

Beschrijving bij het derdejaars project

De Pizzaschuiver



M.J.G. van de Molengraft

R.J.E. Merry

J.J.T.H. de Best

M.G.J.M. Maassen

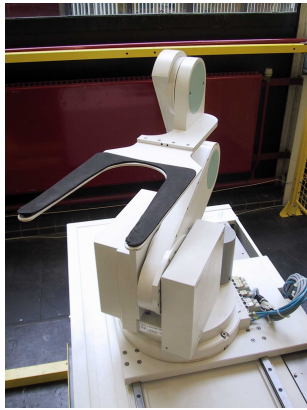
1 september 2006

<http://www.dct.tue.nl/Pizza.htm>

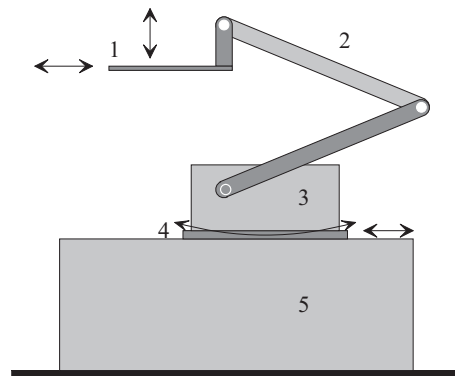
Regeling van een industriële robot

De robot

De industriële robot die tijdens de casus zal worden gebruikt is afkomstig van Philips. Bij Philips werd de robot gebruikt voor het verplaatsen van LCD-schermen. Door verplaatsing van de divisie naar het buitenland is de robot overbodig geworden en kan hij voor dit project worden ingezet. In figuren 1 en 2 zijn een foto en een schematische weergave van de robot te zien. Bij de schematische weergave zijn de vrijheidsgraden van de robot aangegeven. De motoren voor de horizontale en verticale verplaatsing van de vork (1) zijn bevestigd aan onderdeel (3). De assen van het stangenmechanisme (2) worden niet actief aangestuurd, maar dienen slechts als een mechanische overbrenging. Verder is er nog een motor aanwezig voor de rotatie van onderdeel (3) ten opzichte van de tafel (4) en een motor voor de translatie van de tafel (4) ten opzichte van het frame (5).



Figuur 1: foto van de robot



Figuur 2: schematische weergave van de robot

Positiemetingen vinden plaats met behulp van encoders. Deze encoders zijn bevestigd aan de assen van de motoren. De relatie tussen de rotatie van de motor en de verplaatsing van de vork is lineair. Tijdens experimenten is deze lineaire relatie al in de metingen verwerkt.

In tabel 1 zijn de uiterste standen van de verschillende vrijheidsgraden weergegeven. Deze posities zijn weergegeven ten opzichte van een initieel punt $[0,0,0,0]$ voor elke motor.

<i>Vrijheidsgraad</i>	<i>Minimale positie</i>	<i>Maximale positie</i>
Verticale verplaatsing vork	-0.2418 [m]	-0.0268 [m]
Horizontale verplaatsing vork	0.0231 [m]	0.5301 [m]
Rotatie onderdeel 3	0.0108 [rad]	5.5708 [rad]
Translatie tafel	-0.0106 [m]	0.5294 [m]

Tabel 1: uiterste standen van de verschillende vrijheidsgraden

Door de ontwikkeling van het TUEDACs systeem is het mogelijk om metingen aan de robot uit te voeren met een notebook en ook om de robot aan te sturen met een notebook. Voor de aansturing van de robot zijn 2 TUEDACs-QAD's nodig. Deze QAD's zijn aangesloten op een versterkerkast, die gekoppeld is aan de motoren en de encoders op de robot.

Doel van het project

In dit project zal de robot worden gebruikt voor het verplaatsen van pizza's. Aan beide zijden van de robot zijn rekken geplaatst, waarbij de pizza's van het ene rek naar het andere rek dienen te worden verplaatst. De bedoeling is dat dit zo snel en nauwkeurig mogelijk gebeurt. Aan de hand van experimenten zul je zelf moeten ontdekken wat hierbij de mogelijkheden zijn (systeemidentificatie) en met die kennis zul je een pad moeten definiëren, kwantificeren en vervolgens je doel stellen.

Systeemidentificatie

Om een goede regelaar te ontwerpen heb je kennis van het te regelen systeem nodig. Deze kennis doe je op door het verrichten van metingen en het doen van experimenten. Bij identificatie valt te denken aan identificatie in het tijd- en in het frequentiedomein (overdrachten). Bedenk van tevoren wat je graag wilt weten van het systeem en vooral ook hoe je dit kunt meten. Voer hiervoor simulaties uit, b.v. aan de hand van een eenvoudig model van het systeem. Implementeer na de simulaties je simulatiemodel pas op het echte systeem. Matlab commando's die handig kunnen zijn bij de systeemidentificatie bevinden zich in de Matlab/signal processing toolbox. Enkele voorbeelden zijn o.a.: `cohere.m` en `tfe.m`.

Benodigheden

Hardware:

- Notebook met FAT32 partitie of FAT32 memory stick
- Een FAT32 partitie kun je indien nodig laten maken bij de ICT servicedesk
- TUEdAX Linux Live CD v1.18 (te lenen bij Harrie van de Loo), met o.a.
 - Matlab 6.1 (R12.1) met Simulink en Real-Time Workshop
 - Wintarget
 - Simulink model met de referentie generator (`ref3ma_test.mdl`)
 - Simulink model voor de aansturing (`empty_exp.mdl`)
 - Simulink model voor simulaties (`empty_sim.mdl`)

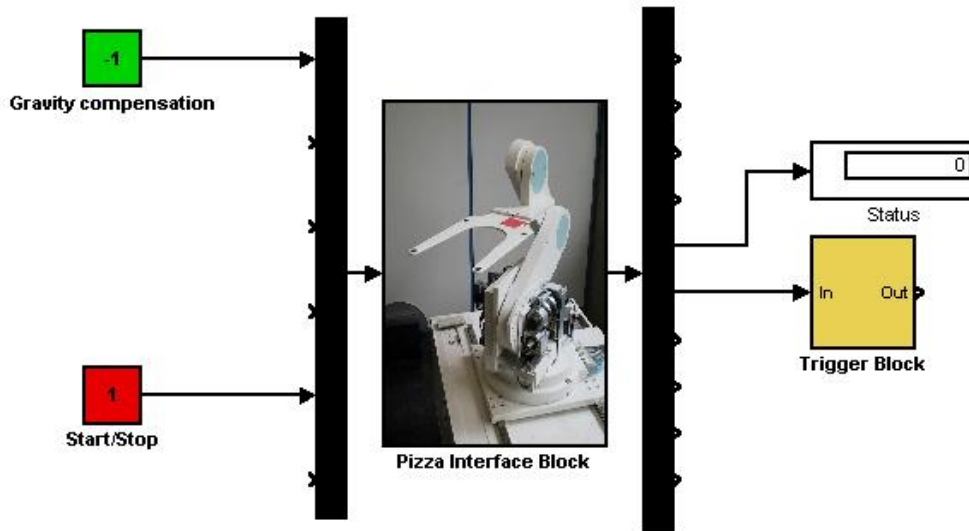
De laatste informatie betreffende het project is te vinden op de internetpagina van het project (<http://www.dct.tue.nl/Pizza.htm>). Met de referentiegenerator (Ref3) kunnen snel en eenvoudig 3^e-graads referentie trajectories worden gemaakt. Om een experiment te kunnen uitvoeren, moet het Simulink model eerst worden gecompileerd met behulp van RTW om er een Real-Time Applicatie (RTA) van te maken. Het compileren van het Simulinkmodel wordt gestart met het commando `RTW build` in het Simulinkmodel. Tijdens het compileren wordt van het simulinkmodel een RTW C-code gegenereerd. Deze C-code wordt vervolgens voor de RTA gebruikt. Het compileren kost enige tijd en kan in het Matlab Command window worden gevolgd. Vervolgens wordt met behulp van Wintarget de RTA aan de QAD's gekoppeld.

Gebruik

Stop de TUEdAX cd in de cd-rom lade van het notebook en start het notebook op vanaf de cd-rom. Afhankelijk van de bios-instelling van het notebook zal de cd-rom als opstartapparaat automatisch worden herkend of is een actie van de gebruiker nodig, b.v. het intoetsen van een karakter zoals aangegeven op het scherm. Als alles goed gaat verschijnt uiteindelijk een full-screen grafische desktop die een vage gelijkenis vertoont met Windows. Dit is de KDE desktop van Linux. Klik één keer met de muis op het Matlab-icoon. Indien je dubbel geklikt hebt, sluit dan beide opgestarte Matlab-sessies af en begin opnieuw. Gedurende het opstarten zullen de beschikbare harde-schijf partities gevonden worden, deze verschijnen als iconen op de desktop (b.v. `hda3`). Indien de harde-schijf partities van het notebook van het type NTFS zijn, kunnen ze niet gebruikt worden onder Linux om data op te slaan. Er kan dan alleen worden geschreven naar de Linux ramdisk (`/ramdisk`) of naar een externe memory stick. Om een experiment te kunnen uitvoeren moet eerst worden gecompileerd met behulp van RTW om een RTA van te maken. Het compileren kost enige tijd en kan in het Matlab Command window worden gevolgd. Vervolgens wordt met behulp van Wintarget de RTA aan de QAD's gekoppeld. De beschikbare simulink-modellen voor experimenten en simulaties en de procedure voor het uitvoeren van experimenten worden hierna beschreven.

Experimenteel model

Het Simulink model (zie figuur 3) voor de aansturing van de robot bevat het pizza-interface blok en een start/stop blok. Het start/stop blok zorgt voor het starten van het experiment. Het pizza-interface blok zorgt voor de communicatie tussen de robot en het gemaakte model.



Figuur 3: simulink model voor experimenten

De ingangen zijn als volgt ingedeeld:

- Ingang 1: motor 1 voor verticale beweging vork [V]
- Ingang 2: motor 2 voor horizontale beweging vork [V]
- Ingang 3: motor 3 voor rotatie [V]
- Ingang 4: motor 4 voor translatie tafel [V]
- Ingang 5: ingang voor activatie via start/stop blok

en de uitgangen:

- Uitgang 1: verticale verplaatsing vork als gevolg van motor 1 [m]
- Uitgang 2: horizontale verplaatsing vork als gevolg van motor 2 [m]
- Uitgang 3: rotatie als gevolg van motor 3 [rad]
- Uitgang 4: translatie als gevolg van motor 4 [m]
- Uitgang 5: geeft de status van de robot weer. Hierbij geven de nummers het volgende weer:
 - 0: Geef start via start/stop knop
 - 1: Initialisatiefase
 - 2: Initialisatiefase
 - 3: Beweging, alleen tijdens deze fase kun je zelf iets ingeven

4 t/m 7: ‘Airbag’ grenzen zijn overschreden. De virtuele ‘airbag’ is een gebied waarin de robot tot stilstand kan komen wanneer deze de eindpositie nadert en daardoor geen schade kan veroorzaken aan de opstelling. Wanneer deze ‘airbag’ bereikt wordt, wordt het experiment gestopt.

Uitgang 6: geeft de tijd na de initialisatiefase weer. Dit gebruik je als trigger-sig-naal van je eigen trajectorie en regeling.

Uitgang 7: uitgangsspanning van motor 1

Uitgang 8: uitgangsspanning van motor 2

Uitgang 9: uitgangsspanning van motor 3

Uitgang 10: uitgangsspanning van motor 4

Simulatie model

Het Simulink model voor het doen van simulaties (zie figuur 4) bevat dezelfde blokken als het Simulink model voor het experimenteren. Het pizza-model blok bevat 8 uitgangen meer dan het pizza-interface blok voor het experimenteren. De eerste 10 uitgangen zijn identiek aan de 10 uitgangen van het pizza-interface blok, de laatste 8 geven de toestand van het systeem en zijn als volgt ingedeeld:

Uitgang 11: verticale verplaatsing vork als gevolg van motor 1 [m]

Uitgang 12: horizontale verplaatsing vork als gevolg van motor 2 [m]

Uitgang 13: rotatie als gevolg van motor 3 [rad]

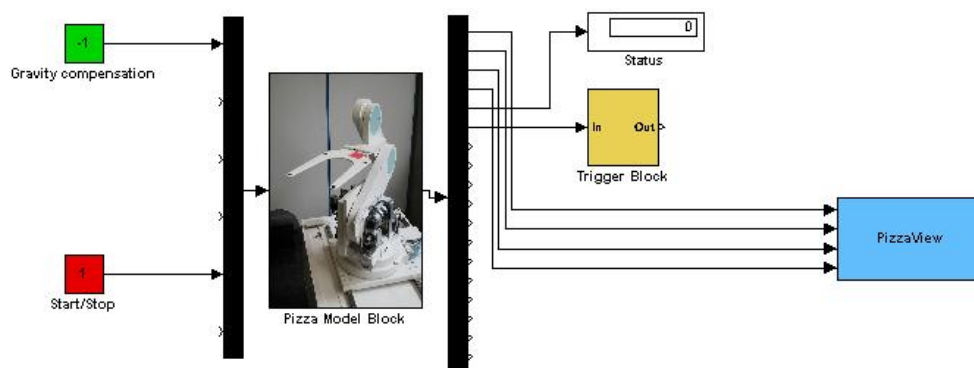
Uitgang 14: translatie als gevolg van motor 4 [m]

Uitgang 15: snelheid van de verticale verplaatsing [m/s]

Uitgang 16: snelheid van de horizontale verplaatsing [m/s]

Uitgang 17: snelheid van de rotatie [rad/s]

Uitgang 18: snelheid van de translatie [m/s]



Figuur 4: simulink model voor simulaties

Uitvoeren van experimenten en veiligheid

Voer voor het doen van experimenten altijd simulaties uit. Maak verstandige keuzes voor ingangssignalen, begin met ‘slappe regelaars’. Aan de versterkerkast van de robot zit een noodstop-knop. Hou deze in de hand tijdens het experiment, zodat je altijd kunt ingrijpen. Sluit de QAD’s aan op de pcmcia-slots van het notebook. De kaart met het hoogste nummer in het bovenste slot. Sluit de QAD’s aan voordat het notebook wordt gestart.

Tijdens experimenten zal de robot altijd eerst een initialisatiefase ondergaan. Tijdens deze fase wordt de robot naar de volgende positie toegebracht:

- motor 1: -0.1 [m]
- motor 2: 0.1 [m]
- motor 3: 2.4 [rad]
- motor 4: 0.2 [m]

Vanuit dit punt wordt na de initialisatiefase het daadwerkelijke experiment gestart. Tijdens de initialisatiefase zal uitgang 5, de uitgang die de status van de robot weergeeft, een 1 of een 2 weergeven. Uitgang 6, de trigger uitgang, wordt getriggerd na de initialisatiefase. Zet een trigger/enable-blok (Simulinkblok) in je regelblok en in de referentiegenerator en verbindt deze met uitgang 6. Hierdoor zullen beide blokken worden getriggerd om na de initialisatiefase te starten en zal jouw experiment worden uitgevoerd.

Bij het uitvoeren van een experiment dienen de volgende stappen te worden gevolgd:

1. Controleer of alles aanwezig is en of alles goed is aangesloten.
2. Zet de versterker van de robot aan.
3. Sluit de pcmcia-kaarten van de QAD’s aan op het notebook (kaart met hoogste nummer in het bovenste slot).
4. Start notebook op.
5. Start Matlab.
6. Laad het benodigde simulink model.
7. Toets: [Control + b](RTW build proces) - In het Matlab window zal uiteindelijk: **### Successful completion of Real-Time Workshop build procedure for model: simulinkmodel**, komen te staan.
8. Open een command prompt (*terminal program* in menubalk) in Knoppix.
9. Typ `su` (superuser) om te benodigde rechten te verkrijgen.
10. Typ `wt_run ./empty_exp` voor normale mode en `wt_run ./empty_exp -w` voor de externe mode om het experiment op te starten. Wanneer je het simulink model een andere naam hebt gegeven, dient `empty_exp` in de voorgaande commando’s vervangen te worden door deze nieuwe naam.
11. Indien voor externe mode is gekozen moet een connectie gemaakt worden tussen het Simulink model en de real-time applicatie en manueel opgestart worden vanuit Simulink. Dit kan gedaan worden via *Connect to target* en

Start real-time code uit het *Simulation* menu in het simulink schema.

12. Verander het start/stop blok van een 0 naar een 1.
13. Na het beëindigen van een experiment kan de verwerking van de meetdata in Matlab gebeuren.

Bij het stoppen van een experiment dienen de volgende stappen te worden gevolgd:

1. Verander de 1 in een 0 in het start/stop blok.
2. Wacht even en stop vervolgens het simulink model.
3. Sluit programma's en notebook af.
4. Verwijder de pcmcia-kaarten van de QAD's uit het notebook.
5. Zet de versterker uit.

In de aangeleverde files is de sample-frequentie ingesteld op 500 Hz. Verander deze niet. Gebruik geen *To file* voor het opslaan van data, omdat bij het gebruik van dit blok het real-time gedrag van het experiment verloren gaat.

Data opslaan op externe memory stick

Het opslaan van data, figuren of andere Matlab files op een externe memory stick kan gedaan worden via de volgende stappen:

- Op de desktop staat een icoon met de naam *Hard Disk Partition [sda1]*, dit is de externe memory stick. Indien de memory stick niet op de desktop aanwezig is, staat de bios waarschijnlijk niet op de default instelling.
- Wanneer bij het icoontje van de externe memory stick rechtsonder een groene driehoek staat, is de memory stick al gemount. Indien de memory stick niet gemount is, kan dit gedaan worden door met de rechter muisknop op het icoon te klikken en 'mount' te kiezen.
- Om data op de memory stick te kunnen bewaren, moet deze beschrijfbaar worden gemaakt. Dit kan gedaan worden door met de rechter muisknop op het icoon te klikken en bij 'actions' te kiezen voor 'change read-write mode'.
- De locatie van de memory stick is nu `/mnt/sda1`. Deze locatie kan in Matlab gebruikt worden om data te bewaren.

Linux commando's

Hieronder volgen enkele Linux commando's die vanuit een command prompt uitgevoerd kunnen worden.

- `cd` om van directory te wisselen, b.v. `cd /ramdisk/home/knoppix`
- `cp` om bestanden te kopiëren, b.v. `cp empty_exp.mdl /mnt/hda3/empty_exp`
- `rm` om bestanden te verwijderen, b.v. `rm empty_exp.mdl`
- `ls` om de inhoud van een directory te bekijken, b.v. `ls`

Vragen

Voor vragen kun je terecht bij:

Martijn Maassen, M.G.J.M.Maassen@student.tue.nl

Jeroen de Best, J.J.T.H.d.Best@tue.nl, tel 4227, WH -1.126

Roel Merry, R.J.E.Merry@tue.nl, tel 4850, WH -1.124

Projectcoördinator:

M.J.G. van de Molengraft, M.J.G.v.d.Molengraft@tue.nl, tel 2998, WH -1.141

Veel succes!