

Labboek Task 4

Leonie du Bois, Axel Bremer, Max Filtenborg, Pepijn Groenen
Pascal van Hees

Juni 2016

1 Inleiding

De afgelopen week hebben we ons gebogen over het probleem om NAO-robots de polonaise te laten dansen. De software die we gebruiken om de robots te programmeren is Choregraphe, versie 2.1.4. De volgende tekst biedt duiding bij ons werkproces, en het resultaat dat dit leverde.

Onderzoeksvraag: Is het mogelijk NAO-robots een polonaise te laten lopen?

In eerste instantie lijkt dit misschien een onderzoek dat puur vermakelijke doeleinden heeft. Maar buiten de entertainende factor van ons eindresultaat heeft ons onderzoek ook zeker wetenschappelijke relevantie. Het verbroederende effect van de menselijke polonaise kan namelijk zomaar iets te maken hebben met de mate van samenwerking die vereist is om de polonaise uit te voeren. De robot polonaise vereist ook veel samenwerking doordat de robots elkaar moeten vinden, elkaar niet om mogen duwen en op elkaar in de gaten moeten houden wanneer de polonaise omdraait. De robot polonaise is kortom goed experiment om uit te testen of de robots in staat zijn om een fysieke menselijke samenwerking na te bootsen.

Deelvragen:

Ten eerste willen we ervoor zorgen dat de robots elkaar herkennen en vervolgens achter elkaar aan lopen. Essentieel is dat de robots niet omvallen tijdens het lopen van de polonaise. Onze derde prioriteit is dat ze elkaar in de gaten houden en om kunnen draaien als hun voorganger dat ook doet. Om dit te bereiken hebben we de volgende deelvragen opgesteld:

- Kunnen de robots elkaar herkennen, zodat we ze achter elkaar kunnen laten lopen?

Hypothese: Met behulp van de video camera die de robots op hun gezicht

hebben, zijn ze in staat om herkenningspunten te herkennen. Als we deze herkenningspunten op de robots plakken, kunnen we ervoor zorgen dat de robots achter de herkenningspunten aan lopen en op deze manier dus achter elkaar aan lopen.

- Hoe kunnen we zorgen dat de robots op dezelfde afstand van elkaar blijven lopen?

Hypothese: Als de we de grote van de herkenningspunten verwerken in onze code, moeten de robots uit kunnen rekenen op welke afstand zij zich van het herkenningspunt bevinden. Als de robots dit weten kunnen ze dus ook een gelijke afstand bewaren.

- Hoe kunnen we ervoor zorgen dat de robots hun evenwicht bewaren terwijl ze lopen, één arm op hun voorgangers schouder en de andere arm in de lucht zwaaien?

Hypothese: Waarschijnlijk zal het bewaren van dit balans een moeilijke opgaven worden aangezien de robots bij het uitvoeren van voorgeprogrammeerde programma 's al vaak omvallen. Maar als we ervoor zorgen dat de ene arm niet te dicht bij de schouder van de andere robot komt en de andere arm niet te uitbundig zwaait, moet dit wel mogelijk zijn.

- Kunnen we de robots op elkaar laten reageren, dus als de voorste robot in de polonaise stopt met lopen en zich omdraait, draaien de andere robots zich dan ook om?

Hypothese: Als we twee herkenningspunten gebruiken, kunnen we het ene herkenningspunten op de rug plakken en het andere herkenningspunt op de buik. We kunnen vervolgens de voorste robot telkens één van die twee herkenningspunten laten zien en zorgen dat ze bij het zien van dat herkenningspunt vooruit lopen. Bij zien van het tweede herkenningspunt zal de eerste robot dan omdraaien, waarbij hij het "omdraai herkenningspunt" aan de tweede robot zal laten zien en op deze manier kan heel de rij omdraaien. Het persoon dat de polonaise leid kan dan om de robots heen lopen en de voormalige laatste robot, nu de voorste dus, weer vooruit leiden met het eerste herkenningspunt.

Voortgangsrapportage

Maandag 20-6-2016

Het hoofdprobleem van de polonaise is om ervoor te zorgen dat de robots elkaar kunnen herkennen, en vervolgens achter elkaar aan kunnen lopen. Dit eerste probleem hebben we opgelost met behulp van ALLandMarkDetection. Dit is een ingebouwde module in de Choregraphe software. Hiermee kan de robot een 'landmark' herkennen, en vervolgens hier naar toe bewegen.

Dinsdag 21-6-2016

We hebben de beweging voor de zwaai uit een bestaande beweging geknipt. In die timeline hebben we beweging frames toegevoegd zodat de beweging vloeiender is. Dit is zodat de robot zijn evenwicht niet verliest. We hebben de robots achter elkaar aan kunnen laten lopen door een landmark op de rug van de robots te plakken.

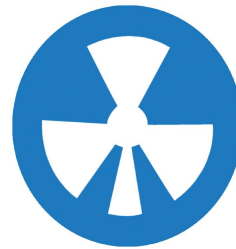


Figure 1: De sticker gebruikt voor herkenning

Woensdag 22-6-2016

Nu de basis functionaliteit om de polonaise te lopen werkt, zijn we aan de slag gegaan, de robots tijdens de polonaise te laten omkeren. Eerst hadden we bedacht om 2 verschillende stickers te gebruiken, een op de rug van de robots en een op de voorkant. Als de robot de soort sticker op de rug ziet, dan beweegt hij naar deze toe. Als de robot de soort sticker ziet die op de borstkas zit, draait de robot zich om. Hiermee zouden we de robot stoet kunnen omdraaien. Uiteindelijk hebben we niet voor deze methode gekozen, maar een simpelere opzet. De uiteindelijke oplossing is geworden om maar één sticker te gebruiken. Een van onze groepsleden gaat voor de robots staan met een sticker, om zo de stoet te leiden. De robots lopen naar deze sticker toe (elke robot heeft deze ook op zijn rug, zodat ze achter elkaar aan lopen). En als de voorste robot de sticker bereikt heeft, stoppen ze, vervolgens halen we de sticker weg uit het zicht van de robot, waardoor de voorste robot zich zal omdraaien. Als de voorste omgedraaid is, ziet de robot achter deze robot de sticker ook niet meer, en zal deze ook omdraaien. Dit zorgt ervoor dat de stoet robots zich omdraaien, en vervolgens de andere kant op gaan lopen.

Donderdag 23-6-2016

Nu hebben we geïmplementeerd dat de robots gaan draaien als ze de sticker niet meer zien. Dit hebben we met behulp van het (output) event targetLost, van de ALLandMarkDetection opgelost. Deze hangt aan een 180 graden draaiing beweging. Voordat dit werkte moesten we echter nog een 'standInit' beweging tussen dit event en de draaiing zetten, omdat de robot niet kan draaien terwijl

hij nog in een beweging naar voren zit.

De robots kunnen zich nu omdraaien. Dit verloopt echter niet helemaal soepel. Omdat de robots de linker arm vooruit steken, en de rechter arm in de lucht zwaaien (als dans beweging), vallen de robots snel om. Als de robot zich gaat omdraaien, gaat de robot gewoon door met het zwaaien van de arm, en houdt ook zijn linker arm in de lucht. Om de draaiing stabiel te maken hebben we ervoor gezorgd dat de robot zijn armen eerst langs zijn lichaam houdt.

Behaviour voor de NAO Polonaise

```
opstaan;
zwaaiend = true;
volgend = true;
while Volgend do
  | if LandMark in beeld then
  |   | Beweeg richting LandMark;
  | else
  |   | Draaimethod;
  | end
end
while Zwaaiend do
  | linkerarm vooruit;
  | rechterarm omhoog en zwaaiend;
end
Draaimethod:
  zwaaiend = false;
  volgend = false;
  stop met lopen;
  handen langs lichaam;
  draai 180 graden;
  volgend = true;
  zwaaien = true;
```

Discussie

Proces

Aan het proces van ons onderzoek zijn de volgende verbeteringspunten op te merken:

- Ten eerste zijn we veel tijd kwijt geraakt aan het opstarten van de apparatuur. Niet al de robots die we tot onze beschikking hadden werkten even goed en vrijwel elke dag hebben we te maken gehad met problemen die zich voordeden in het programma, dat voorheen wel werkend leek te zijn. Dit hadden we kunnen voorkomen door de NAO's op tijd te reserveren.
- Ook hebben we vaak het probleem gehad dat we twee aspecten van een bepaalde functionaliteit werkend hadden gemaakt maar dat deze niet tegelijkertijd werkte. Dit hadden we minder problematisch kunnen maken door de twee aspecten al in een eerder stadium samen te testen.
- Omdat we met vijf mensen tegelijk aan het onderzoek hebben gewerkt zijn we tegelijkertijd oplossingen gaan zoeken voor meerdere deelvragen. Dit hadden we beter stap voor stap aan kunnen pakken de verschillende functionaliteiten elkaar in de weg zaten waardoor de robots omvielen. Een concrete verbetering zou in ons geval dus zijn dat we met minder mensen dit project hadden moeten beginnen.

Door de landmark tracker functie in de NAO software hebben we een redelijk goed antwoord kunnen vinden op de eerste twee deelvragen. Ze kunnen de herkenningspunten op zo'n drie meter afstand herkennen en lopen er heen, met een snelheid die afhankelijk is van de afstand waarop ze verwijderd zijn van het herkenningspunt. Als ze dichterbij komen houden ze een afstand die net voldoende is om elkaar niet aan te raken en deze afstand wordt zeer constant bewaard. Een van de problemen die zich voordoen met de landmark trackers, is dat de robots af en toe omvallen als het herkenningspunt te hoog in de lucht wordt gehouden. Ook is het heel soms zo dat een landmark tracker niet herkent wordt.

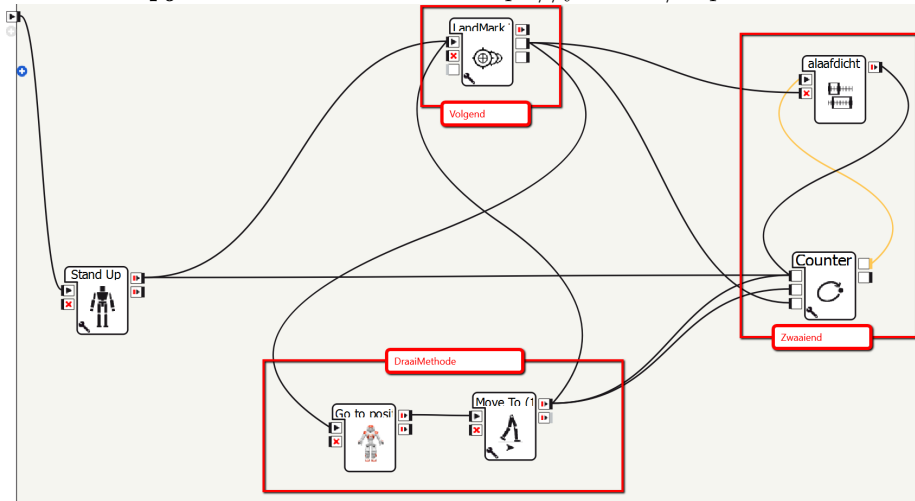
Uiteindelijk is het ons gelukt om de robots hun evenwicht te laten bewaren tijdens het uitvoeren van de polonaise. Het probleem dat echter nog wel bestaat ten aanzien van de armen is dat ze door onbekenden reden hun armen soms niet meer omhoog doen.

Een goede verbetering aan de polonaise zou zijn om de voorste robot autonoom de polonaise te laten sturen. Omdat we nu handmatig een sticker gebruiken om de stoet te leiden, resulteert dit in onstabielheid, en kan het gebeuren dat de voorste robot de sticker niet herkent. Handmatig de sticker voorhouden zorgt ook voor schokkige bewegingen waardoor de voorste robot hier slecht op reageert, en eventueel om kan vallen.

Later kunnen we ook proberen om de robots directer met elkaar te linken. Zodat ze niet met zicht, maar door signalen over het netwerk merken wat de ander doet. Dit zou tot snellere reactietijden kunnen leiden. Zo zullen de robots minder snel tegen elkaar aan botsen.

Appendix

Filmpje van de Robonaise: <https://youtu.be/0squiYaxZVA>



Flowchart voor Polonaise Behaviour