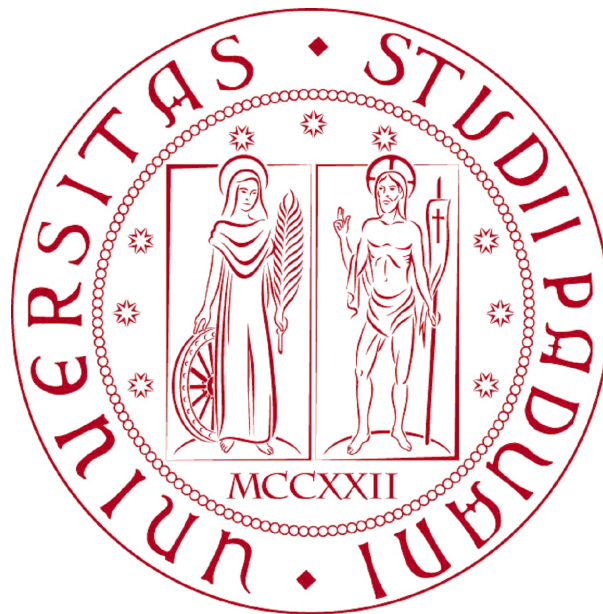


UNIVERSITA' DEGLI STUDI DI PADOVA

CORSO DI ROBOTICA AUTONOMA

ESPERIENZA 3



GRUPPO 11:

Marco Bertagnoli

Matteo Mastellaro

Angelo Trevisol

Anno Accademico 2015-2016

# Indice

1	Introduzione	2
2	Obiettivi e Procedimento	2
3	Simulazioni	2
4	Creazione della Mappa	2
5	Test in laboratorio	3
6	Conclusioni	4

# 1 Introduzione

Il laboratorio di *Intelligent Autonomous Systems Laboratory (IAS - Lab)* di Padova è fornito di un robot umanoide NAO, autonomo e programmabile, sviluppato da Aldebaran Robotics. Esso è spesso utilizzato per scopi di ricerca e didattica in tutto il mondo, ed è inoltre stato utilizzato nella *RoboCup* delle edizioni 2008 e 2009; una importante competizione calcistica internazionale tra robot. Esso dispone di interfacce come wireless e Ethernet per poter essere comandato da remoto e svolgere determinati compiti.



Figura 1: Umanoide NAO.

## 2 Obiettivi e Procedimento

L'obiettivo di tale esperienza è quello di pianificare il moto di un NAO robot in un ambiente 2D popolato da ostacoli. Il robot dovrà dunque essere comandato per camminare lungo un percorso precalcolato, evitando collisioni con gli ostacoli posti nelle vicinanze. Dunque si dovrà, nell'ordine:

- Caricare una mappa, utilizzando inizialmente quella predefinita
- Calcolare il percorso con un software di path planning
- Far camminare il robot simulato, all'interno dell'ambiente senza collidere
- Muovere il robot reale, usando la stessa mappa e lo stesso percorso generato nella simulazione.

Dopodichè si dovrà caricare una mappa diversa da quella predefinita, e ripetere i punti precedenti.

## 3 Simulazioni

Attraverso i tool *2D Pose Estimate* e *2D Nav Goal* è possibile realizzare la fase di path planning, con la quale verrà calcolato il percorso ottimo, che verrà poi comunicato al NAO, il quale sarà interfacciato con il programma in esecuzione attraverso una rete Ethernet. Per prima cosa, in un ambiente software simulativo, si assegna una posizione di inizio e fine percorso per il NAO, all'interno di una mappa preassegnata comprendente alcuni ostacoli.

Si procede dunque all'inizializzazione dell'ambiente simulativo con la mappa predefinita, alla quale si va a comunicare la posizione iniziale del robot e quella finale. Come viene visualizzato in Figura 3a, il percorso tra posizione iniziale e finale viene calcolato e viene dunque simulato il movimento del robot. Allo stesso modo, viene caricata una diversa mappa, e ripetuta la simulazione.

## 4 Creazione della Mappa

Per la creazione della mappa si è scritto un semplice programma (*make\_map.cpp*) avviabile con il comando `ROS rosrundesperienza3 make_map` una volta installato il pacchetto. Il programma cerca il file *mappa.yaml* all'interno del pacchetto *esperienza3*, in questo file un qualsiasi utente può specificare:

- Dimensione della mappa in termini di numero celle in altezza e larghezza.

- Il numero di ostacoli nella mappa e le loro coordinate  $x$  ed  $y$ .
- La lunghezza in metri del lato di una singola cella.
- Il numero di pixel con cui si andrà a disegnare il lato della cella.

Una volta recuperati questi dati crea un'immagine in bianco e nero corrispondente ai dati inseriti nel file *YAML* e la salva all'interno della cartella *maps* del pacchetto *footstep\_planner* così da poter essere direttamente utilizzata durante l'esperienza prima di terminare l'esecuzione visualizza l'immagine creata.

In Figura 2 è rappresentata la mappa restituita dal programma *make\_map* ottenuta con la seguente configurazione, dove i contorni neri e gli assi cartesiani sono stati aggiunti successivamente per rendere la visualizzazione più semplice; essi non sono invece presenti nella mappa che si utilizza per l'esperienza.

- $altezza = 5$ ;
- $lunghezza = 10$ ;
- $numero\_ostacoli = 10$ ;
- Posizione ostacoli:
  - $x = [1, 2, 2, 2, 4, 4, 4, 5, 6, 6]$ ;
  - $y = [1, 1, 2, 3, 0, 2, 4, 1, 2, 3]$ ;

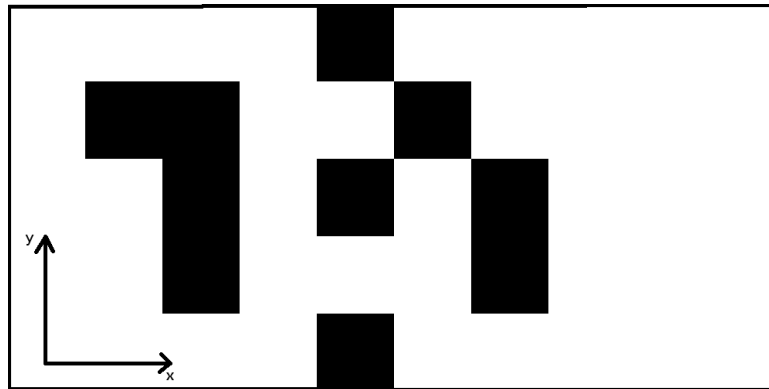
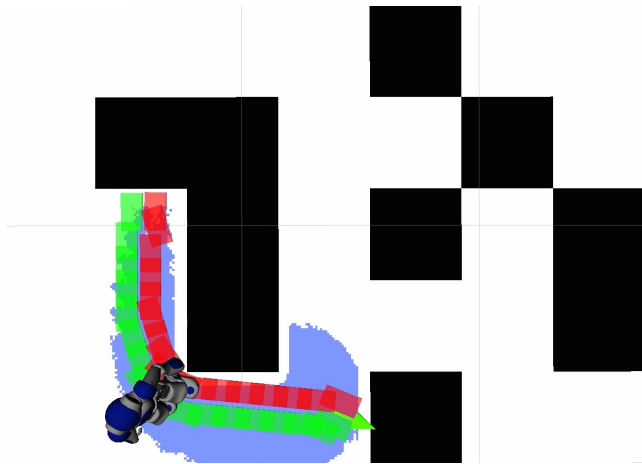


Figura 2: Mappa restituita dal programma *make\_map*.

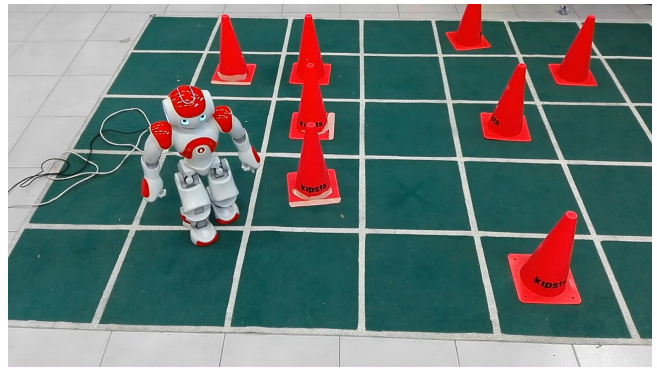
## 5 Test in laboratorio

A questo punto è possibile procedere con il test in ambiente reale. Per far collegare il NAO con il programma che eseguirà path planning e movimento del robot è necessario innanzitutto collegare entrambi alla stessa rete LAN. Ulteriore passo da fare per far lavorare il NAO nell'ambiente reale, è quello di assicurare che i motori dei giunti siano attivi, in modo che il robot possa inizialmente reggersi in piedi restando fermo.

Per prima cosa si procede al calcolo del percorso ottimo su mappa predefinita attraverso l'utility software già menzionata. Dopodiché il percorso calcolato viene comunicato al NAO, che nel frattempo avvia il suo movimento all'interno della mappa predefinita reale, appositamente ricreata, dove si sono posti dei conetti stradali a rappresentazione gli ostacoli (Figura 3b).



(a) Nella mappa viene mostrato il percorso ottimo, calcolato tra inizio e fine.



(b) Mappa ricreata in ambiente reale, in Laboratorio.

Figura 3: Movimento del Robot.

## 6 Conclusioni

Con la presente esperienza si è fatto pratica con i concetti di path planning, obstacle avoiding e con il robot umanoide NAO, con il quale è stato necessario ad esempio risolvere il suo interfacciamento con l'host di simulazione, così come problemi iniziali nel mantenere il robot in equilibrio nella sua posizione iniziale. Le simulazioni del robot su software non hanno riscontrato particolari problematiche, mentre i test sul robot reale hanno evidenziato ricorrenti problemi nel seguire le traiettorie prepianificate, principalmente per il fatto che un continuo stress dei motori, portava questi ultimi ad un eccessivo surriscaldamento. Per far fronte a questo problema si è preferito scegliere una traiettoria che non fosse troppo lunga in modo da non sollecitare eccessivamente il robot. Dopo ciò anche i test in ambiente reale hanno conferito risultati soddisfacenti.