

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

UČENIE SA PERIPERSONÁLNEHO PRIESTORU
HUMANOIDNÉHO ROBOTA
BAKALÁRSKA PRÁCA

2023

KAROLÍNA PISOŇOVÁ

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

UČENIE SA PERIPERSONÁLNEHO PRIESTORU
HUMANOIDNÉHO ROBOTY

BAKALÁRSKA PRÁCA

Študijný program: Informatika
Študijný odbor: Informatika
Školiace pracovisko: Katedra aplikovanej informatiky
Školiteľ: prof. Ing. Igor Farkaš, dr.

Bratislava, 2023
Karolína Pisoňová



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Karolína Pisoňová
Študijný program: informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Učenie sa peripersonálneho priestoru humanoidného robota
Learning the peripersonal space in a humanoid robot

Anotácia: Učenie sa peripersonálneho priestoru je jednou zo základných zložiek, ktoré sa vyvíjajú v dozrievajúcom stelesnenom agentovi (dieťať). To zahŕňa znalosti o dosiahnuteľnom priestore okolo tela, ktoré sú užitočné z viacerých dôvodov. Takáto schopnosť umožňuje agentovi (ktorým môže byť aj robot) robiť predikcie o jeho blízkom prostredí (pomocou motorických predstáv).

Cieľ:

1. Preštudujte si literatúru týkajúcu sa robotických modelov peripersonálneho priestoru.
2. Implementujte vybraný model na báze neurónovej siete, trénujte ho a otestujte pomocou simulovaného humanoidného robota NICO.
3. Analyzujte presnosť modelu a jeho vlastnosti.

Literatúra: Nguyen P. et al. (2020). Sensorimotor Representation Learning for an “Active Self” in Robots: A Model Survey. *KI - Künstliche Intelligenz*, 35,9–35
Rohmer E., Singh S., Freese M. (2013). Coppeliasim (formerly V-REP): a Versatile and Scalable Robot Simulation Framework. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1321–1326. doi:10.1109/IROS.2013.6696520.
Kerzel M. et al. (2017). NICO — Neuro-inspired companion: A developmental humanoid robot platform for multimodal interaction. In *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, doi:10.1109/ROMAN.2017.817228

Vedúci: prof. Ing. Igor Farkaš, Dr.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: prof. Ing. Igor Farkaš, Dr.
Dátum zadania: 20.10.2021

Dátum schválenia: 27.10.2021

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

PodĀkovanie: Týmto by som chcela poĀakovať môjmu vedúcemu bakalárskej práce prof. Ing. Igorovi Farkašovi, Dr. za uvedenie do oblasti neurónových sietí, ochotu a odborné rady pri písaní tejto práce.

Abstrakt

Dôležitú úlohu v robotike zohráva schopnosť humanoidných robotov spracovať podnety zo svojho peripersonálneho priestoru. Na nadobudnutie tejto schopnosti je pre humanoidných robotov kľúčové vytvorenie si vhodnej reprezentácie peripersonálneho priestoru. V tejto práci sa zameriavame na využitie umelých neurónových sietí pri vytváraní reprezentácie peripersonálneho priestoru humanoidného robota. Využívame tréning neurónových sietí na naučenie sa modelu doprednej kinematiky a inverzného modelu. S využitím týchto modelov a poznatkov o ich chybovosti nakoniec vyhodnocujeme schopnosť humanoidného robota odhadovať hranice svojho peripersonálneho priestoru.

Kľúčové slová: Humanoidný robot, peripersonálny priestor, dopredná a inverzná kinematika, umelé neurónové siete, nepriamo kontrolované učenie umelých neurónových sietí

Abstract

The ability of humanoid robots to process stimuli from their peripersonal space plays an important role in robotics. For humanoid robots to acquire this capability, it is crucial to develop an appropriate representation of the peripersonal space. In this work, we focus on the use of artificial neural networks in creating a representation of a humanoid robot's peripersonal space. We use neural network training to learn a forward kinematics model and an inverse model. Using these models and knowledge of their error rates, we finally evaluate the ability of a humanoid robot to estimate the boundaries of its peripersonal space.

Keywords: Humanoid robot, forward and inverse kinematics, artificial neural networks, distal teacher learning

Obsah

Úvod	1
1 Teoretické základy	3
1.1 Schéma tela	3
1.2 Peripersonálny priestor	4
1.2.1 Modely peripersonálneho priestoru	4
1.3 Humanoidný robot	6
2 Umelé neurónové siete	9
2.1 Princíp viacvrstvového perceptrónu	9
2.2 Učenie sa umelých neurónových sietí	10
2.2.1 Nekontrolované učenie	11
2.2.2 Kontrolované učenie	11
2.2.3 Nepriamo kontrolované učenie	12
3 Implementácia	15
3.1 Humanoidný robot NICO	15
3.2 Definícia problému	17
3.2.1 Zjednodušenie nepriamo kontrolovaného učenia	17
3.2.2 Použité označenia	18
3.3 Model doprednej kinematiky	18
3.3.1 Implementácia kontrolovaného učenia	18
3.4 Inverzný model	19
3.4.1 Implementácia nepriamo kontrolovaného učenia	19
4 Výsledky	23
4.1 Trénovanie modelu doprednej kinematiky	23
4.2 Trénovanie inverzného modelu	24
4.2.1 Trénovanie inverzného modelu použitím dosiahnutelných bodov	26
4.2.2 Trénovanie inverzného modelu s použitím aj nedosiahnutelných bodov	28

4.2.3	Porovnanie natrénovaných inverzných modelov	30
4.3	Odhadovanie dosiahnuteľnosti bodov	31
	Záver	35
	Literatúra	37

Zoznam obrázkov

1.1	Schéma dopredného a inverzného modelu	5
2.1	Viacvrstvový perceptrón s dvomi skrytými vrstvami: šípky reprezentujú orientované spojenia, ktoré sú ováňované	10
2.2	Schéma nepriamo kontrolovaného učenia: cieľové hodnoty sú známe pre nepriame premenné (výstupy), nie však pre priame premenné (akcie)	13
2.3	Systém nepriamo kontrolovaného učenia zložený z inverzného a dopredného modelu	14
3.1	Jednoduchá scéna s NICO v robotickom simulátore CoppeliaSim	16
3.2	Schéma zjednodušeného systému pri nepriamo kontrolovanom učení	17
3.3	Schéma modelu doprednej kinematiky	20
3.4	Schéma učenia sa inverzného modelu	21
4.1	Graf vývoju priemernej chyby so štandardnou odchýlkou pri tréovaní modelu doprednej kinematiky	24
4.2	Znázornenie testovacích chýb modelu doprednej kinematiky: pozícia [0, 0, 0] zodpovedá začiatočnej pozícii koncového efektoru	25
4.3	Distribúcia chýb modelu doprednej kinematiky	25
4.4	Graf vývoju priemernej chyby so štandardnou odchýlkou pri tréovaní inverzného modelu iba na dosiahnuteľných bodoch	27
4.5	Znázornenie testovacích chýb inverzného modelu tréovaného na dosiahnuteľných bodoch: pozícia [0, 0, 0] zodpovedá začiatočnej pozícii koncového efektoru	27
4.6	Distribúcia chýb inverzného modelu tréovaného na dosiahnuteľných bodoch	28
4.7	Graf vývoju priemernej chyby so štandardnou odchýlkou pri tréovaní inverzného modelu aj na nedosiahnuteľných bodoch	29
4.8	Znázornenie testovacích chýb inverzného modelu tréovaného aj na nedosiahnuteľných bodoch: pozícia [0, 0, 0] zodpovedá začiatočnej pozícii koncového efektoru	29

4.9	Distribúcia chýb inverzného modelu trénovaného aj na nedosiahnuteľných bodoch	30
4.10	Určenie dosiahnuteľnosti bodu pri perfektných modeloch	31
4.11	Pravidlo 1: Určenie dosiahnuteľnosti bodu iba na základe predikovanej performačnej chyby	32
4.12	Pravidlo 2: Určenie dosiahnuteľnosti bodu aj pomocou vystretosti ruky	32

Zoznam tabuliek

2.1	Prehľad používaného označenia	10
2.2	Prehľad chýb nepriamo kontrolovaného učenia	14
3.1	Reprezentácia používaných premenných	18
3.2	Parametre umelej neurónovej siete určenej na výpočet doprednej kinematiky pravej ruky robota NICO.	19
3.3	Parametre umelej neurónovej siete reprezentujúcej inverzný model pravej ruky robota NICO.	20
4.1	Porovnanie výsledkov testovania inverzných modelov	31

Úvod

Jedným z cieľov robotiky v dnešnej dobe je schopnosť robotov reagovať na okolité situácie podobným spôsobom ako človek, aby sme dosiahli bezpečnosť interakcie robotov s inými objektmi alebo ľuďmi. To znamená hlavne schopnosť vedieť reagovať na okolité objekty aj pri zmene prostredia a pri pohybe. Mohli by sme vytvoriť určitú sadu pravidiel, ktorými by sa mal robot riadiť v rôznych situáciách, pokryť však všetky možnosti by bolo veľmi náročne, priam až nemožné. Namiesto toho by sme teda radšej chceli, aby bol robot schopný učiť sa reagovať na rôzne situácie a následne bol schopný využívať nadobudnuté poznatky aj pri reagovaní na úplne nové situácie.

Inšpiráciu môžeme hľadať pri procese takéhoto učenia sa u ľudí. Ľudia taktiež hneď po narodení nemajú poznatky o tom, ako by mali reagovať na rôzne situácie a ako sa správať v rôznych situáciách. Novorodenec musí najskôr nadobudnúť základné poznatky o sebe a o svojich schopnostiach. Jedna z prvých úloh na nadobudnutie schopnosti interakcie s okolitým prostredím je učenie sa o sebe samom. Základom je poznanie vlastnej schémy tela a schopnosť spracúvať informácie zo svojho najbližšieho okolia - peripersonálneho priestoru.

V tejto práci sa zameriame hlavne na vytváranie reprezentácie peripersonálneho priestoru humanoidného robota NICO pomocou umelých neurónových sietí. Na získavanie poznatkov o jeho blízkom okolí budeme informácie získané vykonávaním náhodných pohybov v simulátore spracúvať pomocou umelej neurónovej siete. Cieľom je natrénovanie umelej neurónovej siete tak, aby sme na základe zadaného pohybu na vykonanie vedeli určiť výslednú pozíciu robota NICO. Následne príde na rad učenie sa predikovania akcie potrebnej na dosiahnutie požadovanej pozície. Nakoniec budeme skúmať schopnosť odhadnúť dosiahnuteľnosť objektov na základe doteraz naučených poznatkov, teda schopnosť rozoznať hranice svojho peripersonálneho priestoru.

Kapitola 1

Teoretické základy

V tejto kapitole sa bližšie oboznámime s niektorými dôležitými konceptmi pre túto prácu, ako napríklad schéma tela, peripersonálny priestor, alebo humanoidný robot. Využijeme hlavne poznatky z článku [1].

1.1 Schéma tela

Pojem schéma tela predstavuje senzomotorickú reprezentáciu stavby tela využívanú na plánovanie a vykonávanie pohybov tela [2]. Reprezentácia schémy tela sa u novorodencov vyvíja už vo veľmi skorom štádiu. Táto reprezentácia vzniká na základe dotykových a vizuálnych vnemov, ktoré novorodenec prijíma a propriopcepce.

Propriocepčia je zmysel, ktorý nám umožňuje uvedomovať si pozíciu tela a jeho pohyb v okolitom priestore. Toto nám umožňujú zmyslové orgány propriocepcie, ktorými sú proprioceptory - senzorické neuróny umiestnené vo svaloch, šľachách a kĺboch [3]. Propriocepčia zohráva dôležitú úlohu pri riadení pohybu tela a udržiavaní správnej polohy.

Na takzvanú kalibráciu schémy tela u novorodencov slúžia informácie, ktoré novorodenec získava pomocou sebapoznávania. Novorodenec získava lepšie a presnejšie informácie o vlastnom tele pomocou zrakových vnemov, ale aj hmatom, pomocou náhodných alebo cielených interakcií s vlastným telom.

Schéma tela je vytváraná pomocou dlhodobých informácií, ktoré sú pomerne stabilné aj v dlhšom čase, ako napríklad dĺžka končatiny, ale aj pomocou krátkodobých informácií, ktoré je potrebné neustále aktualizovať, ako napríklad uhly, v ktorých sú v danom momente končatiny zohnuté.

V tejto práci sa však nebudeme zameriavať na vytváranie reprezentácie schémy tela. Budeme uvažovať, že v ľubovoľnom momente je robot schopný podať informáciu o stave svojho tela, ako hlavne uhly jednotlivých kĺbov a relatívnu pozíciu svojich jednotlivých častí tela vzhľadom na jeho aktuálnu pozíciu.

1.2 Peripersonálny priestor

Peripersonálny priestor definuje oblasť bezprostredne obklopujúcu naše telo. Ak sa v tejto oblasti nachádzajú nejaké objekty, dokážeme ich dosiahnuť a manipulovať s nimi. Objekty mimo peripersonálneho priestoru nevieme dosiahnuť bez toho, že by sme sa pohli smerom k nim, prípadne oni ku nám [4]. Na to, aby sme vedeli interagovať s objektmi a prípadne inými ľuďmi v našom peripersonálnom priestore, potrebujeme mať vytvorenú reprezentáciu peripersonálneho priestoru, ktorá je schopná sa prispôbovať zmenám okolia.

Reprezentácia peripersonálneho priestoru vzniká z pohľadu prvej osoby. Dôležitú úlohu pri jej vytváraní má už spomínaná schopnosť propriocepcie. Reprezentácia peripersonálneho priestoru je rôzna pre jednotlivé časti tela, a tieto reprezentácie sa následne upravujú samostatne pri pohybe jednotlivých častí tela. Nás bude ďalej bližšie zaujímať reprezentácia peripersonálneho priestoru rúk robota. Vytvorená reprezentácia pomáha s manipuláciou objektov v peripersonálnom priestore, uľahčuje ich uchopenie, alebo aj pohyb so schopnosťou vyhýbať sa prekážkam v blízkom okolí. Je taktiež overené, že peripersonálny priestor zohráva kľúčovú úlohu pri obrannom správaní [5].

1.2.1 Modely peripersonálneho priestoru

Učenie sa peripersonálneho priestoru väčšinou zahŕňa identifikáciu a kalibrovanie kinematických modelov, čo sú matematické modely, ktoré popisujú pohyb jednotlivých častí tela. Tieto modely vznikajú hlavne na základe propriocepcie, alebo prípadne aj hmatových vnemov. Následne sa vytvárajú súvislosti medzi pohybmi robota a udalosťami v jeho peripersonálnom priestore, čo už zvyčajne vyžaduje aj zrakové vnemy.

Na vytváranie kinematických modelov je potrebné mať základné informácie o tele robota v podobe parametrizovaných funkcií ako napríklad dopredná kinematika, alebo inverzná kinematika.

Dopredná a inverzná kinematika

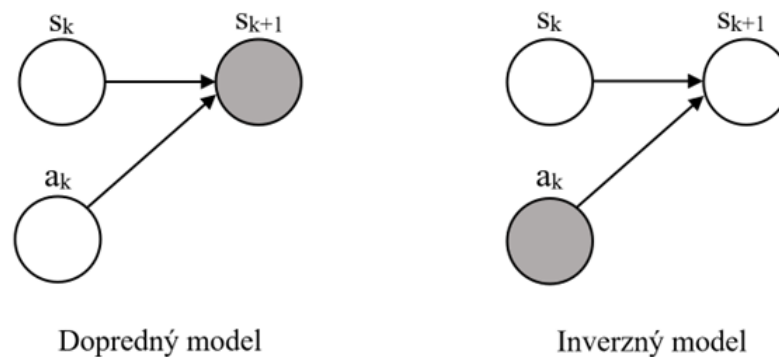
Dôležitým nástrojom na plánovanie, predpovedanie a kontrolovanie pohybov robota je dopredná kinematika. Úlohou doprednej kinematiky je na základe zadaných parametrov jednotlivých kĺbov určiť pozíciu koncového efektoru robota [6]. Parametrami jednotlivých kĺbov sú uhly, v ktorých sa aktuálne nachádzajú. Výsledkom doprednej kinematiky je teda mapovanie stavu jednotlivých kĺbov na výslednú pozíciu koncového efektoru v karteziánskom priestore.

Opačnú úlohu k doprednej kinematike má inverzná kinematika. Model inverznej kinematiky nám umožňuje určiť parametre jednotlivých kĺbov robota, ktoré sú potrebné na dosiahnutie konkrétnej pozície koncového efektoru [7].

Dopredný a inverzný model

Dopredný model je výpočtový model zameraný na úmyselnú motorickú kontrolu. Zdôrazňuje dôležitosť porovnania zámerov akcií a ich skutočných výstupov [8]. Cieľom dopredného modelu je predikcia nasledujúceho stavu na základe aktuálneho stavu a zadanej akcie. Schéma dopredného modelu je na obrázku 1.1. Jeho vstupmi sú aktuálny stav s_k a akcia a_k . Výstupom je stav s_{k+1} , ktorý nastane vykonaním akcie a_k .

Úlohou inverzného modelu je na základe súčasného stavu a žiadaného stavu predikovať akciu, ktorú je potrebné vykonať, aby sa súčasný stav zmenil na požadovaný stav. Schéma na obrázku 1.1 znázorňuje aj tento model, jeho vstupmi sú aktuálny stav s_k a požadovaný stav s_{k+1} . Výstupom je akcia a_k , ktorú je potrebné vykonať, aby sme dosiahli žiadanú zmenu stavu.



Obr. 1.1: Schéma dopredného a inverzného modelu

Inverzný model býva častokrát zložitejší ako dopredný model, pretože inverzný model nemusí byť vždy dobre definovaný, čo znamená, že pre konkrétny vstup môže existovať viacero rovnako dobrých výstupov [9]. V našom prípade budeme využívať inverzný model, ktorý bude predikovať potrebnú zmenu uhlov na to, aby sa koncový efektor dostal do požadovanej pozície. Inverzný model nie je dobre definovaný v prípade redundantného systému, kedy je počet stupňov voľnosti (v našom prípade päť) väčší ako dimenzia karteziánskeho priestoru (tri).

Normatívny model peripersonálneho priestoru

Pozrieme sa teraz bližšie na konkrétny model peripersonálneho priestoru, ktorým bude normatívny model peripersonálneho priestoru [10].

Úlohou tohto modelu je určiť vplyv pohybujúceho sa objektu na telo robota v čase $t + \Delta t$ vzhľadom na jeho pozíciu v čase t a na rýchlosť daného objektu.

Predpokladajme, že sa externý objekt priamočiaro pohybuje smerom k telu robota, alebo smerom od neho, so stálou rýchlosťou $v(t)$. Ďalej predpokladajme, že robot má schopnosť na základe zmyslových vnemov s určitou presnosťou predikovať pozíciu a

rýchlosť objektu v určitom čase.

V čase t má externý objekt pozíciu $x(t)$, ktorá predstavuje vzdialenosť od tela robota a pohybuje sa rýchlosťou $v(t)$ smerom k telu robota v prípade kladných hodnôt, smerom od tela robota v prípade záporných hodnôt. Predikovanú hodnotu pozície robotom budeme označovať ako $x'(t)$, pričom predpokladáme Gaussovskú distribúciu chýb a poznáme štandardnú odchýlku σ_x predikcie tejto hodnoty a robotom predikovanú hodnotu rýchlosti ako $v'(t)$, taktiež s predpokladom Gaussovskej distribúcie chýb so štandardnou odchýlkou predikcie σ_v .

Poznajúc štandardnú odchýlku predikovaných hodnôt budeme brať do úvahy aj jej hodnotu a namiesto samotných predikovaných hodnôt budeme uvažovať normálne distribúcie $N(x'(t), \sigma_x)$, $N(v'(t), \sigma_v)$.

Ďalším krokom je vypočítanie predpokladaného posunu objektu za časový rozdiel Δt . To môžeme zaznačiť nasledujúcim spôsobom

$$\Delta t = N(\Delta t \cdot v'_t, \Delta t \cdot \sigma_v). \quad (1.1)$$

Následne vieme teda predikovanú pozíciu $X'(t + \Delta t)$ vyjadriť ako

$$X'(t + \Delta t) = N(x'(t) + \Delta t \cdot v'(t), \sqrt{\sigma_x^2 + (\Delta t \cdot \sigma_v)^2}). \quad (1.2)$$

Teraz by sme mohli určiť pravdepodobnosť vplyvu objektu na robota ako

$$P(Y|(x'(t), \sigma_x), (v'(t), \sigma_v)) \quad (1.3)$$

kde $Y \in \{0, 1\}$ reprezentuje, či nastane kolízia objektu s telom ($y = 1$), alebo nie ($y = 0$). Pravidlo na určenie vplyvu objektu na robota môžeme následne vyjadriť ako

$$P(y = 1|(x'(t), \sigma_x), (v'(t), \sigma_v)) = P(X'(t + \Delta t) \leq 0). \quad (1.4)$$

Môžeme si všimnúť, že toto pravidlo zahŕňa prípady, keď kolízia nastane v ľubovoľnom okamihu medzi t a Δt .

1.3 Humanoidný robot

Väčšina robotov je aj v súčasnosti využívaná v priemysle, kde vykonávajú preddefinované úlohy v špecializovaných prostrediach, aby nedochádzalo k potenciálne škodlivým interakciám s ľuďmi. Humanoidní roboti, na rozdiel od toho, sú navrhnutí na to, aby vedeli reagovať v dynamickom prostredí a preto potrebujú byť schopní sa prispôbovať aj nezvyčajným prostrediam alebo situáciám, prípadne často majú schopnosť učiť sa z ľudských inštrukcií.

Humanoidný robot je budovaný tak, aby svojou konštrukciou pripomínal stavbu tela človeka a mal by mať sensorimotorické schopnosti čo najviac podobné tým ľudským. To

následne umožňuje testovanie neurálno-kognitívnych teórií na vývoj biologicky inšpirovaných výpočtových modelov. Na to, aby sa dala skúmať interakcia robota s prostredím je potrebné aby spĺňal aj ďalšie podmienky, napríklad by mal mať primeranú veľkosť tak, aby vedel interagovať s okolitým prostredím (napríklad uchopovanie a presúvanie objektov, zdvihnutie predmetu z nábytku štandardnej veľkosti a podobne), či zmyslové vnímanie podobné ľudskému - kamery na mieste očí s vysokým rozlíšením, či stereo mikrofóny umiestnené na oboch stranách hlavy [11].

Rozdelenie humanoidných robotov

V súčasnosti existuje veľké množstvo rôznych humanoidných robotov a prototypov, dajú sa však rozdeliť do troch základných kategórií. Prvou sú semi-humanoidné roboty, ktoré majú ľudskú hornú časť tela, a buď sa nevedia presúvať, alebo sú doplnené iba kolesami namiesto nôh. Ďalšou sú malé humanoidné roboty, ktoré majú veľkosť okolo 50 centimetrov a poslednou sú stredne veľké humanoidné roboty, ktoré majú veľkosť až do 150 centimetrov.

Kapitola 2

Umelé neurónové siete

Na vytvorenie modelu doprednej kinematiky, ale aj inverzného modelu budeme využívať umelé neurónové siete.

Umelé neurónové siete sú abstraktné výpočtové modely, ktoré sú zostavené na základe biologického nervového systému. Slúžia na spracúvanie vstupov a následné predikovanie výstupu podľa naučených pravidiel. V súčasnosti majú rôzne využitia, ako napríklad na rozpoznávanie tvárí a reči, klasifikáciu údajov, analýzu dát a mnohé iné.

Existuje viacero rôznych typov neurónových sietí, nás bude bližšie zaujímať viacvrstvový perceptrón. Viacvrstvový perceptrón sa využíva napríklad na klasifikáciu dát, ak je potrebné dáta rozdeliť do viacerých tried, rozpoznávanie vzorov, alebo aproximáciu funkcií. Okrem toho sa často využíva aj na regresné úlohy, kedy je cieľom predikcia kontinuálnych hodnôt, čo bude aj našou úlohou.

2.1 Princíp viacvrstvého perceptrónu

Viacvrstvový perceptrón pozostáva z troch hlavných častí, a to zo vstupnej vrstvy, jednej alebo viacerých skrytých vrstiev a výstupnej vrstvy. Každá z týchto vrstiev pozostáva z určitého počtu formálnych modelov neurónov, ktoré sú navzájom prepojené orientovanými spojeniami. Spojenie medzi neurónom j vo vrstve $m - 1$ a neurónom i vo vrstve m má priradenú číselnú váhu w_{ij}^m . Toto spojenie prenáša výstupnú hodnotu z neurónu j vo vrstve $m - 1$. Výstupnú hodnotu z neurónu j vo vrstve m označíme ako V_j^m . Výstupmi neurónov vstupnej vrstvy sú priamo jednotlivé vstupné hodnoty, teda ak si vstup označíme ako X , tak $V_i^0 = X_i$. Pre prehľadnosť je toto značenie uvedené v tabuľke 2.1.

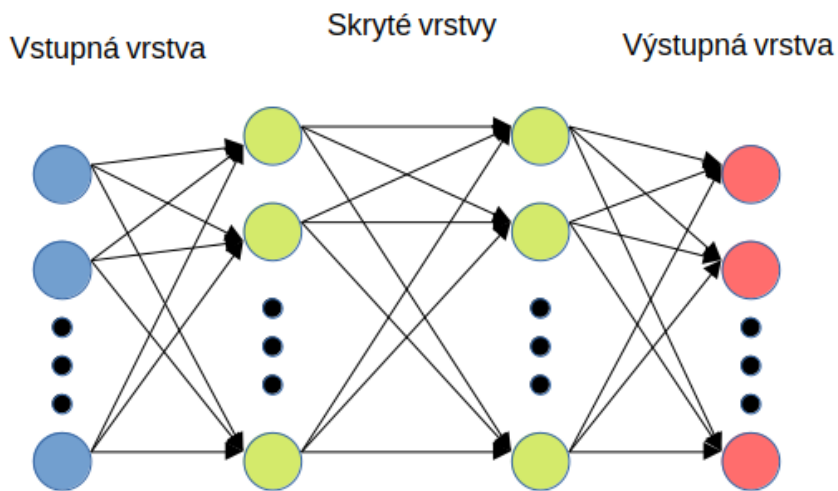
Každý neurón i skrytej alebo výstupnej vrstvy s číslom m následne vypočíta výstup na základe jeho vážených vstupných hodnôt podľa vzorca

$$V_i^m = g\left(\sum_j w_{ij}^m V_j^{m-1}\right) \quad (2.1)$$

Označenie	Význam
w_{ij}^m	Váha spojenia neurónu j z vrstvy $m - 1$ s neurónom i z vrstvy m
V_j^m	Výstupná hodnota neurónu j vo vrstve m
X_i	i -ta vstupná hodnota

Tabuľka 2.1: Prehľad používaného označenia

kde g je aktivačná funkcia. Typicky využívané aktivačné funkcie sú napríklad hyperbolický tangens alebo sigmoidná funkcia. Viacvrstvový perceptrón funguje postupným šírením signálu zo vstupu cez jednotlivé vrstvy až k výstupu. Jednoduchá schéma viacvrstvého perceptrónu je na obrázku 2.1



Obr. 2.1: Viacvrstvový perceptrón s dvomi skrytými vrstvami: šípky reprezentujú orientované spojenia, ktoré sú ováňované

2.2 Učenie sa umelých neurónových sietí

Učenie neurónových sietí prebieha prispôbovaním váh jednotlivých spojení neurónov tak, aby sme dosiahli čo najlepšie výsledky. Najbežnejšími metódami učenia neurónových sietí sú kontrolované učenie (z anglického označenia supervised learning), a nekontrolované učenie (z anglického výrazu unsupervised learning). Špeciálnym druhom učenia neurónových sietí je nepriamo kontrolované učenie (v angličtine používané ako distal teacher learning). Terminológia je prebratá z publikácie [12].

2.2.1 Nekonontrované učenie

Tento spôsob učenia sa využíva pri dátach, ktoré nie sú vopred klasifikované, teda k vstupným údajom nepoznáme priradenie správnych výstupov. Cieľom takéhoto učenia je odpozorovať a popísať vzory a štruktúry vo vstupných dátach s cieľom získať o nich ďalšie informácie. [13] Pri nekontrolovanom učení teda zvyčajne nemáme žiadnu konkrétnu výstupnú hodnotu, ktorú je potrebné predikovať.

Bežným využitím nekontrolovaného učenia sa je napríklad zhlukovanie, teda zhromažďovanie dát do rôznych skupín na základe určitých podobností. Môže byť využité aj na detekciu anomálií, kedy sa algoritmus naučí, čo je bežné pre určitú množinu dát, a následne zvláda detekovať anomálie v dátach. V súčasnosti má mnoho využití v rôznych oblastiach, hlavne v dátovej vede, počítačovom videní alebo v spracúvaní prirodzeného jazyka.

2.2.2 Kontrolované učenie

Kontrolované učenie je princíp strojového učenia, pri ktorom je prítomný istý učiteľ. To znamená, že je potrebná existencia dostatočného množstva trénovacích údajov, pričom k týmto údajom sú známe správne výstupy [12].

Pri kontrolovanom učení sa na určenie chyby výstupu umelej neurónovej siete v závislosti od správneho výstupu využívajú rôzne chybové funkcie. Voľba chybovej funkcie závisí od toho, aký problém práve riešime. Pri problémoch klasifikácie sa bežne využíva napríklad binárna alebo kategorickej krížová entropia. Pri regresných problémoch sa najčastejšie využíva absolútna chyba alebo stredná kvadratická chyba, ktorú budeme využívať aj my. Vzorec na výpočet strednej kvadratickej chyby je nasledujúci

$$MSE(Y', Y) = \frac{1}{n} \sum_{i=1}^n (Y_i - Y'_i)^2 \quad (2.2)$$

kde Y' predstavuje predikovaný vektor výstupných hodnôt, Y predstavuje správny vektor výstupných hodnôt a n je počet hodnôt výstupného vektora.

Bežnou metódou na následné šírenie chyby jednotlivými vrstvami a upravovanie váh je princíp spätnej propagácie, ktorý bol prvýkrát spomenutý už v roku 1986 [14].

Spätná propagácia chýb

Budeme používať podobné označenie ako v 2.1, teda číslo vrstvy, ktoré bude od 0 po M , bude reprezentovať premenná m , V_i^m bude predstavovať výstup i -teho neurónu v m -tej vrstve, V_i^0 bude i -ty vstupný údaj, O^k bude správny výstup ku k -temu vstupu a w_{ij}^m bude predstavovať váhu spojenia neurónu j z vrstvy $m - 1$ a neurónu i z vrstvy m .

Po doprednom prechode informácie neurónovou sieťou sa následne vypočíta chyba pomocou vybranej chybovej funkcie, ktorej vstupnými hodnotami sú správny výstup a

výstup predikovaný umelou neurónovou sieťou. Okrem toho je potrebné na vylepšenie modelu upraviť jednotlivé váhy, čo prebieha nasledovným spôsobom [15]. Ako prvé vypočítame adaptačné hodnoty δ_i^M pre výstupnú vrstvu

$$\delta_i^M = g'(\sum_j w_{ij}^M V_j^{M-1}) [O_i^k - V_i^M] \quad (2.3)$$

Potom pomocou spätnej propagácie chýb vypočítame adaptačné hodnoty aj pre predchádzajúce vrstvy

$$\delta_i^{m-1} = g'(\sum_j w_{ij}^{m-1} V_j^{m-2}) \sum_i w_{ji} \delta_j^m \quad (2.4)$$

Všetky váhy upravíme nasledovným spôsobom

$$\Delta w_{ij}^m = \alpha \delta_i^m V_j^{m-1} \quad (2.5)$$

$$w_{ij}^{m(t+1)} = w_{ij}^{m(t)} + \Delta w_{ij}^m \quad (2.6)$$

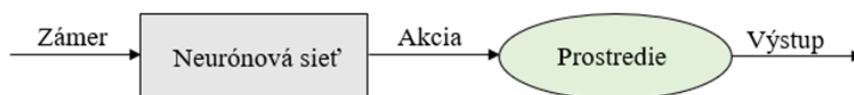
kde α je rýchlosť učenia (bežne používaný anglický názov je learning rate) a premennú t využívame na vyjadrenie časovej následnosti.

Algoritmus spätnej propagácie postupne upravuje váhy na dosiahnutie lepších výsledkov. Toto zlepšovanie prebieha až kým nedosiahneme dostatočne malú výstupnú chybu, ktorú počítame pomocou zvolenej chybovej funkcie, alebo prípadne, kým ešte dochádza k dostatočne výraznej zmene výstupnej chyby.

2.2.3 Nepriamo kontrolované učenie

Špeciálnym druhom učenia neurónových sietí je nepriamo kontrolované učenie (ang. distal teacher learning). Tento koncept bol uvedený v [16] v snahe rozšíriť oblasť použitia kontrolovaného učenia. Všeobecný scénar nepriamo kontrolovaného učenia je ukázaný na schéme 2.2. Zámer, ktorý chce byť dosiahnutý, je poskytnutý umelej neurónovej sieti ako vstup. Zámery sú neurónovou sieťou transformované na akcie, ktoré sú následne vykonané v prostredí, vďaka čomu získavame výstup. Akcie sú priame premenné, teda také, ktoré získavame priamo ako výstup neurónovej siete. Výstupy sú však nepriame premenné, ktoré neurónová sieť ovplyvňuje iba nepriamo, pretože na ich dosiahnutie je potrebné ešte vykonanie akcií v prostredí. Počas učenia sa sú prístupné cieľové hodnoty pre nepriame premenné (výstupy), nie pre priame, ako to funguje pri bežnom kontrolovanom učení. Úlohou neurónovej siete je teda nájsť mapovanie zo zámerov, ktoré chcú byť dosiahnuté na akcie, ktoré je na to potrebné vykonať. Keďže správna akcia na dosiahnutie požadovaného výstupu nie je známa, nevieme priamo určiť, ako chybná bola akcia predikovaná neurónovou sieťou. Vieme však zistiť chybu ako rozdiel medzi zámerom a celkovým výstupom a snažiť sa minimalizovať túto chybu.

Jednou zo všeobecných možností riešenia takéhoto problému je rozdelenie učenia sa modelu na dve časti.



Obr. 2.2: Schéma nepriamo kontrolovaného učenia: cieľové hodnoty sú známe pre nepriame premenné (výstupy), nie však pre priame premenné (akcie)

Prvou fázou v tomto postupe je natrénovanie vnútorného (dopredného) modelu, ktorého úlohou je naučenie sa mapovania akcií na ich výstupy. Možnosťou učenia sa vnútorného (dopredného) modelu je skúmanie rôznych akcií a ich výstupov, keďže zvyčajne nemáme dopredu presne dané toto mapovanie. Toto vie prebiehať klasickou metódou kontrolovaného učenia, keďže k akciám vieme zvyčajne jednoducho zistiť očakávaný výstup.

Ak už existuje aspoň čiastočne natrénovaný vnútorný model, je možné ho nepriamo využiť pri učení sa mapovania zo zámerov na akcie, ktoré je potrebné vykonať na dosiahnutie daných zámerov. Pomocou vnútorného modelu je následne možné vyhodnotiť chybu medzi výstupom vnútorného modelu na základe zvolenej akcie a pôvodným zámerom.

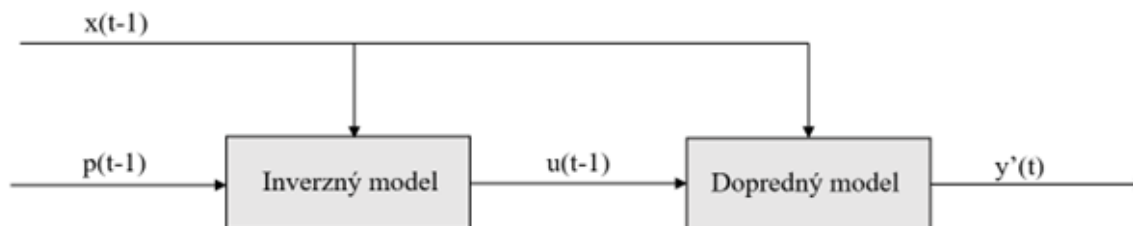
Formalizácia nepriamo kontrolovaného učenia sa

Využijeme označenia a definíciu problému z článku [16]. Úlohou inverzného modelu je na základe aktuálneho stavu $x(t-1)$ a zadaného cieľa $p(t-1)$ predikovať akciu $u(t-1)$ potrebnú na dosiahnutie cieľa $p(t-1)$. Výstup získaný vykonaním predikovanej akcie v prostredí označíme ako $y(t)$, výstup predikovaný dopredným modelom na základe akcie predikovanej inverzným modelom označíme $y'(t)$. Premenná t ako index vyjadruje časovú postupnosť.

Predpokladom na učenie dopredného modelu je možnosť pozorovať a vnímať stavy, akcie a výstupy. Na základe toho je možné dopredný model klasickým kontrolovaným učením naučiť predikovať výstupy $y'(t)$ na základe stavu $x(t-1)$ a akcie $u(t-1)$.

Predstavme si teraz systém, ktorý pozostáva z inverzného a vopred natrénovaného dopredného modelu, ako na obrázku 2.3. Tento systém bude fungovať nasledovným spôsobom. Inverzný model najskôr na základe stavu $x(t-1)$ a cieľovej pozície $p(t-1)$ predikuje akciu $u(t-1)$. Táto akcia následne slúži ako vstup pre dopredný model, ktorý na základe stavu $x(t-1)$ a akcie $u(t-1)$ predikuje výstup $y'(t)$. Skutočný výstup, ktorý by sme získali vykonaním akcie $u(t-1)$ v stave $x(t-1)$ označujeme ako $y(t)$. Pri možnosti použitia dokonalého dopredného modelu by následne mohlo učenie sa prebiehať na základe rozdielu medzi cieľovou pozíciou $p(t-1)$ a pozíciou prediko-

vanou dopredným modelom $y'(t)$, ktorá by teda bola zhodná so skutočnou hodnotou $y(t)$, ktorú by sme dosiahli vykonaním akcie $u(t - 1)$. Potrebovali by sme upraviť už len inverzný model, čiže pri učení sa musíme upraviť algoritmus spätnej propagácie chyby tak, aby zachovával dopredný model a upravoval už iba inverzný. V skutočnosti však väčšinou nemáme dokonalý dopredný model, ak však máme dostatočne kvalitný dopredný model, môžeme aj tak aplikovať tento spôsob, avšak musíme ďalej rátať s tým, že vznikne chyba spôsobená nepresnosťou dopredného modelu.



Obr. 2.3: Systém nepriamo kontrolovaného učenia zložený z inverzného a dopredného modelu

Druhy chýb pri nepriamo kontrolovanom učení

Chybu systému zloženého z inverzného a dopredného modelu môžeme vyhodnotiť rôznymi spôsobmi. Môžeme vyhodnotiť chybu medzi celkovým predikovaným výstupom $y'(t)$ a cieľovou hodnotou $p(t - 1)$, čo je predikovaná performačná chyba. Druhá možnosť je zistiť skutočný výsledok po vykonaní predikovanej akcie $u(t - 1)$, teda $y(t)$ a ten porovnať s cieľovou hodnotou $p(t - 1)$, čo je performačná chyba. Posledná možnosť je porovnať predikovaný výstup so skutočným výstupom, teda $y'(t)$ s $y(t)$, čo je chyba predikcie. Tieto tri druhy chýb postupne vyhodnocujú celkovú chybu systému, chybu inverzného modelu a chybu dopredného modelu. Pre prehľadnosť sú tieto chyby uvedené aj v tabuľke 2.2.

Názov chyby	Vzorec	Referovaný objekt
Predikovaná performačná chyba	$p(t - 1) - y'(t)$	Celý systém
Performačná chyba	$p(t - 1) - y(t)$	Inverzný model
Chyba predikcie	$y(t) - y'(t)$	Dopredný model

Tabuľka 2.2: Prehľad chýb nepriamo kontrolovaného učenia

Kapitola 3

Implementácia

Naším cieľom je skúmanie peripersonálneho priestoru humanoidného robota NICO, s ktorým budeme pracovať prostredníctvom robotického simulátora CoppeliaSim.

3.1 Humanoidný robot NICO

NICO [11] je humanoidný robot, ktorý síce nie je navrhnutý tak, aby bol podobný ľuďom v úplne každom smere, ale dostatočne na to, aby sa dala skúmať interakcia človeka a robota, a aby sa dali implementovať a skúmať neurálno-kognitívne modely. Keďže malý humanoidný robot nemá dostatočnú veľkosť na vykonávanie reálnych ľudských činností a existujúce stredne veľké modely robotov majú zvyčajne implementované zmyslové vnímanie, ktoré nie je dostatočne podobné ľudskému, tak bol NICO navrhnutý tak, aby mal veľkosť dospelujúceho dieťaťa (NICO je vysoký približne meter), vďaka čomu je dostatočne veľký aj na vykonávanie rôznych ľudských činností a je možné jednoducho implementovať zmyslové vnímanie podobné ľudskému.

Jeho hlavnými účelmi sú interakcia s ľuďmi, schopnosť prijímania podnetov podobne ako ľudia, alebo tiež uchopovanie objektov. Zrakové vnemy NICO prijíma pomocou dvoch paralelných kamier ako očí. Ako sluchové receptory mu slúžia mikrofóny umiestnené po oboch stranách hlavy, ktoré sú umiestnené v 3D modelovaných ušniciach, ktoré podobne ako u ľudí, umožňujú lokalizáciu zdroja zvuku a rozpoznávanie reči. Okrem toho ešte má tri LED pásy v priestore úst a nad očami, ktoré umožňujú zmenu výrazu tváre. Dobré uchopovanie predmetov mu umožňujú ruky so siedmimi stupňami voľnosti a dlane s tromi prstami. Dva ukazovacie prsty sú ovládané naraz a protistojný palec je kontrolovaný samostatne.

Existujú aj virtuálne prostredia, v ktorých je možné tréning robota NICO. Tieto je vhodné využívať na testovanie predtým, ako nejakú funkciu implementujeme priamo pre robota, pretože testovanie priamo na robotovi NICO by mohlo byť v niektorých prípadoch nebezpečné pre okolie, spôsobuje zbytočné opotrebovávanie mechanických

častí robota a vykonanie konkrétnej akcie trvá dlhšie, ako trvá výpočet toho, čo sa má udiat. Vďaka virtuálnym prostrediam teda vieme zaručiť vyššiu bezpečnosť, ušetriť čas a vyhnúť sa zbytočnému mechanickému opotrebovaniu. NICO má podporu pre robotický simulátor CoppeliaSim.

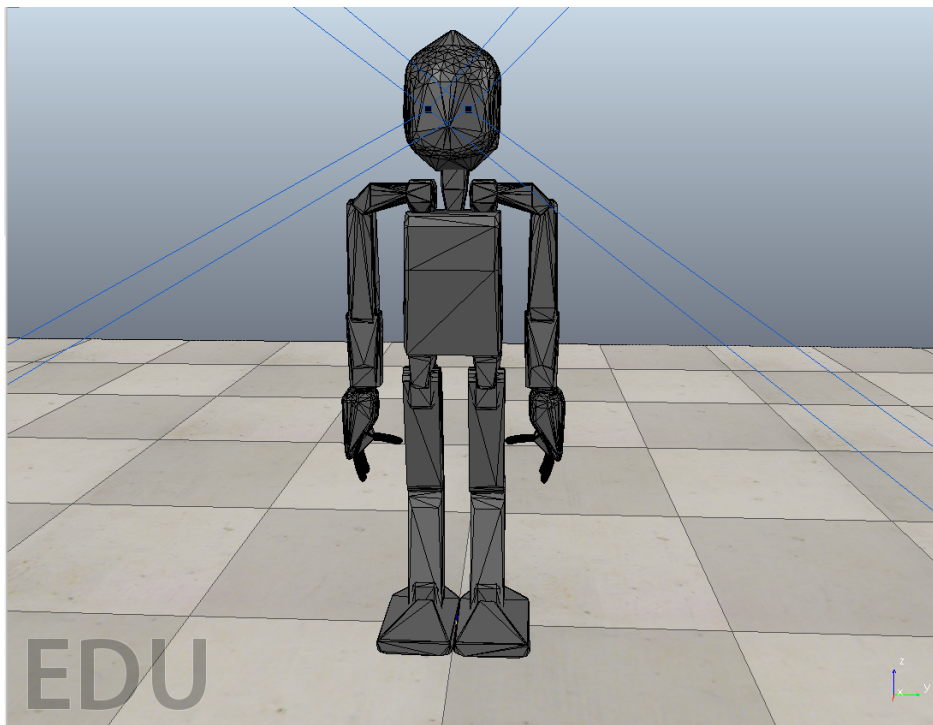
CoppeliaSim

CoppeliaSim je virtuálny robotický simulátor pre simuláciu rôznych programov určených pre robotov. Existujú rôzne možnosti pre vytvorenie simulácie, napríklad v podobe skriptov, alebo pluginov.

Simulácia je vytvorená scénou, ktorá môže obsahovať samotného simulovaného robota ale aj ďalšie objekty, ako napríklad rôzne útvary, senzory zraku, svetlá, cesty a podobne. Niektoré objekty môžu byť prepojené pomocou spojov alebo kĺbov, ktoré majú niekoľko stupňov voľnosti, v ktorých sa môžu nachádzať. Vo vytvorenej scéne môže byť simulovaný napríklad pohyb robota a jeho vplyv na okolité objekty.

V simulátore je možné pomocou jednoduchých Python skriptov napríklad ovládať pohyb robota nastavením konkrétnych uhlov, zistenie aktuálnych pozícií jednotlivých častí tela, pridávanie objektov do scény a manipulácia s nimi.

Ukážku robota NICO v robotickom simulátore CoppeliaSim vidno na obrázku 3.1



Obr. 3.1: Jednoduchá scéna s NICO v robotickom simulátore CoppeliaSim

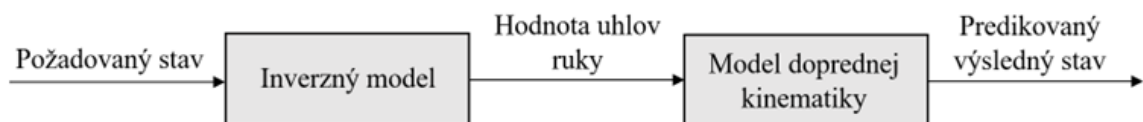
3.2 Definícia problému

Na skúmanie peripersonálneho priestoru a vytvorenie si predstavy o svojom okolí sme zvolili zjednodušený proces nepriamo kontrolovaného učenia.

3.2.1 Zjednodušenie nepriamo kontrolovaného učenia

Bežné nepriamo kontrolované učenie prebieha pomocou systému pozostávajúceho z inverzného a dopredného modelu, pričom tento systém prijíma ako vstupy požadovaný stav a aktuálny stav prostredia. Úlohou inverzného modelu je na základe súčasného stavu predikovať akciu, ktorú je potrebné vykonať na dosiahnutie požadovaného stavu. Úlohou dopredného modelu je na základe súčasného stavu a zadanej akcie predikovať nasledujúci stav.

Pre zjednodušenie inverzného modelu budeme vychádzať zakaždým zo začiatočného stavu, v ktorom sú všetky uhly ruky nulové. V takom prípade teda predikovaná akcia - zmena uhlov, bude zároveň reprezentovať reálnu hodnotu jednotlivých uhlov, ktorú chceme dosiahnuť. Napriek tejto úprave by sme stále vedeli jednoducho dopočítať potrebnú zmenu uhlov na základe aktuálneho stavu jednotlivých uhlov, ako rozdiel predikovaných uhlov a súčasného stavu uhlov. Vďaka tejto úprave následne vieme nahradiť vnútorný dopredný model modelom doprednej kinematiky. Zjednodušenie systému nepriamo kontrolovaného učenia je znázornené na obrázku 3.2.



Obr. 3.2: Schéma zjednodušeného systému pri nepriamo kontrolovanom učení

Učenie sa teraz môžeme rozdeliť na dve časti. Prvá časť je učenie sa doprednej kinematiky na základe náhodných pohybov ruky robota a získavanie informácií o polohe pomocou simulátora. Úlohou modelu doprednej kinematiky je vedieť na základe zadaných uhlov predikovať pozíciu koncového efektoru ruky.

Druhá časť je učenie sa inverzného modelu pomocou nepriamo kontrolovaného učenia, ktoré bude využívať naučený model doprednej kinematiky. Úlohou inverzného modelu zas bude na základe zadanej cieľovej pozície predikovať hodnoty uhlov ruky, potrebných na dosiahnutie tejto pozície.

3.2.2 Použité označenia

Vysvetlenie použitého označenia sa nachádza v tabuľke 3.1. Index t predstavuje časovú postupnosť.

Uhly kĺbov ruky	$u(t)$	Rozsah kĺbov
Cieľová pozícia koncového efektoru	$p(t - 1)$	Karteziánsky priestor
Pozícia koncového efektoru	$y(t)$	Karteziánsky priestor
Predikovaná pozícia koncového efektoru	$y'(t)$	Karteziánsky priestor

Tabuľka 3.1: Reprezentácia používaných premenných

3.3 Model doprednej kinematiky

Na implementáciu modelu doprednej kinematiky sme zvolili viacvrstvovú umelú neurónovú sieť. Učenie sa doprednej kinematiky pre pravú a ľavú ruku sú symetrické problémy, ktoré sa dajú riešiť samostatne. Bez ujmy na všeobecnosti sme si zvolili implementovať model doprednej kinematiky pre pravú ruku. Cieľový priestor sme obmedzili na polpriestor pred robotom v dosahu pravej ruky.

3.3.1 Implementácia kontrolovaného učenia

Na vytvorenie modelu doprednej kinematiky sme zvolili spôsob kontrolovaného učenia, keďže existuje jednoduchý spôsob, ktorým vieme na základe vstupu zistiť očakávaný výstup pomocou simulátora.

Vstup

Vstupom umelej neurónovej siete je päť uhlov pravej ruky (tri stupne voľnosti v ramene, jeden stupeň voľnosti v lakti a jeden stupeň voľnosti v zápästí). Využívaný model robota NICO má síce až 7 stupňov voľnosti v ruke, ale nevyužité dva stupne voľnosti ovplyvňujú pozíciu koncového efektoru iba zanedbateľne, preto neboli brané do úvahy.

Uhly zadávané v stupňoch boli následne preškálované do rozpätia $[0, 1]$. Možné rozpätie jednotlivých uhlov bolo zvolené na základe ich skutočného rozpätia.

Parametre

Aktivačná funkcia skrytých vrstiev je hyperbolický tangens v rozpätí $[-1, 1]$. Výstupná funkcia je sigmoid s rozpätím $[0, 1]$. Skutočné súradnice pozície v priestore získame až následným preškálovaním. Zvyšné parametre sú uvedené v tabuľke 3.2

Parameter	Hodnota
Počet skrytých vrstiev	6
Počet neurónov skrytej vrstvy	200
Rýchlosť učenia	0.001
Aktivačná funkcia skrytých vrstiev	Hyperbolický tangens
Aktivačná funkcia výstupnej vrstvy	Sigmoid
Chybová funkcia	Stredná kvadratická chyba

Tabuľka 3.2: Parametre umelej neurónovej siete určenej na výpočet doprednej kinematiky pravej ruky robota NICO.

Výstup

Výstupom umelej neurónovej siete sú predikované karteziánske súradnice pozície koncového efektoru pravej ruky, za ktorý sme si zvolili koniec tretieho prsta.

Učenie sa

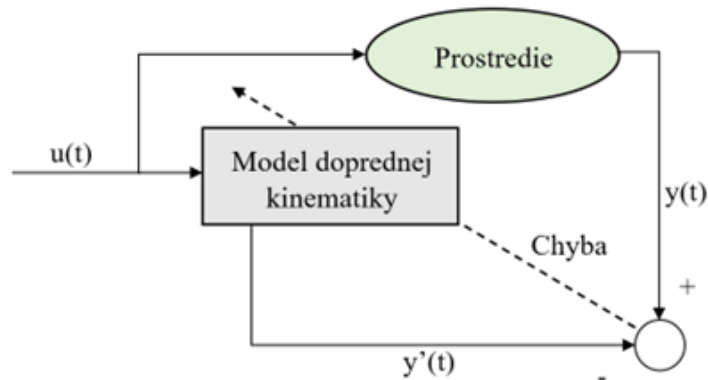
Mapovanie z konkrétneho stavu uhlov ruky na očakávanú pozíciu koncového efektoru sa dá naučiť pomocou kontrolovaného učenia. Modelu doprednej kinematiky sú poskytované páry uhlov ruky $u(t)$ a k nim prislúchajúca pozícia koncového efektoru $y(t)$. Uhly ruky sú generované náhodne z určitého rozsahu a pozícia koncového efektoru je získaná zo simulátora vykonaním potrebného pohybu. Pár je považovaný za validný, pokiaľ pozícia koncového efektoru leží v rovine pred robotom. Uhly sú poskytnuté neurónovej sieti, ktorej výstupom je predikovaná pozícia koncového efektoru $y'(t)$. Rozdiel medzi skutočnou pozíciou koncového efektoru $y(t)$ a predikovanou pozíciou $y'(t)$ vieme jednoducho vypočítať a vieme ho použiť ako chybu pre klasický proces spätnej propagácie chyby na učenie sa siete. Znázornenie schémy učenia sa doprednej kinematiky je na obrázku 3.3.

3.4 Inverzný model

Inverzný model bude opäť reprezentovaný pomocou viacvrstvovej umelej neurónovej siete. Podobne ako dopredná kinematika je to symetrický problém pre pravú a ľavú ruku. Keďže máme implementovaný model doprednej kinematiky pre pravú ruku, tak budeme implementovať aj inverzný model pre pravú ruku.

3.4.1 Implementácia nepriamo kontrolovaného učenia

Na vytvorenie inverzného modelu využijeme zjednodušený princíp nepriamo kontrolovaného učenia, pričom budeme využívať naučený model doprednej kinematiky.



Obr. 3.3: Schéma modelu doprednej kinematiky

Vstup

Vstupom pre umelú neurónovú sieť sú v tomto prípade karteziánske súradnice pozície koncového efektoru, ktorú chceme dosiahnuť. Zadávané polohy sú taktiež z polpriestoru pred robotom, avšak nie nutne v dosahu pravej ruky. Súradnice sú taktiež preškáľované do rozpätia $[0, 1]$.

Parametre

Aktivačná funkcia skrytých vrstiev je opäť hyperbolický tangens v rozpätí $[-1, 1]$. Výstupná funkcia je sigmoid s rozpätím $[0, 1]$. Skutočné uhly ruky, ktoré sú potrebné na dosiahnutie požadovanej pozície získame teda až následným preškáľovaním. Zvyšné parametre sú uvedené v tabuľke 3.3.

Parameter	Hodnota
Počet skrytých vrstiev	10
Počet neurónov skrytej vrstvy	200
Rýchlosť učenia	0.001
Aktivačná funkcia skrytých vrstiev	Hyperbolický tangens
Aktivačná funkcia výstupnej vrstvy	Sigmoid
Chybová funkcia	Stredná kvadratická chyba

Tabuľka 3.3: Parametre umelej neurónovej siete reprezentujúcej inverzný model pravej ruky robota NICO.

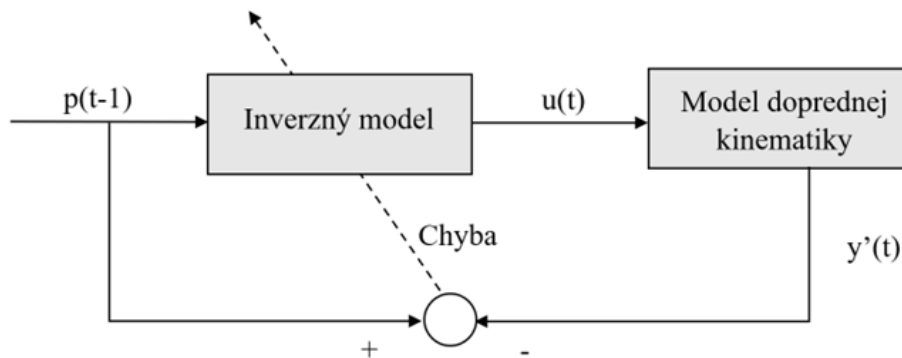
Výstup

Výstupom umelej neurónovej siete je predikovaná päťka uhlov ruky, potrebných na dosiahnutie požadovanej pozície koncového efektoru.

Učenie sa

Ako sme spomínali, pri našom inverznom modeli neexistuje jednoznačné priradenie uhlov ruky ku pozíciám, ktoré chceme dosiahnuť, preto v tomto prípade nevieme využiť klasické kontrolované učenie. Vieme však využiť nepriamo kontrolované učenie. Inverznému modelu sú zadávané náhodné cieľové pozície $p(t - 1)$. Uhly ruky predikované inverzným modelom $u(t)$ sú následne použité ako vstup modelu doprednej kinematiky, ktorého výstupom je predikovaná pozícia $y'(t)$. Následne vieme určiť chybu ako rozdiel pozície predikovanej modelom doprednej kinematiky a cieľovej pozície. Túto chybu vieme ďalej využiť v procese spätného šírenia chyby. Schéma učenia sa inverzného modelu je znázornená na obrázku 3.4.

Úlohou inverzného modelu je aj v prípade nedosiahnuteľného bodu vedieť určiť uhly ruky tak, aby bol koncový efektor čo najbližšie k cieľovej pozícii. Vzhľadom na tento cieľ sme implementovali dva inverzné modely. Jeden model bol tréňovaný iba na dosiahnuteľných bodoch, druhý bol tréňovaný aj na nedosiahnuteľných bodoch.



Obr. 3.4: Schéma učenia sa inverzného modelu

Kapitola 4

Výsledky

V tejto kapitole sa pozrieme na výsledky tréningovania jednotlivých modelov. Konkrétne sa pozrieme na výsledky tréningovania modelu doprednej kinematiky, inverzného modelu iba na dosiahnuteľných bodoch a inverzného modelu aj na nedosiahnuteľných bodoch a na porovnanie ich výsledkov. Nakoniec sa pozrieme na to, ako by sa naučené modely dali využiť pri odhadovaní hraníc peripersonálneho priestoru.

4.1 Tréningovanie modelu doprednej kinematiky

Prvým krokom je tréningovanie modelu doprednej kinematiky. Ako sme spomínali v kapitole 3, v tomto prípade využívame kontrolované učenie a prispôsobovanie váh jednotlivých spojení prebieha pomocou klasickej spätnej propagácie chýb. Vstupné údaje boli získavané zo simulátora náhodnými pohybmi ruky.

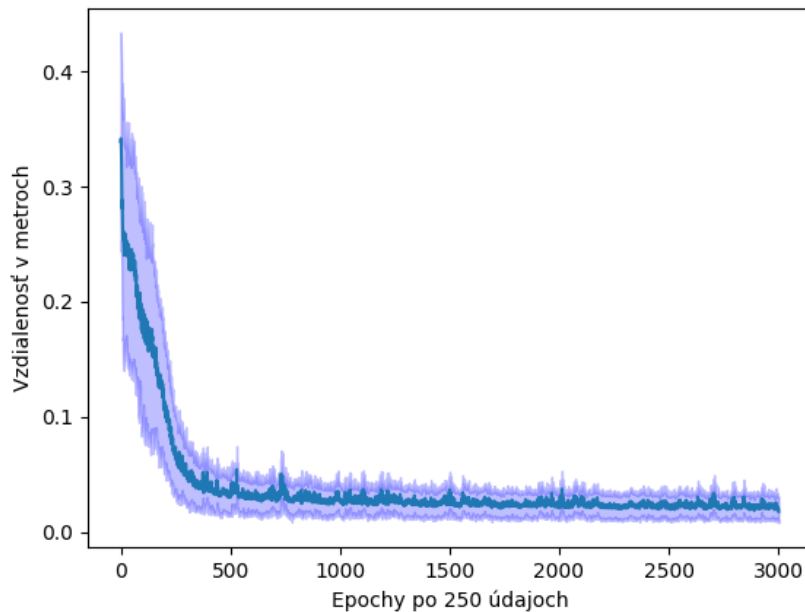
Nepresnosť údajov

Pri prvých pokusoch tréningovania vychádzala priemerná chyba približne 6 centimetrov, pričom dĺžka ramena robota je približne 50 centimetrov. Pri následnom testovaní modelu doprednej kinematiky sa ukázalo, že väčšina chýb je pomerne nízka, ale existujú výrazné odchýlky v nezanedbateľnom množstve dát. Priestorový vzor v údajoch s výraznou odchýlkou nebol jasne viditeľný. Toto viedlo k bližšiemu skúmaniu dát s vysokou odchýlkou, a po opätovnej simulácii so zadaním daných uhlov sa ukázalo, že mnohé z údajov s vysokou odchýlkou boli nepresné. Nepresnosť údajov zo simulátora vzniká kvôli nestabilite robota počas pohybu. V niektorých prípadoch aj pri zadaní uhlov ruky z povolených rozsahov jednotlivých kĺbov došlo k problému vzhľadom na kombináciu jednotlivých uhlov. V takých prípadoch došlo ku kolízií robota so samým sebou, alebo úplnej strate stability, čo ovplyvnilo výslednú pozíciu v niektorých prípadoch iba mierne, avšak v niektorých prípadoch bola táto nepresnosť veľmi výrazná. Aj napriek snahe kontroly zbieraných údajov nebolo možné vždy odhaliť vznik takejto nepresnosti

bez vizuálnej kontroly vykonávaného pohybu. Na dodatočnú kontrolu dát sme využili natrénovaný dopredný model a opätovné simulácie dát s najväčšími odchýlkami, vďaka čomu sa podarilo vyradiť alebo nahradiť mnohé nepresné údaje.

Výsledky tréovania

Následné tréovanie po čiastočnej kontrole údajov prinieslo o viac ako 50% lepšie výsledky s priemernou testovacou chybou 1.89 centimetra a štandardnou odchýlkou 1.03 centimetra. Tréovanie prebiehalo v epochách po 250 tréovacích údajoch. Vývoj chyby v priebehu tréovania možno vidieť na grafe 4.1. Možno vidieť rýchle učenie sa v prvých 500 epochách, s minimálnym zlepšovaním sa v ďalších epochách.

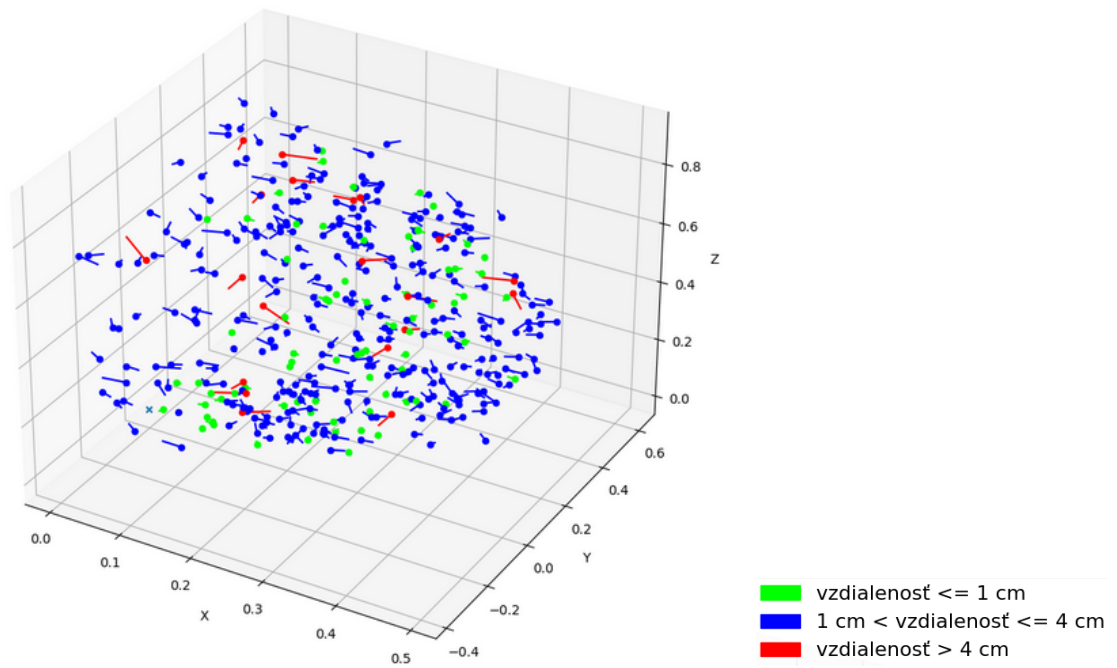


Obr. 4.1: Graf vývoju priemernej chyby so štandardnou odchýlkou pri tréovaní modelu doprednej kinematiky

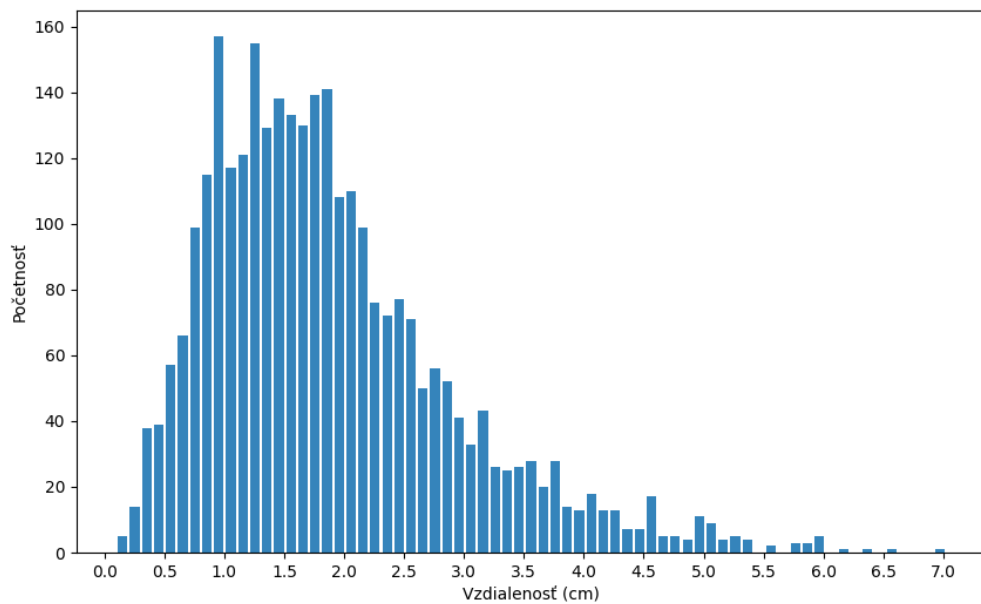
Grafické spracovanie chýb pri testovaní náhodne vybraných údajov je znázornené na obrázku 4.2. Na grafe 4.3 zas môžeme vidieť rozloženie chýb pri testovaní podľa ich veľkostí.

4.2 Tréovanie inverzného modelu

Pri tréovaní inverzného modelu sme využívali natrénovaný model doprednej kinematiky a tréovanie prebiehalo metódou nepriamo kontrolovaného učenia. Upravovanie váh jednotlivých spojení prebiehalo pomocou upraveného princípu spätnej propagácie chýb, ktorý zachováva dopredný model v rovnakom stave a upravuje iba váhy inverzného modelu. Testovanie oboch modelov prebiehalo v dvoch fázach.



Obr. 4.2: Znázornenie testovacích chýb modelu doprednej kinematiky: pozícia $[0, 0, 0]$ zodpovedá začiatkovej pozícii koncového efektoru



Obr. 4.3: Distribúcia chýb modelu doprednej kinematiky

V prvej fáze boli modely testované na dosiahnuteľných cieľoch, pričom vyhodnocovaná bola predikovaná performačná chyba, performačná chyba a chyba predikcie.

V druhej fáze boli modely testované na menšej sade nedosiahnuteľných cieľov. Podobne ako v prvej fáze sme vyhodnocovali predikovanú performačnú chybu, performačnú chybu a chybu predikcie. Vyhodnocovaný bol aj uhol medzi priamkou, ktorá vznikla predĺžením ruky robota a priamkou na ktorej sa nachádza rameno robota a cieľová pozícia. Tento uhol reprezentuje, či rameno robota smeruje ku vzdialenému cieľu. Okrem toho však dôležitým faktorom pri určovaní presnosti modelu na nedosiahnuteľných cieľoch je aj vystretosť ruky. Pri tomto testovaní je síce zjavné, že robot nedosiahne na zvolené ciele, ale stále je jeho úlohou, aby sa dostal do pozície, ktorá je čo najmenej vzdialená od cieľového bodu. Bez ohľadu na konkrétnu pozíciu neodsiahnuteľného cieľa vieme s istotou povedať, že na dosiahnutie ideálnej pozície je potrebné, aby bola ruka úplne vystretá, čo predstavuje uhol 0 stupňov v lakti a v zápästí. Preto sme okrem spomínaných chýb v tejto fáze vyhodnocovali aj ohnutosť ruky v lakti a v zápästí.

4.2.1 Trénovanie inverzného modelu použitím dosiahnuteľných bodov

Trénovanie prebiehalo v epochách po 250 tréningových údajoch. Vývoj predikovanej performačnej chyby v priebehu tréningovania možno vidieť na grafe 4.4. Podobne ako pri modeli doprednej kinematiky vidno rýchly pokles chyby hlavne v prvých 500 epochách.

Testovanie na dosiahnuteľných bodoch

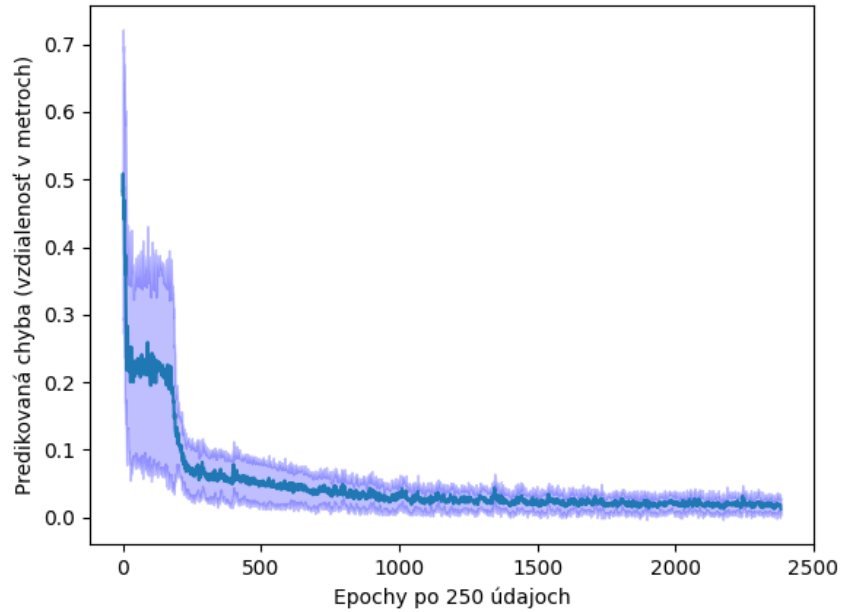
Priemerná predikovaná performačná chyba pri testovaní bola 1.39 centimetra so štandardnou odchýlkou 1.59 centimetra. Priemerná performačná chyba inverzného modelu bola 2.61 centimetrov so štandardnou odchýlkou 2.02 centimetrov a priemerná chyba predikcie bola 2.42 centimetrov so štandardnou odchýlkou 2.81 centimetrov.

Grafické spracovanie predikovaných performačných chýb pri testovaní náhodne vybraných údajov možno vidieť na obrázku 4.5. Na grafe 4.6 je zas znázornené rozloženie predikovaných performačných chýb pri testovaní podľa ich veľkosti.

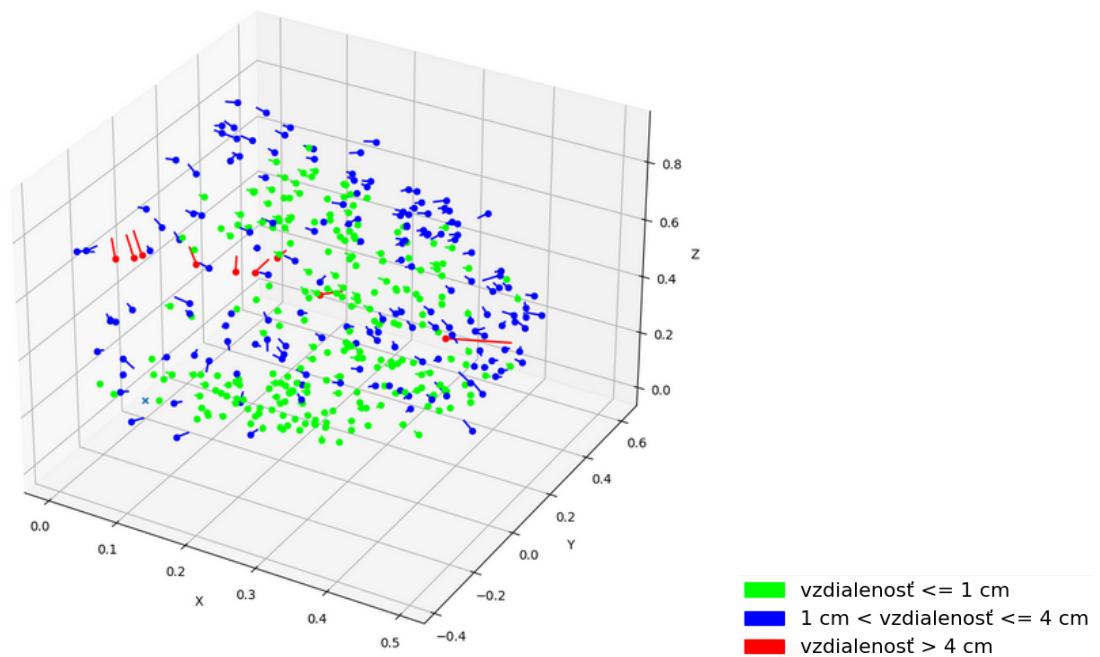
Testovanie na nedosiahnuteľných bodoch

Pri testovaní na nedosiahnuteľných bodoch sú očakávané vyššie chyby, keďže aj ideálne pozície sú v určitej vzdialenosti od požadovaného cieľa. Veľkosť týchto chýb závisí práve od zvolených cieľov a samostatne z nich nevieme určiť úspešnosť modelu, uvedieme ich ale pre možnosť porovnania presnosti inverzných modelov.

Priemerná predikovaná performačná chyba pri testovaní bola 36.75 centimetrov so štandardnou odchýlkou 20.33 centimetrov. Priemerná performačná chyba inverzného



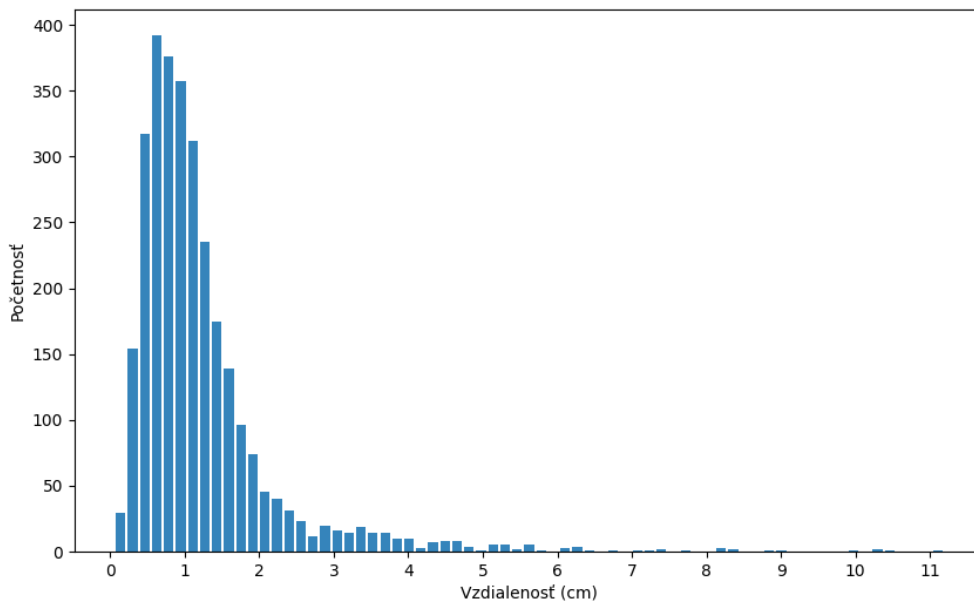
Obr. 4.4: Graf vývoju priemernej chyby so štandardnou odchýlkou pri tréovaní inverzného modelu iba na dosiahnuteľných bodoch



Obr. 4.5: Znázornenie testovacích chýb inverzného modelu tréovaného na dosiahnuteľných bodoch: pozícia $[0, 0, 0]$ zodpovedá začiatkovej pozícii koncového efektoru

modelu bola 36.49 centimetrov so štandardnou odchýlkou 20.23 centimetrov a priemerná chyba predikcie bola 2.27 centimetrov so štandardnou odchýlkou 1.53 centimetrov.

Priemerná odchýlka smerovania ruky robota od cieľového bodu v uhloch bola 9.5



Obr. 4.6: Distribúcia chýb inverzného modelu trénovaného na dosiahnuteľných bodoch

stupňov so štandardnou odchýlkou 6 stupňov.

Čo sa týka vystretosti ruky, priemerný uhol ohnutia v lakti bol 18.5 stupňov so štandardnou odchýlkou 9.6 stupňov, a priemerný uhol ohnutia v zápästí bol 9 stupňov so štandardnou odchýlkou 12.3 stupňov.

4.2.2 Trénovanie inverzného modelu s použitím aj nedosiahnuteľných bodov

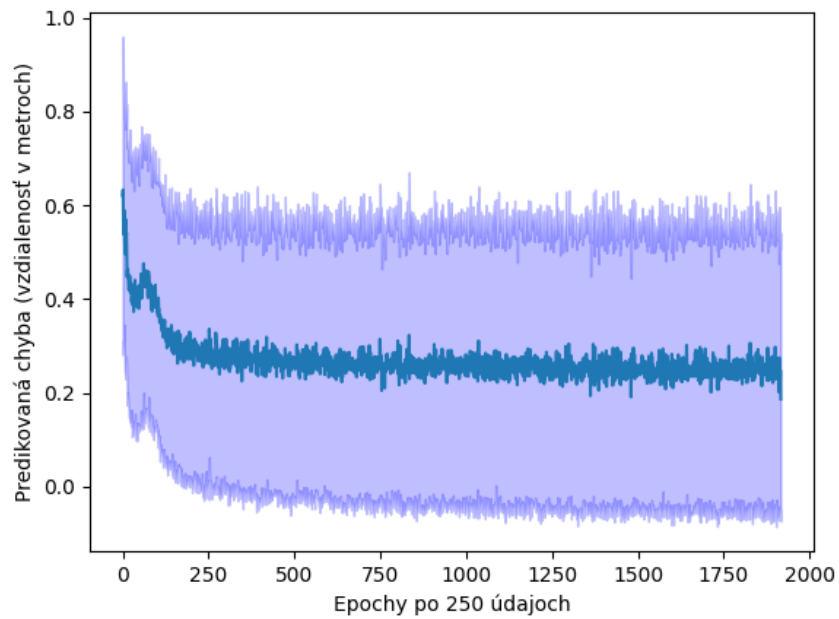
Trénovanie prebiehalo v epochách po 250 trénovacích údajoch. Vývoj predikovanej performačnej chyby v priebehu trénovania možno vidieť na grafe 4.7. Tu možno vidno pomerne vysokú chybu počas celého procesu trénovania, čo je spôsobené práve nedosiahnuteľnými bodmi, pri ktorých nevie chyba klesnúť nižšie, ako je ich reálna vzdialenosť od ideálnej pozície koncového efektoru.

Testovanie na dosiahnuteľných bodoch

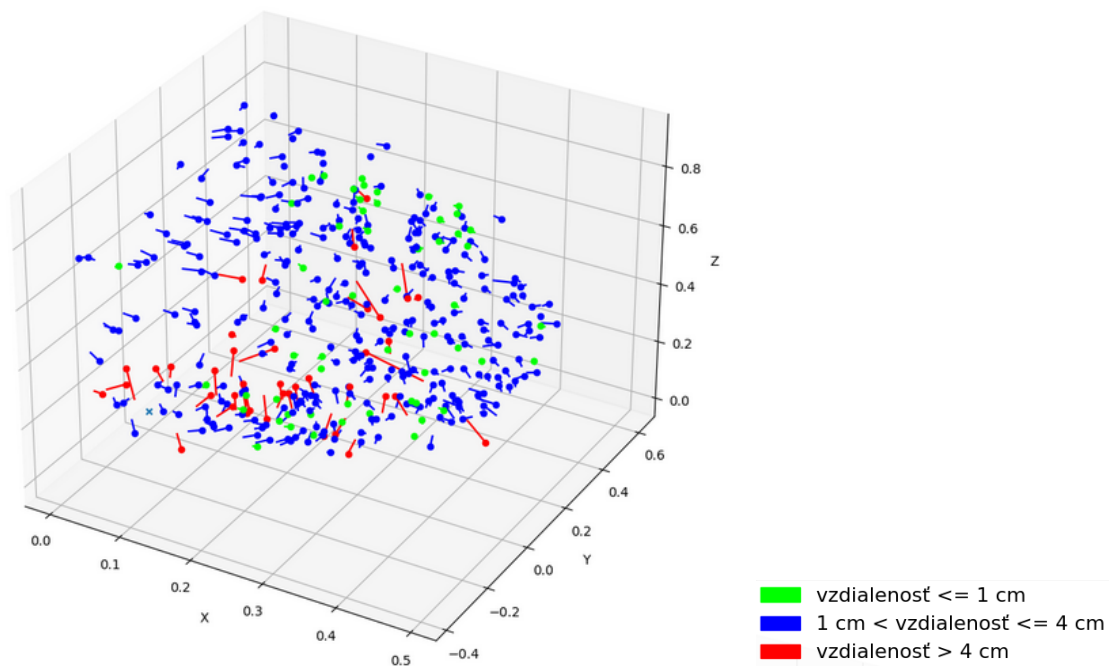
Priemerná predikovaná performačná chyba pri testovaní bola 2.21 centimetrov so štandardnou odchýlkou 1.61 centimetrov. Priemerná performačná chyba inverzného modelu bola 3.26 centimetrov so štandardnou odchýlkou 2.48 centimetrov a priemerná chyba predikcie bola 2.43 centimetrov so štandardnou odchýlkou 2.47 centimetrov.

Grafické spracovanie predikovaných performačných chýb pri testovaní náhodne vybraných údajov možno vidieť na obrázku 4.8. Na grafe 4.9 zas možno vidno rozloženie

predikovaných perforamčných chýb pri testovaní podľa ich veľkosti.



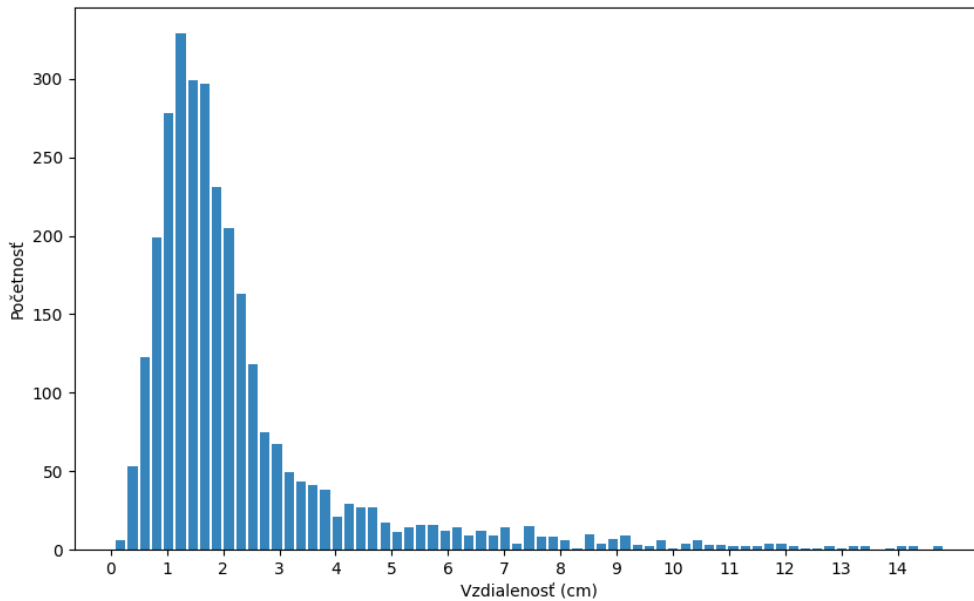
Obr. 4.7: Graf vývoju priemernej chyby so štandardnou odchýlkou pri tréovaní inverzného modelu aj na nedosiahnuteľných bodoch



Obr. 4.8: Znázornenie testovacích chýb inverzného modelu tréovaného aj na nedosiahnuteľných bodoch: pozícia $[0, 0, 0]$ zodpovedá začiatkovej pozícii koncového efektoru

Testovanie na nedosiahnuteľných bodoch

Priemerná predikovaná perforačná chyba pri testovaní bola 36.74 centimetrov so štan-



Obr. 4.9: Distribúcia chýb inverzného modelu trénovaného aj na nedosiahnuteľných bodoch

dardnou odchýlkou 20.92 centimetrov. Priemerná performačná chyba inverzného modelu bola 37.13 centimetrov so štandardnou odchýlkou 20.96 centimetrov a priemerná chyba predikcie bola 2.13 centimetrov so štandardnou odchýlkou 1.13 centimetrov.

Priemerná odchýlka smerovania ruky robota od cieľového bodu v uhloch bola 11.9 stupňov so štandardnou odchýlkou 7.4 stupňov.

Čo sa týka vystretosti ruky, priemerný uhol ohnutia v lakti bol 7.6 stupňov so štandardnou odchýlkou 6.9 stupňov, a priemerný uhol ohnutia v zápästí bol 3 stupne so štandardnou odchýlkou 6.7 stupňov.

4.2.3 Porovnanie natrénovaných inverzných modelov

Prehľad údajov testovania inverzných modelov je uvedený v tabuľke 4.1, pričom model *A* označuje inverzný model, ktorý bol trénovaný iba na dosiahnuteľných bodoch a model *B* označuje inverzný model, ktorý bol trénovaný aj na nedosiahnuteľných bodoch. Vzďialenosti v 4.1 sú udávané v centimetroch, uhly v stupňoch.

Môžeme si všimnúť, že pri oboch inverzných modeloch vo väčšine prípadov platí, že performačná chyba je mierne vyššia ako predikovaná performačná chyba. Toto je spôsobené nepresnosťou využívaného modelu doprednej kinematiky pri tréovaní.

Pridaním vzdialených bodov do tréovacej množiny inverzného modelu *B* sa oproti modelu *A* mierne zvýšila nepresnosť pri predikovaní uhlov pre dosiahnuteľné body,

Vyhodnocovaná chyba	A	B
Predikovaná performačná chyba (dosiahnuteľné)	1.39 ± 1.59	2.21 ± 1.61
Performačná chyba (dosiahnuteľné)	2.61 ± 2.02	3.26 ± 2.48
Chyba predikcie (dosiahnuteľné)	2.42 ± 2.81	2.43 ± 2.47
Predikovaná performačná chyba (nedosiahnuteľné)	36.75 ± 20.33	36.74 ± 20.92
Performačná chyba (nedosiahnuteľné)	36.49 ± 20.23	37.13 ± 20.96
Chyba predikcie (nedosiahnuteľné)	2.27 ± 1.53	2.13 ± 1.13
Odchýlka smerovania ruky (nedosiahnuteľné)	9.5 ± 6	11.9 ± 7.4
Uhol v lakti (nedosiahnuteľné)	18.5 ± 9.6	7.6 ± 6.9
Uhol v zápästí (nedosiahnuteľné)	9 ± 12.3	3 ± 6.7

Tabuľka 4.1: Porovnanie výsledkov testovania inverzných modelov

taktiež sa mierne zhoršil uhol smerovania ruky od cieľovej pozície pri nedosiahnuteľných cieľoch, avšak výrazne sa zlepšila schopnosť vystierania ruky za nedosiahnuteľnými cieľmi.

Model *A* trénovaný iba na dosiahnuteľných bodoch teda dosahuje lepšie výsledky pri dosiahnuteľných bodoch, zatiaľ čo model *B* trénovaný aj na nedosiahnuteľných bodoch preukazuje lepšie výsledky pri nedosiahnuteľných bodoch.

4.3 Odhadovanie dosiahnuteľnosti bodov

Na odhadovanie dosiahnuteľnosti bodov môžeme využiť natrénované modely s využitím poznatkov o ich chybovosti.

Pri predstave perfektne naučených modelov, ktoré by vždy predikovali ideálne akcie, by bolo vyhodnotenie dosiahnuteľnosti veľmi jednoduché. Stačilo by sa pozrieť na to, či je predikovaná performačná chyba ($PPE(p)$) rôzna od nuly, pretože v tom prípade by to skutočne muselo znamenať, že daný cieľ je nedosiahnuteľný.

```

1 def dosiahnuteľný_bod(PPE(p)):
2     if PPE(p) == 0:
3         return True
4     else:
5         return False

```

Obr. 4.10: Určenie dosiahnuteľnosti bodu pri perfektných modeloch

Rátajúc s chybami jednotlivých modelov vieme iba odhadnúť, či je určitý cieľ dosiahnuteľný, alebo nie. Mohli by sme vytvoriť pravidlo na základe predikovanej performačnej chyby, pričom by platilo, že bod je dosiahnuteľný v prípade, že je predikovaná performačná chyba menšia ako súčet priemernej predikovanej performačnej chyby zis-

tenej pri testovaní na dosiahnuteľných bodoch ($PPE(t)$) a hodnoty jej štandardnej odchýlky ($PPE_STD(t)$).

```

1 def dosiahnuteľný_bod(PPE(p), PPE(t), PPE_STD(t)):
2     if PPE(p) < PPE(t) + PPE_STD(t):
3         return True
4     else:
5         return False

```

Obr. 4.11: Pravidlo 1: Určenie dosiahnuteľnosti bodu iba na základe predikovanej performačnej chyby

Vďaka schopnosti učiť sa vystierať ruku za nedosiahnuteľnými bodmi, môžeme pri vyhodnocovaní dosiahnuteľnosti bodov využiť aj informácie o uhloch v lakti a v zápästí a zohľadniť aj informáciu, ako veľmi robot vystieral ruku za zadaným cieľom. Vieme, že pri vzdialených bodoch sa robot snaží vystierať ruku za daným cieľom. Preto by sme mohli vytvoriť pravidlo, ktoré bude brať do úvahy, či je predikovaná performačná chyba väčšia ako priemerná predikovaná performačná chyba pri testovaní na dosiahnuteľných bodoch v súčte s jej štandardnou odchýlkou, ale okrem toho ešte pridáme podmienku, či predikované uhly v lakti a v zápästí ($UL(p)$, $UZ(p)$) sú menšie ako súčty ich priemerných chýb pri tréningu na nedosiahnuteľných bodoch ($UL(t)$, $UZ(t)$) a ich štandardných odchýlok ($UL_STD(t)$, $UZ_STD(t)$). Ak budú splnené tieto podmienky, je pravdepodobné, že ide o vzdialený cieľ.

```

1 def dosiahnuteľný_bod(PPE(p), UL(p), UZ(p), PPE(t), UL(t), UZ(t), PPE_STD(t), UL_STD(t), UZ_STD(t)):
2     if PPE(p) > PPE(t) + PPE_STD(t) AND UL(p) < UL(t) + UL_STD(t) AND UZ(p) < UZ(t) + UZ_STD(t):
3         return False
4     else:
5         return True

```

Obr. 4.12: Pravidlo 2: Určenie dosiahnuteľnosti bodu aj pomocou vystretosti ruky

Pri takejto definícii pravidiel vznikajú problémy hlavne s bodmi na hranici peripersonálneho priestoru. Problémy s hranicami peripersonálneho priestoru sú však očakávané, pretože podobne aj pri procese odhadovania dosiahnuteľnosti bodov u ľudí je táto oblasť kritická a nevieme vždy s úplnou istotou určiť, či je nejaký bod dosiahnuteľný, alebo nie.

Testovanie úspešnosti pravidiel na jednotlivých modeloch

Testovanie prebiehalo na 20000 údajoch, s rovnakým množstvom dosiahnuteľných a nedosiahnuteľných cieľov.

Pri testovaní prvého z pravidiel na modele A , ktorý bol trénovaný iba na dosiahnuteľných bodoch, bola úspešnosť odhadu 93.81%.

Pri testovaní druhého z pravidiel na modele A , bola úspešnosť odhadu 90.44%.

Pri testovaní prvého z pravidiel na modele *B*, ktorý bol trénovaný aj na nedosiahnuteľných bodoch, bola úspešnosť odhadu 89.65%.

Pri testovaní druhého z pravidiel na modele *B*, bola úspešnosť odhadu 93.45%.

Vo všetkých prípadoch sme dostali pomerne vysoké úspešnosti, avšak najlepšie výsledky dalo využitie pravidla, ktoré zohľadňovalo iba predikovanú performačnú chybu na modele *A*, ktorý bol trénovaný iba na dosiahnuteľných bodoch (93.81%) a využitie pravidla, ktoré zohľadňovalo aj vystretosť ruky na modele *B*, ktorý bol trénovaný aj na nedosiahnuteľných bodoch (93.45%).

Záver

V tejto práci sme sa bližšie oboznámili s rôznymi konceptmi dôležitými v oblasti robotiky a neurónových sietí, ako sú napríklad peripersonálny priestor, dopredná a inverzná kinematika, dopredné a inverzné modely, či rôzne metódy učenia sa neurónových sietí.

Implementovali sme konkrétny model doprednej kinematiky založený na princípe neurónovej siete s využitím kontrolovaného učenia pomocou tréningových údajov získaných vykonávaním náhodných pohybov robota NICO v simulátore CoppeliaSim. Úlohou modelu doprednej kinematiky je na základe zadaných uhlov ruky predikovať výslednú pozíciu koncového efektoru.

Taktiež na princípe neurónovej siete sme implementovali inverzné modely, ktorých tréningovanie prebiehalo princípom nepriamo kontrolovaného učenia s využitím natrénovaného modelu doprednej kinematiky. Úlohou inverzného modelu je na základe zadanej pozície predikovať uhly ruky tak, aby pozícia koncového efektoru pri dosiahnutí predikovaných uhlov bola čo najbližšie ku zadanej pozícii. Pri tréningu inverzného modelu sme v prvom prípade využili ako cieľové hodnoty iba dosiahnuteľné pozície, čím sme získali vyššiu presnosť pri predikovaní uhlov ruky pre blízke ciele. V druhom prípade sme pri tréningu využili ako cieľové hodnoty aj nedosiahnuteľné pozície, čím sme dosiahli vyššiu schopnosť vystierania ruky za vzdialenými bodmi.

Napriek ťažko detekovateľným nepresnostiam dát zo simulátora sa nám podarilo dosiahnuť dostatočnú presnosť predikcií daných modelov.

Nakoniec sme ešte navrhli a vyhodnotili spôsoby odhadovania hraníc peripersonálneho priestoru s využitím naučeného modelu doprednej kinematiky, inverzného modelu a poznatkov o ich chybovosti.

Literatúra

- [1] Nguyen P. et al. (2020). Sensorimotor Representation Learning for an “ActiveSelf” in Robots: A Model Survey. *KI - Künstliche Intelligenz*, 35, 9–35.
- [2] D’Angelo, M., di Pellegrino, G., Seriani, S. et al. (2018). The sense of agency shapes body schema and peripersonal space. *Scientific Reports*, 8, 13847.
- [3] Tuthill JC, Azim E (2018). Proprioception. *Current Biology*. 28 (5): R194–R203.
- [4] Holmes NP, Spence C. (2004). The body schema and the multisensory representation(s) of peripersonal space. *Cognitive Processes*; 5(2):94-105.
- [5] Graziano MS, Cooke DF. (2006). Parieto-frontal interactions, personal space, and defensive behavior. *Neuropsychologia*. 44(6):845–859.
- [6] <https://users.cs.duke.edu/~brd/Teaching/Bio/asmb/current/Papers/chap3-forward-kinematics.pdf>
- [7] M. Renaud. (2000). A Simplified Inverse Kinetic Model Calculation Method for all 6R Type Manipulators. *Current Advances in Mechanical Design and Production VII*. Pergamon. Pages 15-25.
- [8] Welniarz Q, Worbe Y and Gallea C. (2021). The Forward Model: A Unifying Theory for the Role of the Cerebellum in Motor Control and Sense of Agency. *Frontiers in Systems Neuroscience*, 15:644059.
- [9] Nguyen-Tuong, D., Peters, J. (2011). Model learning for robot control: a survey. *Cognitive Processes*, 12, 319–340.
- [10] Straka Z, Noel J-P, Hoffmann M. (2022). A normative model of peripersonal space encoding as performing impact prediction. *PLoS Computational Biology*, 18(9): e1010464.
- [11] Kerzel M. et al. (2017). NICO — Neuro-inspired companion: A developmental humanoid robot platform for multimodal interaction. In *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, doi:10.1109/ROMAN.2017.817228.

- [12] Sinčák P., Andrejková G. (1996) Neurónové siete: Inžiniersky prístup (1. diel), Košice: Elfa.
- [13] <https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861>
- [14] Rumelhart, D. E., Hinton, G. E. (1986). Learning representations by backpropagating errors, *Nature* vol. 323.
- [15] Sinčák P., Andrejková G. (1996) Neurónové siete: Inžiniersky prístup (2. diel), Košice: Elfa.
- [16] Jordan, M. I. and Rumelhart, D. E. (1992). Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16:307–354.