

Sistemi Operativi per la Robotica

Enzo Mumolo

Sistemi Operativi per la Robotica

- www.units.it/~mumolo □ sito web del corso (in preparazione)
- Sito web □ programma preliminare, materiale, provetta, risultati etc.
- mumolo@units.it oppure emumolo@computer.org □ indirizzi docente
- 040.558.3861 □ telefono ufficio docente

- orari corso: lunedì dalle 9 alle 11, mercoledì 14-17
- Aule: lunedì: aula Insiel, mercoledì fisica tecnica
- esame: 40% provetta in aula + 60% tesina

Sistemi robotici



Sistemi robotici



Introduzione ai sistemi robotici

- Sistemi meccanici che possono funzionare autonomamente
 - ❑ Sistemi meccanici: sistemi costruiti dall'uomo
 - ❑ Funzionamento autonomo: i sistemi robotici possono prendere decisioni autonomamente, cioè non sotto il controllo di un operatore, in funzione delle letture sensoriali
- La difficoltà principale del funzionamento autonomo sta nel fatto che l'ambiente è solo approssimativamente conoscibile

Introduzione ai sistemi robotici

- Motivazione ai sistemi robotici: ausilio all'uomo
 - in ambienti ostili e in attività noiose o pesanti
- Motivazione ai sistemi robotici: svolgere attività che non può svolgere l'uomo (entrare in luoghi piccoli, attenzione, stanchezza, incendio o gas ...)
- Alcuni campi di attività collegati alla robotica:
 - Architetture software (cognitive model, real time, efficiency...)
 - Computer Vision (pattern recognition...)
 - Haptics & Virtual Reality (grasping, interaction...)
 - Medical Applications (surgery, rehabilitation, prosthesis..)
 - Mobile Robotics (autonomy, walking..)
 - etc

Introduzione ai sistemi robotici

- Modalità operative
 - Terra
 - Aria
 - Superficie
 - Sottomarini
- Componenti
 - Mobilità
 - Percezione
 - Controllo
 - Alimentazione
 - Comunicazione

Introduzione ai sistemi robotici

- Robot terrestri:
 - Robot mobile
 - DARPA Grand Challenge: è una gara tra veicoli autonomi su percorsi e terreno reali; ogni partecipante deve obbedire a regole per quanto riguarda la distanza percorsa e la velocità
 - NASA MER (Mars Exploration Rovers)
 - Umanoide
 - Honda P3, Sony Asimo
 - Animaloide
 - Sony Aibo
 - Robot Miniaturizzati

Introduzione ai sistemi robotici

- Robot aerei
 - Ad ali fisse (Predator)
 - VTOL (veicoli a decollo verticale)
 - UCAV (Unmanned Combat Air Vehicles)
- Robot sottomarini
 - Telecontrollati
 - Autonomi
- Robot che si muovono sulla superficie del mare

Introduzione ai sistemi robotici

- Robot e AI
 - Interpretazione dei sensori
 - Interazione uomo-robot
 - Incertezza
 - Apprendimento
- Qualche area in AI/robotica
 - Rappresentazione della conoscenza
 - Linguaggio naturale
 - Ricerca
 - Visione
 - Pianificazione

Implementazione

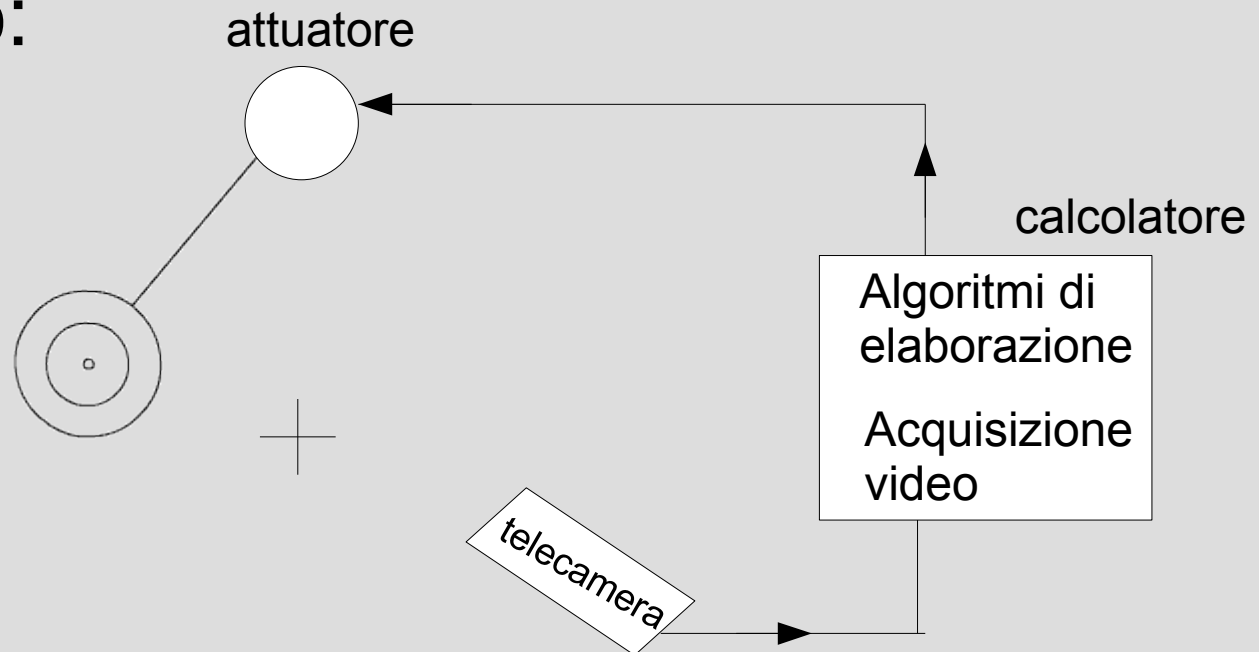
- Caratteristiche generali
 - Sensori
 - Attuatori
 - Architetture
- Sistemi operativi in tempo reale
- Programmazione in Linux RTAI
- Programmazione in Java Real Time

Pianificazione e navigazione

- Rappresentazione dello spazio
- Ricerca di cammini ottimi in un grafo
- Algoritmi di ricerca (Dijkstra, A*)
- Algoritmi di navigazione
 - Localizzazione (Dead-reckoning)
 - Pianificazione della traiettoria
 - Pianificazione mediante algoritmi genetici
- Esercizi ed esempi

Robotica e Tempo reale

- Sistemi in tempo reale: il risultato dipende non solo dal risultato numerico ma anche dal tempo nel quale viene ottenuto.
- Esempio:



Robotica e tempo reale

- I vincoli temporali devono essere garantiti: il tempo di elaborazione e il tempo dell'evento da gestire devono essere noti
- Condizioni necessarie (ma non sufficienti):
 - il sistema deve essere predicibile
 - Condizione necessaria: il tempo di risposta del sistema deve essere molto basso
 - Inoltre: il sistema di calcolo deve essere veloce

Robotica e Tempo Reale

- Le applicazioni di questo tipo sono moltissime, per es.:
 - Telefoni cellulari
 - Lettori DVD e dischi fissi
 - Robotica
 - Reti e sistemi distribuiti
 - Linee di produzione automatizzate
 - Sistemi militari
 - Avionica
 - etc...

Implementazione dei sistemi dedicati

- Un sistema opera in tempo reale soltanto se fornisce i risultati attesi entro predefiniti limiti temporali
- Real-time con vincoli temporali
 - Hard RT se la relativa deadline deve sempre essere rispettata
 - Periodico con frequenza di esecuzione costante
 - Sporadico in caso contrario
 - Soft RT se la relativa deadline può essere disattesa in condizioni di temporaneo sovraccarico
 - Periodico con frequenza di esecuzione costante
 - Aperiodico in caso contrario

Implementazione dei sistemi dedicati

- I sistemi dedicati ('embedded') sono quelli che fanno parte integrante di un dispositivo
 - Ad esempio: un calcolatore PDA, un telefono cellulare, una segreteria telefonica elettronica, una centralina di controllo di un'automobile sono tutti sistemi che vengono attivati all'accensione del dispositivo di cui fanno parte integrale
 - A seconda dei vincoli temporali richiesti dal dispositivo i sistemi possono essere considerati in tempo reale o meno

Implementazione dei sistemi dedicati

- Alcuni SO in tempo reale
 - VxWorks (Motorola, Pentium, StrongArm, Arm)
 - Windows CE .NET (ARM, StrongArm, XScale, MIPS, Pentium)
 - QNX Neutrino RTOS (Pentium, Power PC, ARM, StrongArm, XScale, MIPS, SH-4)
 - pSOSystem 3
 - Arx RTOS <http://arx.snu.ac.kr/html/overview-arx.en.html>
 - AvSys <http://www.avocetsystems.com>
 - CMX RTOS <http://www.cmx.com>
 - Linux Real Time:
 - uCLinux → eliminazione di funzionalità dal kernel Linux standard
 - Montavista's Hard Hat Linux, KURT → modifiche al kernel Linux standard
 - RTLinux, RTAI → Linux all'interno di un sistema real-time

Implementazione dei sistemi dedicati

- Real-Time Java
 - Vantaggi di OOP in Java rispetto a C++:
 - Semplicità
 - Classi disponibili
 - Concorrenza
 - ...
 - Svantaggi di Java:
 - Alto livello di astrazione → bassa predicibilità
 - Parti critiche: thread, monitor, accesso in memoria, rilascio della memoria dinamica (garbage collection)
 - Real-Time Java:
 - Aumenta la predicibilità dei tempi di esecuzione
 - Compatibile con Java tradizionale

Programma del Corso

- Sistemi in Tempo Reale
- Sistemi Intelligenti
- Robotica
- Simulazione di applicazioni robotiche
- Programmazione dei sistemi robotici
- Studio e Sviluppo di sistemi robotici reali

Programma del Corso

- Sistemi in Tempo Reale
 - Introduzione
 - Schedulazione in tempo reale
 - Protocolli di accesso alle risorse condivise

Programma del Corso

- **Sistemi Intelligenti** (Russell, Norvig, “Artificial Intelligence-A Modern Approach”)
 - Introduzione
 - Agenti intelligenti
 - Ricerca informata, ricerca non informata
 - Logica del primo ordine
 - Incertezza, insiemi Fuzzy
 - Apprendimento
 - Reti neurali
 - Apprendimento con rinforzo: ottimizzazione genetica, ottimizzazione Simulated Annealing

Programma del Corso

- Robotica (“Robot Motion Planning” J.C. Latombe,
“Computational principles of mobile robotics”, Dudek, Jenkin)
 - Architetture cognitive
 - Localizzazione
 - Navigazione
 - Spazio delle configurazioni
 - Rappresentazione simbolica

Programma del Corso

- Sistemi a coda d'attesa
 - Analisi e progetto di semplici architetture robotiche mediante la teoria delle code d'attesa

Programma del Corso

- Simulazione di applicazioni robotiche
 - Simulatore OpenSource
 - Realizzazione di semplici algoritmi di robotica

Programma del Corso

- Programmazione dei sistemi robotici
 - Sistema operativo Linux RTAI
 - Principi, programmazione
 - Linguaggio RTJAVA
 - Principi, programmazione

Programma del Corso

- In parallelo alle presentazioni 'teoriche' il corso prevede lo sviluppo di esercitazioni pratiche su piattaforme reali
- Gli studenti vengono divisi in gruppi, ogni gruppo lavora su una applicazione reale (tesina) che viene presentata alla fine del corso.