Upravljanje industrijskim robotom pomoću RoboDK programskog paketa.

Bunić, Alen

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:155682

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-28**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Upravljanje industrijskim robotom pomoću RoboDK programskog paketa

Alen Bunić

Zagreb, 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Mentor:
Doc. dr. sc. Marko Švaco, mag.ing.mech.

Student:
Alen Bunić

Zagreb, 2024.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija te navedenu literaturu.

Zahvaljujem se mentoru, doc. dr. sc. Marku Švaci, mag.ing.mech., na stručnim savjetima i pruženoj pomoći tijekom izrade ovog završnog rada. Također, zahvaljujem se svim kolegama iz CRTA-e, te kolegici Tari Knežević na svakoj pruženoj pomoći i savjetima tijekom izrade rada u laboratoriju.

Naravno, najveće zahvale cijeloj mojoj obitelji i prijateljima, a posebno mami Karmeli na potpori i razumijevanju tijekom studiranja.

Naposljetku, zahvaljujem se svim svojim kolegama koji su na bilo koji način pomogli, obilježili i učinili ovaj studij zanimljivijim i nezaboravnim.

Alen Bunić

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za završne i diplomske ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala i mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa: **602 – 04 / 24 – 06 / 1**	
Ur.broj: **15 – 24 –**	

ZAVRŠNI ZADATAK

Student: **Alen Bunić** JMBAG: **0035233316**

Naslov rada na hrvatskom jeziku: **Upravljanje industrijskim robotom pomoću RoboDK programskog paketa**

Naslov rada na engleskom jeziku: **Industrial robot control using the RoboDK software package**

Opis zadatka:

Komercijalni simulatori industrijskih robota često su vrlo kompleksni i skupi programski alati koji se koriste za provjeru rada industrijskih robota prije puštanja u rad. Danas su razvijeni i dostupni različiti besplatni simulatori industrijskih robota. Jedan od takvih programskih alata za simulaciju rada industrijskih robota je RoboDK koji je proizašao kao rezultat istraživačke grupe iz Kanade. U sklopu ovog rada potrebno je detaljno proučiti mogućnosti RoboDK, ispitati ih u više virtualnih scenarija te opisati glavna ograničenja i mogućnosti programskog alata. Također u radu je potrebno:

- Demonstrirati primjene kao što su praćenje 2D krivulja i kompleksnih 3D oblika izrađenih u tehnologiji 3D printanja ili drugom tehnologijom u simulaciji i na stvarnom robotu.
- Za odabrani industrijski robot ostvariti upravljanje stvarnim robotom u laboratoriju koristeći napisane programe u RoboDK.
- Virtualni radni prostor robota i predložene primjene oblikovane i simulirane u RoboDK potrebno je izvesti na mobilnom postolju u laboratoriju s ciljem izvođenja simulacijskih programa na industrijskim robotima u laboratoriju.

Za razvijene aplikacije potrebno je oblikovati i izraditi sve potrebne robotske alate i druge konstrukcijske elemente. Rad je potrebno validirati na opremi u Laboratoriju za autonomne sustave.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:	Datum predaje rada:	Predviđeni datumi obrane:
30. 11. 2023.	**1. rok:** 22. i 23. 2. 2024. **2. rok (izvanredni):** 11. 7. 2024. **3. rok:** 19. i 20. 9. 2024.	**1. rok:** 26. 2. – 1. 3. 2024. **2. rok (izvanredni):** 15. 7. 2024. **3. rok:** 23. 9. – 27. 9. 2024.

Zadatak zadao:

Doc. dr. sc. Marko Švaco

Predsjednik Povjerenstva:

Prof. dr. sc. Damir Godec

SADRŽAJ

SADRŽAJ I
POPIS SLIKA III
POPIS TABLICA V
SAŽETAK VI
SUMMARY VII
1. UVOD 1
1.1. Industrijski roboti 2
2. ROBODK PROGRAMSKI PAKET 3
2.1. Općenito 3
2.2. Ograničenja 4
2.3. Osnovni parametri robotskih gibanja 5
2.3.1. Gibanje interpolacijom zglobova 5
2.3.2. Linearno gibanje 5
2.3.3. Kružno gibanje 6
2.4. RoboDK Add-In za SolidWorks 6
3. POSTAV ZA UPRAVLJANJE ROBOTOM 7
3.1. UR5e 8
3.2. Hvataljka PZN-plus 200-1 10
3.3. Dio za praćenje 2D krivulje 12
3.4. Dio za praćenje 3D krivulje 13
3.5. Generiranje krivulje za praćenje po plohi 15
4. POVEZIVANJE ROBOTA SA SIMULACIJOM NA RAČUNALU 17
4.1. Uključivanje robota i PLC-a 17
4.1.1. Uključivanje upravljačke jedinice i PLC-a 17
4.1.2. Uključivanje robota 20
4.2. Spajanje računala i robota 22
5. SIMULACIJA 24
5.1. Curve follow project 25
5.2. Mobilno postolje 28
5.3. Izvođenje simulacije 30
6. VALIDACIJA U LABORATORIJU 33
6.1. Kalibracija 33

6.1.1. Kalibracija koordinatnog sustava robotskog alata....33
6.1.2. Kalibracija koordinatnog sustava dijelova za praćenje krivulje....36
6.2. Izvođenje u laboratoriju....39
7. ZAKLJUČAK....42
LITERATURA....43
PRILOZI....44

POPIS SLIKA

Slika 1. Industrijski roboti u automobilskoj industriji. [1] 2
Slika 2. Collision Detection 4
Slika 3. Vrste gibanja [2] 5
Slika 4. Izgled RoboDK Add-In-a u SolidWorks-u. 6
Slika 5. Robotska stanica 7
Slika 6. e-Serija Universal Robotsa [3] 8
Slika 7. Zglobovi robota [4] 9
Slika 8. Schunk PZN-plus 200-1 hvataljka [5] 10
Slika 9. Hvataljka sa šiljkom u SolidWorks-u. 10
Slika 10. Dio za praćenje 2D krivulje u SolidWorks-u. 12
Slika 11. Izgled dijela za 2D krivulju u RoboDK. 13
Slika 12. Kreiranje konture za dio 14
Slika 13. Izgled part-a u SolidWorks-u 14
Slika 14. Izgled part-a u SolidWorks-u. 15
Slika 15. Generiranje krivulje u SolidWorks-u. 16
Slika 16. Projicirana krivulja na plohu. 16
Slika 17. Lokacija upravljačke jedinice. 17
Slika 18. Uključivanje glavnog PLC-a. 18
Slika 19. Lokacija PLC-a. 18
Slika 20. Sklopka PLC-a. 19
Slika 21. Sklopka za uključivanje robota FANUC CRX-10iA 20
Slika 22. Robot FANUC CR-15iA 21
Slika 23. Sklopa za uključivanje robota FANUC CR-15iA 22
Slika 24. Izled teach pendant-a kada je robot spojen. 23
Slika 25. Izgled RoboDK nakon spajanja na robot. 23
Slika 26. Izgled stanice u RoboDK 24
Slika 27. „Curve follow project“ postavke 25
Slika 28. Odabir krivulje za ubacivanje u RoboDK. 26
Slika 29. Koncept mobilnog postolja. 28
Slika 30. Mobilno postolje prikazano u RoboDK. 29
Slika 31. Prikaz simulacije za vrijeme praćenja 2D krivulje sa okomitim normalama. 31
Slika 32. Simulacija praćenja 3D krivulje. 31
Slika 33. Simulacija praćenje 3D krivulje na plohi. 32
Slika 34. Kalibracija alata TCP-a. 34
Slika 35. Kalibracija TCP-a u RoboDK. 34
Slika 36. Potvrda za kalibraciju alata. 35
Slika 37. Kalibracija koordinatnog sustava dijela. 36
Slika 38. Točka ishodišta koordinatnog sustava dijela u stvarnom okruženju. 37
Slika 39. Točka ishodišta koordinatnog sustava u RoboDK. 38
Slika 40. Definiran koordinatni sustav dijela. 38
Slika 41. Praćenje 2D krivulje u laboratoriju. 39
Slika 42. Praćenje po plohi u laboratoriju. 40
Slika 43. Praćenje 3D krivulje u laboratoriju. 40

Slika 44. Usporedba greške kod praćenja krivulje po plohi u laboratoriju. 41

POPIS TABLICA

Tablica 1. Specifikacije robota UR5e 9
Tablica 2. Tehnički detalji i specifikacije hvataljke 11

SAŽETAK

Cilj ovog završnog rada je detaljno proučiti mogućnosti i ograničenja programskog paketa RoboDK te demonstrirati njegovo upravljanje industrijskim robotom. Također, potrebno je ostvariti upravljanje, odnosno gibanje industrijskim robotom preko simulacije kreirane u RoboDK. U početku je dan pregled i uvid u dijelove koji se koriste za rad u Laboratoriju za autonomne sustave, kao i način na koji su ti dijelovi kreirani. Nakon toga dan je pregled kako se program RoboDK spaja s računalom. Simulacija u RoboDK vidljiva je u idućem naslovu, zatim prikaz virtualnog radnog prostora, do ostvarivanja gibanja i korištenja naredbi za gibanje. Također, prije same implementacije, potrebno je napraviti kalibraciju alata kao i kalibraciju koordinatnih sustava dijelova, na kojima se nalaze krivulje koje robot prati, odnosno, na kojima ostvaruje gibanja koja su prethodno definirana u samoj simulaciji. Naposljetku, dan je uvid u izvedbu u laboratoriju na stvarnom industrijskom robotu, u ovom završnom radu koristi se robot marke Universal robot, UR5e.

Ključne riječi: RoboDK, UR5e, industrijski robot, simulacija, kalibracija, praćenje krivulje, naredba „*curve follow project*“

SUMMARY

The goal of this final thesis is to study in detail all the capabilities and limitations of the RoboDK software package and to control an industrial robot with it. Also, it is necessary to achieve control, that is, motion of an industrial robot through a simulation created in RoboDK.. At the beginning, an overview and insight into the parts used for work in the Laboratory for Autonomous Systems, as well as the way in which these parts are created, was also given. After that, there is also an overview of how the RoboDK program connects to the computer. The simulation in RoboDK is visible in the next title, from the creation of the entire simulation, the display of the virtual workspace, to the realization of motion and the use of commands. Also, before the actual execution, it is necessary to calibrate the tools as well as the calibration of the coordinate systems of the parts, on which there are curves that the robot follows, on which it makes the movements previously defined in the simulation itself. Finally, an insight into the performance in the laboratory on a real industrial robot is given, which is Universal robot, UR5e.

Keywords: RoboDK, UR5e, industrial robot, simulation, calibration, curve tracing, instruction „*curve follow project*“

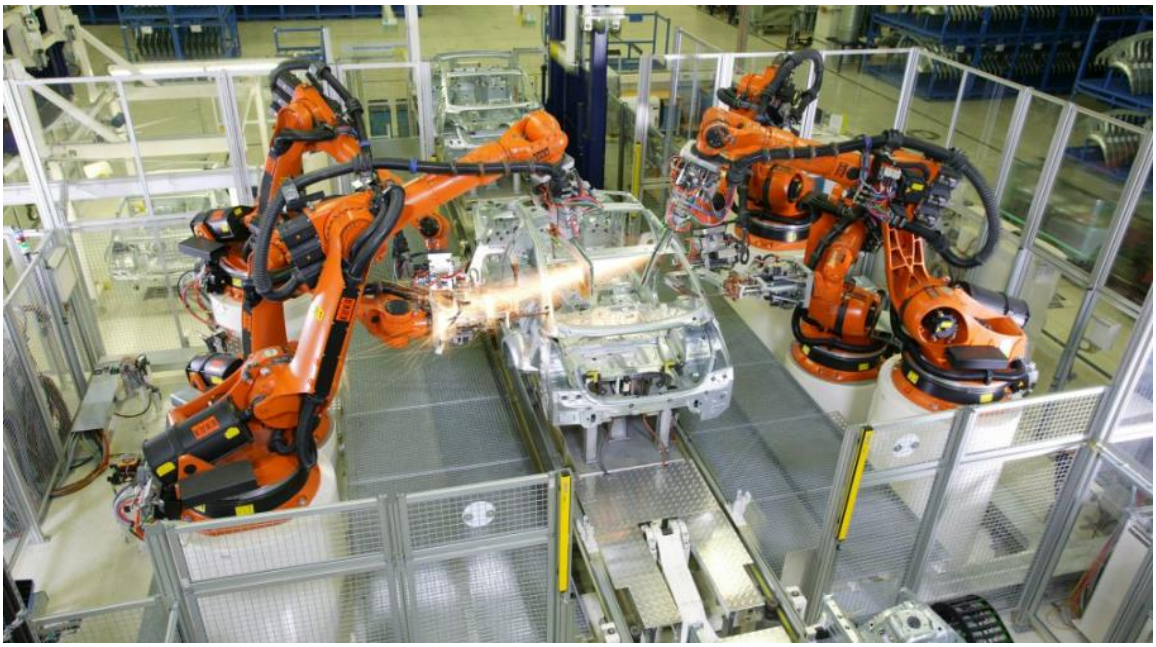
1. UVOD

Robotika kao interdisciplinarno područje predstavlja sinergiju strojarstva, elektrotehnike, mehanike i računarstva s velikom matematičkom podlogom. Osim standardnih industrijskih robota sa 6 stupnjeva slobode pa sve do SCARA i DELTA robota, postoje i vojni roboti, uslužni roboti koji se koriste u restoranima i ostalim uslužnim djelatnostima i mnogi drugi. Roboti su danas u industriji veoma zastupljeni i kao takvi utječu na efikasnost, sigurnost rukovoditelja i ostalih radnika. Naravno, još neke od najvećih prednosti robota su veća preciznost i ponovljivost u odnosu na rad čovjeka. Tijekom godina, rastom robotike i većom njihovom primjenom u industriji, cijena je postala manja, dok je dostupnost robota postala veća, te po predviđanjima, taj trend će se nastaviti i dalje. Većina robota u jednom dijelu moraju ostvariti interakciju s predmetom, kao i s ljudskom osobom. Osim što simulacije omogućuju *offline* programiranje, tako da proizvodnja ne pati za vrijeme programiranja istog, uvelike olakšavaju prikaz rukovanja i potencijalne opasnosti, kao i prostor za napredak. Tako je i nastao RoboDK programski paket, koji omogućuje prikaz i izgled stanice te ono najvažnije, mogućnost da se stvarne naredbe prenesu na stvarnog robota koji će izvoditi naredbe koje su ranije definirane u samoj simulaciji.

Na tom principu temelji se i ovaj završni rad. Potrebno je proučiti same mogućnosti i ograničenja programskog paketa RoboDK kao i mogućnost vizualizacije radnog prostora iz Laboratorija za autonomne sustave. Naravno, osim toga, mogućnost da robot izvrši naredbe i zahtjeve korisnika koji su prethodno definirani u simulaciji.

1.1. Industrijski roboti

Industrijski roboti su automatizirani mehanički uređaji koji su programirani kako bi obavljali različite zadatke u proizvodnim linijama i industrijskim postrojenjima koje im čovjek zadaje. Roboti su također ključni elementi moderne proizvodnje, zbog toga što povećaju efikasnost, preciznost i sigurnost u raznim proizvodnim procesima i zahtjevima. Također, mogu se koristiti u velikom broju namjena, posebice u automobilskoj industriji i metalurgiji, no, također se koriste i u elektronici, prehrambenoj i farmaceutskoj industriji. S razvojem tehnologija poput umjetne inteligencije, računalnog vida i strojnog učenja, industrijski roboti postaju još pametniji i sposobniji za obavljanje složenijih zadataka, otvarajući nove mogućnosti za automatizaciju u različitim industrijama.



Slika 1. Industrijski roboti u automobilskoj industriji. [1]

2. ROBODK PROGRAMSKI PAKET

RoboDK programski paket nastao je u siječnju 2015. godine u CoRO laboratoriju na ETS fakultetu u Montrealu, na jednom od najprestižnijem robotskom laboratoriju u Kanadi. Osnivač je Albert Nubiola. Programski paket je pušten u korištenje nakon što je prvi kupac, privatnik sa sjedištem u Novom Zelandu, želio koristiti RoboDK za kalibraciju ABB robota za robotsko glodanje. RoboDK je komercijalni softverski proizvod, pa je njegova upotreba podložna plaćanju licenci ili pretplatnih troškova, pa je tako određene značajke potrebno nadograditi jer nisu u osnovnoj verziji. RoboDK nema posebne hardverske zahtjeve jer može raditi na većini računala, uključujući Windows, Linux pa i Mac sustave. Minimalni hardverski zahtjevi za pokretanje RoboDK su sljedeći: Windows 7 pa na dalje, Mac OS (64-bitna), Ubuntu 16 ili Ubuntu 18 (64-bitna). Također, potrebno je imati najmanje 2GB RAM memorije i 15 GB prostora na tvrdom disku.

2.1. Općenito

RoboDK programski paket je softver namijenjen za *offline* programiranje i simulaciju industrijskih robota. Prednost korištenja RoboDK simulacije i *offline* programiranje je ta što ona omogućuje programiranje robota izvan proizvodnog okruženja. Odnosno, robot može biti programiran u simulaciji za vrijeme rada robota. Dakle, ne gube se zastoji u proizvodnji koji bi bili uzrokovani programiranjem u radnji. Softver za simulaciju ima mnoge primjene i mnoge projekte. Neki od mogućnosti su simulacija glodanja, zavarivanja, paletizacije, bojanje, kalibracije i mnogo više toga. U biblioteci je dostupno preko 50 različitih proizvođača industrijskih robota i alata.

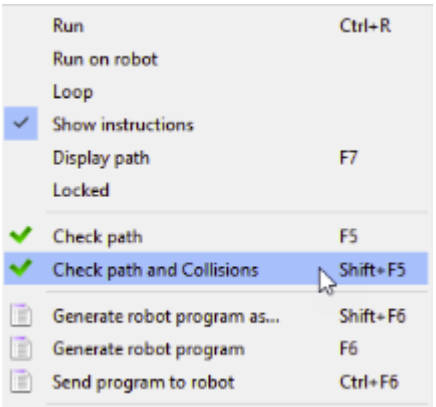
Također, RoboDK dolazi s nekoliko dodataka, odnosno Add-In. To su prilagođena proširenja koja pružaju određene značajke određenoj aplikaciji, ovisno za koju aplikaciju je Add-In namijenjen. Dodaci pružaju jednostavan način prilagodbe softvera specifičnim zadacima, što omogućuje veću automatizaciju i olakšava obavljanje poslova unutar RoboDK. Oni su razvijeni pomoću RoboDK API, koji pruža skup klasa i funkcija za interakciju sa softverom. Tako već postoje gotovi dodaci za određene poslove koje robot obavlja koji su u industriji česti, poput paletizacije, zavarivanja, CNC. API (Application Program Interface) je dostupan u nekoliko programskih jezika, uključujući Python, Matlab, C#. Najčešće korišteni su Python i Matlab. Knjižnica Robolink je direktna veza između RoboDK i Pythona. Svaki robot, predmet, koordinatni sustav ili bilo koji entitet koji se nalazi u RoboDK može se kao objekt prikazati u

Pythonu te u obliku objektno orijentiranog programiranja može se s njima raditi niz programskih manipulacija.

Osim toga, jedna od iznimno korisnih osobina samog softvera je ta što omogućuje uvoz 3D modela napravljenih u nekom od drugih CAD paketa pa tako postoje i dodaci za CAD/CAM softvere. RoboDK podržava velik broj CAD/CAM softvera, samo neki od njih su Fusion 360, Inventor, Rhino, SolidWorks... U ovom završnom radu koristi se dodatak za SolidWorks, čiji pregled je dan u podnaslovu [2.4.].

2.2. Ograničenja

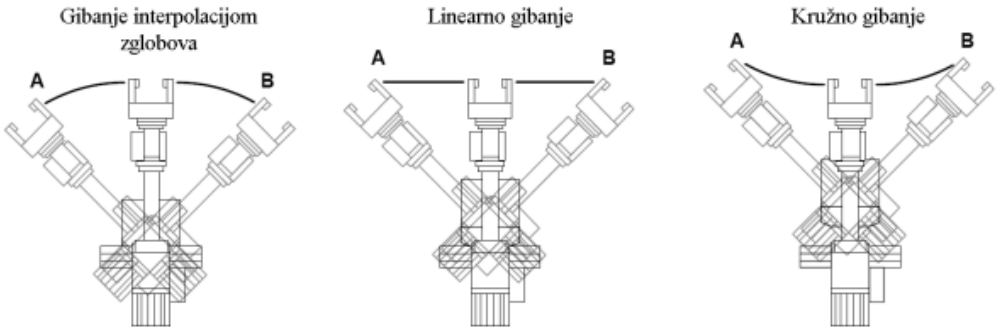
Premda je RoboDK moćan alat za simulaciju i programiranje robota, ipak postoje ograničenja na koja je potrebno obratiti pozornost. Navedeni softver je primarno namijenjen *offline* programiranju i simulaciji virtualnog prostora, tako da u stvarnom vremenu i interakciji s fizičkim robotom tijekom operacije dolazi do određenih oscilacija. Kada se istovremeno izvodi simulacija i na robotu i u simulaciji, stvarni robot se ne giba egzaktno onako kako je programiran, već s određenim zastajkivanjima, što je vidljivo i na simulaciji i na robotu. Isto tako, točnost simulacije i kalibracije ovisi i o točnosti modela robota. Također treba napomenuti i značajku *Collision Detection* koja omogućava provjeru mogućih kolizija. Ako su CAD modeli točnih dimenzija, kao i ostali radni prostor, moguća je simulacija i provjera unutar simulacija koja pomaže u sprječavanju sudara u stvarnom postavu u kojem se nalazi robot. Provjera kolizija može se koristiti na više načina poput vizualne provjere sudara, automatskog izbjegavanja sudara za robota i ostale.



Slika 2. *Collision Detection*

2.3. Osnovni parametri robotskih gibanja

U okviru RoboDK programskog paketa, gibanje je moguće ostvariti kao tri vrste. Najčešće korišteno je gibanje interpolacijom gibanja, no postoje još i linearno gibanje te kružno gibanje.



Slika 3. Vrste gibanja [2]

2.3.1. Gibanje interpolacijom zglobova

Gibanje interpolacijom zglobova (eng. *Joint movement*), je gibanje kod kojeg se sve osi robota pokreću u isto vrijeme. Trajektorija koja je dobivena nije jednostavnog geometrijskog oblika, a ono se definira tako da se izračuna najduže vrijeme koje je potrebno najsporijoj od šest osi, odnosno, onoj koliko robot posjeduje sloboda gibanja, kako bi napravio gibanje definiranom brzinom. Ta os naziva se još i ograničavajuća os. Bitna stvar kod gibanja interpolacijom zglobova je ta da je ono neosjetljivo na pojave singulariteta. Primjena ovog način kretanja moguća je u različitim aplikacijama poput zavarivanja, montaže, rukovanje materijalima i druge industrijske zadatke. Naredba se u RoboDK kreira tako da je u izborniku *Program* potrebno odabrati naredbu *Move Joint instruction.*

2.3.2. Linearno gibanje

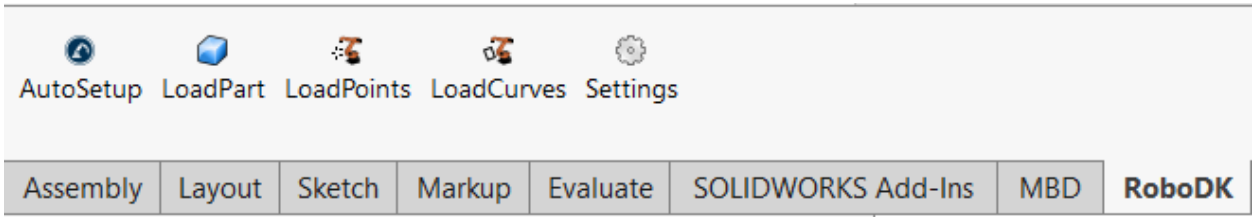
Linearno gibanje (eng. *Linear movement*) je gibanje gdje se središte alata giba linearnom interpolacijom. Središte samog alata giba se po radnoj liniji sve od početne pa do završne točke definiranom brzinom. Od početne pa do završne točke, orijentacija alata se mijenja vrlo blago ili gotovo ništa. Za razliku od gibanja interpolacijom zglobova, linearno gibanje je osjetljivo na singularnost i ograničenja zakreta zglobova robota, pa tako šestero osni roboti ne mogu proći kroz singularnost koristeći linearno gibanje.

2.3.3. Kružno gibanje

Kod kružnog gibanja (kružni luk) (eng. *Circular movement*) alat se giba kružnom interpolacijom. Središte alata robota giba se po kružnom luku od početne točke, zatim kroz prolaznu točku pa sve do završne. Za kružno gibanje potrebno je odrediti prolaznu točku koja određuje radijus kružnog luka. Isto tako, nije moguće napraviti cijeli krug u simulaciji sa samo jednom naredbom kružnog gibanja, već za puni krug mora biti razdvojen u dvije odvojene naredbe za kružno gibanje.

2.4. RoboDK Add-In za SolidWorks

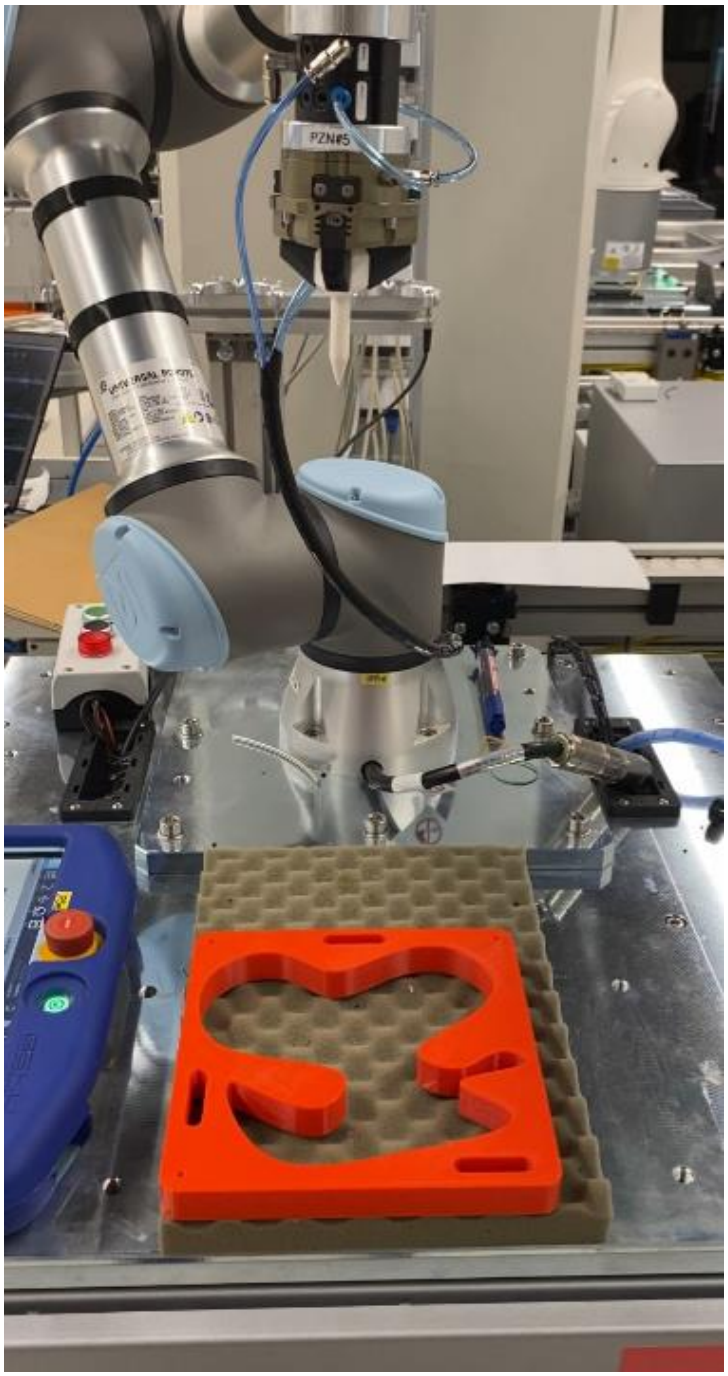
RoboDK Add-In, odnosno dodatak za SolidWorks, omogućuje kombiniranje značajki za 3D modeliranje u SolidWorks-u s RoboDK za *offline* programiranje i simulaciju robota. Dodatak RoboDK za SolidWorks radi za SolidWorks 2016 i novije. RoboDK SolidWorks plug-in je besplatan, međutim, određene opcije u RoboDK-u zahtijevaju kupnju licence. Pomoću dodatka, jednostavno se mogu učitati 3D modeli koji su izrađeni u SolidWorksu. Osim 3D modela, moguće je učitavati i točke i krivulje. S obzirom na odabir referentne površine kod učitavanja krivulje, moguće je učitati krivulju kod koje su normale okomite i uspravne, a moguće je i generirati i normale koje su okomite na plohu. Programi robota mogu se također generirati izravno iz skupine krivulja ili točaka. Također može se izraditi NC program kao što je G-kod ili APT datoteku i učitati ga u RoboDK.



Slika 4. Izgled RoboDK Add-In-a u SolidWorks-u.

3. POSTAV ZA UPRAVLJANJE ROBOTOM

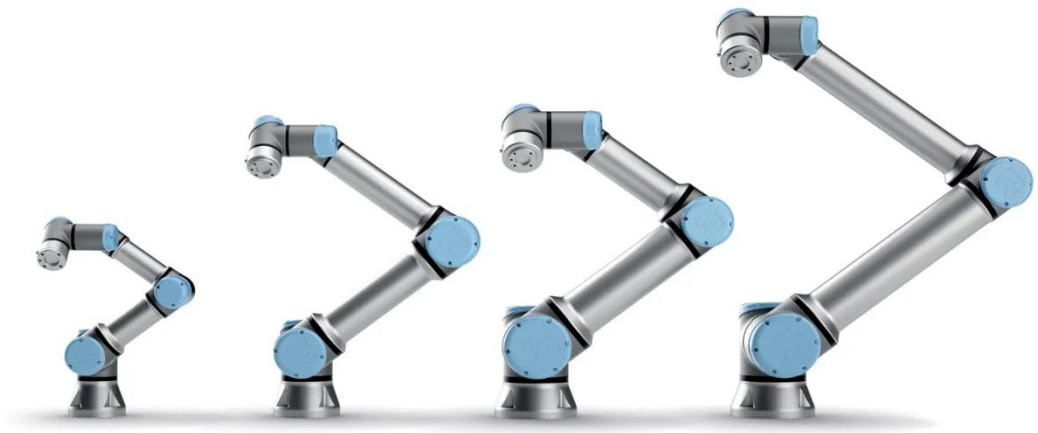
Za potrebe ovog završnog rada složen je postav u kojem se koriste robot UR5e, na koji je pričvršćena pneumatska hvataljka PZN-plus 200-1. Rad se izvodi na dva dijela (eng. *part)* koji su 3D printati u laboratoriju CRTA-e na modelu printera PRUSA. Robot je povezan preko Ethernet kabla, dok su upute za povezivanje robota s računalom na kojem se nalazi RoboDK dane na [4].



Slika 5. Robotska stanica

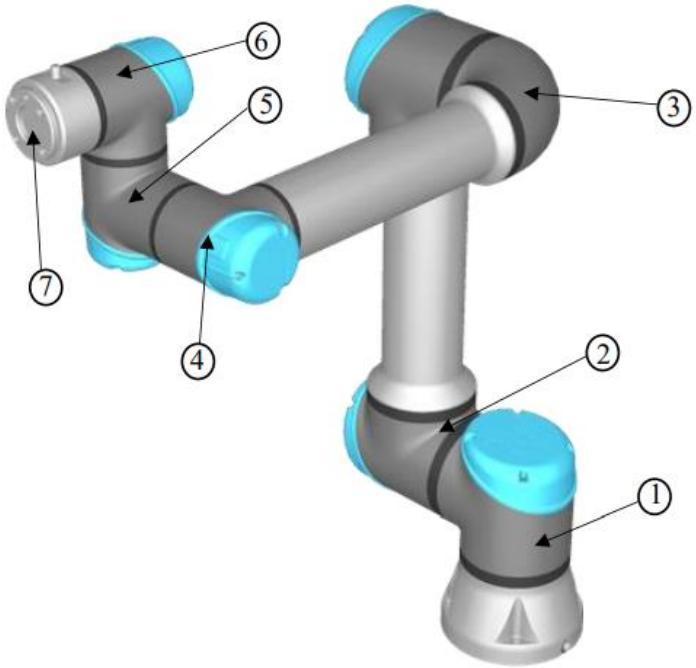
3.1. UR5e

Universal Robots tvrtka sa sjedištem u Danskom Odenseu, sastoji se od preko 650 zaposlenika. Osim sjedišta u Danskoj, imaju preko 27 ureda u 18 zemalja svijeta. U svijetu postoji preko 400.000 njihovih kolaborativnih robota. UR5e robot je kolaborativni robot. Ova robotska ruka je prvo proizvedena 2008. godine. Naziv UR5e, potječe od toga koliku nosivost robot može podnijeti. Tako ovaj UR5e robot ima nosivost od 5 kilograma. Ostali roboti te tvrtke su UR3e, UR10e, UR16e, pri čemu UR3e ima nosivost 3 kilograma, UR10e nosivost od 10 kilograma, dok robot UR16e ima najveću nosivost među tom obitelji robota od 16 kilograma. Kao kolaborativni robot, surađuje s čovjekom, odnosno, ima sigurnosne funkcije koje omogućuju sigurno kretanje robota koje osiguravaju od sudaranja, bilo samog robota s okolinom ili radnika koji upravlja njime.



Slika 6. e-Serija Universal Robotsa [3]

Upravljanje UR robotima moguće je preko nekog od robotskih softvera koji su kompatibilni sa samim robotom, te manualno preko interaktivne jedinice, odnosno, eng. *teach pendant* koji dolazi u paketu s robotom. Jedan od programski paketa kojima je moguće upravljati ovim robotom je i RoboDK, čija svojstva u upravljanju ovim robotom su opisana u ovom završnom radu u nastavku. Robot se sastoji od 6 zglobova koji mu omogućuju 6 stupnjeva slobode gibanja.



Slika 7. Zglobovi robota [4]

Na slici iznad prikazani su i označeni zglobovi robota. Zglob 1 naziva se baza robota (eng. *base*). Zglob 2 je rame robota (eng. *shoulder*). 3. zglob je lakat (eng. *elbow*), dok su zglobovi od 4. do 6. redom nazvani zglob 1. (eng. *wrist 1*), zglob 2. (eng. *wrist 2*), zglob 3. (eng. *wrist 3*). Zadnji zglob naziva se eng. *flange*, odnosno krajnja jedinica.

Tablica 1. Specifikacije robota UR5e

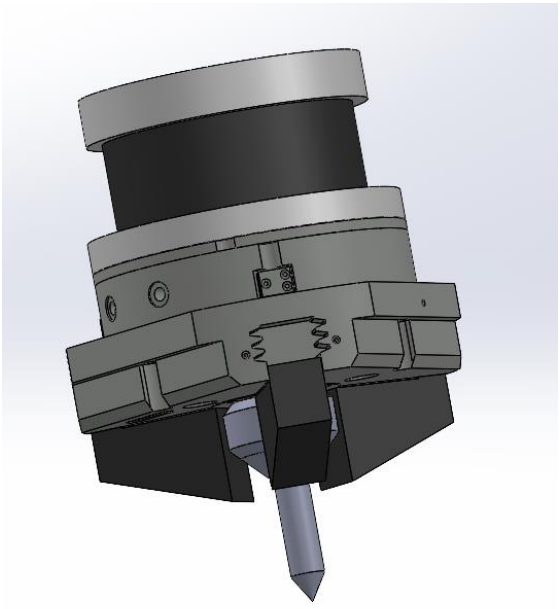
Maksimalna nosivost robota	5 kg
Doseg	850 cm
Stupnjevi slobode	6
Snaga	570 W
Sigurnost	17 konfiguracijski sigurnosnih funkcija

3.2. Hvataljka PZN-plus 200-1

PZN-plus 200-1 je pneumatska hvataljka marke Schunk. Sila hvatanja je aritmetički zbroj pojedinačne sile koja se primjenjuje na svaku od 3 čeljusti. Te 3 čeljusti omogućuju centriranje alata kojeg hvata u sredinu. Sila zatvaranja je 7100 N, dok je sila otvaranja 7540 N. Maksimalna temperatura okoline za rad je do 90 °C. Hvataljka se pričvršćuje na robot s pomoću zraka, odnosno pneumatski.

Slika 8. Schunk PZN-plus 200-1 hvataljka [5]

Na slici ispod prikazana je kompletna hvataljka s alatom, odnosno šiljkom koji je pričvršćen na hvataljku za praćenje krivulje. Slika je iz Solidworks-a, u kojem je modelirana kompletna hvataljka kao i šiljak.



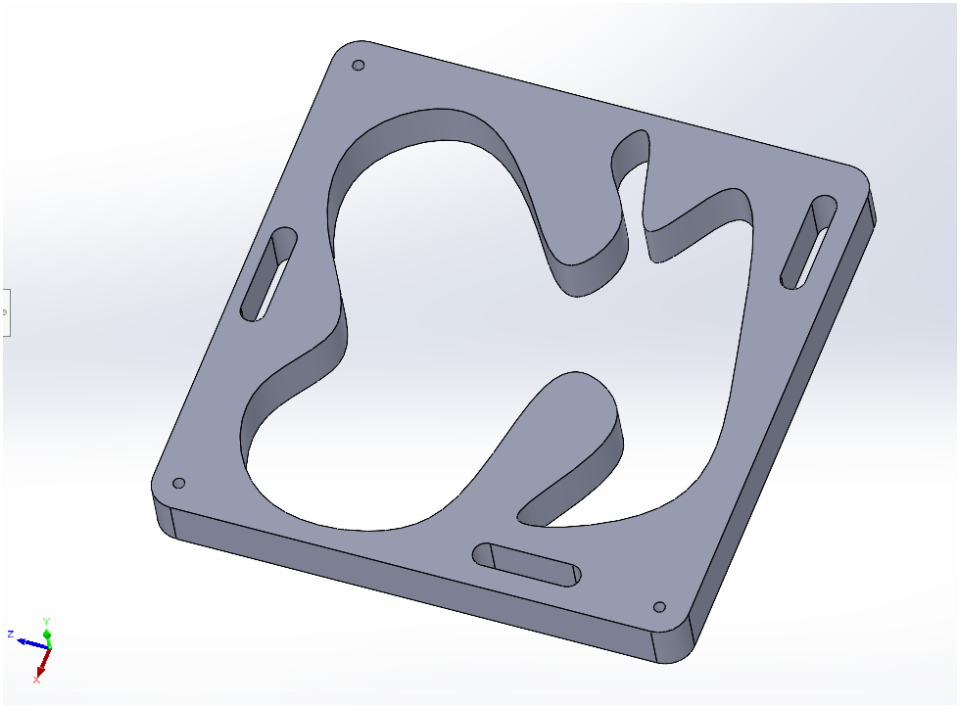
Slika 9. Hvataljka sa šiljkom u SolidWorks-u.

Tablica 2. Tehnički detalji i specifikacije hvataljke

Sila zatvaranja [N]	7100
Sila otvaranja [N]	7540
Masa [kg]	11
Preporučena težina obradka [kg]	35,5
Minimalan pritisak tlaka za rad [bar]	2
Maksimalan pritisak tlaka za rad [bar]	8
Vrijeme otvaranja [s]	1,2
Vrijeme zatvaranja [s]	1,2

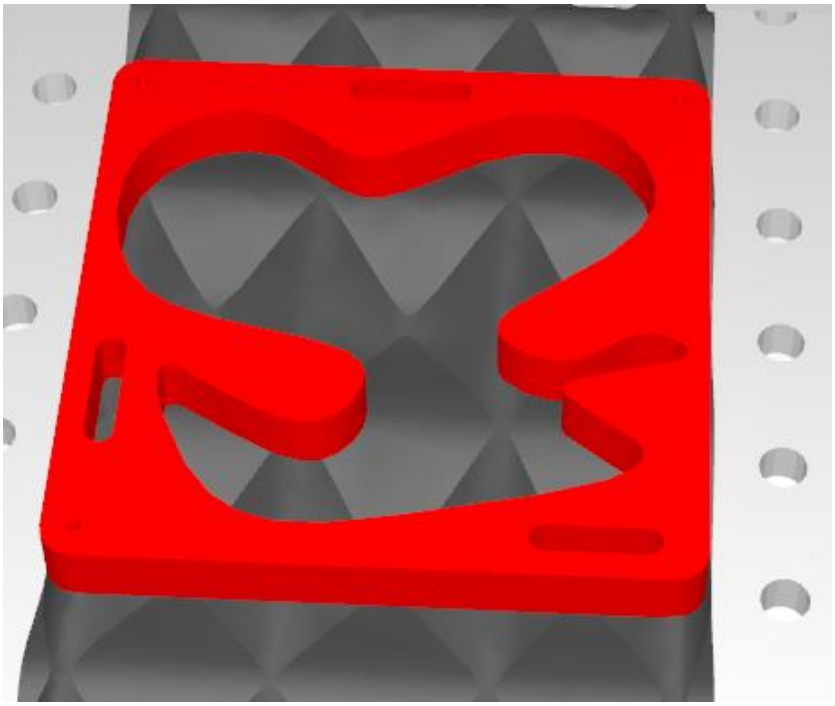
3.3. Dio za praćenje 2D krivulje

Dio, odnosno *part* za praćenje 2D krivulje, izrađen je u SolidWorks-u. Sastoji se od ravnih i zakrivljenih dijelova. Dio je dimenzija 195 mm x 195 mm x 20 mm. Dimenzije dijela određene su mogućnostima 3D printerom PRUSA koji se nalazi u laboratoriju te je na njemu printan dio koji se koristi za validaciju same simulacije. Na dijelu, osim 2D krivulje koju je potrebno pratiti, nalaze se i 3 prolazne izdužene rupe kao i 3 rupe za kalibraciju, no one nisu potpune. 3 prolazne rupe napravljene su za M8 vijak, kako bi se dio mogao pričvrstiti za stol ili za neku plohu na kojoj bi se nalazio, u slučaju da je potrebno da dio bude fiksiran. 3 ne potpune rupe služe za kalibraciju dijela, preko te 3 točke robot kreira koordinatni sustav, preko koje on egzaktno saznaje gdje u prostoru se nalazi dio te kojih je dio dimenzija.



Slika 10. Dio za praćenje 2D krivulje u SolidWorks-u.

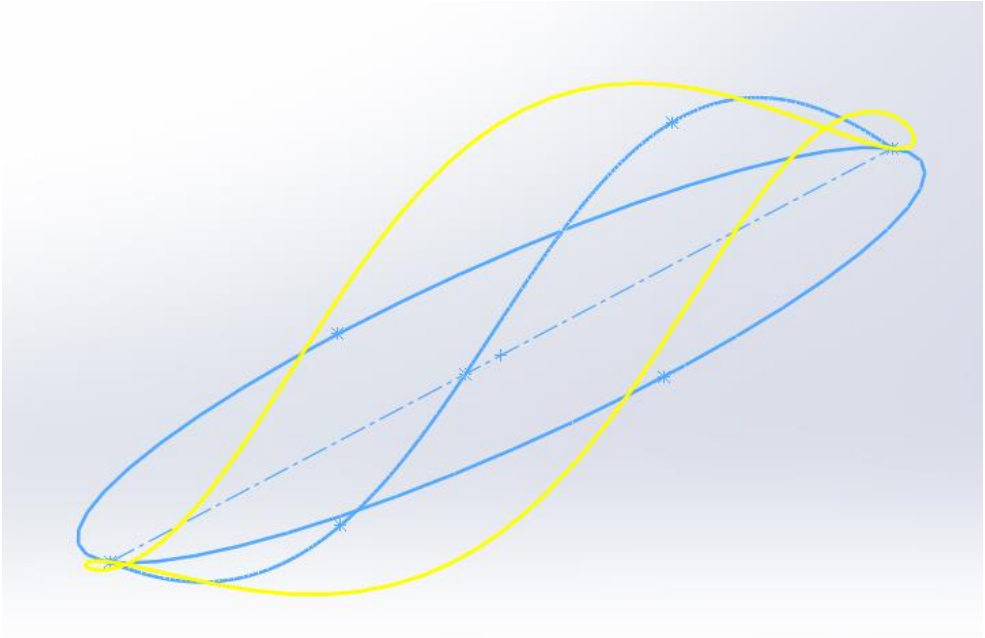
Također, isti dio je korišten i u simulaciji, no, u RoboDK je ubačen kao .stl file kako bi ga RoboDK mogao učitati.



Slika 11. Izgled dijela za 2D krivulju u RoboDK.

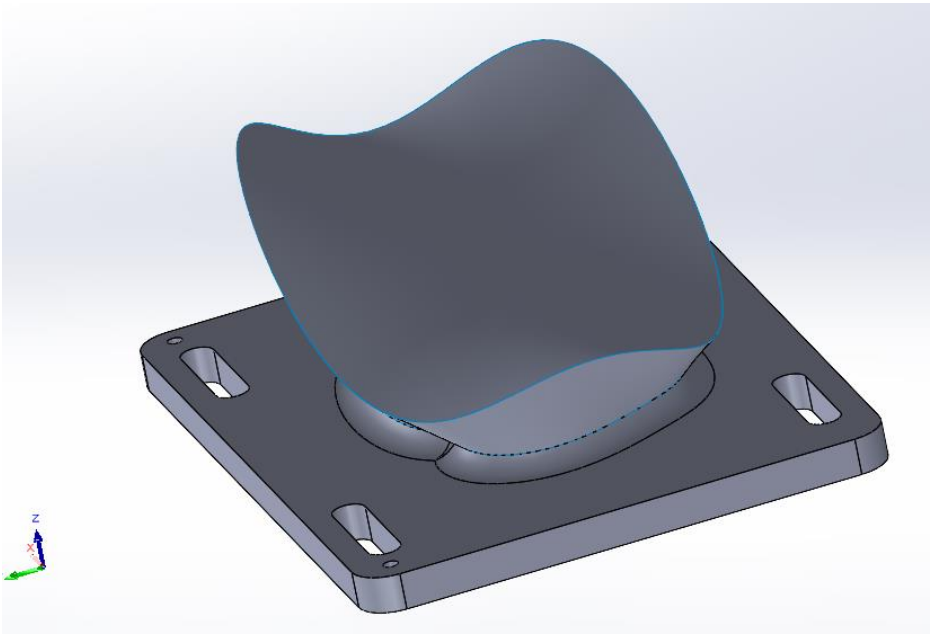
3.4. Dio za praćenje 3D krivulje

Dio za praćenje 3D krivulje, također je rađen u SolidWorks-u. Dimenzije komada su 200 mm x 145 mm x 75 mm, koji je također printat s PRUSA 3D printerom iz laboratorija. Dio se sastoji od baze, na kojoj se nalazi 4 prolazne izdužene rupe kao i 3 rupe za kalibraciju. Sva 4 provrta su za napravljena od dimenzija za M8 vijak, kako bi se dio mogao pričvrstiti vijcima, da bude fiksiran. Ostala 3 provrta, odnosno 3 djelomična provrta, također su napravljena zbog kalibracije dijela. Na bazi, nalazi se dio koji je rađen pomoću *surface design-a*. Prvo su kreirana dva *sketch-a*, iz kojih je generirana krivulja. Plavom bojom prikazani su *sketch-evi*, dok je žutom bojom prikazan oblik koji se dobije nakon ta dva *sketch-a*.



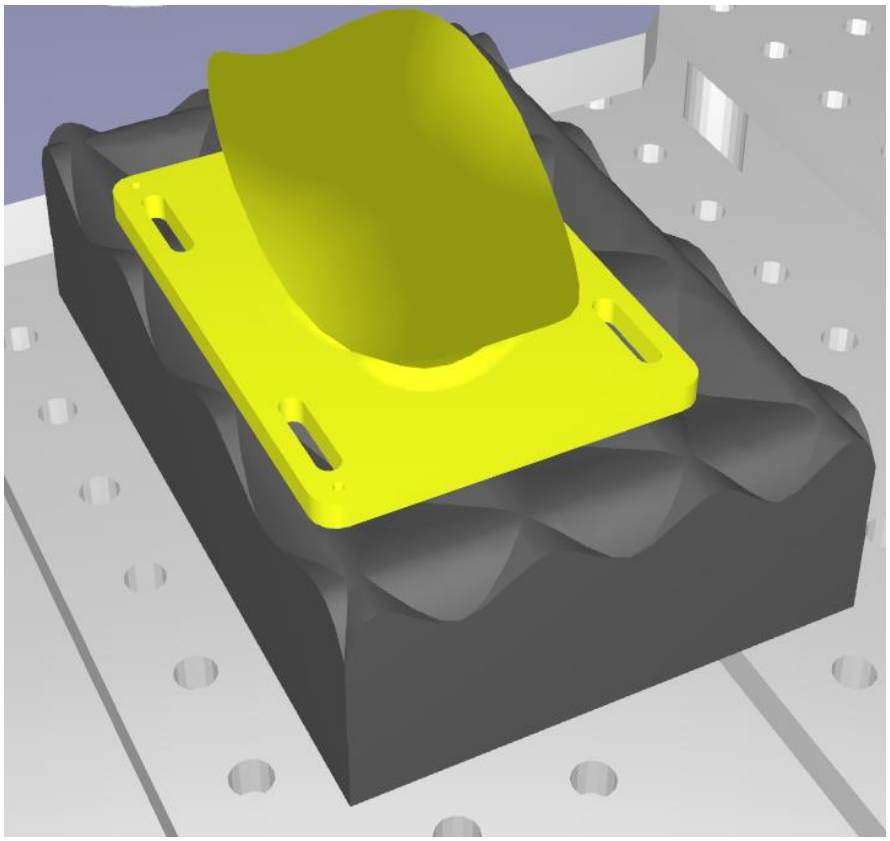
Slika 12. Kreiranje konture za dio

Nakon izrade konture, naredbom *Filled Surface* izrađena je cijela ploha. Tom naredbom konture više nisu „prazne“, već su popunjene, odnosno ploha je puna materijala. Iduća naredba koja je korištena je naredba *Boundary Surface*. To je naredba koja omogućuje kako bi se kontura spojila s bazom dijela, odnosno, kako bi prazninu između popunila materijalom, no u obliku konture koja je već kreirana. Naravno, na samim rubu gdje se spaja baza s gornjim dijelom, dodano je zaobljenje, kako bi se smanjila naprezanja, a samim time i šansa za pucanje materijala, odnosno *part-a* na tom dijelu.



Slika 13. Izgled *part-a* u SolidWorks-u

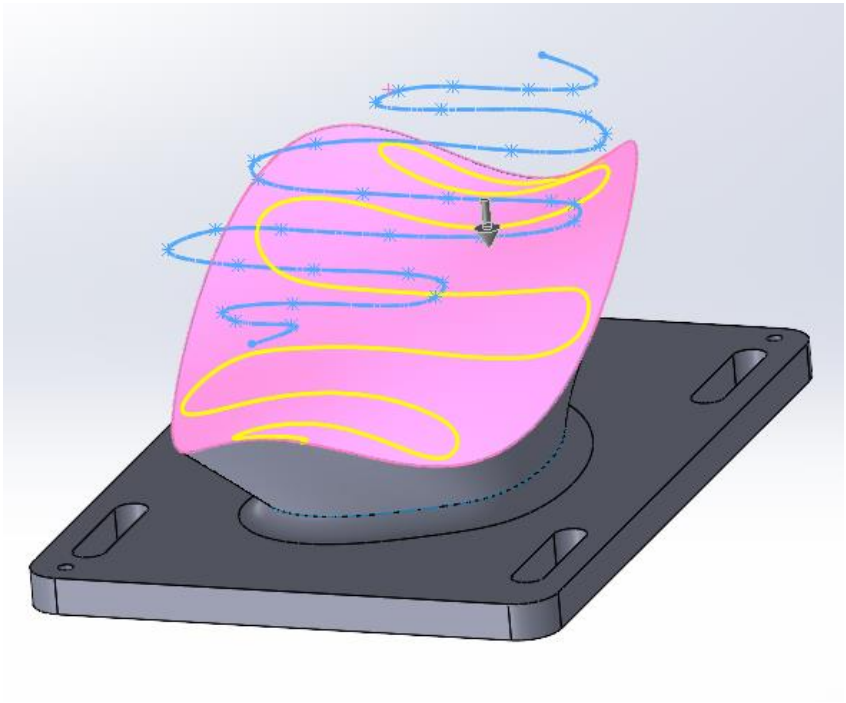
Nakon izrade dijela u SolidWorksu, te 3D printa, dio je prebačen u simulaciju u RoboDK. Prikaz kreiranog dijela vidljiv je na slici. Navedeni dio je printat 3D tehnologijom, isti je postavljen u laboratorij te je na njemu vršena validacija.



Slika 14. Izgled *part-a* u SolidWorks-u.

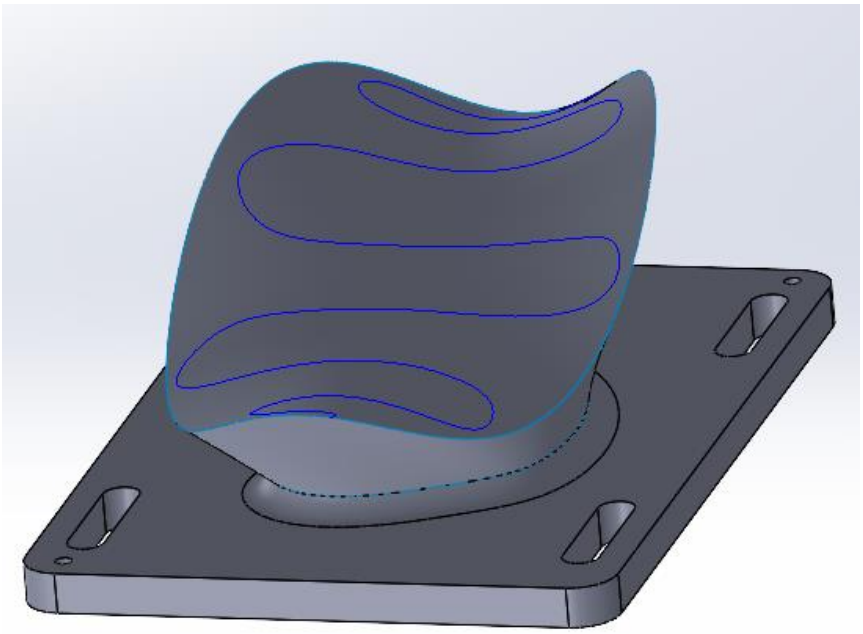
3.5. Generiranje krivulje za praćenje po plohi

Još jedan scenarij ispitan je kada robot upravljan programom RoboDK prati kompleksnu krivulju po plohi. Krivulja je kreirana u SolidWorks-u, te su normale okomite i uspravne na ravninu. Za početak je kreirana proizvoljna slobodna krivulja koja je prema izmjerama unutar plohe na koju se krivulja projicira. Nakon toga, odabrana je naredba *„Projected Curve“.* Unutar te naredbe odabrana je ranije spomenuta krivulja, te je odabrana ploha na koju je potrebno projicirati krivulju. Scenarij u kojem robot prati krivulju po plohi na kojoj je generirana krivulja, čest je u industriji. Najsličniji primjeri tome su brušenje i lakiranje po površini, kod kojih je potrebno da robot zna krivulju po kojoj se mora gibati kako bi zahvatio cijelu površinu, a opet je ne oštetio u slučaju da uđe preduboko u površinu na kojoj se nalazi krivulja.



Slika 15. Generiranje krivulje u SolidWorks-u.

Po završetku naredbe, kreirana je krivulja koja je projicirana na plohu. Nakon što je krivulja kreirana na plohi, potrebno je koristiti RoboDK Add-In kako bi krivulja bila učitana u RoboDK. Nakon otvaranja izbornika za Add-In RoboDK, potrebno je odabrati opciju *LoadCurves*, te odabrati krivulju. Krivulja je tada učitana.



Slika 16. Projicirana krivulja na plohu.

4. POVEZIVANJE ROBOTA SA SIMULACIJOM NA RAČUNALU

Za uspješno povezivanje robota UR5e s računalom na kojem se nalazi simulaciju u programskom paketu RoboDK, potrebno je napraviti sve od iduće navedenih koraka.

4.1. Uključivanje robota i PLC-a

Prvenstveno je potrebno uključiti upravljačku jedinicu, a nakon toga PLC koji je odgovoran za traku koja se nalazi pored UR5e robota. U poglavljima 4.1.1. i 4.1.2. sve sklopke prikazane su u stanju nakon što su uključene i spremne za korištenje, odnosno spajanje robota UR5e na računalo.

4.1.1. Uključivanje upravljačke jedinice i PLC-a

Upravljačka jedinica nalazi se na sredini laboratorija, između robota UR3e i ABB robota, kao što je zaokruženo i prikazano na slici.

Slika 17. Lokacija upravljačke jedinice.

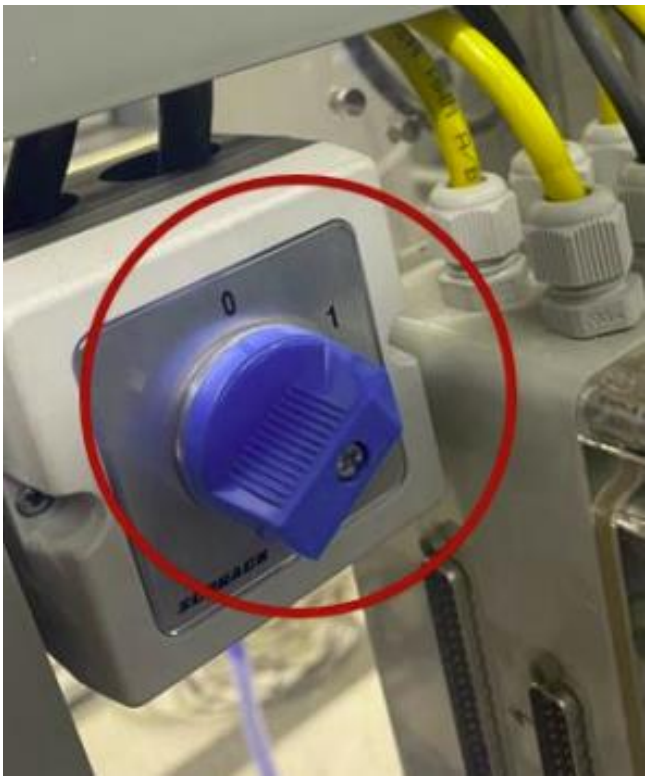
Potrebno je uključiti sklopku, nakon toga stisnuti skroz lijevu tipku *reset* gumb za napon te stisnuti srednji gumb.

Slika 18. Uključivanje glavnog PLC-a.

Nakon uključivanja upravljačke jedinice, potrebno je uključiti i PLC. PLC nalazi se iza FANUC CRX-10iA robota ispod pokretne trake.

Slika 19. Lokacija PLC-a.

Sklopka za uključivanje PLC-a vidljiva je na slici ispod.

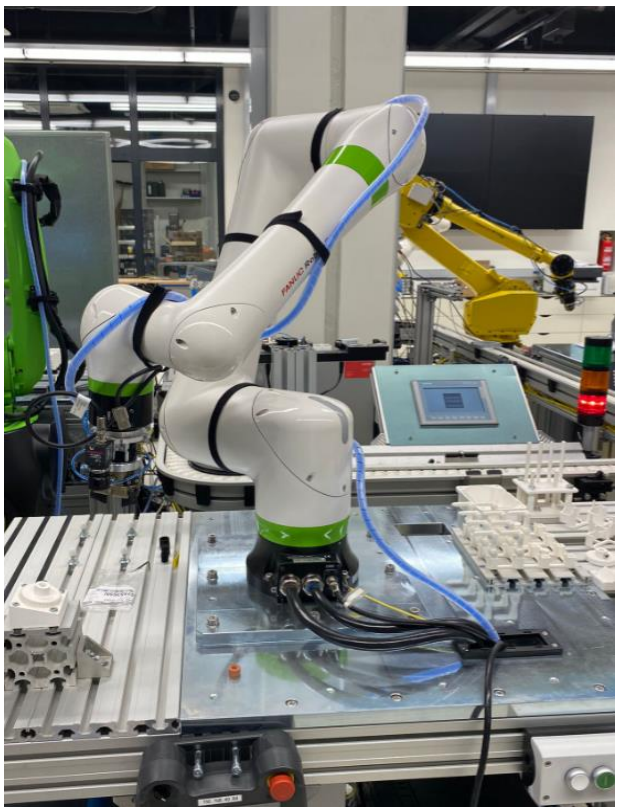


Slika 20. Sklopka PLC-a.

Nakon što su upravljačka jedinica i PLC uključeni, potrebno je uključiti i sve robote koji prethode radu UR5e robota na pokretnoj traci.

4.1.2. Uključivanje robota

Prvi robot koji je potrebno uključiti je FANUC CRX-10iA.

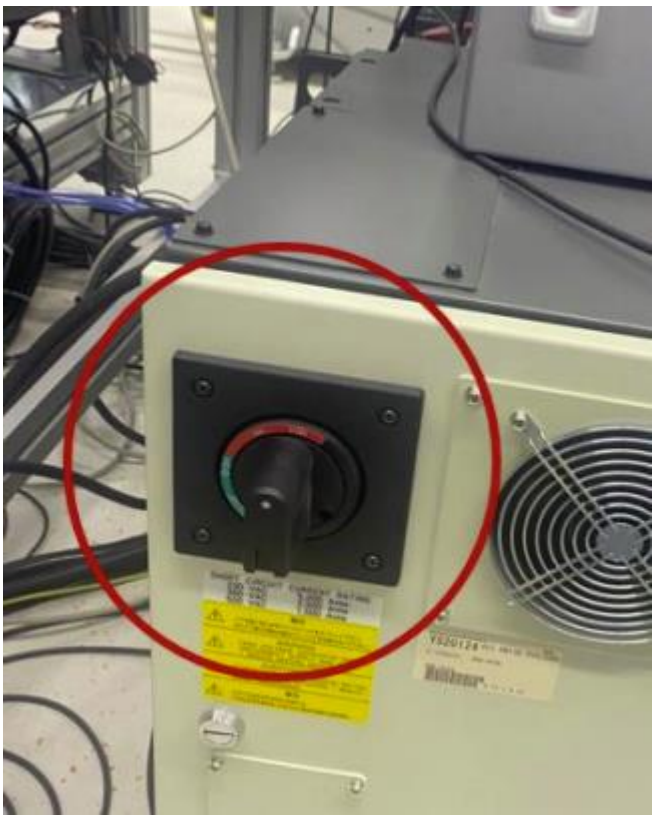


Slika 19. FANUC CRX-10iA

Slika 21. Sklopka za uključivanje robota FANUC CRX-10iA

Idući robot za uključivanje koji je također FANUC CR-15iA te se on nalazi između prethodno FANUC-a i UR5e robota na kojem se izvodi rad.

Slika 22. Robot FANUC CR-15iA



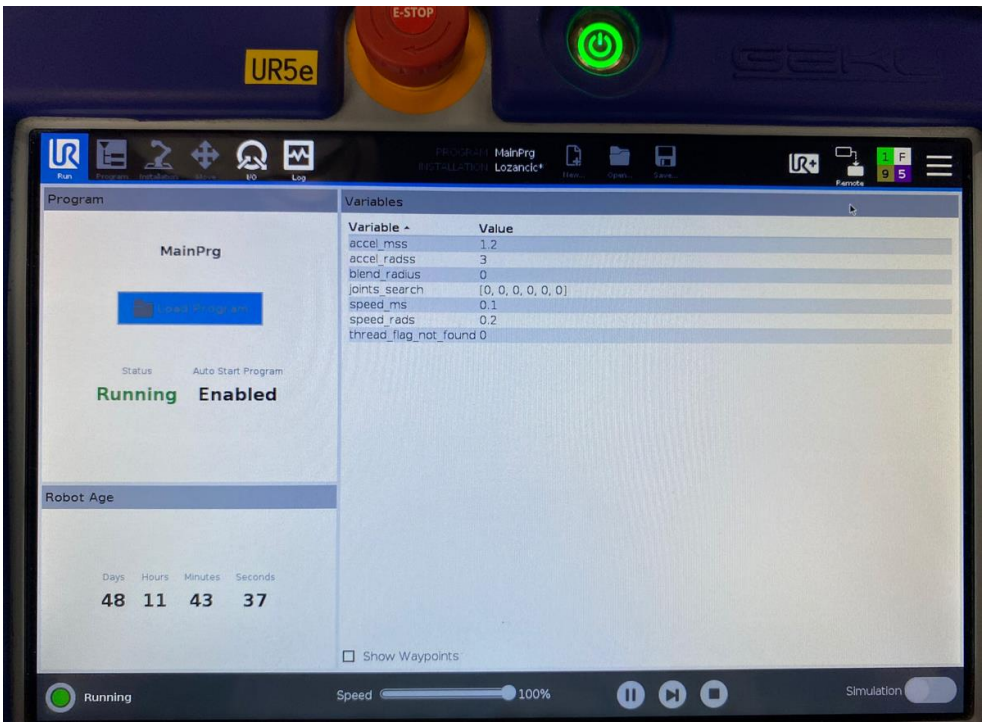
Slika 23. Sklopa za uključivanje robota FANUC CR-15iA

4.2. Spajanje računala i robota

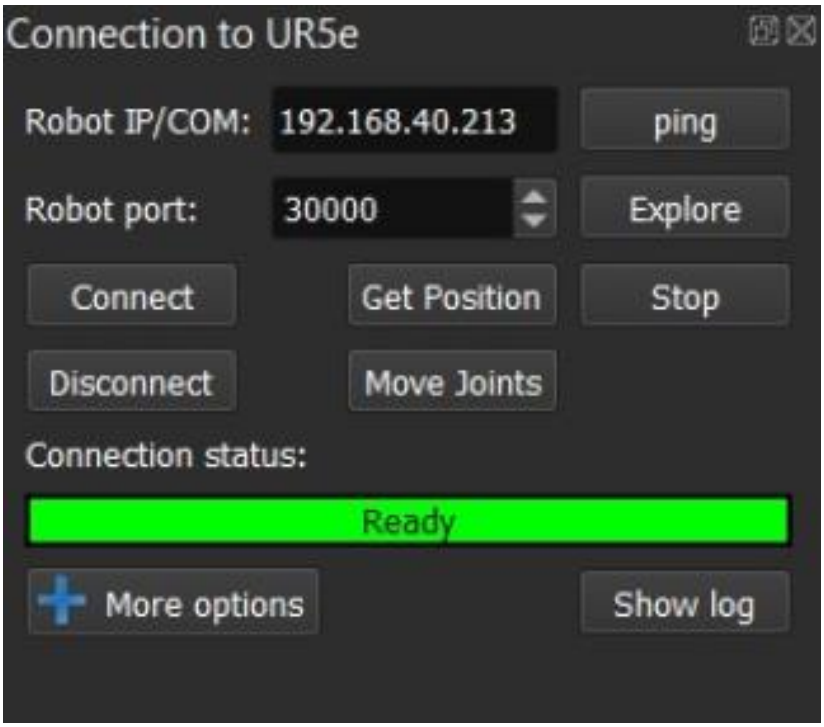
Za spajanje robota s računalom potrebno je da se oni međusobno mogu upariti. Računalo je ethernet kabelom spojeno na isti razvodni stup kao i robot. Nakon uspostavljanja fizičke veze, iduće na redu je provjera IP adrese računala. Potrebno je da i računalo i robot imaju isti ID mreže. IP adresa vidljiva je na naljepnici na bazi robota, no može se pronaći i u postavkama na *teach pendant-u.* Potrebno je ući u *settings*, te tamo u izborniku je potrebno odabrati *system,* a zatim *network,* tamo je vidljiva IP adresa robota, koja za navedeni robot UR5e iznosi ’192.168.40.14’.

Posljednje što je potrebno napraviti kako bi omogućili da robot i računalo budu spremni za povezivanje, je to da se na *teach pendant-u* odabere opcija *remote control*. Slika [24] prikazuje *teach pendant* kada je robot spojen s programom RoboDK.

Za povezivanje robota potrebno je još odabrati u RoboDK opciju *connect*, te tamo upisati IP adresu od robota s kojim se spaja. Nakon uspješnog spajanja, dobije se potvrda u obliku zelene notifikacije na kojoj piše *connected.*



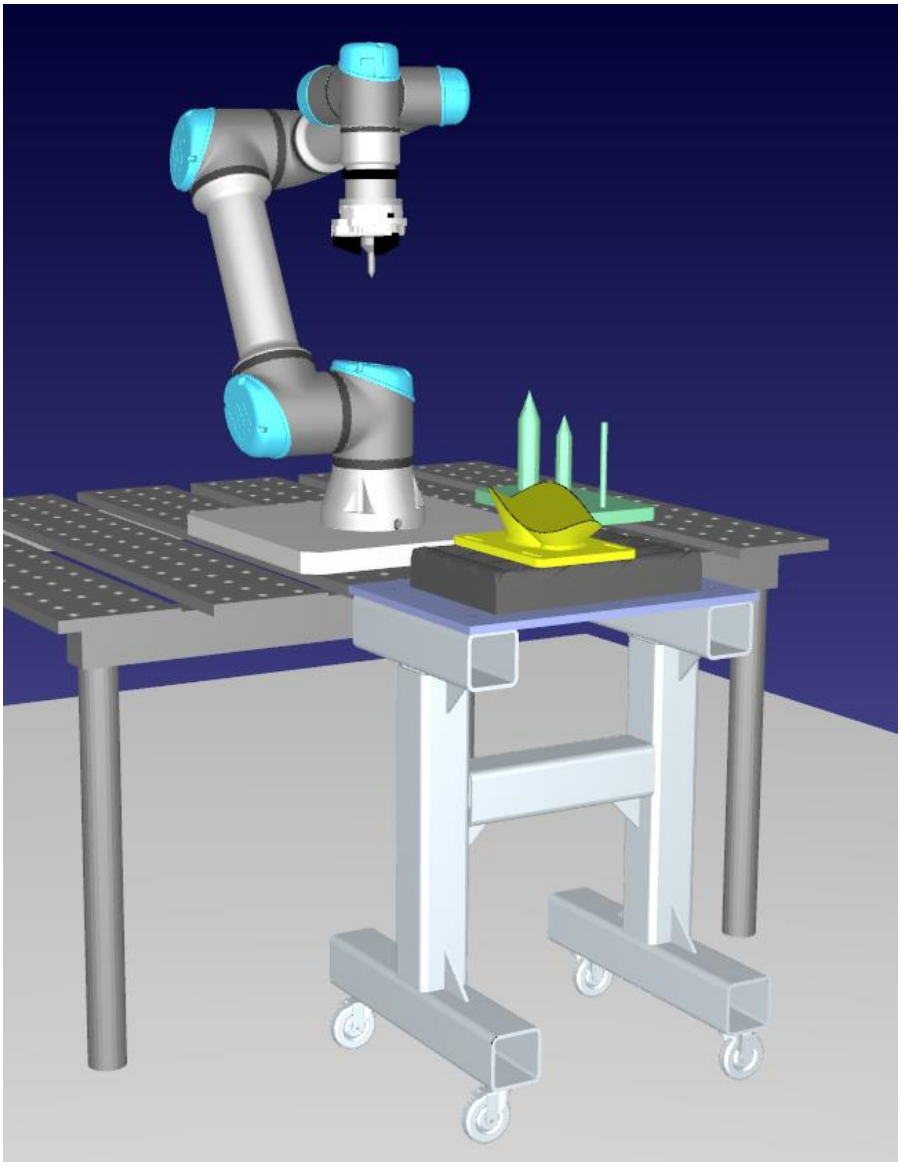
Slika 24. Izled *teach pendant-a* kada je robot spojen.



Slika 25. Izgled RoboDK nakon spajanja na robot.

5. SIMULACIJA

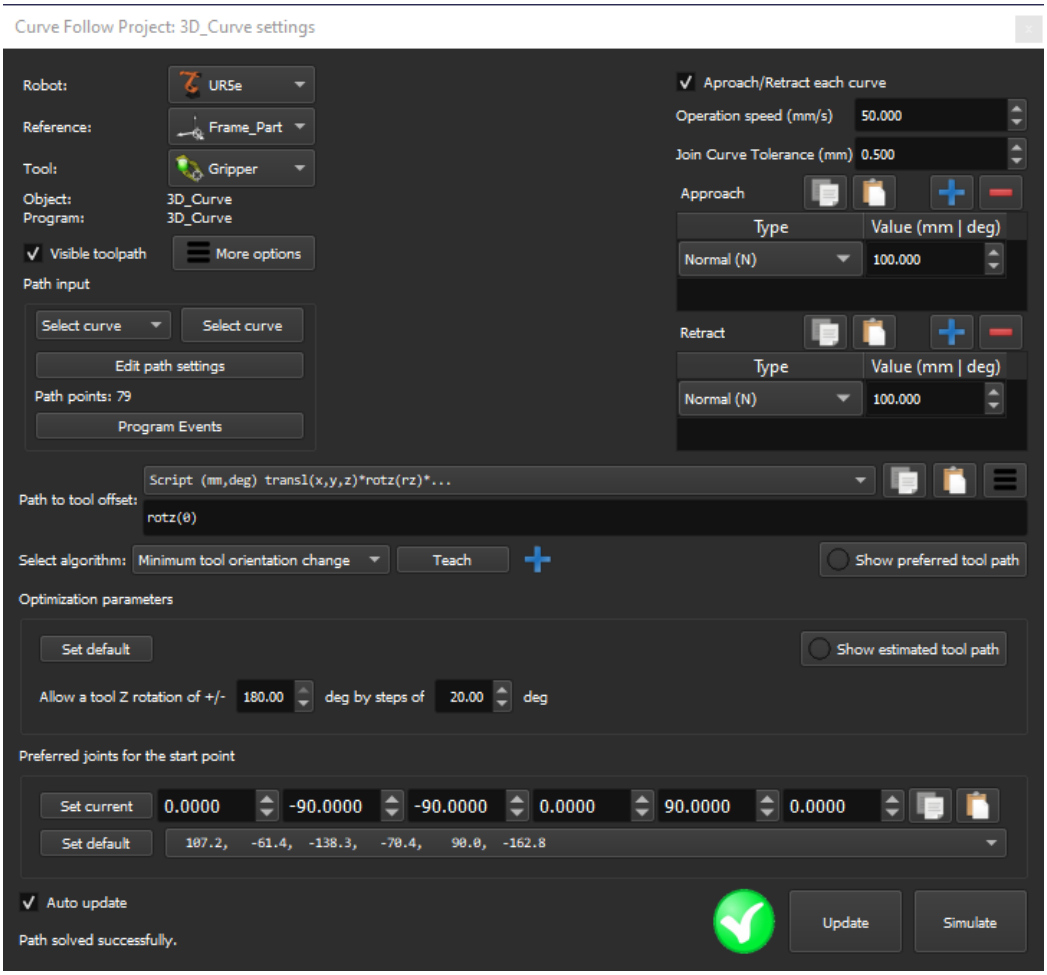
Za početak simulacije u programskom paketu RoboDK, potrebno je za početak kreirati novu stanicu i odabrati robota. Robot se može dodati klikom na *file* te odabirom *open robot library*. Nakon toga, otvara se službena stranica od RoboDK gdje je dan popis svih robota koji se mogu koristiti u programu, te s nekim od njihovih glavnim karakteristikama, kao što su marka, model, težina, nosivost… U ovom završnom radu korišten je robot UR5e, koji je prethodno opisan. Odabirom robota UR5e, on se otvara i ubacuje u program. Iduće na redu je ubacivanje ostalih dijelova koji zajedno čine stanicu, najsličnije stanici koja je u laboratoriju CRTA-e, odnosno, Regionalnog centra izvrsnosti za robotske tehnologije, u kojem je validiran završni rad. Ubačeni dijelovi koji se koriste su pod, stol na kojem se nalazi postolje za robot UR5e, robot s hvataljkom, spužva na kojoj se nalazi dio za praćenje krivulje te kalibracijski šiljci.



Slika 26. Izgled stanice u RoboDK

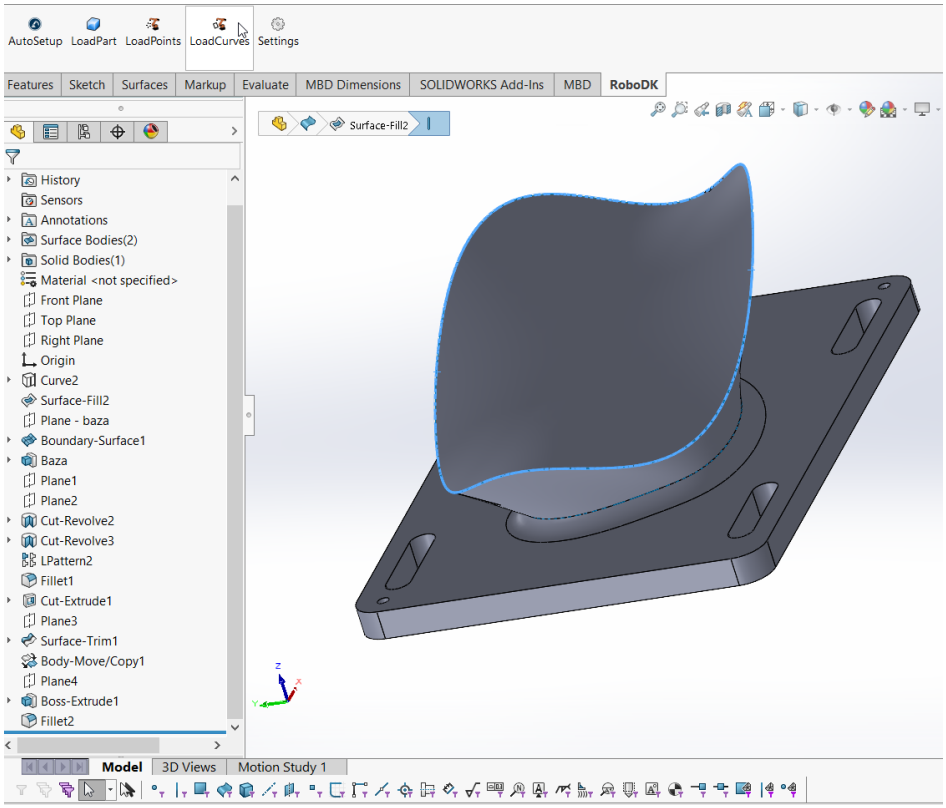
5.1. Curve follow project

Klikom na *Utilities* na alatnoj traci otvara se padajući izbornik gdje je potrebno odabrati naredbu *Curve follow project*. Navedena naredba koristi se za praćenje krivulje, te se otvori cijeli dodatan prozor u kojemu je moguća konfiguracija i postavke za praćenje.



Slika 27. „*Curve follow project*“ postavke

Postavke kreću od odabira robota. Obzirom da je u RoboDK moguć rad s više robota, odnosno da se stanica za rad sastoji od više robota koji istovremeno mogu izvršavati neki rad, tako je za ovu naredbu potrebno odabrati robota. Odabran je jedini robot koji se nalazi u stanici, UR5e. Referenca za koordinatni sustav odabran je *Frame_Part*, to je koordinatni sustav u kojem se nalazi 3D dio, koji prati navedeni robot. Iduće na redu osnovnih postavki je odabir alata, za koji je odabrana hvataljka PZN-plus 200-1. Zatim su na redu postavke za samo krivulju, koju je potrebno odabrati kako bi robot znao koju krivulju treba pratiti. Krivulja je ubačena iz SolidWorks-a, unutar kojeg je aktiviran *Add In* za RoboDK. Klikom na *Load Curves* i označivanjem ruba kojeg želimo da robot prati, krivulja se ubacuje u RoboDK. Isti postupak potrebno je napraviti za ubacivanje 2D krivulje za njezino praćenje.



Slika 28. Odabir krivulje za ubacivanje u RoboDK.

U postavkama za krivulju, iduće na redu su opcije za prilaz i povlačenje od krivulje. Moguće je odabrati na kojoj udaljenosti korisnik želi da da se robot s vrhom alata približava samoj krivulji i pokretanju izvedbe simulacije. Također, odabirom na izbornik *type*, moguća je izmjena načina prilaza te je moguća izmjena kuta pod kojim korisnik želi pristupiti krivulji. Sve te izmjene i opcije dostupne su i za povlačenje od krivulje nakon izvedbe naredbe za praćenje krivulje, odnosno, nakon što robot od prati cijelu krivulju i izvrši program. Također, moguća je izmjena brzina kojom je moguće izvršiti program za praćenje. Sljedeće su postavke za alat i izvedbu same simulacije. *Path to tool offset* označuje udaljenost krivulje od alata. Udaljenost je moguće odrediti kao translacija po bilo kojoj od osi, kao i nagib za neki egzaktni proizvoljni kut koji korisnik želi.

Jedan od najbitnijih postavki jeste algoritam po kojem sve robot može pratiti krivulju koja je prethodno odabrana. Prema zadanim postavkama, RoboDK koristi algoritam za minimalnu promjenu orijentacije alata. U ovom radu korišten je algoritam *Minimum tool orientation change*. To znači da se orijentacija alata održava što je moguće konstantnijom duž putanje alata. Općenito, ova je opcija prikladna za operacije kao što je glodanje, jer minimizira pokrete zglobova robota. Naravno, dostupni su i neki drugi algoritmi za praćenje krivulje, a jedan od njih

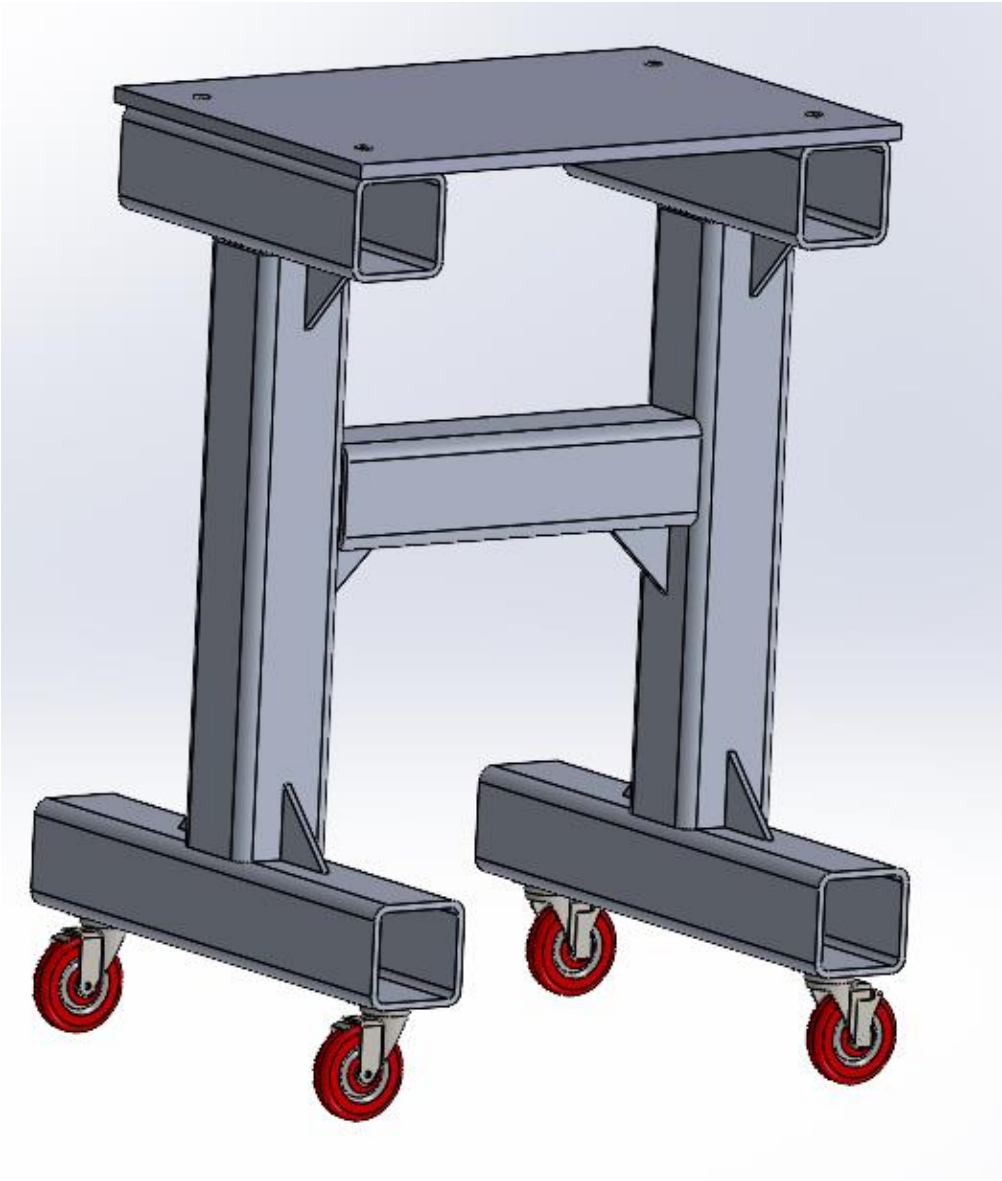
je *Tool orientation follows path*. To je algoritam u kojem orijentacija alata slijedi putanju alata (korisno za primjene rezanja, gdje oštrica mora ostati tangentna duž putanje). Idući mogući algoritam je *Robot holds object*, odnosno, robot koji drži objekt ako se putanja alata mora pratiti pričvršćena na robotski alat (na primjer, za aplikacije nanošenja ili poliranja.). Posljednji mogući algoritam je kombinacija zadnja dva navedena, a to je *Robot holds object and follows path,* gdje robot i drži objekt te istovremeno i slijedi putanju.

Parametri optimizaciji jedni su od posljednjih postavki koje je potrebno urediti prije pokretanja programa za praćenje krivulje. U tom dijelu, s obzirom na željenu orijentaciju alata, robot može imati određenu slobodu okretanja oko Z osi alata. To omogućuje robotu da izvrši program izbjegavajući singularnosti robota, ograničenja zglobova i osiguravajući da su sve točke dostupne. Također, RoboDK automatski odabire početnu konfiguraciju koja je najbliža željenim početnim zglobovima. Od te točke, nadalje, robot će pratiti linearna kretanja.

Nakon što se podese svi od navedenih parametara, potrebno je kliknuti na dugme *Update*, kako bi se ažurirale sve postavke. Naravno, prilikom bilo kakvih parametara i postavki, prije izvršavanja simulacije, potrebno je kliknuti navedeni gumb, kako bi se osvježila simulacija. Pored tog gumba nalazi se još jedan, opcija *Simulate*, te pritiskom na njega, pokreće se simulacija i ona je vidljiva na ekranu unutar programa RoboDK. Klikom na opciju *Simulate,* automatski se kreiraju u stablu s lijeve strane, dvije stavke. Prva od njih su postavke za praćenje krivulje, te klikom na te postavke, se otvara ponovno isti prozor te je u njemu moguća izmjena bilo kakvih parametara. Iduće što se pojavi u stablu je program, na čiji dvostruki klik se pokreće simulacija. Također, isti taj program je moguće ubaciti u neki drugi program kao pod program, te je taj program moguće prebaciti na robot kao i njegovo izvođenje na samom robotu.

5.2. Mobilno postolje

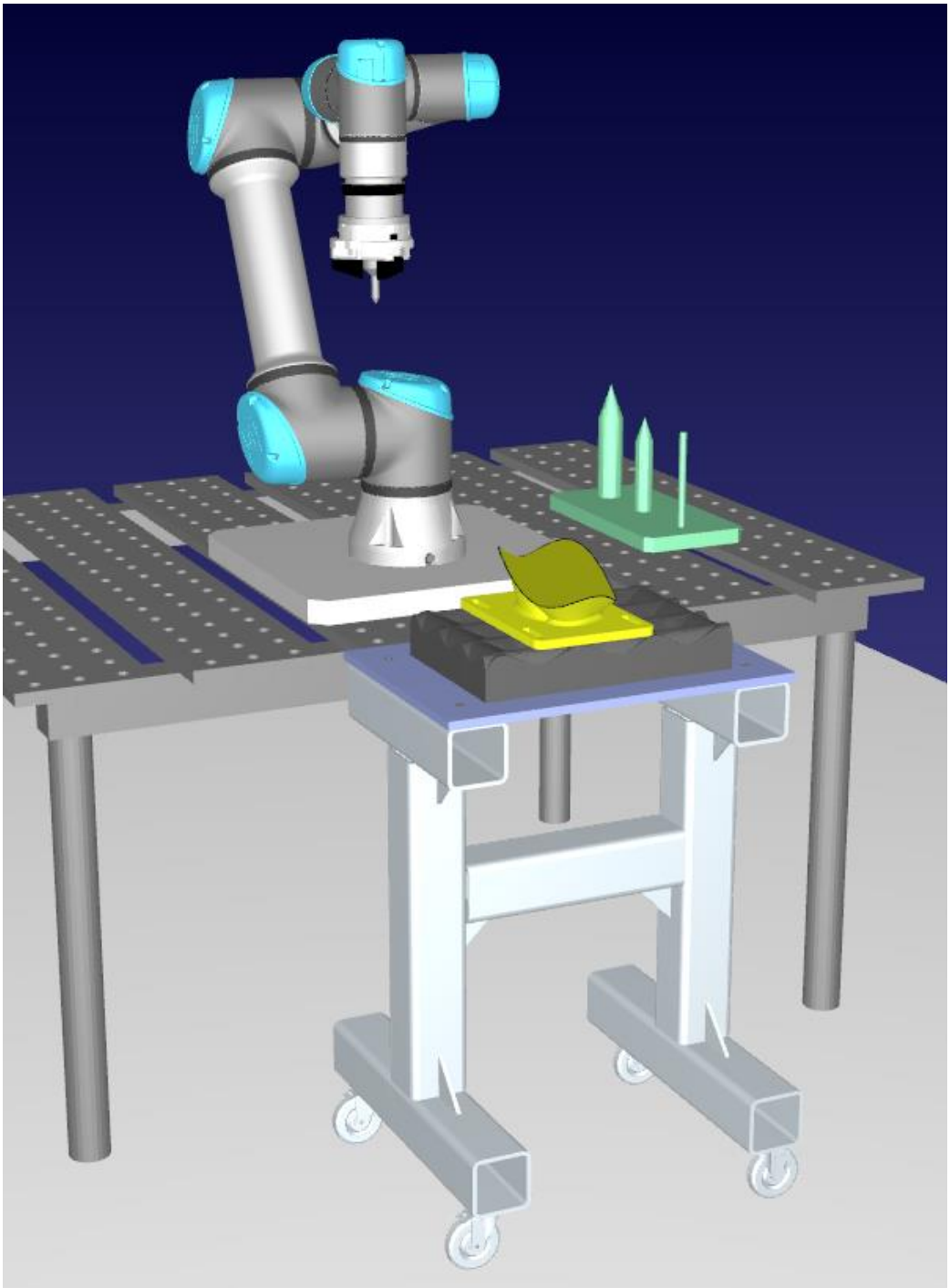
Unutar simulacije korišteno je mobilno postolje na kojem se nalaze fiksirani dijelovi na kojima su krivulje koje robot s vrhom alata prati. Koncept mobilnog postolja napravljen je u SolidWorks-u i prikazan je na slici. Svrha mobilnog postolja je da je ono jednostavno, no opet robusno, kako bi se moglo pomicati po laboratoriju, te kako bi bilo koji robot mogao pratiti navedene krivulje.



Slika 29. Koncept mobilnog postolja.

Ovakav koncept mobilnog postolja moguće je napraviti i u stvarnosti. Profili za postolje mogu se napraviti od aluminija, ali i od nekog nehrđajućeg čelika. Spajanje je moguće zavarivanje, dok se kotači montiraju vijcima na donji profil. Ploča je s profilima povezana s četiri vijka s upuštenom

glavom. Dimenzije su usklađene s visinom stola na kojem se nalazi robot, tako da robot ima doseg i radni obujam da može izvršiti svaku naredbu nad dijelom koji se nalazi na postolju. Ploča na koju se stavlja spužva zbog sigurnosti i dio za praćenje krivulje ne mora biti velike debljine, zato što nije potrebna velika nosivost. Također, na kotače je moguće staviti mehanizam kako bi se kotači mogli zakočiti, samim time bi i postolje bilo stabilno.



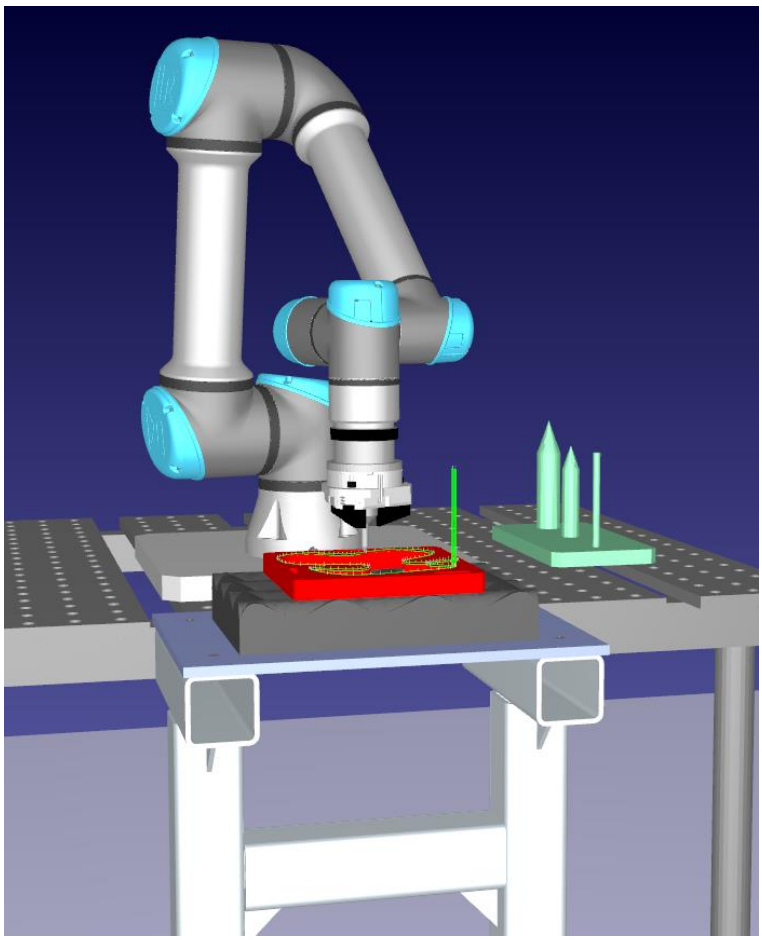
Slika 30. Mobilno postolje prikazano u RoboDK.

5.3. Izvođenje simulacije

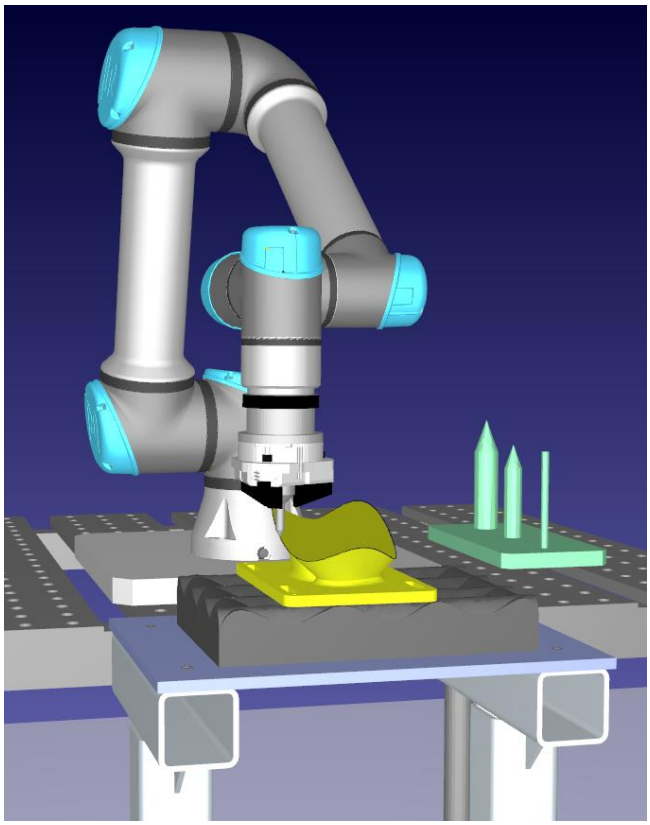
Za izvođenje kompletne simulacije, kreiran je novi program *Main_Program.* Pozivom tog programa, pokreće se simulacija, te je isti taj program prenesen na robot UR5e. Program se sastoji nekoliko dijelova koji zajedno čine glavni program. Prvi od njih je *Move Joint* kretanja, gdje robot odlazi za početne pozicije do pozicije prilaska, odnosno, u poziciji iz koje se linearno spušta prema dijelu čiju krivulju prati. Također, potrebno je definirati neke postavke programa, koje se automatski generiraju. Prva naredba, odnosno parametar je *Smooth*, koji omogućuje da robot izvrši kretanje kroz više točaka bez zaustavljanja u svakoj točki Također, postavljena je i brzina izvođenja, koja je podešena na 200 mm/s. Za referentni koordinatni sustav odabran je koordinatni sustav baze robota, a za alat je odabrana hvataljka PZN-plus 200-1. U program je moguće dodati i druge naredbe, a sve od njih vidljive su na slici. Osim dodavanja programa i gibanja, koji su već dodani u ovom programu u ovom završnom radu, moguće je dodavanje poruke kao instrukcije, dodavanje pauze, brzina izvođenja simulacije, kao i rad s ulaznim i izlaznim signalima. Nakon tih parametara, pozvan je pod program za praćenje krivulje, odnosno, program koji je ranije definiran. Po završetku pod programa, dodana je naredba da se robot vrati u početnu poziciju.

U sklopu izvođenja simulacije u RoboDK, ispitana su tri postava. Prvi od njih je praćenje 2D krivulje. 2D krivulja nalazi se na dijelu koji je napravljen u SolidWorks-u, te je kao .step file prebačen u RoboDK, dok je u laboratoriju printan 3D tehnologijom te je na njemu izvršena validacija. Normale na *Part-u* nalaze se okomito na ravninu i na krivulju. Algoritam koji je odabran za praćenje krivulje je *Minimum tool orientation change*. Zelenom bojom na slici je prikazan put kojim robot ide i slijedi krivulju, odnosno putanju.

Iduća simulacija je praćenje 3D kompleksnog oblika. 3D krivulja koju robot prati nalazi se na rubu *Part-a*. Krivulja koja je generirana iz SolidWorks-a s *Part-a,* također sadrži normale koje su okomite na podlogu, te robot slijedi krivulju tako da je vrh alata konstanto okomit na tu krivulju. U ovoj simulaciji, također je korišten već spomenuti algoritam *Minimum tool orientation change*. Na slici [31.] je prikazano kako hvataljka prati krivulju na rubu, te su normale krivulje okomite na površinu, kao i na vrh alata.

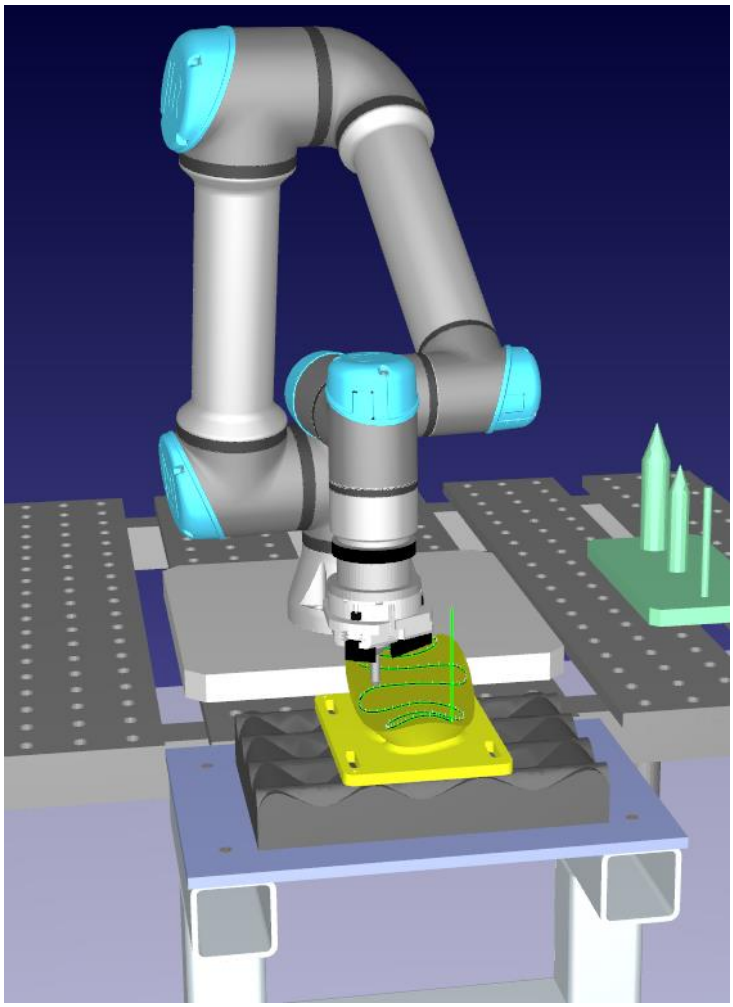


Slika 31. Prikaz simulacije za vrijeme praćenja 2D krivulje sa okomitim normalama.



Slika 32. Simulacija praćenja 3D krivulje.

Za treći primjer praćenja krivulje, napravljena je simulacija u programu tako da robot sa vrhom svog alata prati površinu 3D dijela. Krivulja za praćenje koju prati robot, generirana je u Solidworks-u, te kao takva je napravljena po svim ne ravninama i obrubima kako bi pratila zakrivljenu plohu.



Slika 33. Simulacija praćenje 3D krivulje na plohi.

6. VALIDACIJA U LABORATORIJU

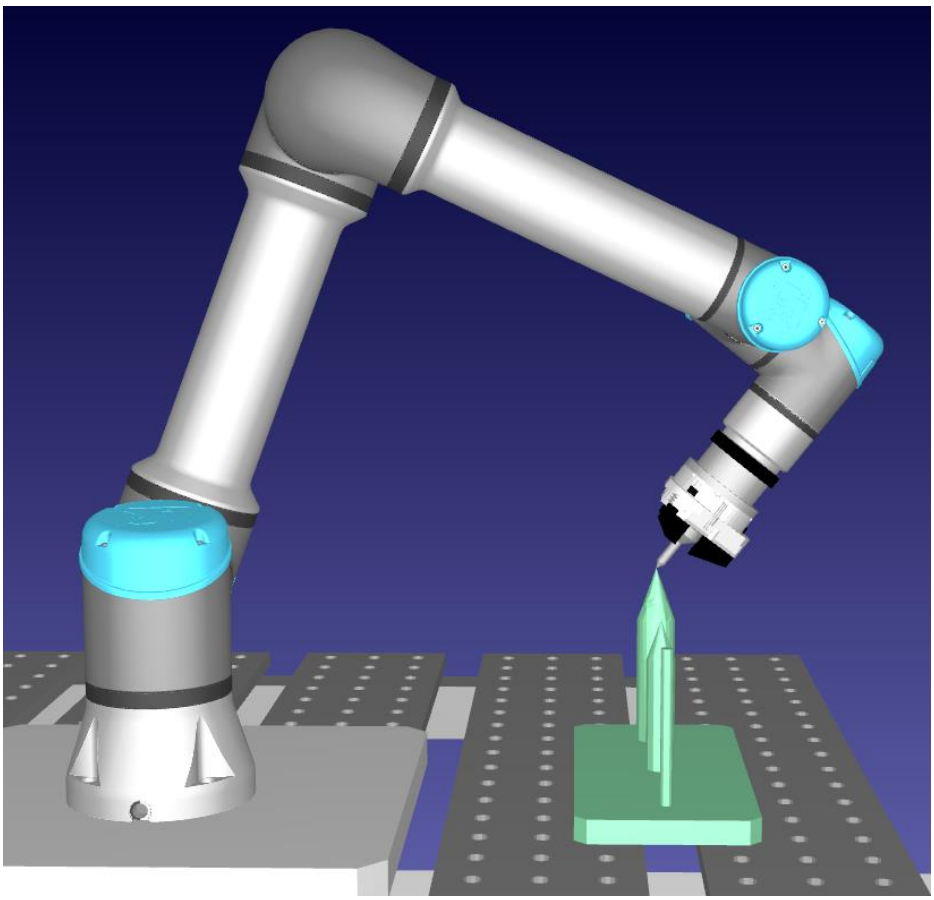
Po završetku spajanja robota iz laboratorija s računalom te kreiranja simulacije u RoboDK programskom paketu, potrebno je simulaciju prenijeti na robot te ju validirati u laboratoriju na stvarnom robotu sa stvarnim dijelovima u stvarnom okruženju.

6.1. Kalibracija

Za početak je potrebno napraviti kalibraciju alata, te kalibraciju koordinatnih sustava oblika po kojem robot prati krivulju. Kalibracija alata u kontekstu robotike i automatizacije je postupak usklađivanja koordinatnih osi softverskih modela u simulaciji, sa stvarnim modelima koji se koriste u laboratoriju u stvarnom okruženju. Ovaj proces je krucijalan za garanciju preciznosti i točnosti robotskih operacija. Kalibracija omogućava robotu da zna točnu poziciju i orijentaciju alata u svom radnom prostoru, kao i modela, što je neophodno za precizno manipuliranje objektima, izvođenje zadataka kao što su zavarivanje, rezanje, ili montaža, te za optimalno iskorištavanje robotskog sustava.

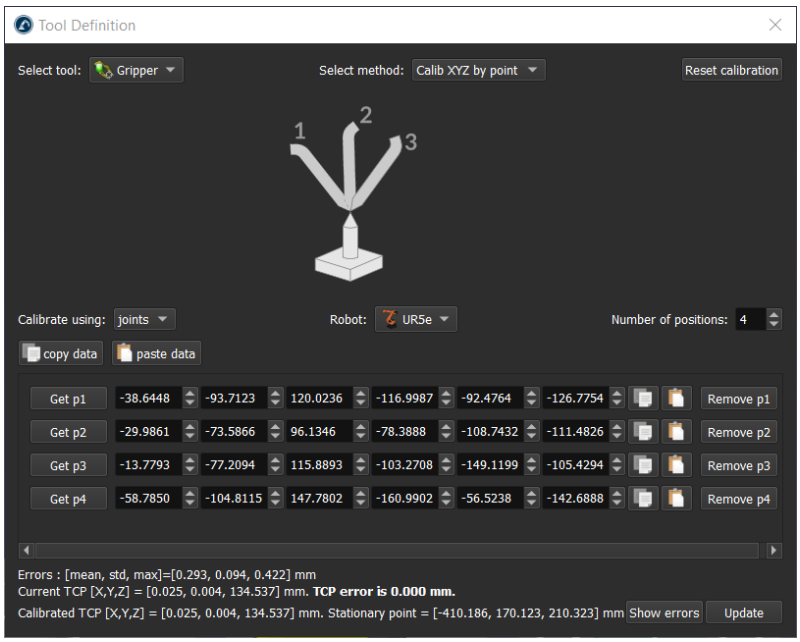
6.1.1. Kalibracija koordinatnog sustava robotskog alata

Prvenstveno je potrebno napraviti kalibraciju alata (TCP - *Tool Center Point*). TCP je specifična točka na vrhu alata koja se koristi kao referenca za sve operacije koje se izvode. Precizno definiranje TCP-a je ključno za uspješnu kalibraciju, te za izvođenje svih radnji na robotu. Matrica homogene transformacije alata određena je nepromjenjivo u odnosu na bazni koordinatni sustav robota. Za kalibraciju alata, odnosno TCP, potrebno je alat približiti nekoj fiksno definiranoj točki s više pozicija. Konkretno, u ovom radu, kalibracija alata, odnosno šiljka, napravljena je tako da je taj isti šiljak kalibriran u odnosu na drugi kalibracijski šiljak koji je postavljen fiksno. Alat je pristupljen s četiri različite strane, s četiri različita nagiba, kako bi definiranje alata bilo što preciznije, naravno, kako bi greška bila što manja. Na slici [34.] dan je prikaz kalibracije alata u simulaciji unutar RoboDK programskog paketa. Dakle, vrh alata prilazi kalibracijskom šiljku, te se u odnosu na taj fiksno postavljeni šiljak određuju četiri točke, i sve te 4 točke se snime kao pozicija u RoboDK.



Slika 34. Kalibracija alata TCP-a.

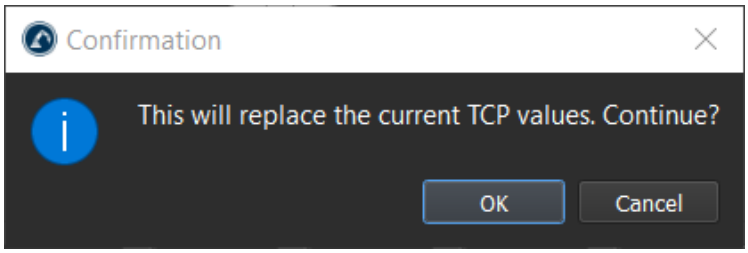
Nakon što budu snimljene 4 točke, potrebno je definirati TCP unutar samog RoboDK. Za kalibraciju alata u RoboDK potrebno je kliknuti na alatnoj traci na *utilities* i nakon toga u padajućem izbornika kliknuti na stavku *Define Tool Frame (TCP*). Otvara se dodatan prozor sa svim postavkama koje je potrebno prilagoditi za kalibraciju.



Slika 35. Kalibracija TCP-a u RoboDK.

Za početak je potrebno definirati za koji alat se vrši kalibracija. Jedini alat u ovoj simulaciji je *Gripper*, pa je stoga potrebno odabrati taj. Za kalibraciju je moguće odabrati dvije metode, *Calib XYZ by Points* i *Calib XYZ by Plane*. Kao što je ranije napomenuto da će se kalibracija raditi preko točaka, potrebno je odabrati metodu *Calib XYZ by Points*. Isto tako potrebno je odabrati da će se kalibracija raditi preko točaka, a ne preko poza. Za robot je odabran UR5e. Također, postoji i mogućnost gdje je moguće odabrati s koliko točaka se radi kalibracija, naravno, što više točaka je odabrano za kalibraciju, to će ona biti točnija. Za potrebe ovog rada napravljena je kalibracija s 4 točke. Za kraj je potrebno uskladiti točke koje su snimljene oko kalibracijskog šiljka, s točkom kalibracija, odnosno p1, p2, p3 i p4, tako da se robot postavi u točku, te zatim klikne na ikonu *Get p1,* i tako za sve četiri točke.

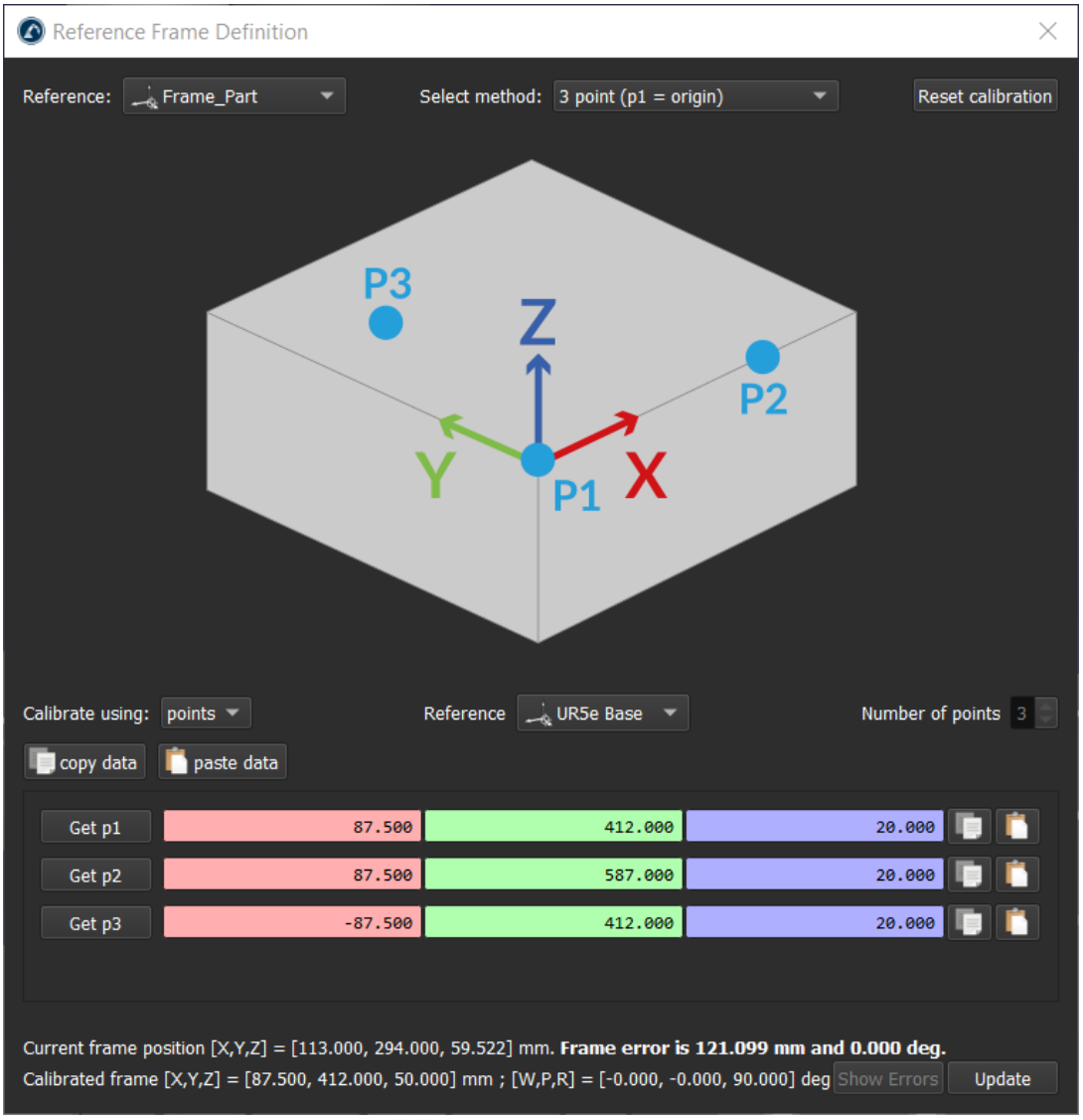
Po završetku svih navedenih postavki, potrebno je kliknuti na *update*, nakon kojeg se pojavi notifikacija za potvrdu. Nakon što se potvrdi sa *OK*, alat je potpuno kalibriran i spreman za korištenje.



Slika 36. Potvrda za kalibraciju alata.

6.1.2. Kalibracija koordinatnog sustava dijelova za praćenje krivulje

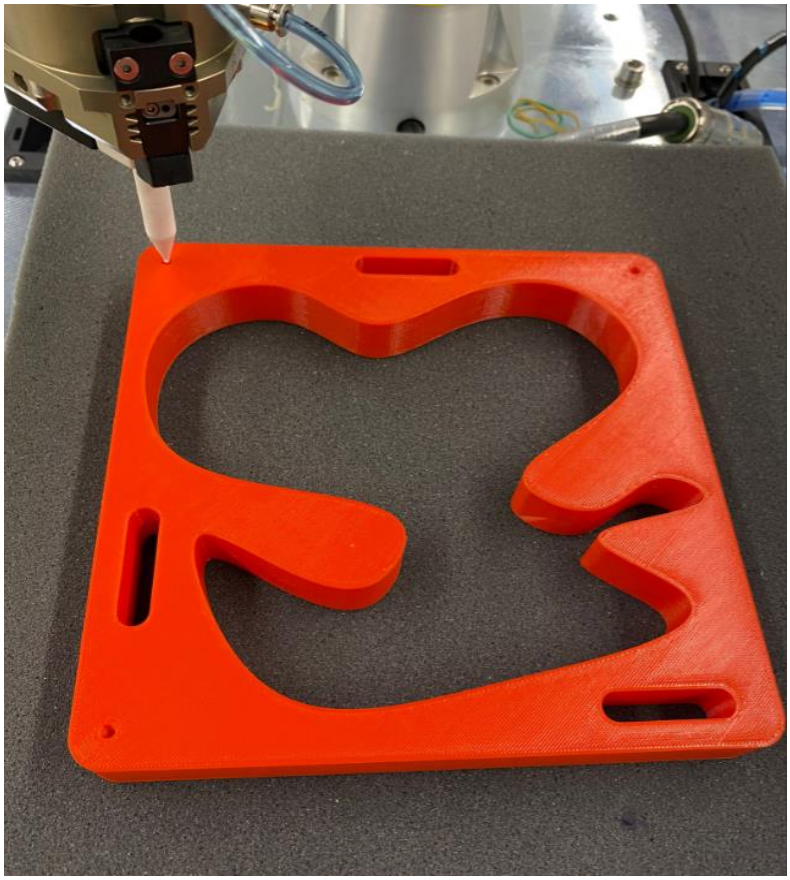
Nakon kalibracije alata, potrebno je kalibrirati i koordinatne sustave dijelova na kojim se nalaze krivulje koje robot prati. Kalibracija je potrebno iz razloga kako bi robot precizno znao gdje točno u prostoru se nalazi dio, kao i krivulja koju prati. Za kalibraciju koordinatnog sustava je potrebno kliknuti na opciju *utilities* na alatnoj traci, te odabrati *Define Reference Frame (User Frame)*.



Slika 37. Kalibracija koordinatnog sustava dijela.

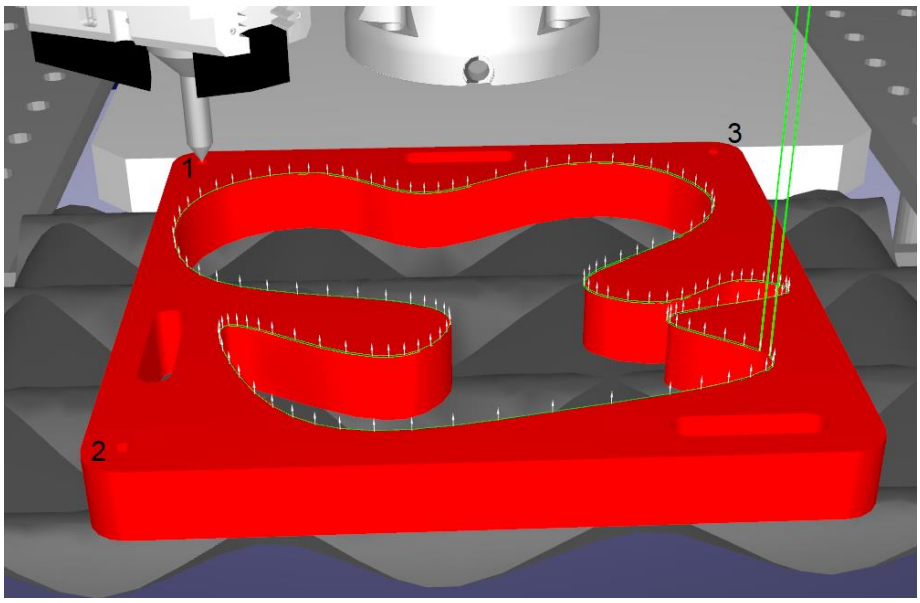
Nakon otvaranja prozora za definiranje koordinatnog sustava, potrebno je odabrati referencu, u ovom slučaju je to novo kreirani koordinatni sustav, koji će se vezati na dio za praćenje krivulje. S obzirom na to da postoje 4 metode za kalibraciju, potrebno je odabrati jednu od njih. U ovom radu korištena je metoda *3 point (p1 = origin)*. Kao što i sam naziv govori, potrebno je odabrati 3 točke, prema kojem će se tada kreirati koordinatni sustav. Također, bitno je odabrati način kalibracije, preko točaka ili preko zglobova. Potrebno je odabrati kalibraciju preko točaka. Iz tog razloga je bilo izuzetno potrebno napraviti ispravnu kalibraciju alata u prethodnom poglavlju. Iduće na redu je snimanje točaka za kreiranje koordinatnog sustava.

Bitna stavka za stvaranje koordinatnog sustava, odnosno kalibraciju, jest da sam robot na kojem se provodi kalibracija, u ovom slučaju UR5e, bude spojen sa simulacijom sa RoboDK. Dakle, potrebno je postaviti robot u laboratoriju, na prvu točku za kalibraciju, ta točka je prema definiciji ishodište koordinatnog sustava.



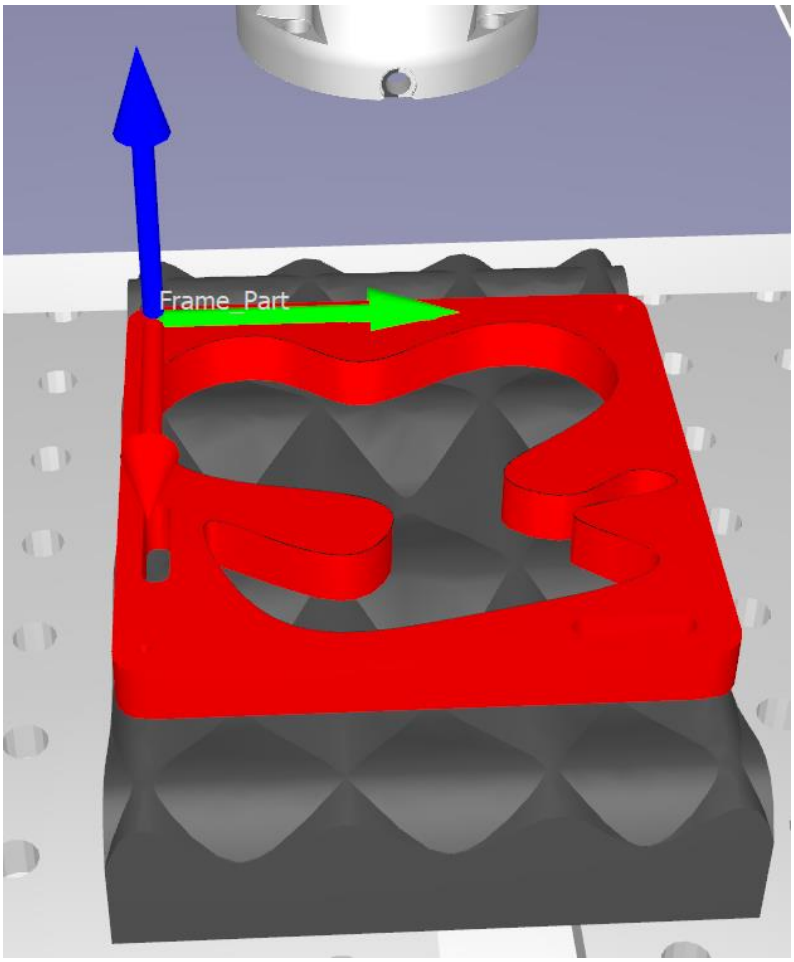
Slika 38. Točka ishodišta koordinatnog sustava dijela u stvarnom okruženju.

Nakon što je ovako postavljen alat u laboratoriju, tada je potrebno u RoboDK odabrati opciju *Get position*, tada se robot postavi u točno tu udaljenost u simulaciji, kako je i on u stvarnom okruženju u laboratoriju. Tu točku potrebno je snimiti i kreirati, ta točka kasnije će biti korištena za kreiranje odnosno kalibraciju koordinatnog sustava.



Slika 39. Točka ishodišta koordinatnog sustava u RoboDK.

Isti je postupak potrebno napraviti za preostale dvije kalibracijske točke. Prvenstveno je potrebno definirati vektor X između točke 1 i 2. Zatim je potrebno definirati točku 3, koja će biti u smjeru y osi.

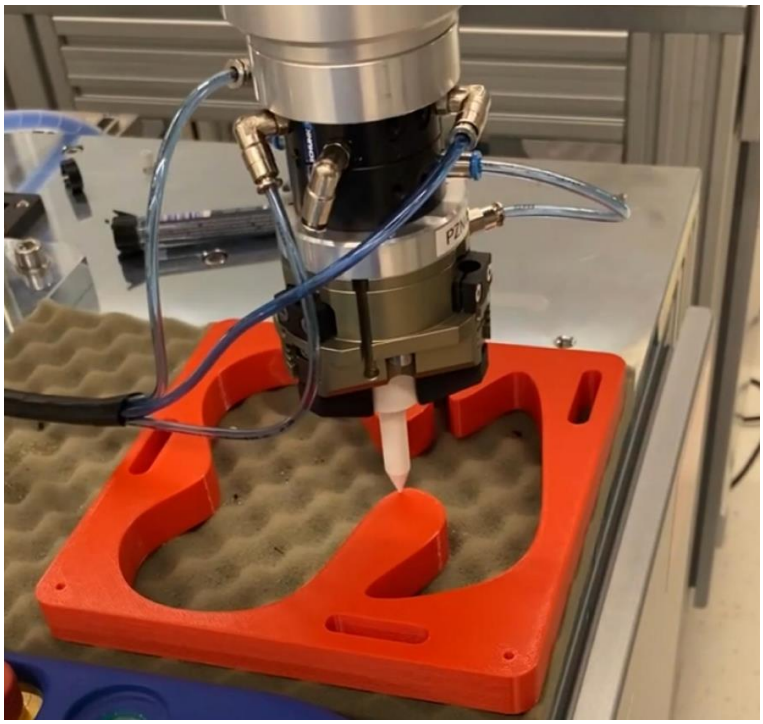


Slika 40. Definiran koordinatni sustav dijela.

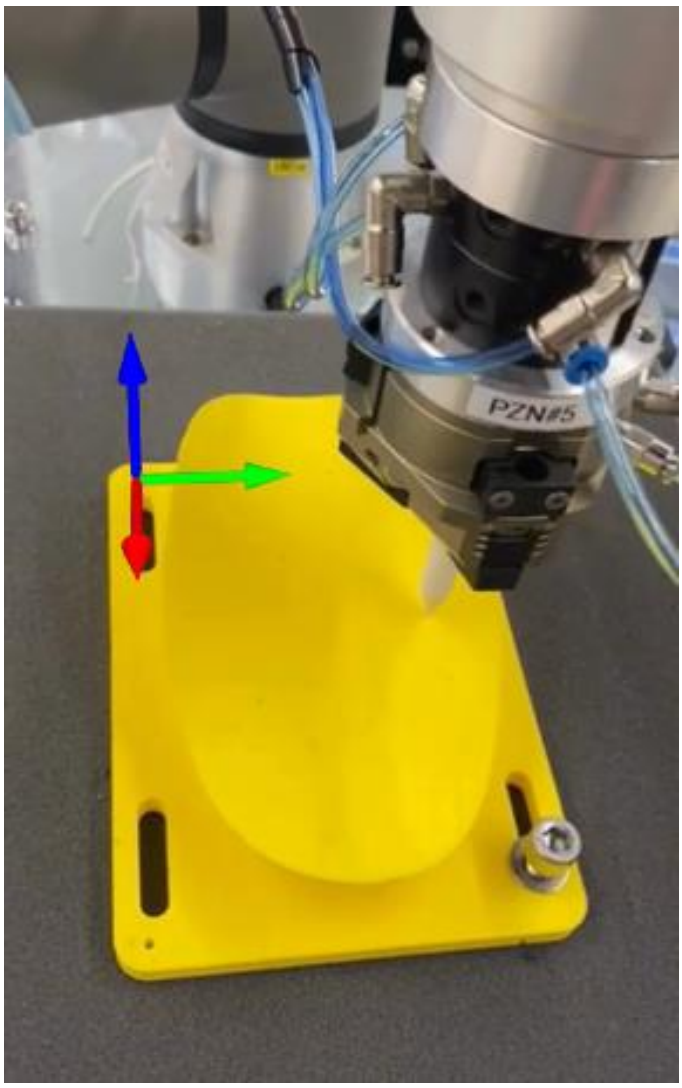
Nakon definiranja koordinatnog sustava, on će imati svoju poziciju i orijentaciju u prostoru. Na slici iznad je prikazan definiran i kalibriran koordinatni sustav. Tek kada je to napravljeno, tek tada su povezani i kalibrirani svi potrebni koordinatni sustavi i dijelovi te je moguće simulaciju napravljenu u RoboDK, prebaciti na stvarni robot i izvesti u laboratoriju na stvarnom okruženju. Potpuno identičan postupak, sa svim koracima, potrebno je napraviti i za 3D dio za praćenje krivulje, s obzirom na to da je on druge geometrije.

6.2. Izvođenje u laboratoriju

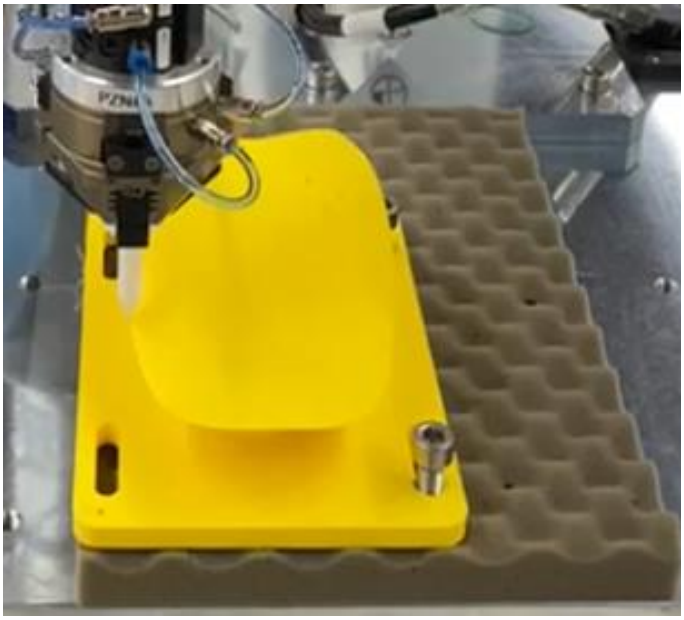
Nakon što su svi prethodni koraci odrađeni, od povezivanja robota sa simulacijom, preko izrade simulacije pa sve do kalibracije u stvarnom okruženju, slijedi izvedba i validacija u laboratoriju na stvarnom robotu. Robot je uspio odraditi sva tri od ispitanih okruženja, i praćenje 2D krivulje, kao i praćenje 3D krivulje.



Slika 41. Praćenje 2D krivulje u laboratoriju.

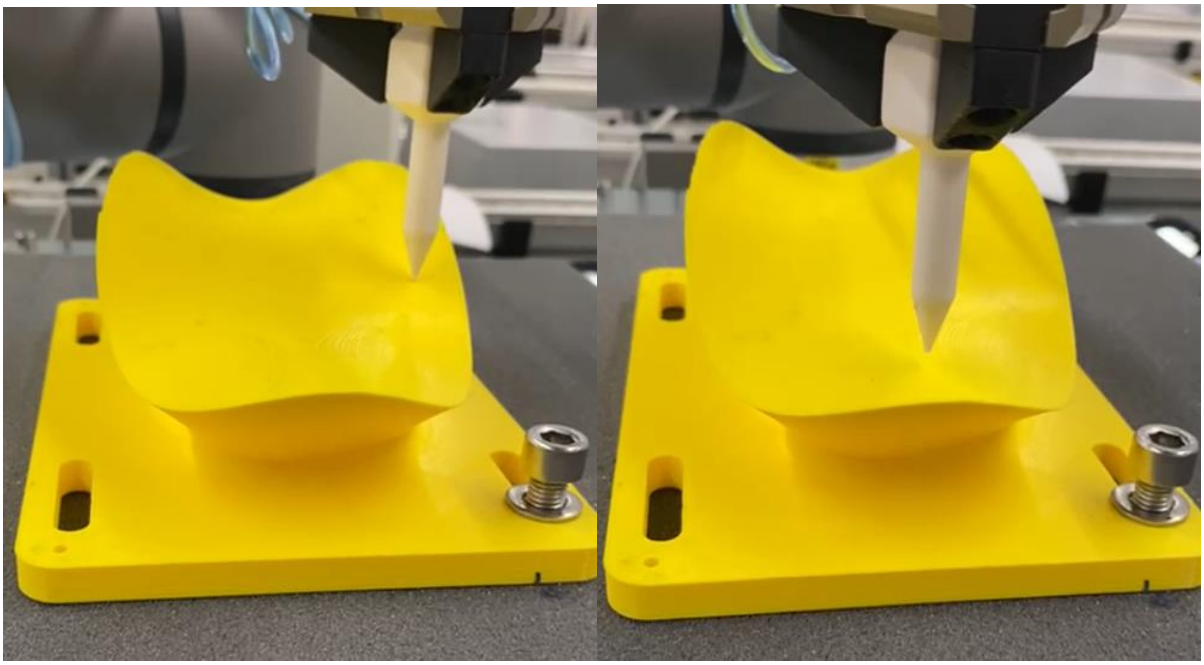


Slika 42. Praćenje po plohi u laboratoriju.



Slika 43. Praćenje 3D krivulje u laboratoriju.

Kod praćenja 2D krivulje, greške nije bilo, dok je greška kod praćenja 3D krivulje bila minimalna. Najveća greška bila je uočljiva kod praćenja krivulje po plohi. Zaključak je taj da što je krivlja kompleksnija, greška je veća. Robot će u stvarnosti i dalje izvršiti praćenje te krivulje, no postojat će mjesta gdje će se robot podići po osi Z u odnosu na simulaciju, na sami *part* i na samu krivulju.



Slika 44. Usporedba greške kod praćenja krivulje po plohi u laboratoriju.

Na lijevoj strani slike vidljivo je kako na jednom dijelu robot sa vrhom svog alata skoro pa savršeno prati plohu, dok je na desnoj strani slike vidljivo kako je robot sa vrhom alata ipak podignut u odnosu na plohu i krivulju koju prati.

7. ZAKLJUČAK

U završnom radu predstavljeno je upravljanje industrijskim robotom UR5e korištenjem programskog paketa RoboDK. Robot je programiran offline, što znači da su svi koraci programiranja izvedeni na računalu prije implementacije na robotu. Prije simulacije i realizacije u laboratoriju, detaljno je opisan proces kreiranja dijelova za praćenje krivulje, kao i komponente koje su korištene u ovom radu. Simulacijski prostor omogućuje izradu kompletnog proizvodnog procesa i evaluaciju različitih robotskih konfiguracija prije stvarne nabave opreme ili robota.

Unutar RoboDK programskog paketa, razvijena su tri scenarija koji su simbolički predstavljali stvarne postavke u laboratoriju. Korištenjem funkcionalnosti Curve follow project, uspješno je ostvareno praćenje svake od tri zadane krivulje, uz sitnu grešku, pokazujući da je robot UR5e pogodan za ovakve vrste projekata. Prednost korištenja simulacija leži u mogućnosti predviđanja različitih scenarija i zadataka koje bi robot trebao izvršiti u stvarnom okruženju, kao i identifikaciji potencijalnih problema, opasnosti i područja za poboljšanje.

Dodatna prednost RoboDK programskog paketa je mogućnost uvoza CAD modela u .step i .stl formatima, što je omogućilo besprijekornu integraciju i vizualizaciju. U budućim istraživanjima na temu ovog završnog rada, moguće je unaprijediti ga dodavanjem kamera i korištenjem tehnologije računalnog vida.

LITERATURA

[1.] https://www.themanufacturer.com/articles/bmw-places-major-order-for-5000-industrial-robots/ , pristupljeno 2.2.2024.

[2.] Doc. dr. sc. Marko Švaco, Podloge za vježbe - Programiranje robota i izrada simulacijskih robotskih programa u programskom paketu RoboDK

[3.] https://robotsguide.com/robots/universal , pristupljeno 4.2.2024.

[4.] https://www.universal-robots.com/products/ur5-robot/ , pristupljeno 22.1.2024.

[5.] https://schunk.com/de/en/gripping-systems/centric-grippers/pzn-plus/pzn-plus-200-1/p/000000000000303315 , pristupljeno 23.1.2024.

[6.] https://robodk.com/doc/en/Basic-Guide.html , pristupljeno 7.1.2024.

PRILOZI

I. OneDrive - Završni rad - https://fsbhr-my.sharepoint.com/personal/ab229332_fsb_hr/_layouts/15/onedrive.aspx?id=%2Fpersonal%2Fab229332%5Ffsb%5Fhr%2FDocuments%2FZavr%C5%A1ni%20rad&ga=1

II. Github - https://github.com/abuni/Zavrsni_rad