

# Robottiohjelmoinnin harjoitustyö

Alexi Paavola  
014141447  
alpa@cs.helsinki.fi

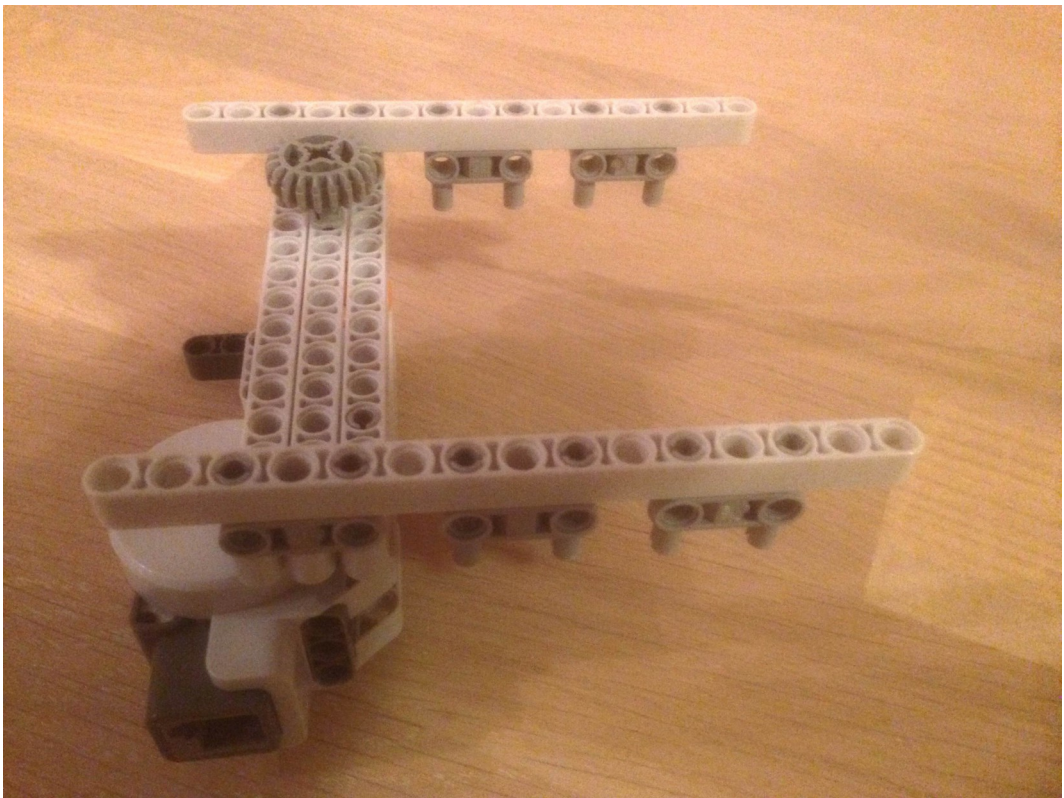
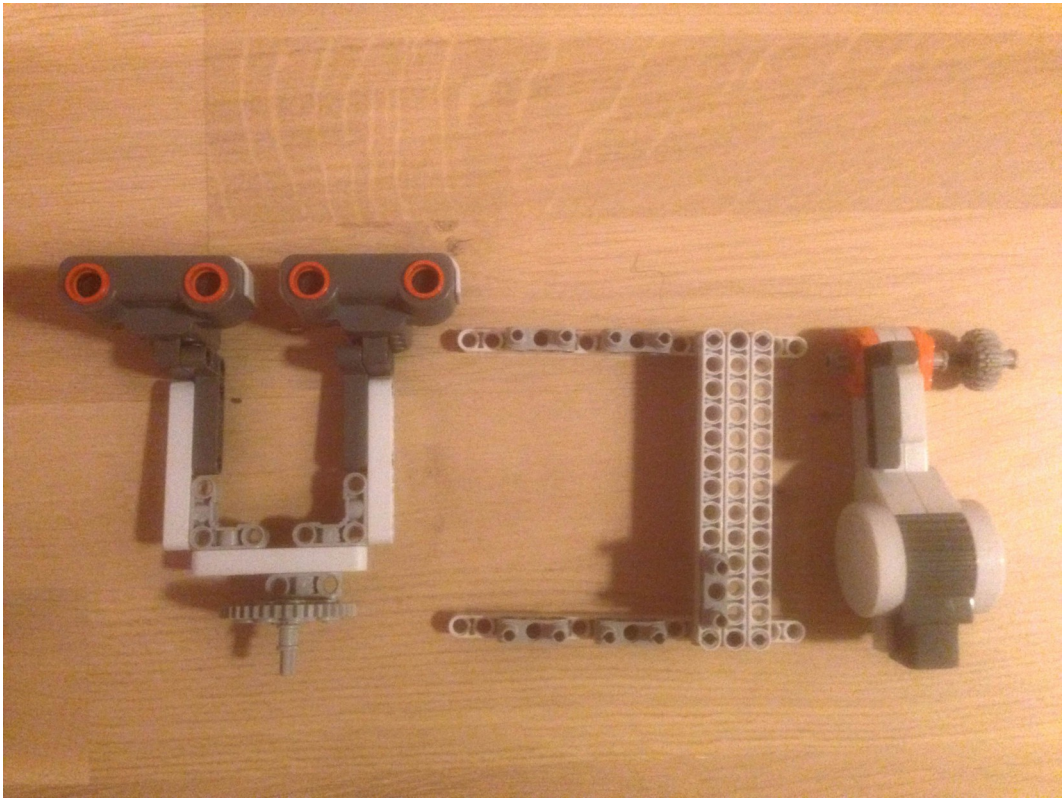
## Kuvaus

Robotti on paikallaan pyörivä tutka-asema, joka aluksi pyörii hiljaa skannaten ympäristöä. Kun robotti havaitsee liikettä, pyrkii se seuraamaan liikkeen kohdetta kääntymällä sitä kohti. Kun kohde on jonkin aikaa paikoillaan, luovuttaa robotti seuraamisen ja jatkaa partiointia.

## Robotin rakenne

Robotti rakentuu alustasta johon BRICK ja moottori kiinnitetään, sekä moottoriin liitetystä tornista. Tornin päässä on kaksi ultraäänisensoria.





# Koodin rakenne

Koodissa on kolme luokkaa:

## 1. Tracker

Pääluokka, vastaa varsinaisesta toiminnallisuudesta. Käyttää moottoria ja sensoreita kahden alemman luokan avulla.

## 2. Turret

Vastaa tornin pyörimisestä, pyörimisnopeudesta ja pitää huolta siitä, että laittomia pyörimisiä ei tapahdu (piuhat menee solmuun!).

## 3. Scanner

Vastaa ultraäänisensorien tiedon keruusta ja käsittelystä. Käyttää sensorien Ping-moodia (mode 1), jossa sensorit luotaavat vain käskystä.

Ohjelma käynnistyy main-luokasta, jossa se luo Trackerin ilmentymän ja kutsuu sen scan()-metodia. Scan pyörittää tornia hitaasti ja luotaa aina välillä ympäristöä, kunnes havaitaan liikettä. Tällöin kutsutaan track()-metodia, jossa pyritään nopealla skannauksella pysymään kiinni aiemmin havaitussa kohteessa. Tämä toteutetaan käyttämällä IsInSights(String usedSensor)-metodia, joka palauttaa true jos kohde on suoraan sensorin edessä. Tämän jälkeen pyritään kääntymään siten, että kohde on molempien sensoreiden edessä (Jos vasen sensori huomaa kohteen, käännetään lisää vasemmalle että myös oikea näkee kohteen).

# Testaus

Testien tarkoituksena on selvittää, miten robotti selviytyy tehtävästään erikokoisissa tiloissa.

## Testitapaus I: Avoin tila

Robotti pyörii skannausmoodissa rutiininomaisesti, lukuvirheitä ei ilmene ja suuntaa vaihdetaan kääntymisalan rajoilla. Havaittuaan hidasta liikettä robotti kääntyy suurimman osan ajasta oikeaan suuntaan ja osaa lukittua kohteeseen määritelmän mukaisesti.

Nopean liikkeen kanssa robotti ei pysy aina perässä ja saattaa lähteä väärään suuntaan tai mennä ohi kohteesta.

## Testitapaus II: Ahdas tila

Robotti pyörii skannausmoodissa rutiininomaisesti, tosin suurissa syvyyseroissa robotti saattaa välillä siirtyä jäljitysmoodiin.

Kohteen seuraaminen kohtaa erään ongelman ahtaissa tiloissa: robotti erehtyy silloin tällöin luulemaan taustaa tai lähettyvillä olevia paikallaan olevia esineitä kohteeksi. Tämä tapahtuu sekä hitaan että nopean kohteen testeissä.

# Rajoitukset ja tulevaisuus

Robotin toiminta on osin rajoittunut sen fyysisestä rakenteesta, osin koodista.

Fyysisellä puolella kaksi korjattavissa olevaa ongelmaa ovat:

1. Ultraäänisensorit ovat liian lähellä toisiaan. Tästä seuraa kohdan 2 kanssa, että nopean liikkeen suuntaa ei pystytä määrittelemään virheettömästi. Voidaan mahdollisesti korjata laventamalla rakennetta, kääntämällä toinen sensori hieman viistoon tai korjaamalla kohta 2.
2. Toteutuksen ultraäänisensoreilla on ping-moodissa kahdenkymmenen millisekunnin viive luotauksen ja tiedon vastaanottamisen kanssa (lejos API:n mukaan). Tämä tekee nopeasti liikkuvan kohteen seuraamisesta vaikeaa. Ongelman korjaamiseksi sensorien määrää voitaisiin lisätä, tai voitaisiin käyttää paketin ulkopuolisia sensoreita.

Ohjelmistopuolella polttavin ongelma on liikkuvan kohteen alkeellinen käsittely. Periaatteessa robotti vain teeskentelee tunnistavansa objektin, sillä se liikkuvan kohteen löydettyään etsii samalla alalla olevia objekteja (tosin se pystyy myös päivittämään kohteen etäisyyttä uusilla löytökerroilla). Tästä johtuen robotti tulkitsee sensoreista yhtä kaukana kuin kohde olevien objektien olevan haettu kohde.

Tälle mahdollinen korjaus olisi toteuttaa löydetylle kohteelle luokka, joka säilyttää etäisyys -ja havaintotietoja kohteesta jotta voitaisiin laskea, voiko jokin havaittu objekti edes olla kohde.

Ehkä paras tapa korjaus olisi kuitenkin lisätä rakenteeseen jokin muu sensori (esim. valo/väri), että kohdetta voitaisiin profiloida tarkemmin.

# Käyttöohje

Robotti on toteutettu Windows 7-käyttöjärjestelmällä käyttäen 32-bittistä eclipse-ohjelmistonkehitysympäristöä 32-bittisellä lejos-ohjelmistolla ja plug-inilla.

Paras tapa ajaa robotti on siis kytkeä se USB:lla tietokoneeseen ja ajaa eclipsellä ”Run as LeJOS NXT Program”.

1. Robotti käynnistyy ajamalla main-tiedosto.
2. Robotti aloittaa skannaamisen kääntymällä hitaasti myötöpäivään. Se vaihtaa suuntaa kun main-luokassa määrätty maksimiaste on saavutettu.
3. Robotti skannaa vain ollessaan paikoillaan. Se ei skannausmoodissa reagoi liikkeeseen.
4. Track-moodin voi käynnistää esim. kävelemällä robotin sensorien ohi sen ollessa paikoillaan.
5. Robotin saa sammutettua painamalla BRICKin tummaa ja oranssia nappia samanaikaisesti.