

INTRODUZIONE

La tesi è stata realizzata nell'ambito della competizione internazionale di calcio robotico *RoboCup 2003*: è stato progettato e costruito un robot calciatore con il ruolo di portiere per la squadra di calcio robotico *MRT* (Milan RoboCup Team) costituita dall'Università di Milano-Bicocca e dal Politecnico di Milano. Come usuale in robotica il lavoro ha coinvolto tematiche relative a discipline tra loro distinte; i contributi principali hanno riguardato:

- la progettazione e realizzazione dell'impianto elettrico del robot;
- la realizzazione di hardware dedicato;
- il software low-level di controllo movimento (in velocità ed in posizione) con problematiche Real-Time;
- i calcoli relativi all'odometria.
- la percezione del mondo per alimentare il sistema di autolocalizzazione;
- l'intelligenza artificiale, per lo sviluppo di comportamenti per il ruolo di portiere ;



**Rabbiati, il robot portiere dell'MRT (Milan Robocup Team),
a SMAU 2003**

Nella figura qui accanto possiamo vedere Rabbiati (questo è il nome dato al nostro robot) in porta durante le competizioni del torneo amichevole svoltosi alla fiera SMAU 2003.

Partendo da un'infrastruttura meccanica realizzata in precedenza dai colleghi di laboratorio, è stato progettato, e successivamente realizzato, l'impianto elettrico del robot. Un *Tablet PC* è montato sul robot per adempiere alle funzionalità di alto livello, ed è energeticamente autosufficiente. Tale PC è stato comunque collegato, attraverso un regolatore di tensione, alle batterie del robot per prolungarne l'autonomia. Le problematiche principali riscontrate riguardano l'alimentazione della logica di basso livello e della

telecamera. Sono stati analizzati diversi tipi di circuiti che prendono in considerazione configurazioni di batterie diverse: batterie per logica e motori separate o un unico pacco batterie per entrambi, con diversi tipi di circuiti atti allo smorzamento dei disturbi elettrici introdotti dai motori. Per realizzare un circuito di alimentazione stabile e privo di disturbi per la logica di controllo e la telecamera sono stati valutati anche diversi tipi di stabilizzatori e DC/DC Converter, tenendo conto dell'assorbimento di corrente dei diversi dispositivi. La scelta è poi ricaduta sull'utilizzo di un unico pacco batterie per l'alimentazione dell'intero robot. L'impianto elettrico è stato realizzato dando la possibilità, grazie ad interruttori distinti, di alimentare separatamente i diversi settori funzionali del robot: l'elettronica di potenza dei motori, il circuito di comando dell'azionamento dei kicker, la logica di controllo dei motori e la telecamera. Ognuno di questi dispositivi può essere quindi alimentato indipendentemente dall'altro. Questa proprietà si è resa particolarmente utile durante la fase di testing, in cui le varie parti del robot potevano essere testate contemporaneamente e indipendentemente l'una dall'altra.

Un'altra fase del lavoro si è occupata della realizzazione di hardware dedicato. Si è reso necessario produrre una scheda di controllo a basso livello dei motori. Sono state prodotte in successione tre versioni della scheda, ognuna l'evoluzione della precedente. La larga parte della progettazione è stata realizzata dal collega Marco Ingrao, mentre un mio contributo alla progettazione è stato fornito nella realizzazione della seconda e della terza versione della scheda. Lo sbroglio del circuito, operazione necessaria per la sua successiva realizzazione fisica, è stato realizzato, utilizzando il software *Protel* della ditta *Altium*, dal collega Lucania. A valle di ciò ho poi svolto il lavoro di realizzazione vera e propria dei vari circuiti stampati, attraverso un processo di fotoincisione di basette presensibilizzate, nonché il montaggio su di esse dei componenti elettronici costituenti il circuito ed il testing delle schede, a valle del montaggio.

Una volta realizzato il circuito è stato perfezionato il software di basso livello che gestisce, in tempo reale, il controllo *in velocità* dei motori [1] e dei kicker e che esegue calcoli odometrici utili per il software di alto livello. La scheda di controllo è basata su un microcontrollore Microchip PIC18F452 a 40 MHz. Per prima cosa sono state individuate le funzioni principali del sistema: queste riguardano la comunicazione con l'alto livello (lettura da seriale e scrittura su seriale) l'interpretazione di comandi (Parsing), la lettura degli encoder posti sui motori, il controllo della velocità dei singoli motori attraverso l'utilizzo di un algoritmo di controllo PI (proporzionale, integrale) ad anello chiuso, la scrittura sui DAC del valore posto in ingresso ai servocontrollori di potenza dei motori e il calcolo dell'odometria. In seguito è stata gestita la corretta esecuzione di queste funzionalità affrontando le problematiche di tempo reale legate ai tempi di esecuzione delle funzionalità stesse. Per fare questo sono stati individuati i vincoli temporali ai quali ogni funzionalità deve sottostare. Tenendo in considerazione questi vincoli sono state assegnate delle priorità alle varie funzioni del sistema. Grazie a queste priorità viene quindi definita l'interrompibilità di una funzione da parte di ognuna delle altre in modo che ogni funzione possa portare a termine il suo compito rispettando i propri vincoli temporali. Il sistema è quindi un sistema ad eventi. Gli

eventi possono essere esterni (ad esempio: arrivo di un byte sulla seriale) o interni (ad esempio: scadenza del timer che gestisce la lettura degli encoder). Gli eventi interni vengono generati da interrupt che determinano il cambiamento di stato del sistema. L'implementazione del sistema di controllo di basso livello del robot è stata poi seguita da un'intensa fase di testing in cui il sistema è stato reso robusto e affidabile in ogni sua funzionalità. Lo sviluppo di questo software, per quanto riguarda la parte di controllo, è stato inizialmente svolto assieme al collega Ingraio; in seguito ho introdotto alcune evoluzioni. Successivamente è stato integrato, al posto del controllo in velocità, un algoritmo di controllo *in posizione* del robot. Questo algoritmo effettua un'interpolazione della traiettoria per far compiere al robot dei percorsi specifici. L'implementazione di questo algoritmo di controllo in posizione è dovuta al collega Giuseppe Cimetti. Il mio compito è stato quello di integrare il nuovo algoritmo con il software preesistente, che comprendeva già tutte le altre funzionalità, in particolare la struttura ad eventi.

Nel sistema di controllo di basso livello è stato integrato anche un algoritmo dedicato al calcolo della *odometria* [2]. L'algoritmo mantiene aggiornata la rototraslazione tra la posizione del robot nel sistema di riferimento iniziale e quella corrente utilizzando i valori degli encoder, letti a intervalli costanti. Data la complessità del calcolo rispetto alle risorse hardware a disposizione, che richiede il calcolo di funzioni trascendenti, è stato necessario effettuare alcune approssimazioni. In conclusione è stato possibile inserire questo calcolo a livello della logica di controllo rispettando i vincoli temporali delle altre funzionalità svolte dalla logica di basso livello; questo lavoro è stato svolto assieme al collega Cavicchioli.

Durante il lavoro di tesi è stato altresì affrontato il problema dell'estrazione di *percezioni visive* utili all'autolocalizzazione del robot. Il robot utilizza come unico sensore un sistema di visione *omnidirezionale catadiottrico* costituito da una telecamera e da uno specchio, che gli consente di vedere a 360° il mondo intorno a sé [3]. Le percezioni consistono in particolari punti dell'immagine omnidirezionale catturata dal sistema di visione del robot. L'estrazione delle percezioni si basa sul fatto che l'ambiente di RoboCup è un ambiente strutturato, cioè costituito da elementi ben precisi e noti al robot: il campo è verde, le linee sono bianche, una porta è di colore giallo e una porta è di colore blu. Esistono inoltre, ai quattro angoli del campo, dei paletti multicolore gialli e blu chiamati *corner*. Le percezioni utili all'autolocalizzazione sono: i punti bianchi che costituiscono le linee del campo, le direzioni in cui si trovano le porte e le direzioni in cui si trovano i corner. Sull'immagine omnidirezionale sono disposti, in 40 corone da 360 elementi, dei *recettori* che costituiscono un sottocampionamento polare dell'immagine [4]. Questi contengono informazioni sul colore di ogni singolo recettore, sull'angolo e sulla distanza dal robot del punto nella scena corrispondente al punto nell'immagine. L'elaborazione avviene sulla struttura dati che contiene l'insieme dei recettori, detta *Mappa dei Recettori*, che ha una dimensione di 40 righe per 360 colonne. La scansione della mappa dei recettori avviene per colonne, questo corrisponde ad una scansione a raggiera dell'immagine omnidirezionale partendo, per ogni raggio, dal punto più vicino al robot e allontanandosi progressivamente da esso. L'estrazione delle

percezioni si basa sulle transizioni di colore tra due o più recettori consecutivi. Per estrarre una percezione di tipo “*punto bianco*” si rileva, durante la scansione, il passaggio da un recettore verde ad uno bianco (*transizione*) e successivamente la transizione da un recettore bianco ad uno verde. La percezione ricavata consiste nella media delle distanze delle due transizioni dal robot e dall’angolo, rispetto allo zero del robot, in cui queste sono state trovate. Le coordinate del punto sono quindi in coordinate *relative* nel sistema di riferimento del robot. Per quanto riguarda i corner e le porte, invece, la percezione indica solamente la direzione in cui sono rilevati questi elementi. Per le porte viene prodotta una percezione quando, scandendo una direzione (cioè un raggio della mappa dei recettori), si incontrano un certo numero (parametrizzabile) di recettori blu per quanto riguarda la porta blu o gialli per la porta gialla. Per i corner si tiene invece traccia delle alternanze dei colori giallo e blu in una stessa direzione.

Le percezioni inviate al sistema di autolocalizzazione sono un sottoinsieme di quelle trovate: prima si inviano quelle più significative (ad es.: i corner) e poi un certo numero delle altre. Questa selezione è necessaria per evitare che il sistema di autolocalizzazione riceva un numero eccessivamente elevato di percezioni.

Un’altra parte del mio lavoro si è occupata dello sviluppo di comportamenti avanzati per il ruolo di portiere. Il sistema di controllo di alto livello adottato è basato su un *Behaviour Engine* a regole fuzzy [6] che elabora i dati in ingresso convertendoli in predicati fuzzy e restituisce delle attuazioni da inviare alla logica di basso livello. L’utilizzo della logica fuzzy è legato alla rappresentazione dell’incertezza di cui è affetto il modello del mondo fornito dai sensori, nel nostro caso il sistema di visione. Sono stati sviluppati quattro comportamenti: *Align*, *GoalKeeping*, *Kick* e *GotoHome*. Il comportamento di *Align* si occupa di mantenere il robot allineato su una sorta di binario immaginario posto ad una distanza predefinita dalla linea di porta. Questo gli consente di muoversi davanti alla porta rimanendo in una sorta di “zona di parata”. Il comportamento di *GoalKeeping* consente al robot di parare seguendo i movimenti della palla. Nel comportamento di *Kick* sono gestite le funzionalità di calcio della palla. Il comportamento di *GotoHome* consente al robot di tornare in porta da qualsiasi punto del campo. In ogni momento questi comportamenti vengono elaborati dal *Behaviour Engine* che decide quale comportamento attivare ed in che misura. Le attivazioni dei comportamenti sono quindi accompagnate da un peso che ne rappresenta la desiderabilità. Le attivazioni vengono successivamente miscelate in funzione dei loro pesi, dando così origine alle attuazioni.

Questo lavoro di tesi mi ha consentito di avere una visione di insieme delle problematiche legate alla progettazione e alla realizzazione di un robot mobile, consentendomi di scendere poi nei dettagli implementativi di diverse parti, sia di alto che di basso livello. Tutto il lavoro di tesi ha comportato una continua fase di studio e progettazione delle diverse soluzioni adottate sul robot e ha richiesto continue e consistenti prove per verificare il corretto funzionamento dell’elettronica e del software.

Rabbiati è il primo esemplare di robot calciatore progettato e costruito nel laboratorio di *Informatica e Robotica per l'Automazione* (IRALab) dell'Università di Milano – Bicocca. Il robot ha partecipato alla competizione internazionale *RoboCup 2003* svoltasi a Padova (luglio 2003), al torneo amichevole svoltosi in occasione della fiera *SMAU 2003* ed alle dimostrazioni, anche contro bambini rigoristi scelti tra il pubblico, effettuate al *Festival della Scienza* a Genova (ottobre 2003), ottenendo risultati soddisfacenti e ripagandoci così degli sforzi prodotti per la sua realizzazione.



The Seventh International Symposium and Competitions

Il Logo di RoboCup 2003, la competizione mondiale tenutasi quest'anno a Padova dal 5 al 9 luglio e a cui Rabbiati ha partecipato come portiere del Milan RoboCup Team



Il Milan RoboCup Team a RoboCup 2003

La tesi è strutturata in diversi capitoli:

- Il **Primo Capitolo** è un'introduzione a RoboCup. Spiega cos'è, lo scopo per cui è nata, come si svolge e un po' di storia della competizione internazionale;
- Il **Secondo Capitolo** riguarda l'impianto elettrico del robot: la progettazione di ogni suo circuito, le funzionalità e le scelte progettuali affrontate. Tutta l'elettronica del robot nei minimi dettagli;
- Il **Terzo Capitolo** parla del sistema di controllo di basso livello del robot. Inizialmente ci si è occupati della realizzazione di hardware dedicato. In questo capitolo viene mostrata la realizzazione pratica della scheda di controllo. In seguito viene illustrato il software di basso livello con cui è stata programmata la suddetta scheda. Il capitolo tratta inoltre gli algoritmi di controllo elaborati, le problematiche e le soluzioni del controllo in tempo reale e la progettazione generale del sistema. In questo capitolo viene infine descritto un algoritmo per il calcolo dell'Odometria del robot e come questo sia stato inserito nel sistema di controllo di basso livello;
- Il **Quarto Capitolo** descrive il sistema di visione del robot e l'algoritmo di estrazione delle percezioni indispensabili per il sistema di autolocalizzazione;
- Il **Quinto Capitolo** descrive il sistema cognitivo del robot, l'intelligenza artificiale che lo guida ed i comportamenti che sono stati creati appositamente per il suo ruolo di portiere.