

# Daljinsko vodenje robota z inercialnimi merilnimi enotami

Gaja Žumer<sup>1</sup>, Jernej Puc<sup>2</sup>, Janez Podobnik<sup>1</sup>, Marko Munih<sup>1</sup>

<sup>1</sup>Laboratorij za robotiko, Fakulteta za elektrotehniko, Univerza v Ljubljani

<sup>2</sup>Fakulteta za matematiko in fiziko, Univerza v Ljubljani

{<sup>1</sup>Tržaška 25, <sup>2</sup>Jadranska 19}, 1000 Ljubljana

E-pošta: {jp4745, gz1671}@student.uni-lj.si, {janezp, marko}@robo.fe.uni-lj.si

## Abstract

We describe a generally applicable approach for intuitive and remote robot control, which relies on estimating the orientation of inertial measurement units that are placed on the upper and lower part of the user's arm. The Unscented Kalman filter, a key component in this estimation, is briefly discussed. Practical counter-measures to typical problems are considered, as well as suggestions for extending the application. The described approach is evaluated on a practical case, comparing the estimated trajectory to that of the robot and of Optotrak Certus' reference.

## 1 Uvod

Odkar se je sledenje človeškega gibanja pojavilo na raziskovalnem področju robotike, se je uveljavilo kot zmogljivo orodje pri vodenju robotskih sistemov; primeri uporabe se vrstijo od daljinskega upravljanja [1] do enostavnega učenja trajektorij [2]. Pri teh je dejanska uporabnost navadno pogojena s sistemom za sledenje (npr. optični ali mehanski), ki uporabnika omejuje z zahtevami po prostoru, neprekinjeni vidnosti sledenega objekta ali drugimi fizičnimi omejitvami, poleg tega pa so lahko natančne konfiguracije cenovno nerazpoložljive [3].

Iz teh razlogov so dostopne, majhne in lahke naprave (ang. *microelectromechanical systems* - MEMS) postale pomemben merilni sistem na področju kinematike in gibanja [4, 5]. Z uporabo naprav, kot so inercialne merilne enote (IME oziroma ang. IMU - *inertial measurement unit*), lahko omejitve sistemov za sledenje gibanja poljubno omilimo glede na robustnost brezžične komunikacije.

V prispevku je privzeta uporaba IME, ki vsebujejo triosni žiroskop, pospeškometer in magnetometer, saj je na podlagi kotne hitrosti, pospeška in gostote magnetnega polja lahko z metodami za senzorno fuzijo (kot je npr. Kalmanov filter) ocenjena orientacija objekta.

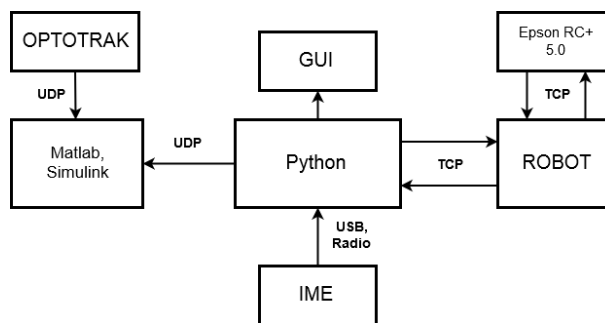
Osrednja ideja aplikacije je povezava lege vrha robota v baznem koordinatnem sistemu z lego dlani v koordinatnem sistemu rame. V nadaljevanju je opisan in ovrednoten postopek vodenja robota, ki realnočasno sledi poziciji dlani, izračunani iz orientacij IME. Postopek omogoča zmogljivost ročnega vodenja (*hand-guiding*) z dodano možnostjo vodenja na daljavo.

## 2 Metodologija

Da bi lahko sklepali o primernosti vodenja robota z uporabo IME, primerjamo tri različne meritve pozicij:

- meritve referenčnih pozicij dlani, ki jih dobimo z uporabo Optotrak Certus sistema,
- s pomočjo Kalmanovega filtra izračunane zelene pozicije robota iz meritev IME in
- prebrane (dejanske) pozicije robota.

Slika 1 prikazuje vse potrebne elemente merilnega sistema in povezave med njimi. Najpomembnejši je program napisan v programskem jeziku Python, ki skrbi za uspešno in usklajeno komunikacijo med ostalimi elementi ter istočasnim računanjem zelenih pozicij z uporabo Kalmanovega filtra.



Slika 1: Diagram komponent potrebnih za opravljanje meritve in vrste povezav med njimi.

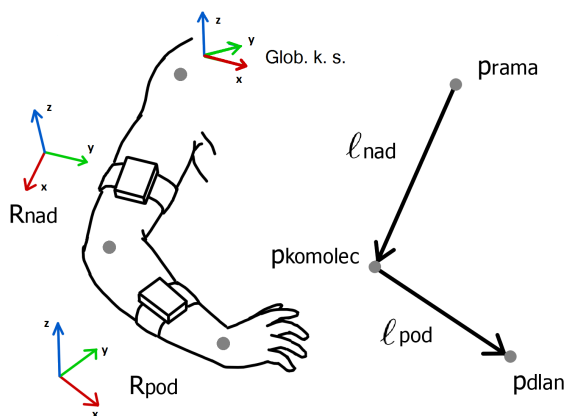
### 2.1 Kinematika

Položaj rame  $\mathbf{p}_{rama}$  definiramo kot koordinatno izhodišče  $[0, 0, 0]^T$  globalnega koordinatnega sistema, ki je poravnano s koordinatnim sistemom robota. Če poznamo dolžini nadlahti  $\ell_{nad}$  in podlahti  $\ell_{pod}$  ter njuni orientaciji  $R_{nad}$  in  $R_{pod}$ ,  $\mathbf{p}_{dlan}$  izračunamo kot:

$$\mathbf{p}_{komolec} = \mathbf{p}_{rama} + R_{nad} \cdot \mathbf{v}_{nad},$$

$$\mathbf{p}_{dlan} = \mathbf{p}_{komolec} + R_{pod} \cdot \mathbf{v}_{pod},$$

pri čemer sta vektorja nadlahti  $\mathbf{v}_{nad}$  in podlahti  $\mathbf{v}_{pod}$  definirana kot  $[\ell_{nad}, 0, 0]^T$  in  $[\ell_{pod}, 0, 0]^T$  za konfiguracijo na Sliki 2.



Slika 2: Razporeditev IME po roki uporabnika s pripadajočo vektorsko obliko, ki določa kinematične lastnosti sistema.

## 2.2 UKF (Unscented Kalman filter)

Med algoritmi za senzorno fuzijo je Kalmanov filter med najbolj uveljavljenimi. Slednji v vsakem trenutku oceni verjetnostno porazdelitev spremenljivk dinamičnega sistema glede na zaporedje meritev z znano statistično napako. V obravnavanem primeru se z integracijo kotne hitrosti po času oceni orientacijo v danem trenutku (*prediction*), nato pa na podlagi meritev pospeška in magnetnega polja oceno dopolni glede na ujemanje z referenčnimi polji (*update*). Orientacija je interno obravnavana v kvaternioni obliki; kvaternioni, ki predstavljajo razširitev kompleksnega prostora na štiri dimenzije, se v splošnem uporabljajo zaradi prednostnih matematičnih lastnosti [6, 7]. Za nelinearne probleme, kot je ocenjevanje orientacije, predpostavke osnovnega filtra niso izpolnjene, zato je v ta namen uporabljena njegova razširitev, t.i. *Unscented Kalman filter* (UKF), ki jih zadovoljivo aproksimira. Ker je teoretično ozadje filtra preobširno za podrobno obravnavo v tem prispevku, je bralec napoten k uporabljenemu viru [8].

Medtem ko lahko  $l_{nad}$  in  $l_{pod}$  enostavno izmerimo za vsakega uporabnika pred inicializacijo, se  $R_{nad}$  in  $R_{pod}$  dinamično posodabljata glede na ocenjeni orientaciji IME.

Orientacijo v diskretnem času  $n$  predstavimo z enotskim kvaternionom

$$q_{m,n} = [q_r \ q_i \ q_j \ q_k],$$

ki se posodablja glede na merjeno kotno hitrost  $\omega_{m,n}$ , kot jo meri  $m$ -ti IME, po znani enačbi 1 [6]:

$$q_{m,n+1} = q_{m,n} + \int \dot{q}_{m,n} dt,$$

$$\int \dot{q}_{m,n} dt = \frac{1}{2} q_{m,n} \otimes [0 \ \omega_{m,n}]^T \Delta t, \quad (1)$$

kjer je  $\otimes$  oznaka za kvaternioni oz. Hamiltonov produkt. Oboje je kompenzirano s korakom korekcije ob posodobitvi ocene Kalmanovega filtra.

Da oceno popravimo, projiciramo orientacijski kvaternion v meritveni prostor, tj. rotiramo referenčni vektorja polj (težnostno in magnetno), nakar je ocena primerljiva z rezultati meritev. Definicija vektorja težnostnega

polja je dokaj splošna, zato za normirani primer zadošča, da gledamo le  $z$ -komponento projekcije, medtem ko je vektor referenčnega (Zemljinega) magnetnega polja težje definirati za vsako lokacijo posebej. Poleg tega je z njim določena usmeritev globalnega koordinatnega sistema, zato bi želeli, da je definirano v skladu s specifično aplikacijo (npr. poravnano s koordinatnimi osmi upravljanega robota).

V ta namen preko enačbe

$$\mathbf{h}_{\text{ref}} = R^T \mathbf{h}$$

inverzno rotiramo merjeno magnetno polje  $\mathbf{h}$  nazaj v referenčno  $\mathbf{h}_{\text{ref}}$  ter ga po preureditvi proglasimo za novo referenco  $\hat{\mathbf{h}}$  [7]. Najbolj smiselno je, da določimo velikost (normo) horizontalne komponente in jo razdelimo med  $x, y$ -osi glede na želeno deklinacijo (kot  $\phi$   $z$ -rotacije med referenčnim poljem in baznim koordinatnim sistemom); ta mora biti eksperimentalno določena (preverjena) pred vsako inicializacijo:

$$h_{\text{hor}} = \|[h_{\text{ref}_x}, h_{\text{ref}_y}]\|,$$

$$\hat{h}_x = h_{\text{hor}} \cos \phi,$$

$$\hat{h}_y = h_{\text{hor}} \sin \phi.$$

Vertikalna komponenta  $\hat{h}_z$  v tem primeru ostaja enaka kot pri  $\mathbf{h}_{\text{ref}}$ . Ko imamo referenco znano, je projekcija v meritveni prostor ponovna rotacija referenčnega polja:

$$\mathbf{h} = R \hat{\mathbf{h}}.$$

Matematični modeli, ki opisujejo stanje in prehodno delovanje Kalmanovega filtra, so lahko poljubno kompleksni (in računsko zahtevni). Za obravnavani primer je najpreprosteje, da se orientaciji IME na nadlahti in podlahti ocenjujeta sočasno s svojim filtrom, za stanje posameznega filtra pa se vzame pripadajoči orientacijski kvaternion.

## 2.3 Vodenje robota in komunikacija

Uporabljen je 6-osni robot Epson PS3-AS00 s krmilnikom RC170.

Na strani robota se izvaja program, napisan v programskem jeziku SPEL, ki skrbi za komunikacijo in vodenje robota. Z računalnikom komunicira po Ethernet protokolu (TCP/IP), pri čemer se na robota pošilja trenutno izračunano pozicijo dlani. Ukazu za linearni premik, ki se izvaja v zanki, je dodan način *CP* (ang. *continuous path control*), ki zagotavlja zvezno gibanje vrha robota.

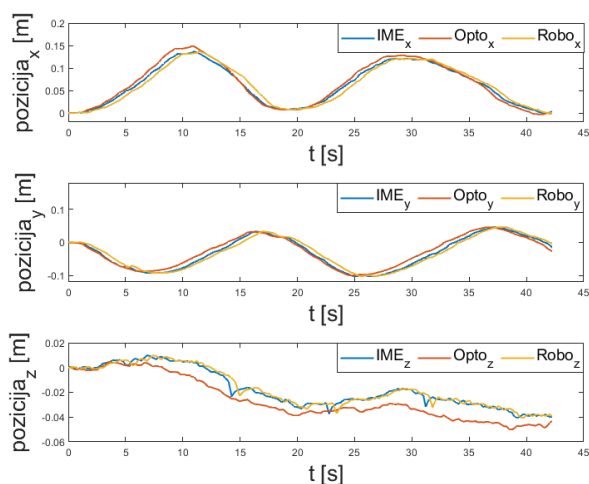
## 2.4 Optotrak Certus

Z Optotrak kamero so bile izmerjene referenčne pozicije dlani in rame. Vzpostavljena je bila UDP komunikacija s Simulink shemo na računalniku, ki je bila namenjena shranjevanju meritev.

Da so izmerjene pozicije dlani podane relativno glede na globalni koordinatni sistem (in ne koordinatni sistem merilne naprave - sistema Optotrak Certus), se od njih odšteje pozicijo rame  $\mathbf{p}_{rama}$ , ki v opisani aplikaciji služi kot izhodišče globalnega koordinatnega sistema.

### 3 Rezultati

Sledeči rezultati so bili pridobljeni s krožnim gibanjem dlani v ravnini  $x$ - $y$  glede na globalni koordinatni sistem (gl. sliko 2). Na sliki 3 so za vsako koordinatno os prikazani grafi referenčnih pozicij dlani (*Opto*), zelenih pozicij vrha robota (*IME*) in prebranih pozicij vrha robota (*Robo*) v odvisnosti od časa.

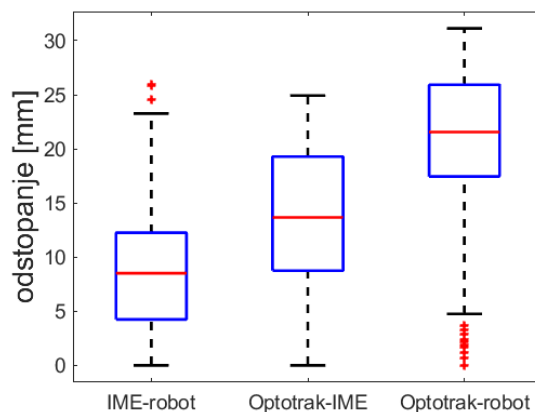


Slika 3: Grafi referenčnih pozicij dlani merjenih z Optotrakom (*Opto*), zelenih pozicij vrha robota (*IME*) in dejanskih pozicij vrha robota (*Robo*) v odvisnosti od časa. Zavoljo preglednosti ordinate grafov niso v istem merilu. Napake po oseh - povprečne vrednosti absolutnih razlik med referenčnimi pozicijami dlani (*Opto*) in dejanskimi pozicijami robota (*Robo*) - so primerljive in znašajo 9.5 mm, 13.3 mm, in 8.9 mm z ozirom na osi  $x$ ,  $y$  in  $z$ .

Iz primerjave zelenih pozicij robota (*IME*) z njegovi dejanskimi pozicijami (*Robo*) je razviden časovni zamik, ki nastane zaradi komunikacije med računalnikom in robotom ter premika robota. Pri primerjavi poteka referenčnih (*Opto*) in izračunanih pozicij dlani (*IME*) je iz grafov vidna razlika, ki nakazuje na pomanjkljivosti računske metode in kinematičnega modela roke, slabe kalibracije *IME*, neugodne vplive iz okolja (npr. vpliv bližine kovinskih objektov na meritve magnetometra), itd. Največje odstopanje se kaže med referenčnimi pozicijami dlani (*Opto*) in pozicijami robota (*Robo*), kar je posledica prej omenjenih pomanjkljivosti. V tabeli so podane izračunane povprečne vrednosti omenjenih napak in pripadajoči standardni odkloni. Osnovna statistična analiza napak je grafično prikazana na sliki 4.

Tabela 1: Povprečne vrednosti in standardni odkloni napak (norm razlik) med izračunanimi pozicijami (*IME*), izmerjenimi pozicijami robota (*Robo*) in izmerjenimi pozicijami dlani (*Opto*).

Primerjane pozicije	povp. vrednost [mm]	std [mm]
IME-Robo	8.7	5.6
Opto-IME	13.6	6.3
Opto-Robo	20.5	7.3



Slika 4: Škatlični diagram za tri različne reprezentacije napak - med zelenimi pozicijami in dejanskimi pozicijami robota (*IME-Robo*), referenčnimi pozicijami dlani ter zelenimi pozicijami robota (*Optotrak-IME*) in med referenčnimi pozicijami dlani in dejanskimi pozicijami robota (*Optotrak-robot*).

Kakovost vodenja je odvisna od večjega števila spremenljivk in dejavnikov, začenši s fizičnimi omejitvami izbranega robota (konfiguracija sklepov, število osi, delovni prostor, maksimalne hitrosti in pospeški, itd.).

Med izvajanjem meritev je bila nastavljena minimalna hitrost premika robota na 900 mm/s. Določena je bila eksperimentalno z namenom, da pri tej robot ni preveč zaozstajal za premiki dlani, saj je potrebno upoštevati časovni zamik pri komunikaciji. Smiselno bi bilo dinamično prilagajanje hitrosti robota glede na izmerjene kotne hitrosti premikanja dlani, vendar v primerjavi z zagotovljeno minimalno vrednostjo ne bi prišlo do izraza.

Mirovanje predstavlja dodaten problem. Zaradi tresenja roke robot sledi šumni trajektoriji, čeprav bi želeli, da miruje na mestu. V tem primeru bi dinamično prilagajanje hitrosti prav tako prišlo v poštev.

Za kompenzacijo tresljajev bi lahko ponovno uporabili Kalmanov filter, ki bi se v svojih fazah zanašal na različne pristope, od določenega dinamičnega modela do izračuna na podlagi orientacij *IME*, ali celo razširil koncept senzorne fuzije in uporabljal podatke kamere ali Optotraka. V vsakem primeru v sistem vnesemo nekaj zamika, hkrati pa dodaten UKF prinaša več zapletov in naredi model precej zahtevnejši. Druga ideja za kompenzacijo tresljajev temelji na predpostavki, da se slednji najbolj izrazijo v času mirovanja, medtem ko so med gibanjem manj problematični. Predlagana rešitev je uporaba sigmoidne funkcije

$$S(x) = \frac{1}{1 + e^{-x}},$$

ki je razširjena s parametri za natančnejše nastavljanje v

$$S(x) = \frac{S_{\max}}{1 + e^{-s(x-x_0)}}; \quad S(x) \in (0, 1),$$

katere namen je zagotoviti zvezen prehod med konstantno preslikavo  $f(x) = 0$  in linearno funkcijo  $g(x) = x$ :

$$f(x) + S(x)(g(x) - f(x)) = \frac{x}{1 + e^{-s(x-x_0)}}.$$

Tako bo glede na velikost premika ta bolj ali manj zadušen. Uvedemo prag blaženja  $d$ , od katerega naprej želimo, da je premik zaznan v celoti. Nastavimo  $x_0 = \frac{d}{2}$  ter strmino prehoda  $s$  kot

$$s = 2 \log \frac{\frac{1}{E_{\max}} - 1}{d}.$$

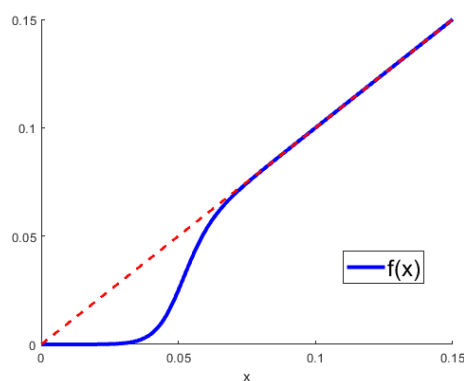
S tem dosežemo, da bo nezveznost na pragu enaka  $E_{\max} = S(0) = 1 - S(1)$ . Odločamo torej med gladkostjo prehodne funkcije in napako nezveznosti. Tako izpeljemo enačbo, s katero zagotovimo stabilizacijo v času mirovanja in glajenje tekom izvajanja premika:

$$p_2 = p_1 + \frac{\Delta p}{e^{-s(\|\Delta p\| - \frac{d}{2})} + 1}.$$

Primer take prenosne funkcije je funkcija s predpisom

$$f(x) = \frac{x}{1 + e^{-200(x-0.05)}}; x \in (0, \infty),$$

katere potek je prikazan na sliki 5.



Slika 5: Graf funkcije  $f(x)$ .

Med pomembnimi lastnostmi robota je tudi način in hitrost komunikacije, ki sta lahko večja povzročitelja napak. Za zagotavljanje zveznega gibanja je namreč ključnega pomena visoka frekvenca izmenjave podatkov. Za obravnavano konfiguracijo se je izkazalo, da se hitrost komunikacije občutno zmanjša, ko med vodenjem poteka tudi branje pozicije vrha robota.

Na podlagi pridobljenih meritev je bil izračunan približen čas, ki preteče med pošiljanjem zelene pozicije, izvedbo premika ter branjem in posredovanjem nove pozicije robota na računalnik. Večino časa, ki znaša 0.45 s, se porabi za dejanski premik in branje nove pozicije vrha robota. Pri izračunu je bila uporabljena križna korelacija med zeleno in dejansko trajektorijo robota, ki ju je bilo potrebno predhodno interpolirati, saj posamezne meritve niso bile enakomerno časovno vzorčene. Robot UR5 bi bil s stališča hitrosti komunikacije primernejši (Ethernet (TCP/IP) komunikacija s pričakovano hitrostjo 500 Hz), vendar predstavlja druge omejitve pri vodenju, ki otežujejo aplikacijo.

## 4 Zaključek

Predstavljena metoda omogoča daljinsko vodenje robota z inercialnimi merilnimi enotami prek sledenja vodilni

trajektoriji, ki jo opisuje dlan upravljalca. Vodenje je realnočasno in intuitivno, a hkrati nenatančno v primerjavi z diskretnimi alternativami.

Omejitve opisanega postopka so povezane z zmogljivostjo robota (hitrost in način komunikacije, hitrost premikanja, konfiguracija sklepov), natančnostjo kalibracije IME, napakami zaradi tresljajev rok in šuma v meritvah uporabljenih senzorjev, itd. V primerih, kjer je natančnost kritičnega pomena, je torej še vedno zaželena in potrebna natančnost eksplicitnega vnosa. Uporabnost metode je tako odvisna od aplikacije; primer bi bilo hitro učenje trajektorij, ki se v naknadnih procesih zgladijo in uredijo, kot sestavni del hibridnega vodenja ali zahtevnejše realnočasno upravljanje na daljavo [1].

Pristop je zlahka razširljiv in nosi potencial za različne aplikacije. Med predlogi za nadaljnji razvoj je hkratna uporaba obeh dlani za boljši nadzor nad položajem in rotacijo robotskega vrha. Alternativno bi druga dlan oblikovala kretnje naučenih ukazov, prepoznavne npr. z *Dynamic time warping* ali kompleksnejšimi algoritmi.

## Zahvala

Študija je bila opravljena v sklopu raziskovalnega programa št. P2-0228, ki ga je sofinancirala Javna agencija za raziskovalno dejavnost Republike Slovenije iz državnega proračuna.

## Literatura

- [1] D. Whitney, E. Rosen, D. Ullman, E. Phillips, S. Tellex. ROS reality: a virtual reality framework using consumer-grade hardware for ROS-enabled robots. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018.
- [2] M. Ortiz-Salazar, A. Rodriguez-Linan, L. M. Torres-Trevino, I. Lopez-Juarez. IMU-based trajectory generation and modelling of 6-DOF robot manipulators. International Conference of Mechatronics, Electronics and Automotive Engineering, 2015.
- [3] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Matric. Motion capture from inertial sensing for untethered humanoid teleoperation, International Conference of Humanoid Robotics, pp. 547–565, 2004.
- [4] O. J. Woodman. An introduction to inertial navigation. University of Cambridge, Computer Laboratory, Technical Report UCAM-CL-TR-696. Avgust, 2007.
- [5] H. Qiang, S. Zhang. Applications of IMU in Humanoid Robot. Humanoid Robotics: A Reference. Springer Netherlands, 2019.
- [6] K. Feng, J. Li, X. Zhang, C. Shen, Y. Bi, T. Zheng, J. Liu. A New Quaternion-Based Kalman Filter for Real-Time Attitude Estimation Using the Two-Step Geometrically-Intuitive Correction Algorithm. Sensors (Basel). September, 2017.
- [7] S. O. H. Madgwick, A. J. L. Harrison, R. Vaidyanathan. Estimation of IMU and MARG orientation using a gradient descent algorithm. IEEE International Conference on Rehabilitation Robotics. Švica, 29. junij - 1. julij, 2011.
- [8] R. R. Labbe Jr. Kalman and Bayesian Filters in Python. Maj, 2018.