Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Pohyby humanoidného robota Lilli

Bakalárska práca

2024
Matúš Bystrický

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Pohyby humanoidného robota Lilli

Bakalárska práca

Študijný program:	Aplikovaná informatika
Študijný odbor:	Informatika
Školiace pracovisko:	FMFI.KAI - Katedra aplikovanej informatiky
Školiteľ:	Mgr. Pavel Petrovič, PhD.

Bratislava, 2024
Matúš Bystrický

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:
Študijný program:

Študijný odbor:
Typ záverečnej práce:
Jazyk záverečnej práce:
Sekundárny jazyk:

Názov:

Anotácia:

Vedúci:
Katedra:
Vedúci katedry:

Dátum zadania:

Dátum schválenia:

garant študijného programu

.....

študent

.....

vedúci práce

Poďakovanie: Týmto by som sa chcel poďakovať .....

Abstrakt

Slovenský abstrakt v rozsahu 100-500 slov, jeden odstavec. Abstrakt stručne sumarizuje výsledky práce. Mal by byť pochopiteľný pre bežného informatika. Nemal by teda využívať skratky, termíny alebo označenie zavedené v práci, okrem tých, ktoré sú všeobecne známe.

Kľúčové slová: jedno, druhé, tretie (prípadne štvrté, piate)

Abstract

Abstract in the English language (translation of the abstract in the Slovak language).

Keywords:

Obsah

Úvod **1**

1 Nazov ——— **3**
1.1 Humanoidný robot 3
1.2 Humanoidný robot Lilli 4
1.2.1 Kostra robota 4
1.2.2 Hardvér 5
1.2.3 Obslužný softvér 5
1.3 3D model robota Lilli 6
1.3.1 URDF formát 6
1.3.2 URDF model Lilli 6
1.3.3 CoppeliaSim 6
1.4 Tréning chôdze v simulácii 7
1.4.1 Genetický algoritmus 7
1.4.2 Tréning chôdze 7

2 LaTeX **11**
2.1 Oprava a úprava robota Lilli 11
2.2 Počítačová aplikácia na ovládanie robota 11
2.3 Testovanie chôdze natrénovanej v simulácii 12
2.4 Spomalovanie a zrýchlovanie 12

Záver **15**

Zoznam obrázkov

1.1 Humanoidný robot WABOT-1 z roku 1973 4
1.2 Ukážka súboru pre laserovú rezačku 5
1.3 Časti nájdenej chôdze v simulácii 9

Zoznam tabuliek

1.1 Zoznam kĺbov robota Lilli 10

Úvod

Cieľom tejto práce je navrhnúť a realizovať systém, resp. postup pre exploratívnu tvorbu kinematických sekvencií reálneho robota Lilli s prvkami automatického generovania a validácie. Postup bude overený na konkrétnych praktických kinematických správaniach, ako napr. rôzne spôsoby kráčania.

Kapitola 1

Nazov ———

V tejto kapitole si priblížime základy o humanoidnćh robotoch. V podkapitole 1.2 si povieme čo o robotovi Lilli vieme. V podkapitole 1.3 oboznámime čitateľa o existujúcom 3d modeli robota Lilli. Text podkapitoly 1.4 obsahuje vysvetlenie postupu pri trénovaní chôdze v simulácii pomocou strojového učenia.

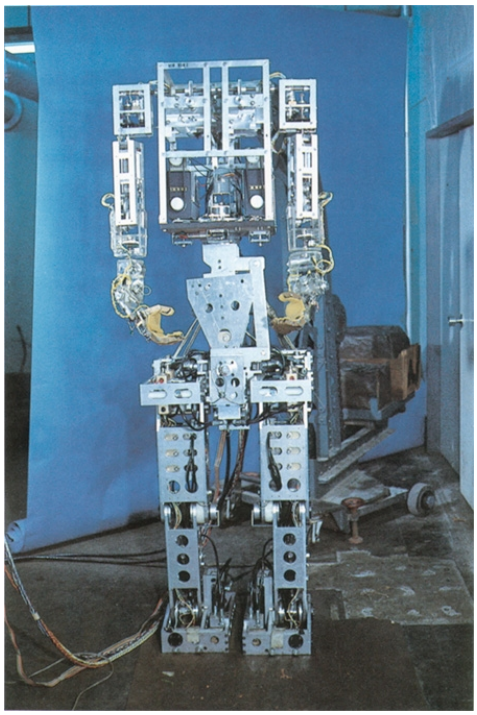
1.1 Humanoidný robot

Humanoidný robot je robot, ktorý má vzhľad podobný človeku. Sci-fi príbehy často zobrazujú roboty s ľudskou podobou, a preto si mnohí ľudia za pojmom robot rovno predstavia humanoidného robota. Na druhej strane, roboty by nemali byť vždy humanoidné. Optimálny tvar robota by mal byť určený a prispôsobený úlohám, ktoré bude vykonávať. Častokrát ľudská podoba nie je potrebná ani efektívna. [2]

Výhodou humanoidných robotov je schopnosť vykonávať úlohy v prostrediach navrhnutých pre ľudí. Ideálne sú riešenia, kde je potrebná vysoká úroveň adaptability a univerzálnosti. Humanoidi by nemuseli byť používaní iba v priemysle, či na ľahké úlohy napríklad v zdravotníctve. Majú potenciál stať sa spoločníkmi v domácnostiach, kde by mohli vykonávať bežné úlohy, ako je upratovanie, varenie alebo starostlivosť o deti a starších.

Za prvého humanoidného robota sa berie *WABOT-1* (pozri obrázok 1.1). Vytvorený bol v roku 1973 na univerzite Waseda. Vedel rozpoznávať objekty videním, pochopiť hovorený jazyk a rozprávať umelým hlasom. Bol schopný primitívnej chôdze na dvoch nohách a dokázal manipulovať s objektami pomocou rúk.

Dvojnohá chôdza nemusí byť stabilná, aj keď máme naplánovaný správny pohyb chôdze. Keďže humanoidi, podobne ako ľudia, majú ťažisko veľmi vysoko, aj malý okolitý vplyv môže spôsobiť pád. Chôdza by preto mala byť stabilizovaná za pomoci spätnej väzby zo senzorov.



Obr. 1.1: Humanoidný robot WABOT-1 z roku 1973 [2, Fig. 1.1]

1.2 Humanoidný robot Lilli

V tejto kapitole si priblížime informácie, ktoré o robotovi vieme. Bohužiaľ pre poruchu disku na fakultnom serveri sme prišli o cenné informácie z čias vyrobenia robota, preto sme ich museli spätne získavať za pomoci reverzného inžinierstva.

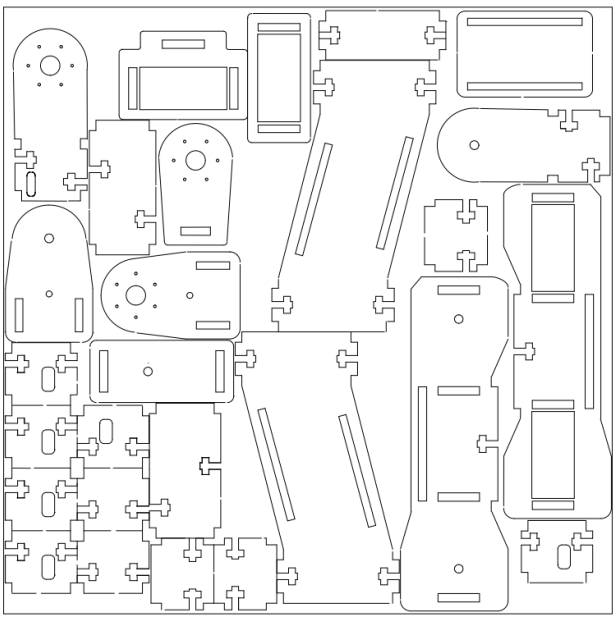
Lilli je humanoidný robot s laserom vyrezávaným dreveným skeletonom. Má výšku 75 centimetrov a 25 voľností pohybu. Ovládaná môže byť akýmkoľvek mikrokontrolérom schopným ovládať 25 PWM servo motorov. Alternatívou je mikrokontrolér podporujúci I2C protokol, ktorým komunikuje s jedným alebo viacerými servo kontrolérmi.

Lilli pôvodne navrhol Per R. Ø. Salkowitsch na výstave Maker Faire vo Viedni v roku 2018. Celý projekt je ako open source dostupný na webovej stránke Wikifactory [4], kde sa nachádzajú súbory potrebné k zostrojeniu vlastného robota. Nájdeme tam súbory pre laserové rezačky vo formáte .dfx (Drawing Exchange Format), pomocou ktorých vyrežeme z 4mm hrubých dosiek jednotlivé diely robota. Ukážka je na obrázku 1.2. Tie následne podľa návodu poskladáme.

Jeden kus robota bol v roku 2018 vyrobený na Fakulte matematiky, fyziky a informatiky Univerzity Komenského v Bratislave. Vyrobili ho študenti na praktickom seminári z robotiky.

1.2.1 Kostra robota

Pozri tabuľku 1.1



Obr. 1.2: Ukážka súboru pre laserovú rezačku [4, Limbs_B1.dxf]

1.2.2 Hardvér

Robot je ovládaný mikrokontrolérom Arduino XXXXX. V oboch stehnách má servo kontrolér Adafruit PCA9685 –OVERIT– každý schopný ovládať 16 servo motorov. Servo kontrolér na pravej strane ovláda pravú ruku, pravú nohu a hlavu. Ľavý kontrolér ovláda ľavú ruku, ľavú nohu a servo v trupe umožnujúce nakláňanie robota do strán.

1.2.3 Obslužný softvér

Robot Lilli je vybavený obslužným softvérom bežiacim na platforme Arduino. Po zapnutí ...

Po prepojení s počítačom pomocou USB kábla, vieme komunikovať cez sériový port. Softvér si pamätá index serva, ktoré užívateľ zvolil ako *aktuálne*. Týmto vybratým servom sa dá následne plynule otáčať do oboch smerov. Nižšie môžeme vidieť príkladný zoznam akcií, ktoré obslužný softvér vykoná na základe poslaného znaku.

- **”1”** - zmení servo nastavené ako *aktuálne* na servo s indexom o jedna menším
- **”9”** - zmení servo nastavené ako *aktuálne* na servo s indexom o jedna väčším
- **”q”** - posunie servo nastavené ako *aktuálne* do polohy o jeden krok väčšej
- **”a”** - posunie servo nastavené ako *aktuálne* do polohy o jeden krok menšej

1.3 3D model robota Lilli

Robot Lilli má definovanú svoju URDF špecifikáciu.

1.3.1 URDF formát

URDF (Unified Robot Description Format) je XML formát používaný na popis robota v robotike. Tento formát poskytuje komplexný opis štruktúry a vlastností robota, vrátane jeho častí, pohyblivých spojov, senzorov a vizuálnych modelov. URDF súbor je štrukturovaný pomocou XML syntaxe a pozostáva z niekoľkých základných prvkov, medzi ktoré patria:

- **robot** - hlavný element súboru, ktorý obsahuje informácie o robotovi
- **link** - reprezentuje jednotlivé časti robota
- **joint** - definuje kĺby medzi časťami robota a ich typy (revolute, planar...)
- **sensor** - popisuje senzory ako napríklad kamery alebo gyroskopy
- **visual** - určuje vizuálny vzhľad jednotlivých častí

Jednotlivé časti robota majú zadefinovanú hmotnosť a ťažisko. Špecifikácia kĺbov zahŕňa rozsahy pohybu, silu a maximálnu rýchlosť.

1.3.2 URDF model Lilli

Všetky kĺby robota Lilli sú typu *revolute*. Tento typ označuje rotáciu okolo určenej osi, čo je presne to, čo rotačné servá robota Lilli robia. Umožňujú teda pohyb iba v jednej rovine.

1.3.3 CoppeliaSim

CoppeliaSim je simulátor.... CoppeliaSim sa používa na simulácie automatizácie výroby, vzdelávanie v oblasti robotiky a prototypovanie [1].

- objekty v hierarchickej štruktúre (meshes, joints, various sensors)

- každý objekt vie byť individuálne kontrolovaný pomocou scriptu, pluginu alebo remote API klienta.

- možné programovat scripty v jazykoch Lua a Python

1.4 Tréning chôdze v simulácii

S robotom Lilli bola v simulácii trénovaná chôdza [3]. Táto podkapitola je venovaná tejto bakalárskej práci. Chôdza bola trénovaná pomocou genetického algoritmu v simulátore CoppeliaSim.

1.4.1 Genetický algoritmus

V genetickom algoritme sa snažíme nájsť riešenie problému vytvorením *populácie* možných riešení, ktoré sú reprezentované ako sady genetických informácií alebo *chromozómov*. Vhodnosť jednotlivých chromozómov sa hodnotí pomocou *fitness* funkcie. Fitness funkcia vracia číslo, ktoré vyjadruje kvalitu riešenia na základe stanovených kritérií.

Proces začína náhodne generovanou populáciou. Následne pomocou genetických operátorov vytvárame nové generácie (množiny riešení). Pomocou fitness funkcie vyberieme najlepších jedincov z generácie. Krížením párov vybraných chromozómov vytvárame nové. Z rodičov takto vytvárame potomkov. Proces zahŕňa aj mutáciu, teda náhodné zmeny v genetickej informácii, aby sa zvýšila diverzita populácie. Tento proces sa opakuje cez veľa generácií, čím by sme sa mali dostávať stále bližšie k optimálnemu riešeniu.

1.4.2 Tréning chôdze

Tréning chôdze bol naprogramovaný v jazyku Lua ako skript v simulátore CoppeliaSim. Bolo vybraných 9 kĺbov, ktorými môže robot hýbať - *l_hip_x*, *r_hip_x*, *l_knee_x*, *r_knee_x*, *l_ankle_x*, *r_ankle_x*, *l_shoulder_y*, *r_shoulder_y* a *chest_z* (pozri tabuľku 1.1). Zahŕňa to nielen servá na nohách umožňujúce chôdzu, ale aj zdvíhanie rúk v ramennom kĺbe a nakláňanie trupu na udržiavanie rovnováhy.

Jedinec v populácii je reprezentovaný postupnosťou polôh kĺbov robota. Inými slovami túto postupnosť môžeme nazvať choreografiou pohybu. Je vyjadrená ako dvojrozmerné pole. V každom kroku postupnosti jedinec obsahuje polohy všetkých kĺbov. Cyklickým opakovaním nájdenej a správnej postupnosti, dostávame chôdzu.

Proces

Na začiatku sa vytvorí náhodná počiatočná generácia. Každý jedinec sa ohodnotí fitness funkciou. Výber jedincov na kríženie je uskutočnený metódou *tournament selection*. Kríženie dvoch jedincov prebieha zvlášť na každej polohe kĺbov, pre každú polohu s 50% pravdepodobnosťou vykonania kríženia. Po krížení sa s nastavenou pravdepodobnosťou vykoná mutácia, v našom prípade zmena polohy náhodného kĺbu o malý uhol náhodným smerom.

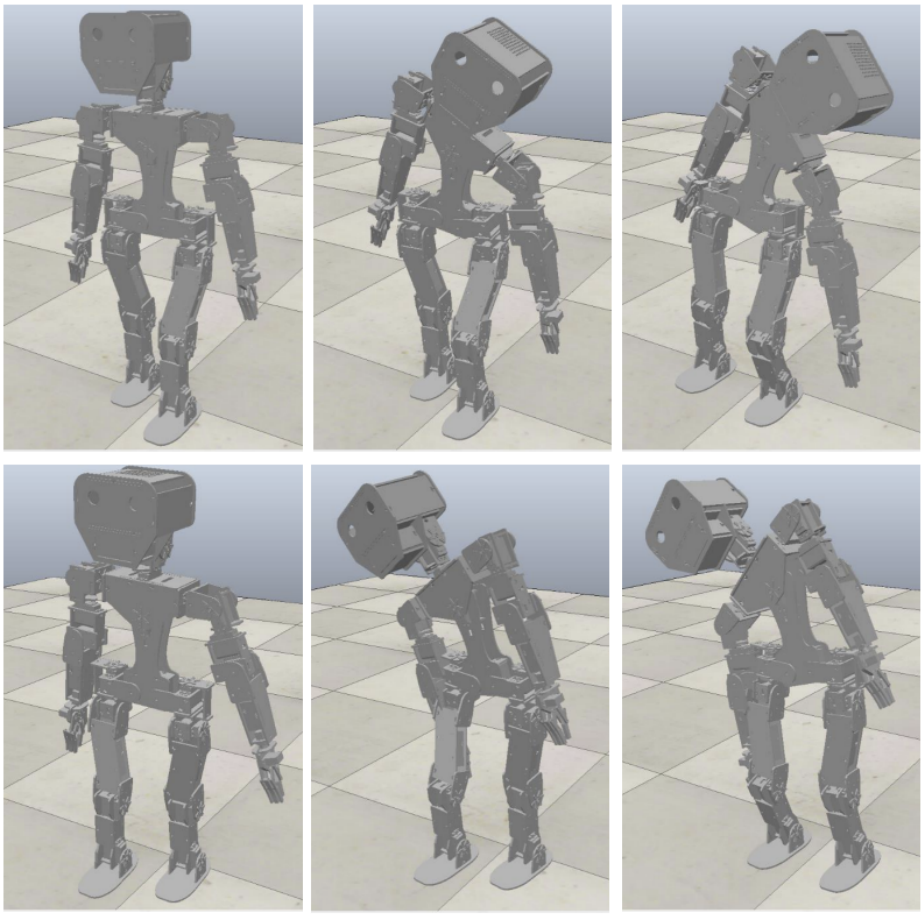
Hodnota fitness jedinca zodpovedá prejdenej vzdialenosti do momentu, kedy robot spadol. Za spadnutú sa Lilli považuje v momente, keď stred jej torza klesne pod výšku 40 centimetrov.

Výsledok

Genetický algoritmus bol spúšťaný s rôznymi vstupnými parametrami. Našli sa jedinci s nasledovnými výsledkami:

- Jedinec vedel spraviť 3 veľmi rýchle kroky bez pauzy medzi krokmi. Štvrtý krok už pre nedostatočnú rýchlosť servo motorov spraviť nedokázal.
- Jedinec vedel spraviť krok so zastavením. Krok robil jednou nohou, druhú ťahal za sebou po zemi. V každom kroku sa ale otáčal do strán, takže táto chôdza nebola veľmi efektívna z hľadiska prejdenej vzdialenosti do jedného smeru.
- Jedinec bol schopný spraviť dva kroky, každý realizovaný inou nohou. Napriek úspešnému vykonaniu tejto postupnosti polôh z génu, zlyhal pri pokuse túto sekvenciu znova opakovať prvým krokom.
- Jedinec vedel vykonať najviac krokov spomedzi ostatných. Krok však vedel robiť iba jednou nohou.

Zistilo sa, že trochu lepšie výsledky dostávame pri symetrickom dopočítaní kroku druhou nohou ako pri prenechaní kontroly druhého kroku genetickému algoritmu. Po rôznych úpravách parametrov sa nakoniec podarilo nájsť chodiaceho jedinca. Časti tejto chôdze sú na obrázku 1.3.



Obr. 1.3: Časti nájdenej chôdze v simulácii

[3, Obrázok 17]

Index	Názov v kóde	Názov v simulátore	Vysvetlenie
0	LEFT_FOOT	l_ankle_x	Ľavý členok
1	LEFT_KNEE	l_knee_x	Ľavé koleno
2	LEFT_HIP_LIFTING_FWD	l_hip_x	Ľavý bedrový kĺb - zdvíhanie dopredu
3	LEFT_HIP_TURNING	l_hip_y	Ľavý bedrový kĺb - otáčanie
4	LEFT_HIP_LIFTING_SIDEWAYS	l_hip_z	Ľavý bedrový kĺb - zdvíhanie doboku
5	LEFT_ELBOW	l_elbow_y	Ľavý lakeť
6	LEFT_SHOULDER_LIFTING	l_shoulder_y	Ľavý ramenný kĺb - zdvíhanie
7	LEFT_ARM_TURNING	l_arm_x	Otáčanie ľavej ruky
8	LEFT_WRIST	l_wrist_x	Ľavé zápästie
9	LEFT_HAND_CLOSING	l_gear_z	Zatváranie prstov na ľavej ruke
10	LEFT_SHOULDER_TURNING	l_shoulder_x	Ľavý ramenný kĺb - otáčanie
11	TORSO	chest_z	Naklánanie trupu do strán
12	RIGHT_FOOT	r_ankle_x	Pravý členok
13	RIGHT_KNEE	r_knee_x	Pravé koleno
14	RIGHT_HIP_TURNING	r_hip_y	Pravý bedrový kĺb - otáčanie
15	RIGHT_HIP_LIFTING_FWD	r_hip_x	Pravý bedrový kĺb - zdvíhanie dopredu
16	RIGHT_HIP_LIFTING_SIDEWAYS	r_hip_z	Pravý bedrový kĺb - zdvíhanie doboku
17	RIGHT_ELBOW	r_elbow_y	Pravý lakeť
18	RIGHT_ARM_TURNING	r_arm_x	Otáčanie pravej ruky
19	HEAD_TURNING	head_y	Otáčanie hlavy
20	HEAD_LIFTING	head_x	Zdvíhanie hlavy
21	RIGHT_SHOULDER_TURNING	r_shoulder_x	Pravý ramenný kĺb - otáčanie
22	RIGHT_SHOULDER_LIFTING	r_shoulder_y	Pravý ramenný kĺb - zdvíhanie
23	RIGHT_HAND_CLOSING	r_gear_z	Zatváranie prstov na pravej ruke
24	RIGHT_WRIST	r_wrist_x	Pravé zápästie

Tabuľka 1.1: Zoznam kĺbov robota Lilli

Kapitola 2

Nazov ...

2.1 Oprava a úprava robota Lilli

Robotovi neskôr praskla jedna drevená časť, konkrétne na kryte serva zabezpečujúceho dvíhanie ľavej nohy dopredu. Všetky diely nájdeme aj jednotlivo medzi súbormi na github robota Lilli, každý v troch formátoch: *.dwg*, *.dxf* a *.stl*. Konkrétne praskol diel označeý v súboroch číslom 40. Bolo ho treba vyrezať na laserovej rezačke a robota rozobrať.

2.2 Počítačová aplikácia na ovládanie robota

Naprogramovali sme počítačovú aplikáciu, ktorá vie ovládať fyzického robota ale taktiež aj robota v simulátore, prípadne obe naraz. Aplikácia obsahuje 25 sliderov (posúvačov na výber hodnoty z rozsahu), pomocou ktorých sa dá nastaviť každý z 25 stupňov voľnosti robota Lilli. Pri zmene akejkoľvek hodnoty sa v reálnom čase informácia o novej polohe serva odkomunikuje do robota.

Robota pripojíme cez USB kábel k počítaču. Aplikácia má prepínač na zapnutie a vypnutie napojenia k robotovi. Po zapnutí tohto prepínača sa sa otvorí sériový port na rýchlosti 115200 baudov a nadviaže sa komunikácia s robotom.

Aplikácia má druhý prepínač, po ktorého zapnutí sa napojí na simulátor CoppeliaSim pomocou remote API.

Aplikácia má okrem živého ovládania aj možnosť ukladať konfigurácie - stav pre každý slider. Postupnosť konfigurácií je možné prehrať v čase. Treba zvoliť čas v milisekundách pre každú konfiguráciu a potom pri spustení takejto choreografie sa kĺby robota plynule presúvajú z aktuálnej konfigurácie na nasledovnú.

2.3 Testovanie chôdze natrénovanej v simulácii

2.4 Spomalovanie a zrýchlovanie

Prejdená dráha pri našej úlohe

$$s = s_{acceleration} + s_{constantSpeed} + s_{deceleration} \tag{2.1}$$

Čas spomaľovania bude rovnaký ako čas zrýchľovania takže

$$s = 2s_{acceleration} + s_{constantSpeed} \tag{2.2}$$

Vzorec na výpočet dráhy počas zrýchľovania z nulovej rýchlosti. s = dráha, a = zrýchlenie, t = čas

$$s = \frac{1}{2}at^2 \tag{2.3}$$

Dosadíme, s tým, že zrýchľovať budeme 10% času

$$s = 2\frac{1}{2}a(0.1t)^2 + s_{constantSpeed} \tag{2.4}$$

$$s = a(0.1t)^2 + s_{constantSpeed} \tag{2.5}$$

rýchlosť počas časti s konštantnou rýchlosťou je

$$v_{max} = a(0.1t) \tag{2.6}$$

potom

$$s_{constantSpeed} = v_{max}(0.8t) \tag{2.7}$$

$$s_{constantSpeed} = a(0.1t)(0.8t) \tag{2.8}$$

$$s_{constantSpeed} = 0.08at^2 \tag{2.9}$$

potom

$$s = a(0.1t)^2 + 0.08at^2 \tag{2.10}$$

$$s = 0.01at^2 + 0.08at^2 \tag{2.11}$$

$$s = 0.09at^2 \tag{2.12}$$

$$a = \frac{s}{0.09t^2} \tag{2.13}$$

Záver

Literatúra

[1] Webová stránka spoločnosti coppelia robotics - developer simulátora coppeliasim. Dostupné na `https://www.coppeliarobotics.com/`.

[2] Shuuji Kajita, Hirohisa Hirukawa, Kensuke Harada, and Kazuhito Yokoi. *Introduction to Humanoid Robotics*. Springer Tracts in Advanced Robotics. Springer Berlin Heidelberg, 1 edition, 2014. eBook ISBN: 978-3-642-54536-8. Hardcover published on August 5, 2014. Softcover published on September 17, 2016. Original Japanese edition published by Ohmsha Ltd., Tokyo 2005.

[3] Lukáš Kostrian. Strojove učenie pohybov humanoidného robota lilli, 2023.

[4] Personal Robotics. Webová stránka o open source projekte lilli humanoid. Dostupné na `https://wikifactory.com/+personalrobotics/lilli-humanoid`.