



UNIVERSITÀ DI BRESCIA
FACOLTÀ DI INGEGNERIA
Dipartimento di Ingegneria dell'Informazione

Laboratorio di Robotica Avanzata **Advanced Robotics Laboratory**

Corso di Robotica Mobile
(Prof. Riccardo Cassinis)

**Interfacciamento tra
manipolatore Kawasaki e sensore
di forza applicato alla tecnica del
Leading by nose**

Elaborato di esame di:

**Tiziana Gargiulo, Davide
Ghilardi, Enrico Piantoni, Giulio
Quarenghi**

Consegnato il:

2 luglio 2014



Leading by nose di QGGP (Giulio Quarenghi, Tiziana Gargulo, Davide Ghilardi, Enrico Piantoni) è distribuito con [Licenza Creative Commons Attribution - Condividi allo stesso modo 4.0 Internazionale](https://creativecommons.org/licenses/by-sa/4.0/).

Sommario

L'implementazione della tecnica del leading by nose adottata nel progetto realizzato consiste nel guidare i movimenti del manipolatore seguendo la forza esercitata sul suo end effector. Più in dettaglio, fornendo il braccio robotico all'altezza del polso di un sensore di forza, è stato possibile sviluppare un programma che gli consentisse di muoversi nella direzione e con la velocità determinate elaborando i dati ricavati.

Il progetto complessivo è stato suddiviso in tre parti: il seguente elaborato tratta esclusivamente del passaggio intermedio, ovvero il rilevamento, l'acquisizione e la rielaborazione dei dati riguardanti la forza. Esulano dagli scopi prestabiliti il collegamento fisico del sensore al braccio e la definizione degli effettivi movimenti che il manipolatore dovrà compiere.

1. Introduzione

In alcune applicazioni pratiche in ambito robotico può risultare utile mostrare al braccio, trainandolo, il percorso da seguire per portare il proprio end effector in una determinata posizione, tenendo traccia del tragitto compiuto.

Il manipolatore utilizzato fornisce due modalità di funzionamento: può essere guidato attraverso il teach pendant oppure controllato mediante apposite istruzioni. Nel primo caso si comunica al braccio la direzione in cui muoversi grazie ad alcuni pulsanti (freccie direzionali) presenti sul terminale. Le istruzioni disponibili, invece, specificano il tipo di movimento da eseguire basandosi su precisi punti (di partenza o di arrivo) espressi secondo posizioni assolute o relative.

Nessuna delle due opzioni previste, però, permette di raggiungere l'obiettivo scelto in modo adeguato poiché, nel caso delle direzioni, sarebbe inevitabile aggiungere mosse superflue per arrivare alla posizione esatta, mentre, attraverso le istruzioni, sarebbe necessario conoscere le coordinate precise di numerosi punti lungo il tragitto, per poterlo percorrere. Si è deciso quindi di dotarsi di un sensore di forza e di posizionarlo alla flangia del polso, per poter trainare il braccio lungo il cammino voluto. Questo approccio, contrariamente ai due precedenti, consente di stabilire la traiettoria da seguire in modo intuitivo e veloce; inoltre offre maggiore accuratezza e risolve il problema della determinazione dei punti.

L'utilizzo del sensore di forza è necessario per poter rilevare i tentativi esterni di traino. Da questo la scelta del suo posizionamento all'altezza del polso, dal momento che le sollecitazioni esterne saranno esercitate sull'end effector. Il primo compito da realizzare per intraprendere il progetto è stato quindi la costruzione di un collegamento meccanico per vincolare il sensore alla flangia.

Ottenuta la struttura fisica, il passo successivo è stato l'implementazione di un meccanismo di comunicazione tra i due dispositivi, basato sullo sfruttamento di connessioni ethernet. I dati rilevati dal sensore riguardano le sei componenti della forza: le tre traslazioni lungo gli assi del riferimento cartesiano interno e i tre momenti torcenti.

Infine, per poter determinare i movimenti da far eseguire al braccio, si è reso necessario una elaborazione dei dati registrati, per ricavare da essi misure che potessero rappresentare delle velocità. Queste ultime sono state ulteriormente manipolate da un programma che, dopo opportuni controlli, determina gli spostamenti da seguire.

La trattazione che viene proposta copre le operazioni che vanno dall'interfacciamento di sensore e manipolatore alla memorizzazione in opportune variabili, interne al manipolatore industriale, dei valori puntuali delle velocità.

2. Il problema affrontato



Fig. 1 - Manipolatore industriale guidato attraverso l'applicazione di una forza sul suo end effector

Il manipolatore industriale Kawasaki è privo di dispositivi in grado di misurare eventuali forze esterne applicate al suo end effector. Per poter realizzare l'obiettivo precedentemente descritto era quindi necessario fornirlo di uno strumento atto a tale scopo. Si è deciso di sopperire a questa mancanza attraverso l'utilizzo di un sensore di forza, da inserire tra il polso e la pinza del braccio.

Da questa premessa è derivato il problema da affrontare: braccio e sensore sono dispositivi disgiunti perciò necessitano di un apposito sistema di comunicazione.

Entrambi gli strumenti da interfacciare hanno la possibilità di sfruttare una connessione ethernet, il manipolatore attraverso il calcolatore che lo controlla ed il sensore tramite un dispositivo di elaborazione e trasmissione ad esso collegato. Nella pratica, l'obiettivo da realizzare è stato quindi sviluppare un meccanismo che consentisse di trasmettere i dati rilevati dal sensore al Kawasaki, mediante la rete interna del laboratorio, applicando ad essi una opportuna rielaborazione per convertirli da misure di forza a valori di velocità.

3. La soluzione adottata

Il progetto consta nella realizzazione di due programmi all'interno di un'architettura client-server. L'applicazione server, eseguita sul calcolatore di cui il robot è provvisto, rimane in ascolto sulla porta UDP prestabilita in attesa di comandi. Nello specifico, attende pacchetti il cui contenuto è formattato come una stringa numerica di 42 caratteri, 7 caratteri per ogni componente della velocità calcolata. Le componenti sono ordinate nel seguente modo: tre velocità lineari (rispettivamente lungo l'asse x, y e z) e tre velocità angolari (rispettivamente relative all'asse x, y e z).

Il pacchetto UDP viene ricevuto dal calcolatore del braccio e il suo contenuto viene suddiviso tramite l'istruzione **\$MID** in sottostringhe numeriche che verranno convertite tramite il comando **VAL** in numeri.

Tali valori verranno in seguito riposti all'interno delle variabili *morz_sens_vx*, *morz_sens_vy*, *morz_sens_vz*, *morz_sens_vrx*, *morz_sens_vry*, *morz_sens_vrz*. Nel caso i valori cambino rispetto allo step precedente si invierà un segnale, tramite il comando **SIGNAL(morz_data_sig)**, il quale verrà gestito dal programma residente sul robot Kawasaki.

La controparte client si articola invece in due sottoprocessi principali, gestendo la mutua esclusione mediante l'utilizzo di semafori binari. Il primo sottoprocesso si occupa della ricezione dei dati dal sensore, dell'applicazione di un filtraggio allo scopo di ridurre il rumore e della definizione di una soglia

sotto la quale i dati vengono considerati nulli. Il secondo sottoprocesso riguarda il calcolo delle velocità derivate dalle forze lineari e torcenti e al relativo invio destinato al manipolatore Kawasaki.

La suddivisione in sottoprocessi consente al programma di parallelizzare la ricezione di dati dal sensore ed il calcolo delle velocità in modo da aumentare l'efficienza del prodotto.

Come accennato in precedenza il sottoprocesso dedicato alla ricezione dei dati dal sensore si occupa anche di effettuarne un filtraggio. Questo viene effettuato ricevendo un certo numero di pacchetti (fissato a tempo di compilazione) e fornendo al secondo sottoprocesso la media di tali dati.

Per quanto concerne il sottoprocesso dedicato al computo dei movimenti, partendo dalle forze, esso ricava le velocità tramite l'uso delle seguenti formule:

$$V_l = \frac{f}{m} \Delta t$$

$$V_a = \frac{f}{i} \Delta t$$

dove:

V_l : rappresenta le velocità lineari

V_a : rappresenta le velocità angolari

f : rappresenta le forze (fornite dal primo sottoprocesso)

m : rappresenta un coefficiente di scala sulle forze lineari

i : rappresenta un coefficiente di scala sulle forze torcenti

Δt : rappresenta l'intervallo di tempo passato dal calcolo precedente

Dopo una valutazione sperimentale si è deciso di mantenere il modello mostrato a sfavore di uno più complesso che tenesse conto anche di un'inerzia derivata dal moto di un grave dotato di massa. Questa scelta è stata dettata dal fatto che impostando il comportamento del manipolatore come un sistema con inerzia il suo movimento risultava eccessivamente prolungato anche quando le forze esercitate sull'end effector erano cessate. Dopo svariati esperimenti, condotti con differenti parametrizzazioni, il risultato non era quello desiderato, ovvero quello di rendere il movimento proporzionale alle forze istantanee.

3.1. I linguaggi di programmazione utilizzati

Lo sviluppo del progetto si è suddiviso in due parti:

- l'implementazione del programma di acquisizione, elaborazione ed infine invio dei dati al manipolatore sviluppato tramite il linguaggio C,
- la creazione del software per l'interfacciamento dei dati inviati mediante la rete e il robot Kawasaki realizzato attraverso il linguaggio AS.

3.2. Elaborazioni sui dati acquisiti

Di seguito vengono mostrati degli esempi di acquisizioni di valori. Nel primo grafico si mostrano le forze lineari mentre sul secondo si mostrano le torcenti.

Le misure sono state effettuate con il sensore montato all'altezza del polso. Le varie prove sono state fatte applicando o meno forze sull'end effector e utilizzando o meno il pacchetto di bias. Da una prima analisi si può vedere come gli andamenti con il filtraggio (valor medio su 3 valori) sono migliori rispetto a quelli presi senza alcun tipo di validazione dei dati (confronto tra figura 2 e 5). In più si può notare che utilizzando il pacchetto di bias le traiettorie, sia di quelle lineari sia le torcenti, hanno range più contenuti (confronto tra figura 2 e 3).

Infine aggiungiamo due dinamiche ottenute applicando forze su tutte e sei le componenti del sensore con e senza filtraggio (figura 6 e 4).

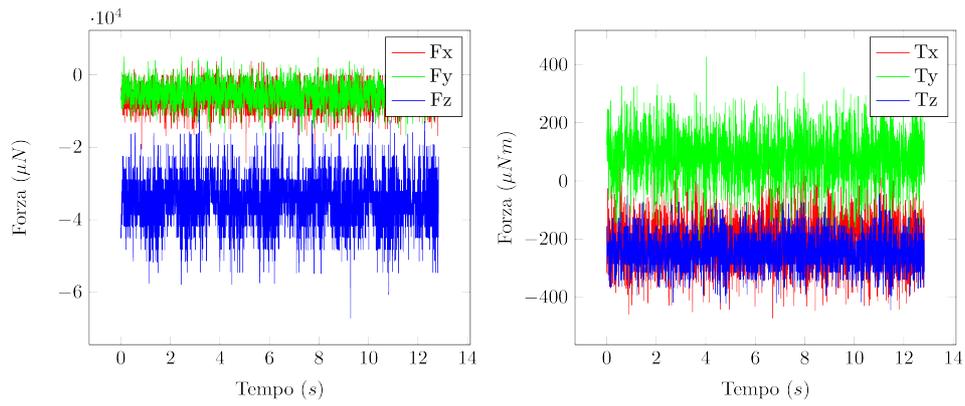


Fig. 2 - Misurazioni effettuate senza sottoporre l'end effector a nessuna forza tranne quella gravitazionale, senza filtraggio e senza inviare il pacchetto di bias

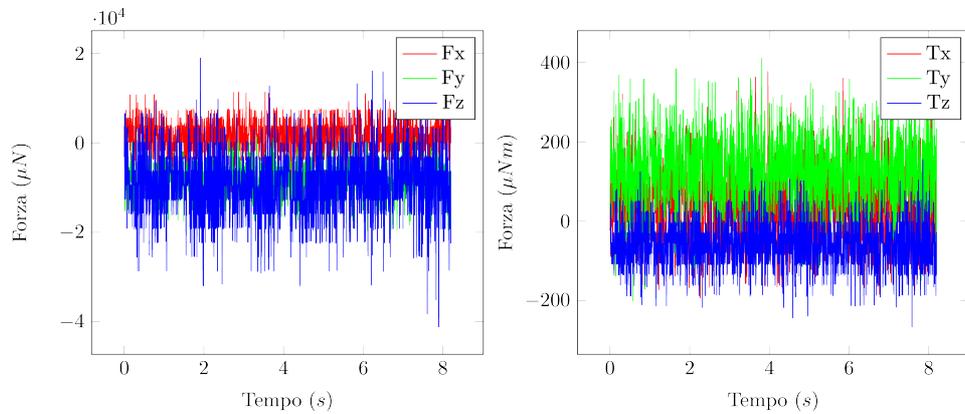


Fig. 3 - Misurazioni effettuate senza sottoporre l'end effector a nessuna forza tranne quella gravitazionale, senza filtraggio e con l'invio del pacchetto di bias

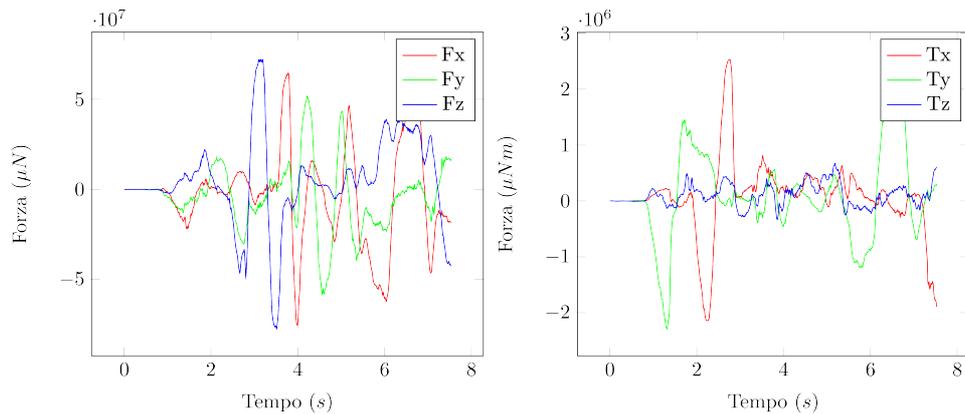


Fig. 4 - Misurazioni effettuate sottoponendo l'end effector ad alcune forze su tutte le sue componenti, senza filtraggio e con l'invio del pacchetto di bias

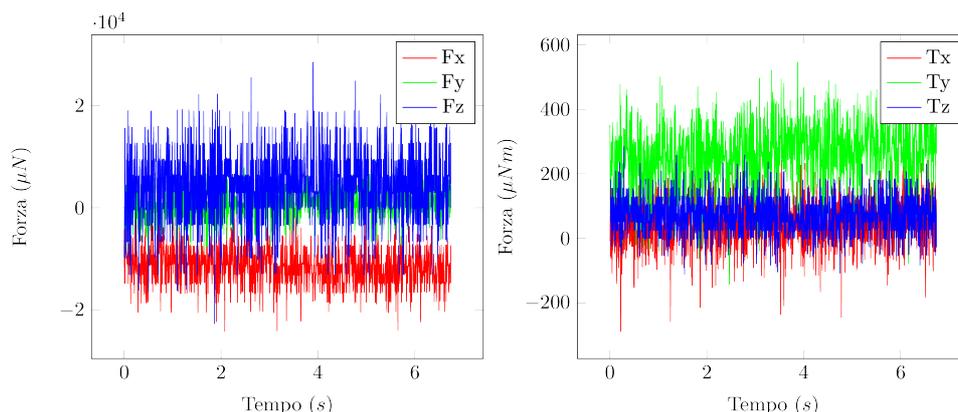


Fig. 5 - Misurazioni effettuate senza sottoporre l'end effector a nessuna forza tranne quella gravitazionale, con filtraggio e con l'invio del pacchetto di bias

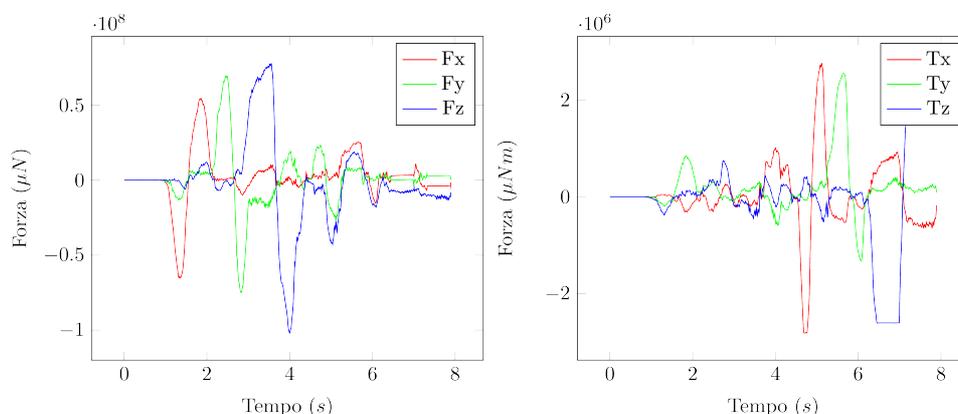


Fig. 6 - Misurazioni effettuate sottoponendo l'end effector ad alcune forze su tutte le sue componenti, con filtraggio e con l'invio del pacchetto di bias

4. Modalità operative

Per dare avvio al sistema, bisogna innanzitutto avviare il programma "leadingbynose" che può essere avviato su una qualsivoglia macchina provvista della libreria pthread e di almeno una connessione alla rete (consigliata porta di rete ethernet). Il programma, sviluppato e testato su sistemi linux (64 bit), realizza la connessione tra il sensore ed il braccio, agendo da client verso entrambi: richiede al sensore i dati rilevati, li elabora e invia i risultati al Kawasaki.

Per rendere completamente operativo l'intero meccanismo, bisogna accendere il manipolatore ed avviare su di esso i programmi *qggp_sensorUdp* e *morz_follow*, il primo attraverso il comando **pc exec qggp_sensorUdp.as** ed il secondo con **exec morz_follow.as**. Questa diversificazione deve essere attuata per ovviare al limite imposto dal calcolatore del braccio di poter eseguire un solo programma alla volta. La prima procedura riceve i dati relativi alle velocità dal programma *leadingbynose* sotto forma di stringa, li converte e li memorizza in particolari variabili numeriche. La seconda, attivata da un opportuno segnale della prima, legge i valori precedentemente salvati e stabilisce i movimenti del manipolatore.

A questo punto il sistema è in azione: applicando una forza all'estremità del braccio il robot si muoverà di conseguenza.

4.1. Componenti necessari

Sono necessari:

- Un manipolatore industriale; nello specifico è stato usato il *Kawasaki RS03N* (visibile in figura 7).

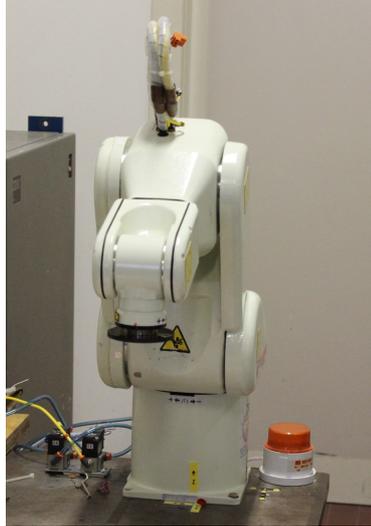


Fig. 7 - Il robot Kawasaki presente nel laboratori ARL

- Un pc linux con librerie *pthread* (standard posix)
- Un sensore di forza con dimensioni compatibili con quelle del polso, dotato di possibilità di connessione alla rete ethernet e che fornisca le sei componenti della forza. In particolare è stato utilizzato il *MINI40* della *ATI industrial automation* con il relativo *Net Box* (visibile nella figura 8).



Fig. 8 - Il sensore di forza con il relativo Net Box presente in laboratorio ARL

- Un'infrastruttura fisica di rete

4.2. Modalità di installazione

Per quanto riguarda l'hardware:

- connettere tramite l'adattatore *M12-RJ45* (visibile nella figura 10) il *Net Box* (visibile nella figura 9) alla rete e connettere l'alimentatore tramite l'apposito cavo.



Fig. 9 - Il Net Box visto frontalmente

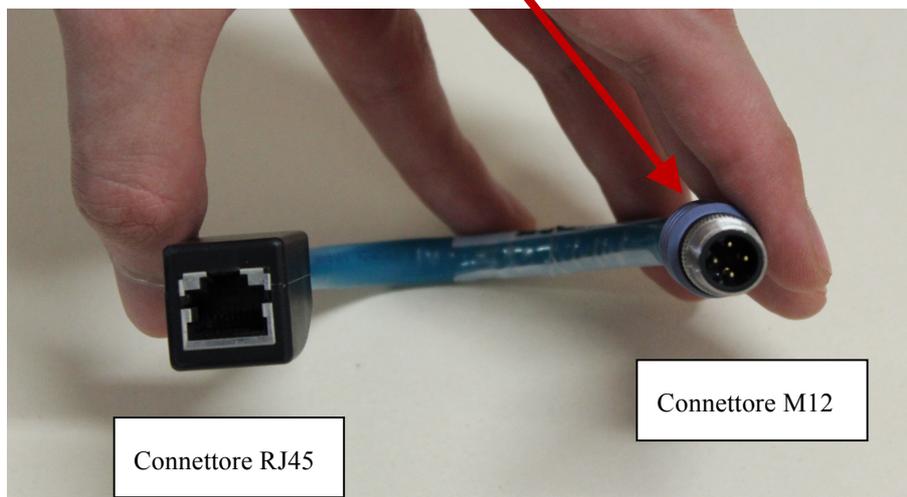


Fig. 10 - Adattatore da connettore M12 a RJ45

- collegare il sensore di forza (visibile nella figura 12) con il Net Box (visibile nella figura 11)

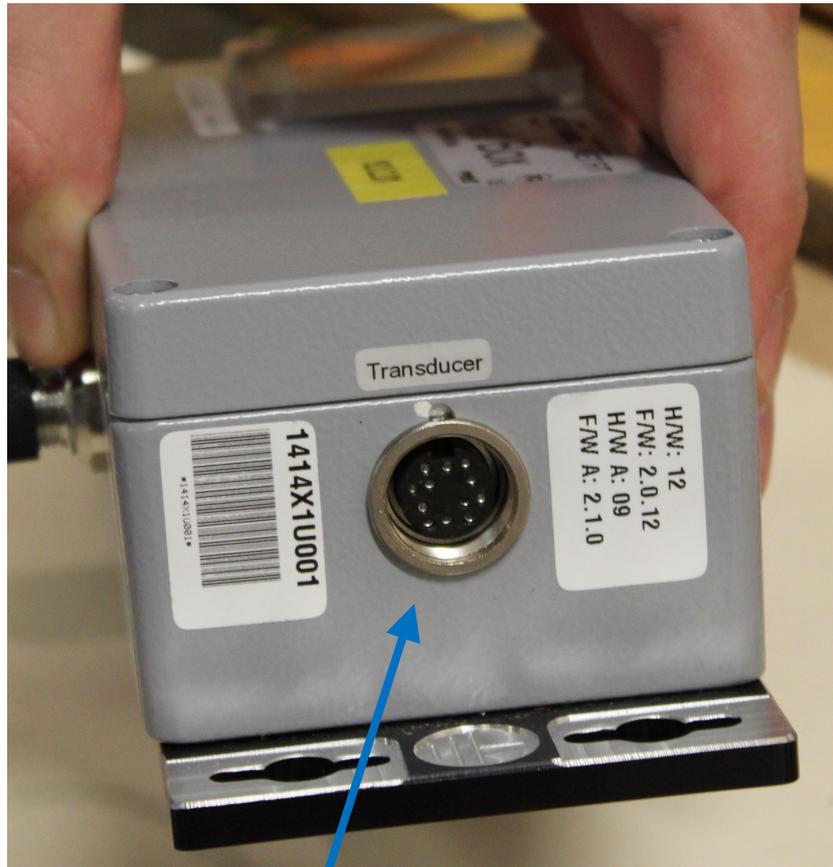


Fig. 11 - Il Net Box visto lateralmente, con la connessione per il sensore



Fig. 12 - Il sensore di forza a destra con il suo connettore

Infine tutti i componenti risulteranno come mostrato nell'immagine 13.

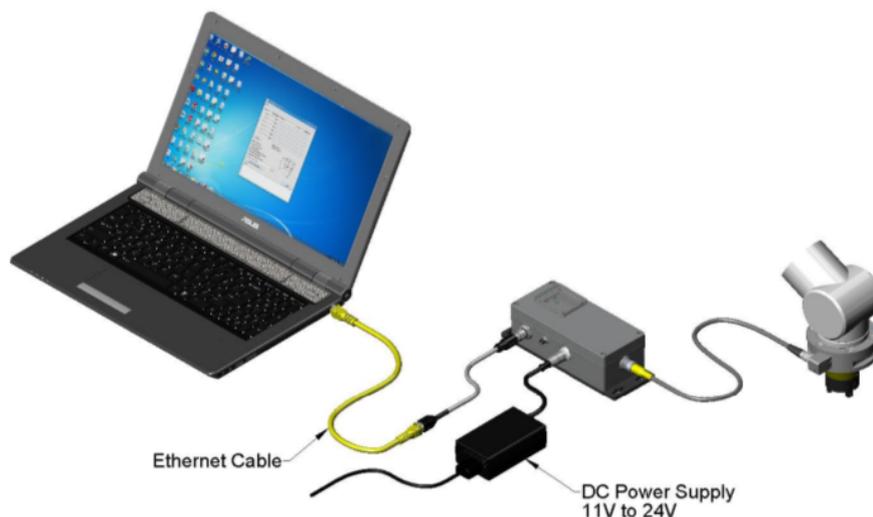


Fig. 13 - Infrastruttura finale

Per quanto riguarda il software:

- caricare il sorgente *qggp_sensorUdp* sul computer del robot Kawasaki ed avviarlo in modalità **pc exec** insieme al programma *morz_follow* avviato in modalità **exec**
- caricare il software *leadingbynose* su un computer linux con librerie **pthread**, il software come già accennato, è stato testato su un computer con architettura di 64 bit. Al fine di generare l'eseguibile bisogna usare il comando **cmake**. Una volta creato l'eseguibile lo si può avviare.

4.3. Configurazione

Al primo utilizzo, dal manuale del sensore si può ricavare l'indirizzo IP di default del *Net Box*, grazie al quale è possibile accedere alla pagina di configurazione tramite browser. Per riuscire a connettersi al sensore è necessario allacciare direttamente il *Net Box* al computer, dopo aver impostato l'indirizzo IP della macchina a **192.168.1.100** e la maschera di sottorete a **255.255.255.0**, ed accedere, nel caso del dispositivo in laboratorio, all'indirizzo **192.168.1.1**.

Se la procedura è andata a buon fine, viene mostrata la pagina principale di configurazione del sensore, dalla quale è possibile accedere alla sezione *communications*.

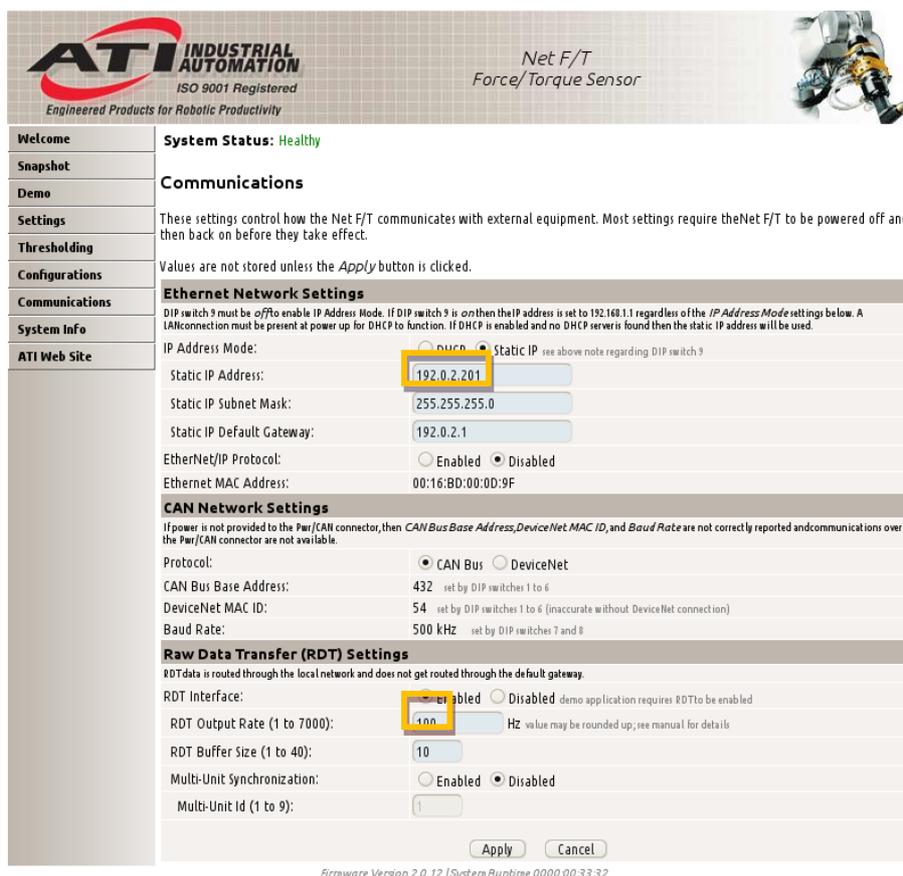


Fig. 14 - Pagina di configurazione del sensore

Dalla pagina visibile nella figura 14 è possibile impostare l'indirizzo IP del sensore e modificare la frequenza massima di invio dei pacchetti di risposta.

Per fare in modo che il sensore utilizzi realmente l'indirizzo scelto invece che quello di default bisogna aprire il Net Box, rimuovendo il suo coperchio (figura 15).



Fig. 15 - Net Box con il coperchio smontato

e spostare l'interruttore del piedino 9 del dip switch da ON a OFF, come in figura 16.

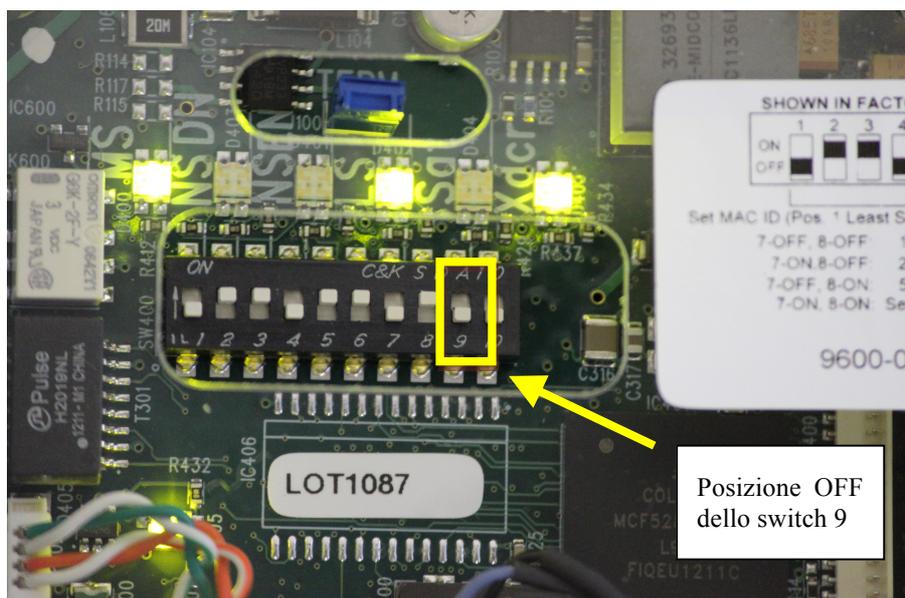


Fig. 16 - Il pin 9 dello switch impostato su OFF

Nel nostro elaborato l'indirizzo IP è stato impostato a **192.168.2.201** e la frequenza di risposta dei pacchetti a **100 Hz**.

Per controllare che ciò che è stato realizzato funzioni correttamente si può sfruttare la sezione *demo* della pagina del sensore a cui ci si è precedentemente collegati, che mette a disposizione un programma che offre una rappresentazione grafica dei valori delle forze registrate.

Inoltre attraverso il comando

```
./leadingbynose <IPsensor> <IProbot> <portRobot>
```

è possibile variare il valore di default di

- indirizzo IP del sensore,
- indirizzo IP e porta del robot,

per l'invocazione del programma che si sta effettuando.

4.4. Avvertenze

- Per l'utilizzo del progetto occorre oltrepassare la barriera di sicurezza, entrando nella zona in cui operano i due manipolatori.
- All'avvio del programma *morz_follow* il braccio esegue alcuni movimenti automatici per enfatizzare il fatto che il Kawasaki è pronto per essere mosso .

5. Conclusioni e sviluppi futuri

I progetti futuri riguarderanno altri lavori in cui il braccio potrà svolgere, attraverso questa sua nuova capacità di interazione con il mondo. In particolare potrà essere guidato rapidamente e intuitivamente in posizioni difficili da raggiungere attraverso le interazioni classiche e potrà avere un feedback sullo sforzo che sta esercitando il suo end effector. Una possibile implementazione riguarda l'utilizzo del braccio come supporto durante un'operazione in ambito medico, in particolare endoscopie.

Bibliografia

- [1] ATI industrial automation: "Net F/T", *Network Force/Torque Sensor System* Documento 9620-05-NET FT, febbraio 2013.
- [2] Kawasaki Heavy Industries: "AS Manual", *Kawasaki robot controller E series* Documento 90209-1022DIA, marzo 2010.
- [3] Kawasaki Heavy Industries: "TCP/IP communication manual", *Kawasaki robot controller E series* Documento 90210-1248DEA, marzo 2009.

Indice

SOMMARIO	1
1. INTRODUZIONE	1
2. IL PROBLEMA AFFRONTATO	2
3. LA SOLUZIONE ADOTTATA	2
3.1. I linguaggi di programmazione utilizzati	3
3.2. Elaborazioni sui dati acquisiti	3
4. MODALITÀ OPERATIVE	5
4.1. Componenti necessari	6
4.2. Modalità di installazione	7
4.3. Configurazione	10
4.4. Avvertenze	12
5. CONCLUSIONI E SVILUPPI FUTURI.....	12
BIBLIOGRAFIA	12
INDICE	13