; % bezpečnost  
    end  
  
    if abs(stoh.poloha(k,2) - fyt(stoh.kraj_x(k,2))) < stoh.sirka(k)*1.3  
        bezkoliznibody(4*k-2,1) = stoh.kraj_x(k,2); % násobeno celou deskou  
        bezkoliznibody(4*k-2,2) = fyt(bezkoliznibody(4*k-2,1));  
        bezkoliznibody(4*k-2,3) = stoh.vyska(k) + bezpecnost; % bezpečnost  
    end
```

Obr. 5.3: Matlab – tvorba bezkolizních bodů

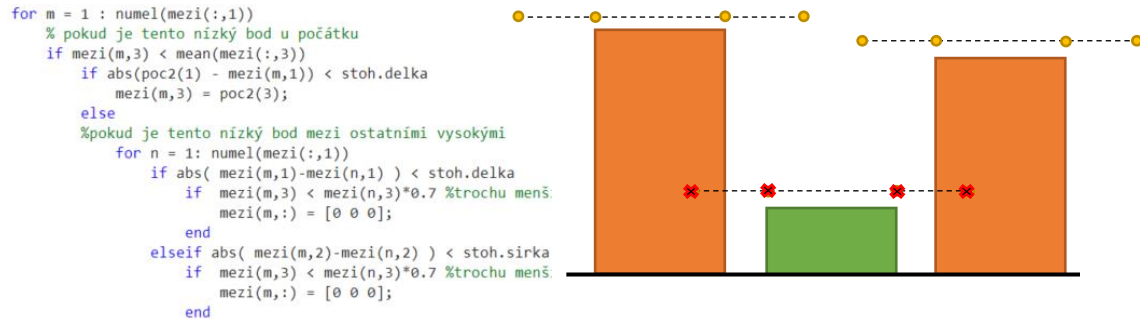
5.4 Úprava bezkolizních bodů

V této podkapitole se budeme na základě možných situací při stohování věnovat úpravě bezkolizních bodů, aby plánování trajektorie nesešlo.

5.4.1 Vymazání bezkolizních bodů

Použijme příklad, kdy jsou tři stohy vedle sebe, krajní stohy jsou vysoké a prostřední stoh je nízký (Obr. 5.4). Pro tvorbu trajektorie jsou bezkolizní body získané od prostředního stohu neúčinné, naopak by mohly zapříčinit

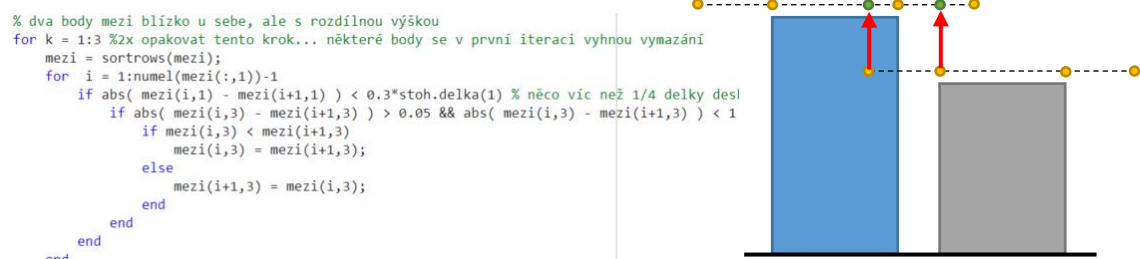
kmitání výsledné trajektorie nebo by trajektorie procházela skrz dva vysoké krajní stohy. Proto je vhodné tyto body z matice bezkolizních bodů vymazat.



Obr. 5.4: Vymazání bezkolizních bodů nízkého stohu

5.4.2 Úprava výšky bezkolizních bodů

Jiná situace nastává v momentě, kdy jsou vedle sebe dva stohy, jejichž výška není stejná, avšak rozdíl mezi výškami je malý (maximální hodnotu tohoto rozdílu si opět určíme, například 1 m). Uvažujme příklad, kdy levý stoh má výšku 2,25 m a pravý stoh má výšku 1,75 m a stohy jsou umístěné vedle sebe. Rozdíl výšek těchto stohů nepřesahuje 1 m, a tak souřadnici z bezkolizních bodů vytvořených dle nižšího stohu upravíme na hodnotu bezkolizních bodů vyššího stohu (Obr. 5.5). Tato úprava však platí pouze pro některé body nižšího stohu, konkrétně pro ty, které jsou od vyššího stohu vzdáleny do určité vzdálenosti. Logicky se tato vzdálenost bude odvíjet od rozměrů přemísťované desky, protože chceme zajistit, aby koncový efektor s deskou do stohu nenarazil. Po určité vzdálenosti bude tedy stoh mimo dosah této desky. Body, které jsou od vyššího stohu dále, než je námi určená vzdálenost, je tedy zbytečné upravovat. Tuto úpravu provádíme místo vymazání proto, abychom měli v daných výškových hladinách co nejvíce bezkolizních bodů. Díky tomu budeme schopni vytvořit nejlepší možnou trajektorii aproximací těchto bodů.

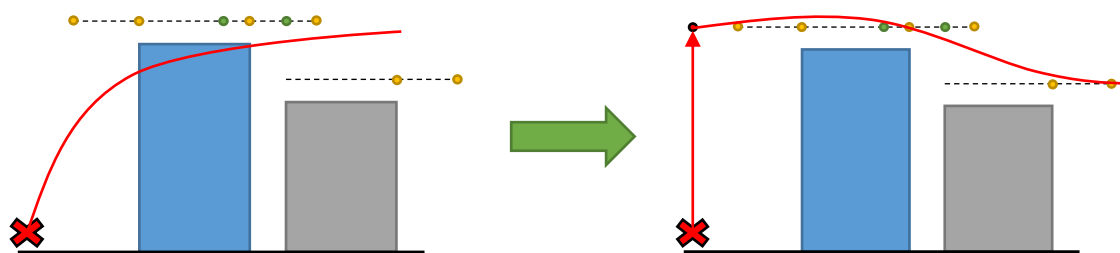


Obr. 5.5: Úprava výšky bezkolizních bodů

5.4.3 Úprava počáteční (koncové) polohy

Představme si situaci, kdy je některý z bezkolizních bodů v rovině xy blízko počáteční poloze. Pokud bude rozdíl jejich souřadnic v ose z velký, může se stát, že polynom nestačí z počáteční polohy do výšky tohoto bezkolizního bodu vystoupat dostatečně rychle a manipulátor by narazil do stohu. Pokud do dané výšky i přesto vystoupá, může se stát, že se trajektorie v další části bude aproximovat bezkolizní body velmi nepřesně, protože používáme polynom maximálně 5. stupně. Z toho důvodu vytvoříme náhradní počáteční polohu, do které bude manipulátor z původní počáteční polohy konat svislý pohyb. Stejný princip platí pro bezkolizní bod blízko koncové poloze.

```
% vysoký stoh na začátku  
mezi = sortrows(mezi);  
for k = 1:stoh.pocet  
    if abs(poc(1)-stoh.poloha(k,1)) < stoh.delka(k)*1.3  
        if abs(stoh.poloha(k,2) - fyt(stoh.kraj_x(k,1))) < stoh.sirka*1.3  
            if stoh.vyska(k) > poc2(3)  
                poc2(3) = stoh.vyska(k) + svisle;  
            end  
        elseif abs(stoh.poloha(k,2) - fyt(stoh.kraj_x(k,2))) < stoh.sirka*1.3  
            if stoh.vyska(k) > poc2(3)  
                poc2(3) = stoh.vyska(k) + svisle;  
            end  
        end  
    end  
end
```



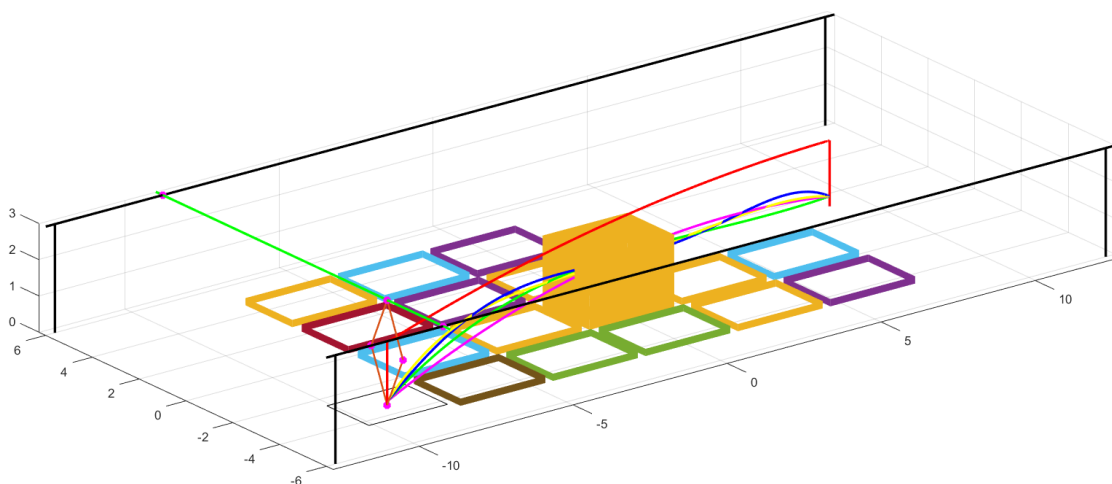
Obr. 5.6: Úprava počáteční polohy

Další situace, kdy je vhodné upravit počáteční (či koncovou) polohu nastává, pokud máme sklad plný nízkých stohů, avšak jeden stoh, který kříží cestu navrhované trajektorii, bude v porovnání s ostatními stohy nadměrně vysoký. Protože jsou polynomy vytvářeny metodou nejmenších čtverců, všechny stohy křížící trajektorii budou mít bezkolizní body nízko (malou hodnotu souřadnice z) a pouze 4 bezkolizní body budou ve výškové hladině vysokého stohu, tak se tyto 4 body do tvorby polynomu téměř neprojeví (Obr. 5.8).



```
% situace, kdy je nějaký stoh nadměrně vysoký oproti ostatním
for m = 1 : numel(mezi(:,1))
    if mezi(m,3) > mean(mezi(:,3))*1.5
        if abs(set(1) - mezi(m,1)) > abs(set(1)-poc(1))/2
            if mezi(m,3)*0.8 > poc2(3)
                poc2(3) = mezi(m,3)*0.8;
            end
            if mezi(m,3)*0.7 > set2(3)
                if abs( set(1) - mezi(m,1) ) < 1.5*stoh.delka
                    set2(3) = mezi(m,3)*0.7;
                end
            end
        elseif abs(set(1) - mezi(m,1)) < abs(set(1)-poc(1))/2
            if mezi(m,3)*0.8 > set2(3)
                set2(3) = mezi(m,3)*0.8;
            end
            if mezi(m,3)*0.7 > poc2(3)
                if abs( poc(1) - mezi(m,1) ) < 1.5*stoh.delka
                    poc2(3) = mezi(m,3)*0.7;
                end
            end
        end
    end
end
```

Obr. 5.7: Vytvoření náhradní počáteční (koncové) polohy z důvodu nadměrně vysokého stohu v porovnání s ostatními stohy



Obr. 5.8: Vytvoření náhradní počáteční a koncové polohy z důvodu nadměrně vysokého stohu v porovnání s ostatními stohy – grafické znázornění

5.5 Vytvoření polynomů

Body, které již prošly algoritmem pro jejich úpravu (nezáleží na tom, zda byly změněny nebo zůstaly původní) si označíme body „mezi“ a sloučíme do matice. Pomocí metody nejmenších čtverců a této matice vytvoříme základní polynomy 2. až 5. stupně (resp. 1. až 5. stupně při zohlednění prázdného skladu). Volba konkrétního stupně polynomu je předmětem kapitoly 5.11.



```
% SOUŘADNICE Z
if isempty(stoh.poloha)
    nz = [poc(3), set(3)];
    m = 1; % stupeň křivky
    p = polyfit(nx, nz, m);
    b1 = p(1);
    b0 = p(2);
    fzt = @(x) c1*x + c0;
else
% Aplikace bezkolizních bodu do polynomu
    nx = [poc2(1), mezi(:,1)', set2(1)];
    ny = [poc2(2), mezi(:,2)', set2(2)];
    nz = [poc2(3), mezi(:,3)', set2(3)];

% POLYNOM ST.2
    n2 = 2; % stupeň křivky
    p = polyfit(nx, nz, n2);
    c2 = p(1);
    c1 = p(2);
    c0 = p(3);
% fz2 = @(x) c2*x^2 + c1*x + c0;

% optimalizace
xt = linspace(poc(1), set(1), 10000);
z = c2*xt.^2 + c1*xt + c0;

C = [xt.^2; xt; ones(size(xt))];
d = z;
A = [];
b = [];

Aeq = [poc2(1)^2 poc2(1) 1;
       set2(1)^2 set2(1) 1];

beq = [poc2(3);
       set2(3)];

coef2 = lsqlin(C,d,A,b,Aeq,beq);

fz2 = @(x) coef2(1)*x^2 + coef2(2)*x + coef2(3);
```

Obr. 5.9: Matlab – Vytvoření polynomu 1. nebo 2. stupně dle splnění podmínky prázdného skladu (vlevo), optimalizace polynomu druhého stupně (vpravo)

Metoda nejmenších čtverců (MNC) všechny body aproximuje, a proto je nutné provést optimalizaci polynomů pro získání fixní počáteční a koncové polohy (resp. fixní náhradní počáteční a koncové polohy dle kapitoly 5.4.3).

Pro souřadnice ve směru osy y máme tedy polynom:

$$f_y(x) = c_{1y}x + c_{2y}. \quad (1)$$

Pro souřadnice ve směru osy z použijme příklad polynomu 2. stupně, kdy po optimalizaci získáme rovnici pro polynom ve tvaru:

$$f_{z2}(x) = c_{1z}x^2 + c_{2z}x + c_{3z}. \quad (2)$$

5.6 Základní svislý pohyb

Představme si dřevěnou desku přemísťovanou z jednoho stohu na druhý. Pokud budeme chtít z klidové polohy přímo konat pohyb po vytvořené trajektorii, tedy ve všech třech souřadných osách najednou, můžeme spodní část této desky a horní část desky pod ní poškodit nebo se od sebe



dané desky vlivem tření nepodaří vůbec oddělit. Proto bude manipulátor konat nejprve svislý pohyb do určité vzdálenosti (například 10 cm) tak, aby bylo možné s deskami manipulovat bez poškození. Pouze svislý pohyb vzhůru však vždy nezajistí překonání tření mezi deskami a tím jejich oddělení. Řešením by mohlo být naklopení desky z vodorovné polohy na počátku pohybu. Tato práce se však zabývá pouze plánováním trajektorie. Problematiku naklopení koncového efektoru je možné rozvíjet v dalším postupu projektu, a proto je zde tato možnost řešení pouze zmíněna.

5.7 Rychlostní profil

Velký vliv na negativní dynamické účinky, které působí na celou soustavu, má průběh pohybu. Je proto důležité se zabývat průběhem rychlosti a zrychlení pro maximální zmírnění těchto negativních účinků, kterými mohou být například překmit jeřábové kočky a nůžkového mechanismu při zastavení pohybu vedoucí k možné srážce se stohem, dále dosažení vlastní frekvence manipulátoru a vznik rezonance, která by vedla k poškození jak manipulátoru, tak i objektů uvnitř skladu a také je zde riziko únavových lomů součástí manipulátoru.

5.7.1 Ideální S-křivka

Podle obrázku 5.11 předpokládáme ideální S-křivku bez periody s lineárním posuvem a vyšetřujeme zrychlování z počáteční rychlosti v_0 na požadovanou rychlost v_s , kdy předpokládáme počáteční rychlost $v_0 = 0$, tedy startujeme z klidu. [6]

- a) Plocha pod konkávní a konvexní částí S-křivky je rovna ploše pod přímkou zrychlení a_r ,
- b) Maximální zrychlení S-křivky (v čase $T/2$, označené a_s) má větší směrnici než zrychlení a_r .



Rovnici pro S-křivkový rychlostní profil předpokládáme ve tvaru:

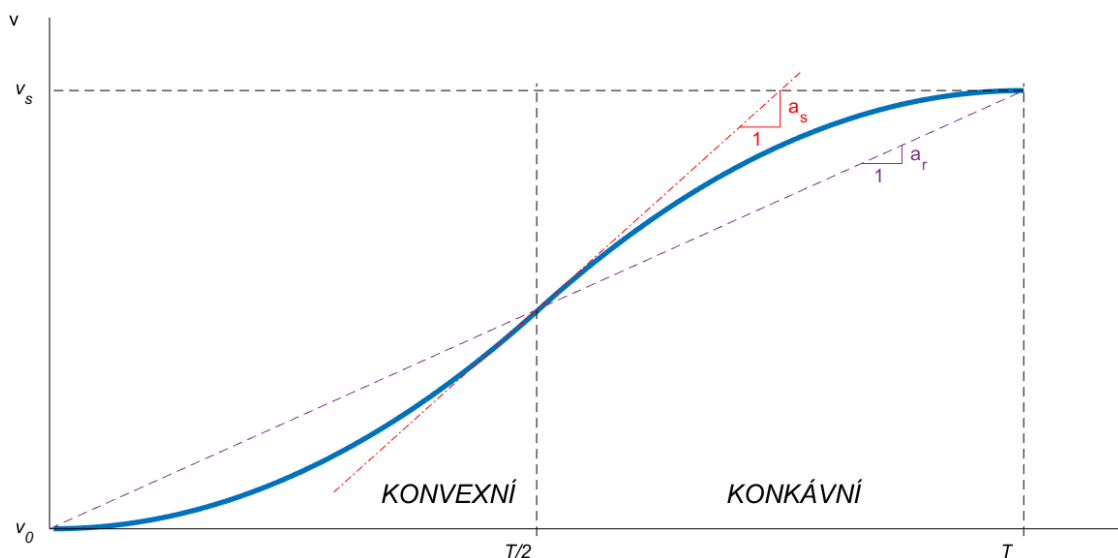
$$v(t) = c_0 + c_1t + c_2t^2 . \quad (3)$$

Dále rovnice pro zrychlení a konstantní jerk:

$$a(t) = c_1 + 2c_2t , \quad (4)$$

$$j(t) = 2c_2 . \quad (5)$$

Perioda zrychlování je rozdělena do dvou částí – konvexní částí následovanou částí konkávní.



Obr. 5.10: S-křivka rychlostního profilu

5.7.1.1 Konvexní část

Okrajové podmínky pro konvexní část jsou: $v(0) = v_0; a(0) = 0;$
 $a\left(\frac{T}{2}\right) = a_s; j(0) = j_m$, kde j_m je hodnota veličiny jerk a a_s je maximální zrychlení S-křivky v inflexním bodě. Po aplikaci těchto okrajových podmínek vyřešíme rovnice (4) a (5) a získáme:

$$c_0 = v_0 , \quad (6)$$



$$c_1 = 0, \tag{7}$$

$$c_2 = \frac{j_m}{2} = \frac{a_s}{T}. \tag{8}$$

Odtud získáváme vztah pro výpočet délky periody, a to:

$$T = \frac{2a_s}{j_m}. \tag{9}$$

Aplikací získaných integračních konstant můžeme dopočítat žádané veličiny, jako jsou poloha (s), rychlost (v) a zrychlení (a):

$$s(t) = v_0 t + \frac{j_m t^3}{6}, \tag{10}$$

$$v(t) = v_0 + \frac{j_m t^2}{2}, \tag{11}$$

$$a(t) = j_m t. \tag{12}$$

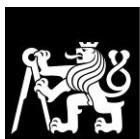
Předpokládá se, že poloha je na počátku rovna nule ($s(0) = 0$) a s tedy reprezentuje spíše změnu polohy než její absolutní hodnotu.

Tyto rovnice mohou být aplikovány pro $0 \leq t \leq \frac{T}{2}$. V čase $t = \frac{T}{2}$ bude $v = \frac{v_s + v_0}{2}$, tedy rychlost bude rovna střední rychlosti periody zrychlování. [6]

5.7.1.2 Konkávní část

Tato část je aplikována pro $\frac{T}{2} \leq t \leq T$. Předpokládejme měření času od hodnoty nula, pak jsou okrajové podmínky následující:

$$v(0) = v_h = \frac{v_s + v_0}{2},$$



$$a(0) = a_s ,$$

$$a(T/2) = 0 ,$$

$$j(0) = -j_m .$$

Zbytek řešení je analogický dle konvexní části.

5.7.1.3 Maximální hodnota jerk

Předpokládejme velikost jerk j_m , počáteční rychlost v_0 a koncovou rychlost v_s , pak můžeme určit rychlosti v_1 a v_2 , které ukončují konvexní část a začínají konkávní část při maximální hodnotě jerk. [6]

$$v_1 = v_0 + \frac{a_s^2}{2j_m} , \tag{13}$$

$$v_2 = v_s - \frac{a_s^2}{2j_m} . \tag{14}$$

Předpokládejme $v_1 = v_2$, pak při daném zrychlení a_s a daném rozdílu rychlostí $\Delta v = v_s - v_0$ získáme vztah pro výpočet maximální hodnoty jerk:

$$j_m = \frac{a_s^2}{\Delta v} . \tag{15}$$

5.8 Aplikace S-křivky

Pro určení rychlostního profilu použijme rovnice (10), (11) a (12), avšak abychom byli schopni ukládat jednotlivé polohy pro daný čas do matice, upravíme tyto rovnice tak, že čas budeme měřit absolutně od počátku, dostaneme tedy:

$$a(t) = a_0 + j_m t , \tag{16}$$



$$v(t) = v_0 + a_0 t + \frac{j_m t^2}{2}, \quad (17)$$

$$s(t) = s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6}. \quad (18)$$

Znaménka v rovnicích a hodnoty jednotlivých konstant jsou získány z počátečních podmínek.

Pro zjednodušení uvažujme:

$$j_{mx} = j_{my} = j_{mz} = j_m,$$

$$a_{0x} = a_{0y} = a_{0z} = a_0,$$

$$v_{0x} = v_{0y} = v_{0z} = v_0,$$

$$s_{0x} = s_{0y} = s_{0z} = s_0.$$

Toto zjednodušení si můžeme dovolit, protože následující rovnice budeme vždy ve směru osy y a osy z násobit příslušnými koeficienty c_{1y}, c_{2y} a c_{1z}, \dots, c_{nz} z rovnic (1) a (2), kde n je stupeň polynomu.

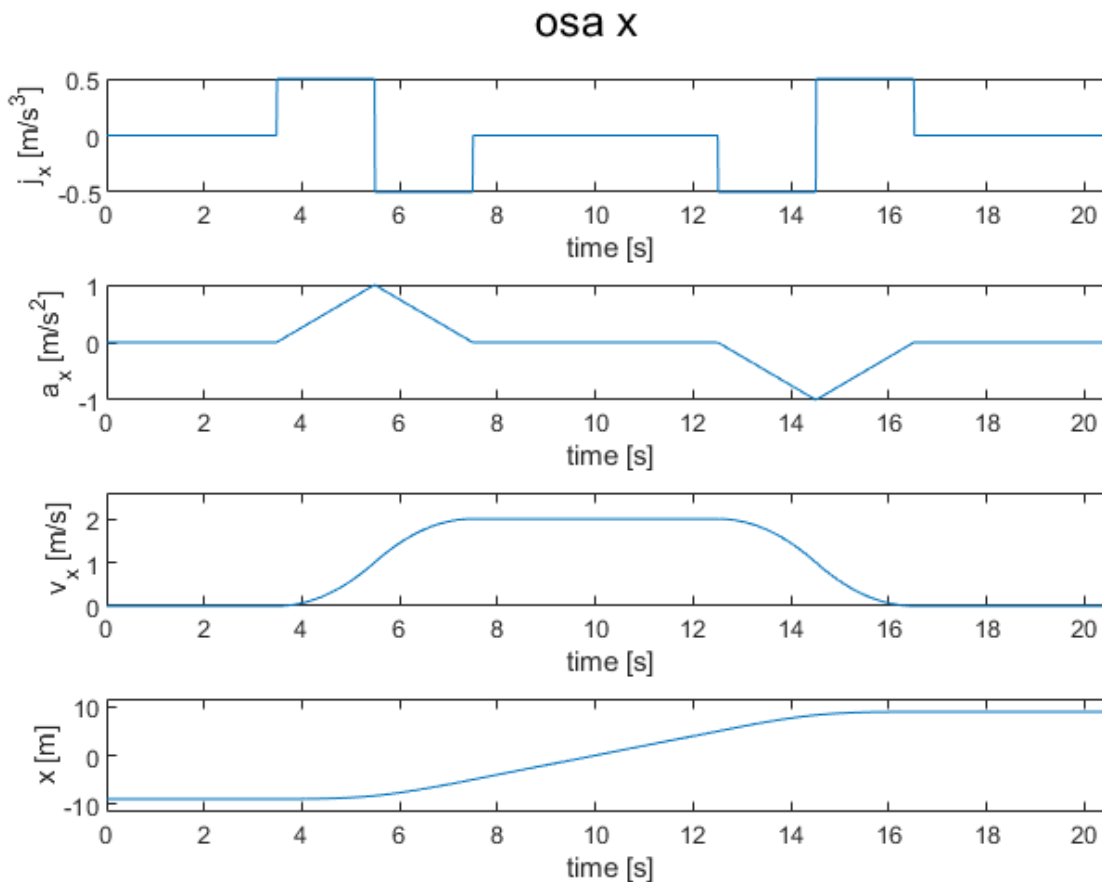
5.8.1 Osa x

Ve směru osy x jsou pohybové rovnice téměř totožné s rovnicemi (16), (17) a (18), pouze je však označíme indexem x :

$$a_x(t) = a_0 + j_m t, \quad (19)$$

$$v_x(t) = v_0 + a_0 t + \frac{j_m t^2}{2}, \quad (20)$$

$$x(t) = s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6}. \quad (21)$$



Obr. 5.11: Grafy pohybových veličin ve směru osy x

5.8.2 Osa y

Rovnici pro souřadnici ve směru y máme určenou polynomem, tedy rovnicí (1). Dosadíme rovnici (21) do rovnice (1) a získáme:

$$f_y(x(t)) = c_{1y}x(t) + c_{2y} = c_{1y} \left(s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6} \right) + c_{2y} = f_y(t). \quad (22)$$

Tím získáme polohu ve směru osy y závislou na čase a derivováním můžeme tedy získat rovnice pro rychlost a zrychlení:

$$v_y(t) = c_{1y} \left(v_0 + a_0 t + \frac{j_m t^2}{2} \right) + c_{2y}, \quad (23)$$



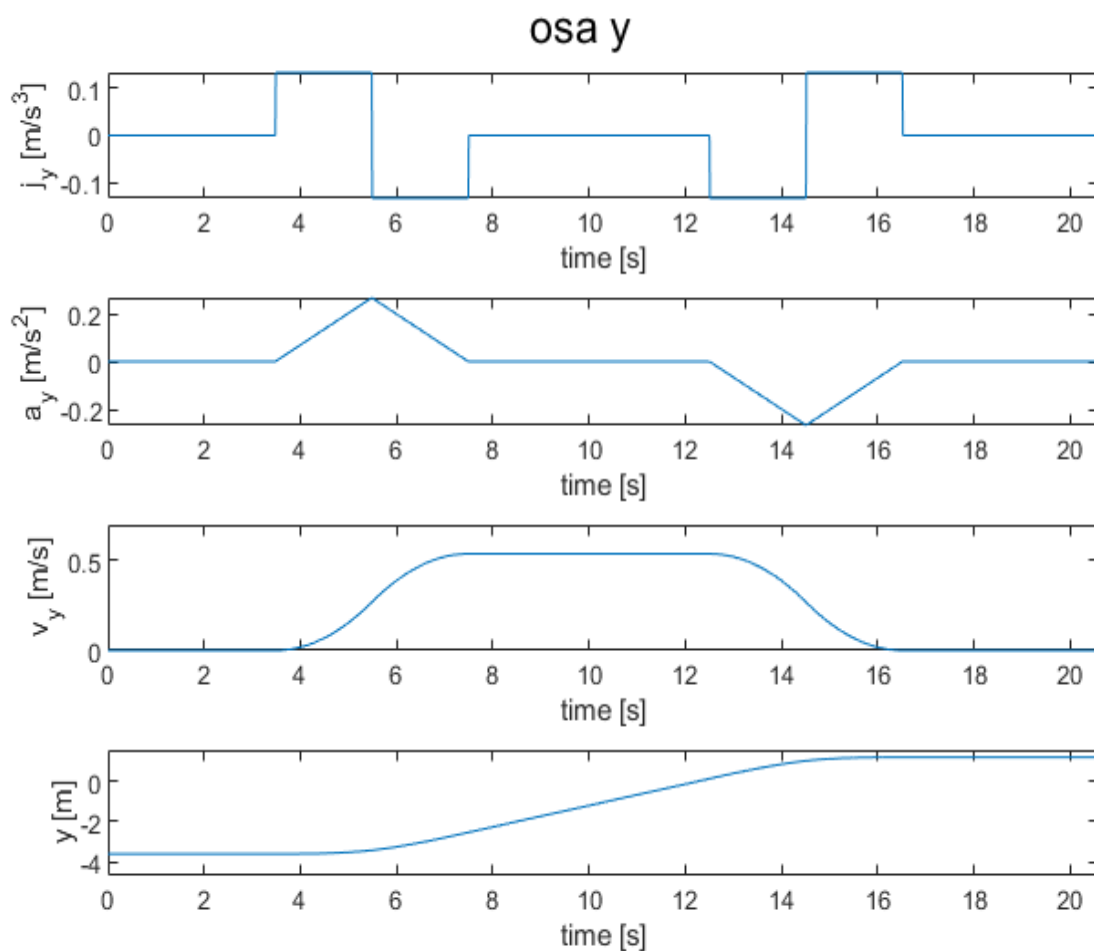
$$a_y(t) = c_{1y}(a_0 + j_m t) + c_{2y}.$$

(24)

Dále označme $y(t) = f_y(t)$ a získáme:

$$y(t) = c_{1y} \left(s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6} \right) + c_{2y}.$$

(25)



Obr. 5.12: Grafy pohybových veličin ve směru osy y

5.8.3 Osa z

Polohu, rychlost a zrychlení ve směru osy z získáme analogicky, jako ve směru osy y. Pro polynom 2. stupně tedy dostaneme:

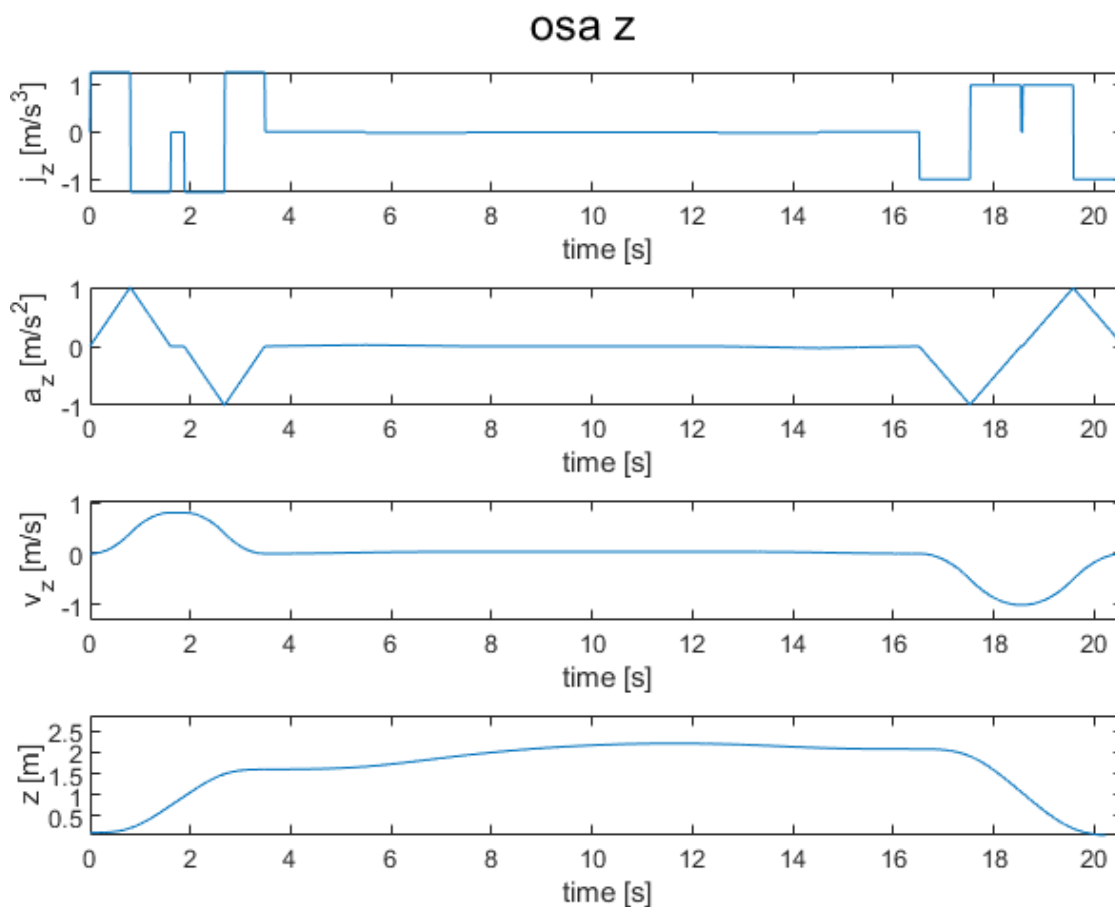


$$z(t) = c_{1z} \left(s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6} \right)^2 + c_{2z} \left(s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6} \right) + c_{3z}, \quad (26)$$

$$v_z(t) = c_{1z} \left(v_0 + a_0 t + \frac{j_m t^2}{2} \right)^2 + c_{2z} \left(v_0 + a_0 t + \frac{j_m t^2}{2} \right) + c_{3z}, \quad (27)$$

$$a_z(t) = c_{1z} (a_0 + j_m t)^2 + c_{2z} (a_0 + j_m t) + c_{3z}. \quad (28)$$

Analogicky bychom získali rovnice pro polynomy 3. až 5. stupně. Při tvorbě matice poloh závislých na čase vytvoříme trajektorii paralelně pro všechny typy polynomů, tedy 2. až 5. stupně.



Obr. 5.13: Grafy pohybových veličin ve směru osy z



5.8.4 Počáteční podmínky jednotlivých úseků

Dle počátečních podmínek pro jednotlivé úseky vypočítáme rovnici rovnice (15) velikost veličiny jerk a dále dle rovnice (9) vypočítáme časovou délku tohoto úseku. Rovnice (19) až (28) aplikujeme na všechny úseky rychlostního profilu, rozdíly však budou v počátečních podmínkách, které jsou následující:

Tabulka 1: Počáteční podmínky pro rychlostní profil ve tvaru S-křivky

Část úseku	Zrychlování		Úsek s konstantní rychlostí	Zpomalování	
	Konvexní část	Konkávní část		Konkávní část	Konvexní část
Pořadí úseku	1	2	3	4	5
Počáteční podmínky	$s_0 = x_{end}$	$s_0 = x_{end}$	$s_0 = x_{end}$	$s_0 = x_{end}$	$s_0 = x_{end}$
	$v_0 = 0$	$v_0 = \frac{v_{max}}{2}$	$v_0 = v_{max}$	$v_0 = v_{max}$	$v_0 = \frac{v_{max}}{2}$
	$a_0 = 0$	$a_0 = a_s$	$a_0 = 0$	$a_0 = 0$	$a_0 = -a_s$
	$j_m = j_m$	$j_m = -j_m$	$j_m = 0$	$j_m = -j_m$	$j_m = j_m$

Kde x_{end} označuje poslední hodnotu v matici polohy ve směru osy x . Pokud jsme na počátku pohybu, bude tato hodnota rovna souřadnici x počáteční polohy. Veličina v_{max} znázorňuje maximální rychlost, které chceme při pohybu dosáhnout. Pro dlouhé vzdálenosti se budeme kromě rychlostního profilu ve tvaru S-křivky pohybovat maximální možnou rychlostí, která bude konstantní. Je tedy jasné, že jerk a zrychlení budou v tomto úseku rovny nule.

5.8.5 Dráha úseku s konstantní rychlostí

Pro výpočet délky dráhy, po které se manipulátor bude pohybovat konstantní rychlostí použijme následující úvahu. Při zrychlování a zpomalování



je profil S-křivky symetrický a dráha, kterou manipulátor překoná při zrychlování a zpomalování bude tedy stejná. Pokud od celkové dráhy, kterou manipulátor překoná odečteme dvojnásobek dráhy potřebné pro zrychlení, získáme vztah pro délku dráhy úseku s konstantní rychlostí. Opět uvažujeme dráhu pouze ve směru osy x , protože do ostatních směrů je dráha přepočítána pomocí rovnic (1) a (2).

$$s_{vkonst} = |x_{poc} - x_{set}| - 2 \cdot |x_{poc} - x_{end}|. \quad (29)$$

Kde s_{vkonst} je dráha, kterou manipulátor překoná pohybem s konstantní rychlostí, x_{poc} je souřadnice x počáteční polohy, x_{set} souřadnice x koncové polohy a x_{end} je poloha x na konci periody zrychlování.

Čas potřebný pro překonání této dráhy je pak:

$$t = \frac{s_{vkonst}}{v_{max}}. \quad (30)$$

5.8.6 S-křivka pro svislý pohyb

Tabulka 1 předpokládá situaci, kdy již konáme pohyb ve všech třech osách současně. Pro svislý pohyb se mění pouze souřadnice z , a proto platí:

$$a_z(t) = a_0 + j_m t, \quad (31)$$

$$v_z(t) = v_0 + a_0 t + \frac{j_m t^2}{2}, \quad (32)$$

$$z(t) = s_0 + v_0 t + \frac{a_0 t^2}{2} + \frac{j_m t^3}{6}. \quad (33)$$



5.9 Nedosažení maximální rychlosti

Předpokládejme, že velikost dráhy je malá a při předepsaném maximálním zrychlení v inflexním bodě S-křivky nedosáhneme na požadovanou rychlost dostatečně rychle. Manipulátor by tedy překonal mnohem větší vzdálenost pro dosažení požadované rychlosti, než je délka dráhy. Použijme tedy iterace pro snížení maximální požadované rychlosti. Dokud platí následující podmínka:

$$|x_{end} - x_{poc}| > \frac{|x_{set} - x_{poc}|}{2}, \quad (34)$$

kde x_{end} je poloha ve směru osy x po periodě zrychlování, pak:

$$v_{max} = 0,9 \cdot v_{maxp}, \quad (35)$$

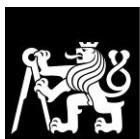
kde v_{max} je nová hodnota požadované maximální rychlosti a v_{maxp} je její původní hodnota. Hodnota nové maximální rychlosti se promítne také do počátečních podmínek.

Jinými slovy: „Pokud vzdálenost, kterou překoná manipulátor při periodě zrychlování, je větší než polovina celé dráhy, snížíme požadovanou maximální rychlost na 90 % její původní hodnoty.“

Toto snižování rychlosti provádíme do chvíle, dokud je vzdálenost pro periodu zrychlování menší nebo rovna polovině celkové dráhy. Z toho vyplývá, že po úpravě požadované maximální rychlosti nebude rychlostní profil obsahovat část s konstantní rychlostí nebo bude mít tato část zanedbatelnou velikost.

```
while abs( X(end,3) - X(1,3) ) > 1/2*abs( poc2(3) - poc(3) )
    vmaxs1 = 0.9*vmaxs1;
    X(2:end,:) = [];
    v(2:end,:) = [];
    a(2:end,:) = [];
    j(2:end,:) = [];
```

Obr. 5.14: Matlab – snížení maximální dosažitelné rychlosti



5.9.1 Svislý pohyb

Z kapitoly 5.6 víme, že pokud nebude potřeba vytvořit náhradní počáteční či koncovou polohu, bude dráha svislého pohybu velmi krátká (například 10 cm). Z toho vyplývá, že pokud bude maximální předepsaná rychlost stejné velikosti, jako pro hlavní pohyb, pak bude potřeba tuto rychlost snížit (za předpokladu zachování proporcí celého skladu, výška (osa z) několika násobně menší než délka (osa x)). Proto se dá předpokládat, že pro svislý pohyb jakékoliv velikosti (při zachování těchto proporcí) bude mít vždy maximální dosažitelná rychlost pro svislý pohyb nižší hodnotu než původní předepsaná maximální rychlost.

Pro svislý pohyb, kdy se mění pouze souřadnice z platí tedy podmínka (analogie k rovnici (34)):

$$|z_{end} - z_{poc}| > \frac{|z_{set} - z_{poc}|}{2}, \quad (36)$$

kde z_{end} je poloha ve směru osy z po periodě zrychlování, z_{poc} je počáteční poloha ve směru osy z a z_{set} je souřadnice z náhradní počáteční polohy.

Analogicky lze vytvořit podmínku pro svislý pohyb na konci trajektorie.

5.10 Velikost maximální dosažitelné rychlosti

Pokud uvažujeme reálný systém, tak hlavní omezení, které jej limituje, je velikost maximálního zrychlení, jehož limitem je zrychlení, kdy nedojde k překonání podtlaku vytvořeného přísavkami a nedojde k odtržení přísavek od desky. Toto nám omezuje pouze velikost maximálního zrychlení, nikoliv velikost maximální rychlosti.

Omezení pro maximální rychlost bychom mohli hledat v maximálních otáčkách elektromotorů. Předpokládejme však, že maximální otáčky elektromotoru budou dostatečně velké a pokud tedy bude maximální rychlost



z konstrukčního hlediska neomezená, budeme její omezení hledat v rychlostním profilu. Vraťme se ke kapitole 5.9, kde snižujeme maximální rychlost na takovou hodnotu, aby dráha, kterou manipulátor překoná na úseku zrychlování byla menší nebo rovna polovině celkové dráhy. Proto pro výpočet veličiny jerk dle rovnice (15) použijeme hodnotu maximální rychlosti dostatečně velkou (například $v_s = v_{max} = 20 [m/s]$), abychom splnili podmínku (34) a (36). Z toho plyne, že všechny průběhy pohybu budou tedy bez části s konstantní rychlostí nebo bude tato část zanedbatelné velikosti.

Pokud však vznikne nějaké konstrukční či jiné omezení na velikost maximální rychlosti, vznikne v rychlostním profilu část s konstantním průběhem rychlosti. Tato práce se nezabývá řešením konkrétních omezení, ale pouze možnými stavy tak, aby byl plánovač trajektorie parametrický a zcela univerzální.

5.11 Volba výsledné dráhy

Po vytvoření trajektorií sledujících polynomy 2. až 5. stupně je nutné vybrat jednu, která bude následně odeslána do řídicího systému. Kritéria pro výběr trajektorie mohou být:

- časově optimální řešení,
- dynamické účinky na soustavu,
- dosažení maximální možné výšky.

Na základě těchto kritérií zvolíme jedno optimální řešení.

5.11.1 Časově optimální řešení

Při výpočtu potřebného času pro překonání dráhy určené polynomem 2. až 5. stupně zjistíme, že všechny polynomy závisí na časové závislosti polohy x . Pohyb ve směru osy x však není závislý na stupni zvoleného



polynomu v rovině xz . Lze tedy říct, že pro trajektorii jakéhokoliv polynomu v rovině xz je čas manipulace totožný.

5.11.2 Dynamické účinky na soustavu

Při zvyšování stupně polynomu docílíme ve většině případů přesnější aproximace bezkolizních bodů a vypočítaná trajektorie se bude blížit pomyslné ideální trajektorii. Současně s tím se dá také očekávat zvýšení počtu prudkých změn pohybu, které při vysoké rychlosti mohou způsobit velké dynamické účinky na celou soustavu. Nabízí se tedy zvolit finální trajektorii pro nejmenší možný stupeň polynomu.

Vraťme se však k principu metody nejmenších čtverců, kdy dané body aproximujeme, a proto není jisté, zda při zvolené bezpečnosti, o jejíž hodnotu je posunut bezkolizní bod nad daný stoh, se nedostane manipulátor do kolize právě s tímto stohem. Proto každý řádek matice polohy závislé na čase porovnáme postupně s polohami a výškami všech stohů a pokud dojde ke kolizi, označíme tento polynom za nepoužitelný pro výslednou trajektorii. Toto opakujeme pro všechny vytvořené polynomy.

```
x2er = 0;
x3er = 0;
x4er = 0;
x5er = 0;
for st = 1 : stoh.pocet
    for ax = 1 : numel(X5(Ls15+1:L5,1))
        if abs( X2(Ls15+ax,2) - stoh.poloha(st,2) ) < stoh.sirka*1.05
            if abs( X2(Ls15+ax,1) - stoh.poloha(st,1) ) < stoh.delka*1.05
                if X2(Ls15+ax,3) < stoh.vyska(st)+0.1
                    x2er = 1;
                end
            end
        end
    end

    if abs( X3(Ls15+ax,2) - stoh.poloha(st,2) ) < stoh.sirka*1.05
        if abs( X3(Ls15+ax,1) - stoh.poloha(st,1) ) < stoh.delka*1.05
            if X3(Ls15+ax,3) < stoh.vyska(st)+0.1
                x3er = 1;
            end
        end
    end
end
```

Obr. 5.15: Matlab – volba výsledné dráhy – kritérium pro bezkolizní trajektorii (zobrazeno pouze pro polynom 2. a 3. stupně)



5.11.3 Dosažení maximální výšky

Pokud by při průjezdu naplánovanou trajektorií dosáhl manipulátor větší hodnoty souřadnice z , než je maximální dovolená výška, bude polynom vyřazen. Toto omezení je vytvořeno z důvodu, aby nedošlo v kinematickém modelu k překlopení nůžkového mechanismu. V praxi bude maximální výška omezena koncovým dorazem na nůžkovém mechanismu, a právě souřadnice z koncového efektoru, kdy nůžkový mechanismus dosáhne této koncové polohy, je v kódu označena „maxvyska“. Způsob označení polynomů jako vyřazené je zobrazeno na obr. 5.16.

```
for ax = 1 : numel(X5(:,1))
    if X2(ax,3) > maxvyska
        x2er = 1;
    end

    if X3(ax,3) > maxvyska
        x3er = 1;
    end
```

Obr. 5.16: Matlab – vyřazení trajektorie při překročení maximální povolené výšky

5.11.4 Vyhodnocení

Z obr. 5.15 a obr. 5.16 vyplývá, že pokud podmínka bezpečného bezkolizního průjezdu trajektorií není splněna (Obr. 5.15) nebo by při pohybu přesáhl manipulátor maximální výšku a dostal by se do singulární polohy (Obr. 5.16), přiřadí se veličině x_{ner} (kde n je stupeň polynomu, $n \in \{2, \dots, 5\}$) hodnota 1. Na obr. 5.17 je zobrazen způsob vyhodnocování použitelných trajektorií, který pracuje na následujícím principu:

Začne od nejmenšího stupně polynomu a pokud je $x_{ner} = 1$ pro daný stupeň polynomu, přeskočí na polynom o stupeň vyšší. Pokud je $x_{ner} = 1$ pro všechny polynomy, žádná trajektorie nebude zvolena a program napíše chybovou hlášku. Tomuto stavu je dále věnována kapitola 5.13.1. Pokud je pro nějaký polynom hodnota $x_{ner} = 0$, je pro výslednou trajektorii vybrán právě tento polynom a matice polohy, rychlosti, zrychlení a jerk je pro výslednou trajektorii převzata z hodnot pro tento polynom. Pro pohyb manipulátoru je důležitá zejména matice polohy. Matice zbylých veličin jsou vhodné spíše



pro kontrolu a simulační účely. Hodnoty těchto veličin pro ostatní polynomy ve výsledné trajektorii použity nebudou.

```
if x2er == 1
    if x3er == 1
        if x4er == 1
            if x5er == 1
                disp('ERROR - DRÁHA NEVYKRESLENA')
            elseif x5er == 0
                Xtraj = X5;
                v(Ls15+1 : L5,3) = v5(Ls15+1 : L5,3);
                a(Ls15+1 : L5,3) = a5(Ls15+1 : L5,3);
                j(Ls15+1 : L5,3) = j5(Ls15+1 : L5,3);
            end
        elseif x4er == 0
            Xtraj = X4;
            v(Ls15+1 : L5,3) = v4(Ls15+1 : L5,3);
            a(Ls15+1 : L5,3) = a4(Ls15+1 : L5,3);
            j(Ls15+1 : L5,3) = j4(Ls15+1 : L5,3);
        end
    elseif x3er == 0
        Xtraj = X3;
        v(Ls15+1 : L5,3) = v3(Ls15+1 : L5,3);
        a(Ls15+1 : L5,3) = a3(Ls15+1 : L5,3);
        j(Ls15+1 : L5,3) = j3(Ls15+1 : L5,3);
    end
elseif x2er == 0
    Xtraj = X2;
    v(Ls15+1 : L5,3) = v2(Ls15+1 : L5,3);
    a(Ls15+1 : L5,3) = a2(Ls15+1 : L5,3);
    j(Ls15+1 : L5,3) = j2(Ls15+1 : L5,3);
end
```

Obr. 5.17: Volba výsledné dráhy – výběr polynomu

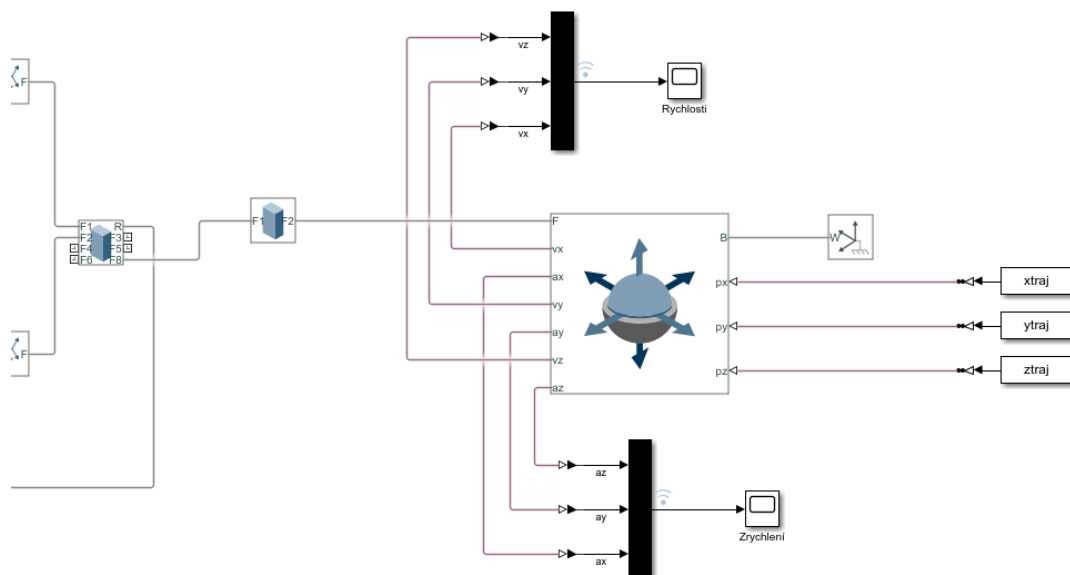
5.12 Odeslání trajektorie

Před odesláním trajektorie do řídicího systému (pro simulační účely do Simscape Multibody pro digitální dvojče) je nutné vytvořit matici časových bodů, a to na základě celkového času, který je potřebný pro manipulaci a také na počtu bodů v matici polohy, který je závislý na zvoleném diskretizačním kroku. Matici polohy dále rozdělíme na matice jednotlivých směrů x , y , z a tyto matice spojíme s maticí časových bodů. Tím získáme vždy matici bodů polohy v daném směru závislých na čase, kterou následně



odešleme do řídicího systému nebo pro simulační účely do vazby 6-DOF Joint digitálního dvojčete.

Na základě struktury řídicího systému nemusíme provádět krok rozdělení matice polohy na jednotlivé složky, ale spojit přímo tuto matici s maticí časových bodů. Toto rozdělení je nutné zejména kvůli simulačním účelům, protože vazba 6-DOF Joint v prostředí Simscape Multibody vyžaduje polohu závislou na čase rozdělenou na jednotlivé složky. Tyto složky jsou zobrazeny na obr. 5.18 jako vstupní data na pravé straně vazby 6-DOF Joint. Na levé straně jsou pak výstupy pro měření rychlosti a zrychlení a také úchyt ke středovému bodu koncového efektoru.



Obr. 5.18: Odeslání trajektorie do vazby se 6 stupni volnosti

5.13 Další vývoj

Dosud jsme se věnovali popisu již vytvořeného plánovače trajektorie. Ten lze nyní považovat za dokončený. Nyní se přesuneme k úvahám a možnostem dalšího vývoje tohoto plánovače.



5.13.1 Error dráha nevykreslena

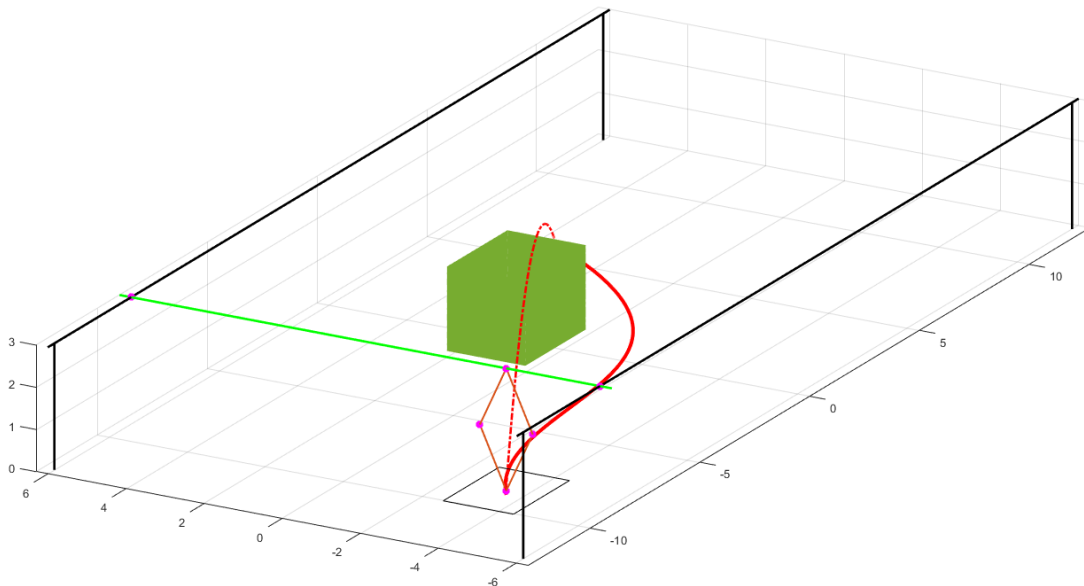
Pokud výběru podle kapitoly 5.11.4 nevyhovuje žádný z polynomů, program napíše: „ERROR, DRÁHA NEVYKRESLENA“. To znamená, že žádný z polynomů nebyl zvolen jako výsledná varianta trajektorie a do řídicího systému tedy nejsou odeslána žádná data. Při testování plánovače trajektorie již byly všechny stavy, které vedly k chybě při výpočtu, prověřeny a byla proti nim provedena opatření v podobě úprav bezkolizních bodů, kterým je věnována kapitola 5.4. V průběhu dalšího testování nebylo při náhodném generování počtu desek v jednotlivých stozích tohoto stavu dosaženo a trajektorie byla vždy zvolena a odeslána. To však nezaručuje možnost, že by takový případ nemohl nastat.

Jako opatření je zde možno použít cyklus *while*, tedy dokud je splněna podmínka, že alespoň některý z polynomů vyhovuje požadavkům z kapitoly 5.11.2, bude plánovač postupovat stejně, jako doposud. Pokud však bude tato podmínka porušena, tzn. žádná z trajektorií nevyhovuje výše zmíněným požadavkům, je zde několik možností dalšího postupu.

- a) Vytvoří se náhradní počáteční i koncová poloha ve výšce nejvyššího ze stohů, který kříží trajektorii a její obálku, tedy i prostor, kterým se bude pohybovat přemísťovaná dřevěná deska, a manipulátor bude mezi těmito náhradními polohami konat pohyb po přímce. Toto je však časově nejnáročnější varianta.
- b) Dalším možným řešením je vytvořit počáteční i koncovou polohu pouze do určité části výšky nejvyššího ze stohů v závislosti na vzdálenosti tohoto stohu od počáteční a koncové polohy a na výšce zbylých stohů, které kříží trajektorii nebo její obálku. Výsledný pohyb by pak byl realizován pomocí polynomu druhého stupně.

5.13.2 Energeticky nejméně náročná cesta

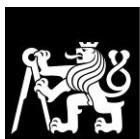
Představme si situaci, kdy máme téměř prázdný sklad, ve kterém se nachází jeden vysoký stoh s dřevěnými deskami, který stojí v cestě mezi počáteční a koncovou polohou (uvažujeme přímkovou trajektorii v rovině xy). Je zřejmé, že zvedat koncový efektor manipulátoru i s dřevěnou deskou je energeticky velmi náročné a nevýhodné, pokud existuje i jiná možnost, jak předejít kolizi. Řešením je vyhnout se stohu v rovině xy , tedy ho „objet z boku“. Tím se však značně zkomplikuje celého plánování, protože se bude v obou rovinách (xy a xz) manipulátor současně pohybovat po polynomech stupně vyššího než 1, avšak z hlediska energetické náročnosti pohybu a možného zkrácení manipulačního času, je vhodné se tomu při dalším vývoji věnovat.



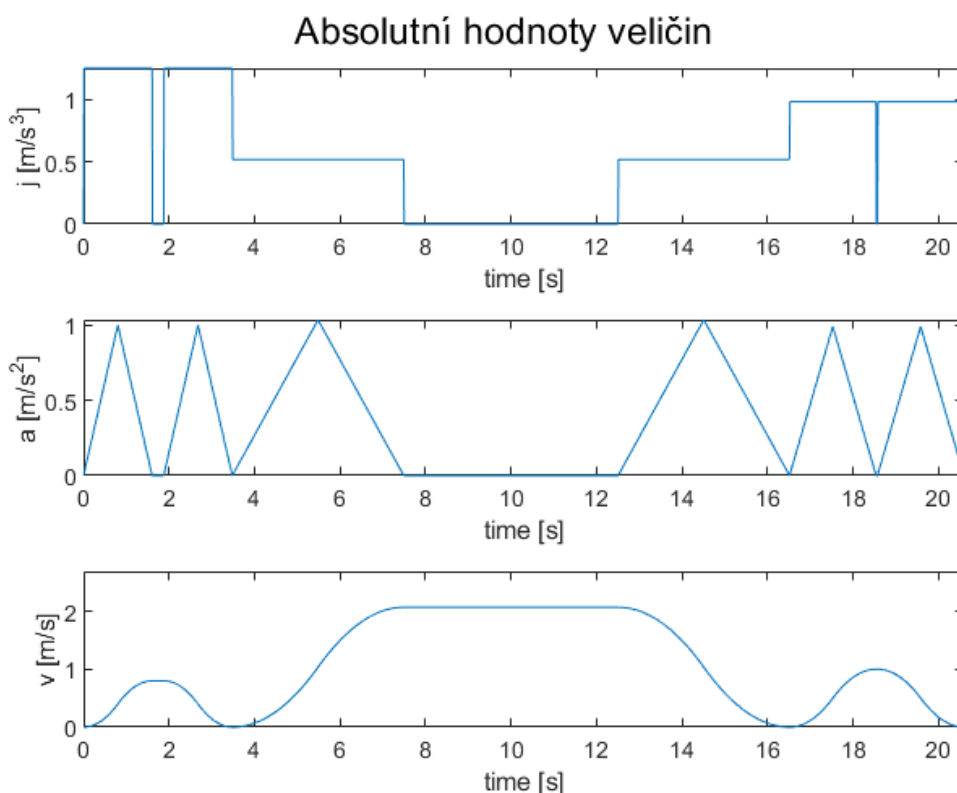
Obr. 5.19: Příklad optimalizace pomocí energeticky nejméně náročné cesty

5.13.3 Maximální absolutní zrychlení

Na obr. 5.11 je z grafu zrychlení ve směru osy x patrné, že tento modelový příklad uvažoval zrychlení v jedné ose maximálně $a_n = 1 [m \cdot s^{-2}]$, kde n jsou složky x, y, z , avšak z obr. 5.20 pro absolutní hodnoty veličin je zřejmé, že absolutní hodnota zrychlení je mírně větší než $1 [m \cdot s^{-2}]$. V těchto



proporcích skladu můžeme díky poměru délky, šířky a výšky skladu předpokládat, že zrychlení v ose x bude dominantní a ve zbylých osách, zejména v ose z , budou mít zrychlení na velikost absolutní hodnoty zrychlení malý (téměř zanedbatelný) vliv. Z hlediska univerzálnosti tohoto plánovače je vhodné se zaměřit i na tento faktor. Pokud by délka skladu nebyla několika-násobně větší než jeho výška, velikost výsledného absolutního zrychlení by byla značně rozdílná (v porovnání k velikosti zadanému maximálnímu zrychlení v jedné ose) a došlo by tak například k odtržení desky od přísavek. Je však vhodné se také zamyslet nad tím, jestli při malých rozdílech délky, šířky a výšky skladu je stále vhodné použít kartézský manipulátor. Plánovač by však měl být univerzální i v tomto ohledu, tedy bez závislosti na typu manipulátoru.



Obr. 5.20: Grafy absolutních hodnot pohybových veličin



6 ZÁVĚR

Cílem bakalářské práce bylo vytvořit automatický plánovač trajektorie pro manipulátor do automatického skladu deskového materiálu. Jelikož je vytvoření plánovače část projektu, kterým se Ústav mechaniky, biomechaniky a mechatroniky na Fakultě strojní ČVUT v Praze zabývá, bylo již ze zadání projektu předurčeno, že se jedná o automatický sklad a pro manipulaci s objekty byl vybrán kartézský manipulátor s nůžkovým mechanismem, kterému je věnována rešerše.

Byly vytvořené simulační modely, a to jak v podobě drátového zobrazení pro snadnou a nenáročnou reprezentaci celého skladu a vykreslení navržené trajektorie v podobě 3D grafu v prostředí Matlab, ale i simulační model v prostředí Simscape Multibody. V tomto modelu jsou použity ideální vazby, jejichž kombinacemi je snaha docílit stejného chování, jako u reálných vazeb a přiblížit tak celý model co nejlépe reálnému manipulátoru. Model zastává pozici digitálního dvojčete manipulátoru a je možné v něm nadále vyvíjet řídicí systém.

Automatické plánování trajektorie je založeno na pohybu po hladkých křivkách, konkrétně polynomech 2. až 5. stupně v rovině xz a po polynomu 1. stupně rovině xy . Existují podmínky, zmíněné v tomto textu, při jejichž splnění je možné se pohybovat v rovině xz také po polynomu prvního stupně. Veškerý pohyb manipulátoru je realizován pomocí S-křivky rychlostního profilu, čímž se docílí nejkratšího možného manipulačního času při maximálním možném zmírnění dynamických účinků na celou soustavu. Plánování trajektorie je parametrické a plánovač je možné použít pro sklad jakýchkoliv rozměrů a měnit vstupní veličiny, kterými jsou zejména velikost objektů, jimiž je manipulováno a maximální zrychlení ovlivňující časový průběh trajektorie.



7 SEZNAM ELEKTRONICKÝCH PŘÍLOH

Příloha 1 – „RobotickyManipulatorDoAutomatickehoSkladu.m“

Příloha 2 – „SimulacniModel_robot.m“

Příloha 3 – „Matice_desky_fcn.m“

Příloha 4 – „GrafickeZobrazeni_ManipulatorProAutomatickySklad.m“

Příloha 5 – „SimulacniModel_robot_carkovane.m“

Příloha 6 – „RobotickyManipulator_Simscape.slx“

Příloha 7 – „RobotickyManipulator_Simscape.slxc“

Příloha 8 – „README.txt“



8 BIBLIOGRAFIE

- [1] KURDILA, Andrew a Pinhas BEN-TZVI. *Dynamics and control of robotic systems*. Hoboken, NJ: Wiley, 2020. ISBN 978-1-119-52483-0.
- [2] SIMPSON, Colin D. *Introduction to robotics*. London: Logic Design Publishing, 2008. ISBN 978-0-9686860-2-7.
- [3] SKAŘUPA, Jiří. *Roboty a manipulátory I*. Ostrava: VŠB-Technická univerzita, 2002. ISBN 80-248-0044-6.
- [4] Industrial Robotics. In: *Engineer On A Disk* [online]. c1993-2013 [cit. 2022-05-06]. Dostupné z: <http://engineeronadisk.com/V3/engineeronadisk-173.html>
- [5] Robot Types. In: *Engineer On A Disk* [online]. c1993-2013 [cit. 2022-05-06]. Dostupné z: http://engineeronadisk.com/V2/book_integration/engineeronadisk-113.html
- [6] BERSCHIED, L. a T. KRÖGER. *Jerk-limited Real-time Trajectory Generation with Arbitrary Target States*. Německo: Karlsruhe Institut of Technology.
- [7] Robotic Manipulator. In: *ScienceDirect* [online]. 2022 [cit. 2022-05-06]. Dostupné z: <https://www.sciencedirect.com/topics/engineering/robotic-manipulator>
- [8] Stewart Platform. In: *GrabCAD* [online]. Estonsko: GrabCAD, c2009-2022 [cit. 2022-05-06]. Dostupné z: <https://grabcad.com/library/stewart-platform-6>
- [9] Mobile Robot Development Kit includes all the basics. In: *Direct Industry* [online]. Marseille: DirectIndustry, 2022 [cit. 2022-05-06]. Dostupné z: <https://trends.directindustry.com/project-18143.html>
- [10] *ResearchGate* [online]. Berlín: ResearchGate, c2008-2022 [cit. 2022-05-06]. Dostupné z: <https://www.researchgate.net/>
- [11] Why Does It Bend Like That?. In: *OpenDog* [online]. 2015 [cit. 2022-05-06]. Dostupné z: <https://opendog.wordpress.com/tag/workspace/>



- [12] TOBOLA, Přemysl. *Automatický sklad deskového materiálu*. Praha, 2021. Diplomová práce. České vysoké učení technické v Praze, Fakulta strojní.
- [13] MACFARLANE, Sonja. *On-Line Smooth Trajectory Planning for Manipulators*. New Brunswick, 1999. The University of New Brunswick, Department of Mechanical Engineering.
- [14] MathWorks Support. In: *MathWorks* [online]. Massachusetts: MathWorks, c1994-2022 [cit. 2022-05-06]. Dostupné z: https://www.mathworks.com/support.html?s_tid=gn_supp
- [15] Robotika od historie po současnost. In: *ElektroPrůmysl* [online]. ElektroPrůmysl, 2012 [cit. 2022-05-06]. Dostupné z: <https://www.elektroprumysl.cz/automatizace/robotika-od-historie-po-soucasnost>