

Vodenje žogice po trajektoriji na platformi s sistemom robotov

Adam Hrastnik, Jaka Kovše

Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, 1000 Ljubljana, Slovenija

E-pošta: ah1223@student.uni-lj.si

E-pošta: jk4684@student.uni-lj.si

Povzetek. Prispevek predstavlja razvoj demonstracijskega sistema treh robotov, katerih cilj je vodenje žogice po podani trajektoriji na okrogli platformi, s pomočjo nagiba. Prikazana je izdelava kinematičnega modela sistema in regulatorja pozicije žogice. Sistem je v celoti zasnovan in simuliran virtualno, v okolju Matlab – Simscape. Na koncu je predstavljena uporaba robotskega vida za sledenje žogici in zunanjsega uporabniškega vmesnika za vnos trajektorije.

Ključne besede: vodenje, trajektorija, regulacija, robot, robotski vid

Trajectory ball control on a platform with a system of robots

This article presents development of a demonstrational system, that consists of three robots, whose goal is to guide a ball on a round platform, based on a given trajectory, by tilting the platform. The procedures for creating the kinematical model and the ball position regulator, are shown. The system is entirely designed and simulated virtually in the Matlab – Simscape environment. Finally, it displays the usages of robot vision for ball tracking and an external user interface for inputting a trajectory.

Keywords: control, trajectory, regulation, robot, robot vision

1 UVOD

Vpliv robotov na naše življenje se večja iz dneva v dan. Z večjo mero integracije robotov v industrijo in naš vsakdan [2], se širi območje uporabe in s tem tudi kompleksnost posameznih aplikacij. Tovrstne aplikacije lahko terjajo zahtevnejše postopke vodenja in medsebojno sodelovanje različnih robotskih sistemov.

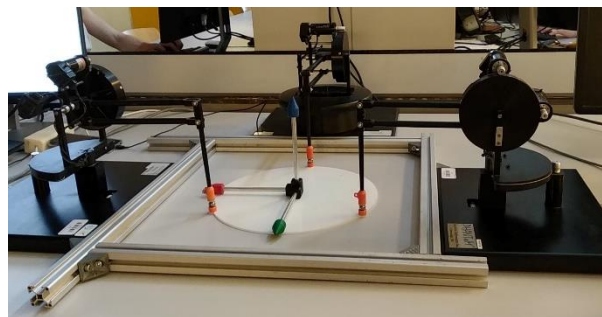
Iz tega razloga, je potrebno neprestano iskati nove načine vodenja in bolj učinkovite povezave robotskih sistemov. Pridobljene rešitve namreč lahko pripomorejo k hitrejši in večji integraciji robotov v naše življenje in industrijo – ter s tem povečajo učinkovitost celotne družbe.

V želji, da prispevamo, k razvoju robotskih sistemov, smo razvili demonstracijski sistem, ki predstavlja kompleksno vodenje skupine robotov. Za sistem smo si zadali naslednje cilje:

- Sistem je sestavljen iz skupine treh enakih robotov, ki vodijo žogico po okrogli platformi, s pomočjo nagiba.
- Žogico vodimo v dveh prostorskih stopnjah.

- Vodenje poteka po trajektoriji, ki je povsem poljubna, saj jo določi uporabnik.
- Vsa interakcija s sistemom, je izvedena preko uporabniškega vmesnika [1].
- Pozicija žogice se zazna s pomočjo robotskega vida [1].
- Vodenje se izvaja s pomočjo spodbujevalnega učenja.

2 SESTAVA ROBOTSKEGA SISTEMA



Slika 1: Postavitev robotov

Robotski sistem je sestavljen iz treh strukturno enakih haptičnih Phantom robotov [3]. Posamezen robot ima tri rotirajoče prostorske stopnje. Konec vsakega robotja je pritrdjen na okroglo platformo, premera 30 cm, ki se nahaja na sredini vseh robotov. Sredina platforme predstavlja vrh robotskega sistema. Kamera se nahaja nad sistemom, tako da ima v vidnem polju celotno platformo.

Na žalost smo bili zaradi pandemije COVID-19, primorani prestaviti celotno izvajanje projekta v simulacijo. Zato naše robote simuliramo s pomočjo poenostavljenih primitivnih modelov, kamera pa je prav tako virtualna.

2.1 Postavitve robotov

Posebno pozornost je potrebno posvetiti postavitvi robotov. Namreč, že zelo majhna neujemanja postavitve z modeli, lahko povzročijo napačno gibanje vrha robota po željenih koordinatah.

V ta namen, smo zgradili »kalibracijski« okvir, ki vedno zagotovi pravilno medsebojno postavitev in orientacijo robotov (slika 1).

3 DIREKTNA IN INVERZNA KINEMATIKA SISTEMA

Poleg izvedbe kinematičnega modela [6] za vsakega robota posebej, je potrebno za ustrezno vodenje vrha sistema (sredine platforme), tudi ustvariti model povezave med platformo in roboti.

3.1 Direktna kinematika

Direktna kinematika je namenjena izračunu pozicije vrha sistema, glede na željene notranje koordinate (rotacije sklepov) robotov.

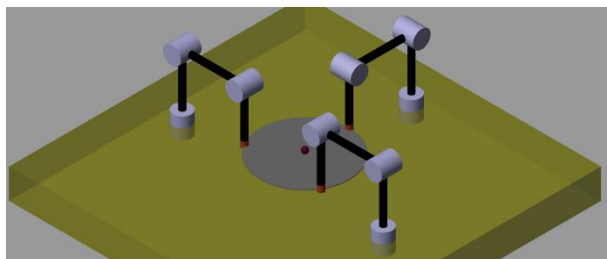
Preden lahko pridobimo vrh sistema, je potrebno izračunati kinematični model za posameznega robota. V tem primeru uporabimo za izračun tega že obstoječo knjižnico, specifično namenjeno za robote tipa Phantom.

3.2 Inverzna kinematika

Inverzna kinematika je namenjena izračunu notranjih koordinat v robotih, glede na željeno pozicijo vrha sistema. Podobno kot pri direktni kinematiki, je tudi tu potreben kinematični model za vsakega robota posebej.

Da premaknemo platformo v določeno lego, potrebujemo znane vektorje med prijemališči robotov in centrom platforme. S transformacijo koordinat dobimo vrhove robotov glede na njihov bazni koordinatni sistem. Te koordinate služijo kot vhod v funkcijo inverzne kinematike za posameznega robota.

4 SIMULACIJA V OKOLJU SIMSCAPE



Slika 2: Simulacija robotskega sistema

Simscape je razširitev okolja Simulink, ki omogoča modeliranje fizičnih sistemov, razvoj krmilnikov in njuno testiranje. Fizični model krmiljenega sistema je bil izdelan s pomočjo Simscape-ove razširitve Multibody, ki predstavlja okolje za modeliranje in simulacijo mehanskih 3D struktur.

Model, zgrajen v okolju Simscape-Multibody predstavlja nadomestek in nadgradnjo matematičnega modela uporabljenega za načrtovanje krmilnika.

5 REGULACIJA POZICIJE ŽOGICE

Za učinkovito načrtovanje regulatorja pozicije žogice je potrebno podrobno poznavanje fizičnega ozadja robotskega sistema. Najprej je potrebno zapisati enačbe gibanja žogice, ki je odvisna od naklona platforme, zato smo se lotili izdelovanja matematičnega modela za sistem z eno rotacijsko prostorsko stopnjo:

$$\left(\frac{J}{r^2} + m\right)\ddot{p} + mg\sin\alpha - mr\dot{\alpha}^2 = 0, \quad (1)$$

ki jo lineariziramo okrog $\alpha = 0$ in dobimo:

$$\left(\frac{J}{r^2} + m\right)\ddot{p} = -mg\alpha \quad (2)$$

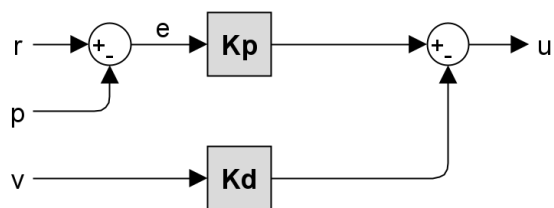
Enačbo (2) pretvorimo v Laplace-ov prostor za pridobitev prenosne funkcije sistema:

$$G(s) = \frac{p(s)}{\alpha(s)} = -\frac{mg}{\left(\frac{J}{r^2} + m\right)s^2} \quad (3)$$

J , r in m predstavljajo vztrajnost, polmer in maso žogice, medtem ko α in p predstavljata vhod (naklon platforme) in izhod (pozicija žogice) iz sistema.

Če sistem vzbujamo s stopnico, hitro ugotovimo, da je sistem nestabilen, zato naredimo povratno zanko (razlika med referenco in dejanskim stanjem na izhodu) in sistem postane mejno stabilen. Željeno obnašanje sistema dosežemo (stabilen sistem, s čim hitrejšim odzivom) s proporcionalnim ojačenjem napake in z diferencialnim ojačenjem odvoda napake. Regulator za sistem z eno prostorsko stopnjo je potrebno še dodatno uglasti $20 \cdot Kd \approx Kp$.

$$u = K_p(p_r - p_x) - K_d\dot{v} \quad (4)$$



Slika 3: Regulator posamezne osi žogice

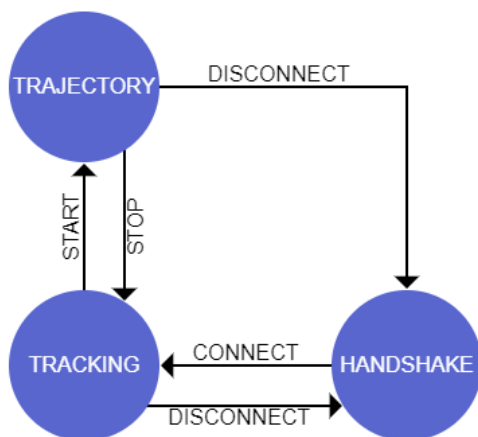
Enačba (4) in slika 3 predstavljata PD-regulator, ki kot vhodne parametre prejme referenčno in dejansko pozicijo žogice ter njeno hitrost, kot izhod pa kot naklona platforme. Premikanje žogice po platformi mora zadostovati vsaj dvema prostorskima stopnjama, zato regulator (slika 3), ki vodi žogico v smeri x-osi razširimo z enakim regulatorjem, ki skrbi za premikanje žogice po y-osi koordinatnega sistema platforme.

6 SLEDENJE ŽOGICI S POMOČJO ROBOTSKEGA VIDA

Določanje pozicije žogice, je izvedeno s pomočjo robotskega vida [1], katerega upravlja ločen program, ki neprestano zajema simulacijo. Program vsebuje uporabniški vmesnik, ki prikazuje trenutni zajem simulacije. Poleg prikaza zajema, vmesnik omogoča vnos referenčne trajektorije, po kateri naj se žogica giblje.

6.1 Komunikacija z vidom

Komunikacija z vidom poteka preko protokola UDP, ki omogoča visoko podatkovno pretočnost [4]. Pri tem projektu je pretočnost izredno velikega pomena, saj je od tega odvisna tudi uspešnost regulacije. Zaradi tega razloga, je potrebno zagotoviti kar se da nizek časovni zamik med zajeto fotografijo simulacije, procesiranjem slike, iskanjem pozicije žogice in prenosom podatka na krmilnik.



Slika 4: Komunikacijski stroj stanj

Logika za komunikacijo je izvedena s pomočjo stroja stanj:

- Program se začne v stanju »HANDSHAKE«. Ko je povezava s programom za vid potrjena, krmilnik povezavo odobri z odgovorom, ki vsebuje pritrdilno vrednost.
- Po uspešnem »rokovanju«, se aktivira stanje »TRACKING«, kjer se aktivno prejema pozicijo žogice.
- V primeru, da uporabnik vnese trajektorijo, se aktivira stanje »TRAJECTORY«, kjer se prejme in nastavi nova referenčna trajektorija za vodenje žogice.
- Če se med delovanjem povezava z vidom prekine, se stroj stanj ponastavi nazaj na začetno stanje »HANDSHAKE«.

6.2 Kalibracija zajema

Za pravilno določanje pozicije žogice in vključitev informacije v povratno zanko, je potrebno določiti

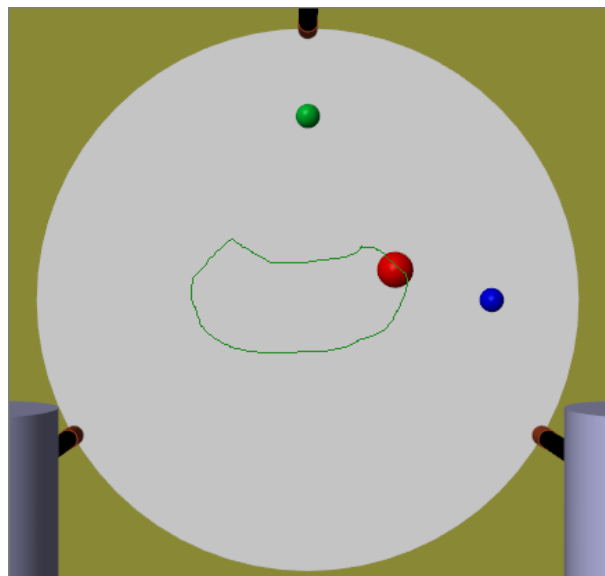
relacijo med parametri značilk žogice v koordinatnem sistemu slike in lokacijo žogice v koordinatnem sistemu platforme. To storimo z izvedbo kalibracije. Navadno kalibracija poteka tako, da v delovno območje postavimo kalibracijski objekt. Ker je bila kamera realizirana v simulacijskem okolju direktno nad platformo, na nespremenljivi poziciji, smo se že na začetku rešili neenakomernosti absolutnih velikosti slikovnih elementov. Prav tako, zaradi virtualne realizacije kamere ni bilo prisotnih vplivov optičnih aberacij kamere.

Težava je nastala pri vstavitvi kalibracijskega objekta v vidno polje kamere, saj smo na platformo želeli postaviti kalibracijsko šahovnico v smislu uvoza tekstur, vendar tega virtualno okolje ne podpira. Problem smo rešili s postavitvijo virtualnih markerjev modre in zelene barve na točno definiranih pozicijah, kot je prikazano spodaj (slika 5). Z uporabo enačb (5, 6) smo naredili preslikavo med pozicijo žogice na sliki in pozicijo žogice v virtualnem okolju, kjer Pr predstavlja razmerje slikovni element/milimeter, Mc , $M1$ in $M2$ pa predstavljajo koordinate izhodišča koordinatnega sistema platforme in markerje. Bp , Y_{offset} in X_{offset} pa pozicijo žogice ter Y in X komponento vektorja, ki premakne koordinatni sistem slike v koordinatni sistem platforme.

$$Pr = 0.1 \cdot \left(\frac{Mc_y - M2_y + M1_x - Mc_x}{2} \right)^{-1} \quad (5)$$

$$Bp = \left((X - X_{offset}), -(Y - Y_{offset}) \right) \cdot Pr \quad (6)$$

7 VODENJE PO TRAJEKTORIJI



Slika 5: Vodenje po trajektoriji

Trajektorijo uporabnik nariše v uporabniškem vmesniku, s pomočjo miške [1]. Trajektorijo tvori zbirka vzorcev, ki je bila generirana ob dogodkih premika računalniške miške. Pri tem je pomembno dejstvo, da operacijski sistem kliče dogodek premika večkrat, bolj

kot je kompleksna oblika trajektorije. To pomeni, da ravne črte vsebujejo manj vzorcev, krivulje pa več.

7.1 Prenos trajektorije

Ko uporabnik zaključi z risanje trajektorije, se začne prenos na krmilnik. Vid s prvimi paketom naznani prenos trajektorije in število vzorcev, ki jih ta vsebuje. Za tem, začne prenašati vse vzorce. Ko je zadnji vzorec prejet, krmilnik preveri, če se število prejetih vzorcev ujema s pričakovanim. Če se števili ujemata, krmilnik nadaljuje na procesiranje trajektorije.

7.2 Procesiranje in izvajanje trajektorije

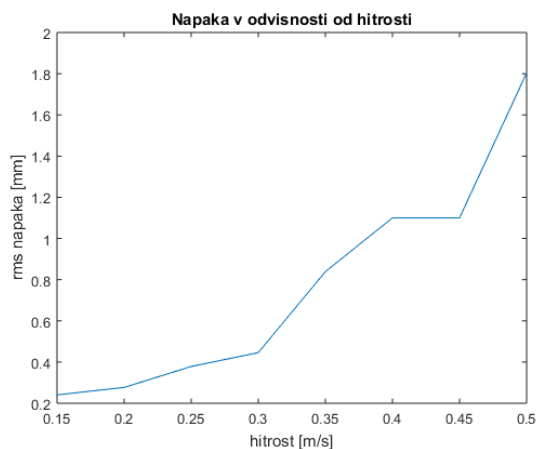
Po uspešnem prenosu trajektorije, se najprej določi vsem vzorcem časovne žige, glede na hitrost, s katero želimo, da žogica potuje po trajektoriji. Ko so žigi določeni, stroj stanj začne z iterativnim posodabljanjem vzorcev. Glede na časovne žige, stroj stanj čaka toliko časa preden nadaljuje na naslednji vzorec, kolikor je časovna razlika med časovnimi žigi vzorcev. Ko iteracija doseže zadnji vzorec se nato vrne na prvega in tako neprestano kroži med vsem vzorci trajektorije.

Posodabljanje vzorcev se izvaja samo znotraj stanja »TRACKING« (Slika 3). V kolikor se stanje spremeni, ostane na izhodu vrednost zadnjega vzorca.

8 VREDNOTENJE VODENJA

Glavna značilka, ki pove uspešnost vodenja žogice, je napaka pozicije žogice od referenčne trajektorije. Z večanjem napake od reference, žogica vse slabše sledi zadani trajektoriji in obratno.

Regulatorje navadno uglašujemo v določeni delovni točki [5], kjer postopek izvajamo, dokler ne dosežemo željenega odziva. Težava v tem primeru nastane, ker se referenca neprestano spreminja, zaradi česar težko določimo odziv, ki bo univerzalno popolnoma ustrezen. Namesto tega smo se osredotočili na prehodne pojave, oz. spremembe reference.



Slika 6 Napaka žogice glede na referenco v odvisnosti od hitrosti premikanja

Opaziti je mogoče, da se s hitrejšo spremembo reference (večjo zeleno hitrost žogice), večja napaka pozicije žogice. Iz tega lahko sklepamo, da potrebujemo za kompleksnejše trajektorije ustrezno prilagoditi željeno hitrost žogice.

9 ZAKLJUČEK

Tekom projekta smo naleteli na mnogo težav. Največje težave smo imeli z migracijo iz realnega robotskega sistema na simulacijo. Kljub temu, nam je na koncu uspelo doseči skoraj vse prvotno zastavljene cilje. Edini cilj, ki nam ni uspel, je bil vodenje s pomočjo spodbujevalnega učenja. Pri tem smo imeli namreč neodpravljljive težave s kompatibilnostjo knjižnic v Matlabu.

Ne glede na dosežene cilje, smo med razvojem naleteli na par pomanjkljivosti, oz. idej za izboljšave, za katere nam je zmanjkalo časa:

- Pri vodenju žogice z višjimi hitrostmi, nastane težava pri ostrih robovih na trajektoriji, ko jih žogica ne more doseči. Težavo se da razrešiti s prilagajanjem hitrosti, glede na ostrost trajektorije.
- Razširitev vodenja žogice na tri prostorske stopnje.
- Uporaba regulatorja stanj, ki bi znatno izboljšal odziv sistema.

Kljub vsemu, smo z rezultati zadovoljni, saj menimo, da smo uspešno razvili demonstracijski sistem za kompleksno vodenje skupine robotov.

LITERATURA

- [1] Adam Hrastnik, Jaka Kovše: Vodenje žogice po trajektoriji na platformi z uporabo robotskega vida, Robotski vid, maj 2020
- [2] International federation of Robotics press conference, 2016
- [3] Uradna stran Phantom robotov: <https://www.3dsystems.com/haptics-devices/3d-systems-phantom-premium>, 5.5.2020
- [4] Xiangning Liu, Lebin Cheng, Bharat Bhargava, Zhiyuan Zhao, "Experimental study of TCP and UDP protocols for future distributed databases", Department of Computer Sciences, Purdue University, July, 1995
- [5] Maja Atanasijević-Kunc: Sistemi in vodenje, Fakulteta za elektrotehniko, 2014
- [6] Marko Munih, Kinematika in dinamika robotov, Fakulteta za elektrotehniko, 2019