Ingeniería Civil en Informática

Taller de Robótica

Juan Carlos Figueroa Durán
jcfigueroa@ubiobio.cl

Motivation

Los entornos inteligentes están destinados a mejorar la experiencia y el rendimiento de la tareas de sus habitantes.

- Automatizar funciones en el hogar
- Proporcionar servicios a los habitantes

- Las desiciones provenientes de quienes toman las desiciones en estos entornos deben ser ejecutadas.
  - Las desiciones requieren acciones a realizar en dispositivos
  - Con frecuencia, las decisiones no son interacciones elementales entre dispositivos, sino que comandos relativamente complejos
  - Las decisiones definen puntos revision que deben ser alcanzados.
  - Decisiones que requieren ser ejecutadas completamente.

Robotics in Intelligent Environments

- Control del entorno físico
  - Cortinas automatizadas
  - Termostatos y conductos de calefacción
  - Puertas automatizadas
  - Particionado automático de habitaciones
- Robots de servicio personal
  - Limpieza de casas
  - Cortadora de pasto
  - Asistencia a adultos mayores y discapacitados
  - Aistentes de oficina
  - Servicios de seguridad

Autonomous Robots

- El control de robots autónomos involucra una serie de subtareas
  - Understanding and modeling of the mechanism
    - Kinematics, Dynamics, and Odometry
  - Reliable control of the actuators
    - Closed-loop control
  - Generation of task-specific motions
    - Path planning
  - Integration of sensors
    - Selection and interfacing of various types of sensors
  - Coping with noise and uncertainty
    - Filtering of sensor noise and actuator uncertainty
  - Creation of flexible control policies
    - Control has to deal with new situations

Problems

- Las técnicas de programación tradicional carecen de capacidades necesarias para la creación de robot industriales en ambientes inteligentes
  - Only limited on-line sensing
  - No incorporation of uncertainty
  - No interaction with humans
  - Reliance(Confianza) on perfect task information
  - Complete re-programming for new tasks

Traditional Industrial Robots

- El control tradicional de robots industriales usa brazos robotizados y movimientos en gran parte pre configurados
  - Programming using “teach box”
  - Repetitive tasks
  - High speed
  - Few sensing operations
  - High precision movements
  - Pre-planned trajectories and task policies
  - No interaction with humans

Problems

- Traditional programming techniques for industrial robots lack(carece) key capabilities necessary in intelligent environments
  - Only limited on-line sensing
  - No incorporation of uncertainty
  - No interaction with humans
  - Reliance on perfect task information
  - Complete re-programming for new tasks

Requirements for Robots in Intelligent Environments

- Autonomia
  - Robots have to be capable of achieving task objectives without human input
  - Robots have to be able to make and execute their own decisions based on sensor information
- Interfaces Humano-Robot intuitivas
  - Use of robots in smart homes can not require extensive user training
  - Commands to robots should be natural for inhabitants
- Adaptación
  - Robots have to be able to adjust to changes in the environment

Robots for Intelligent Environments

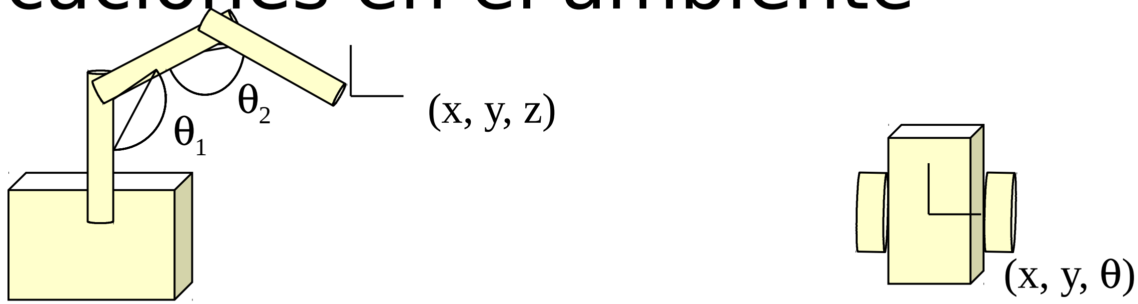
- Robots de servicio
  - Security guard
  - Delivery
  - Cleaning
  - Mowing (Siega)
- Robots asistentes
  - Mobility
  - Services for elderly and People with disabilities

Autonomous Robot Control

- Para programar robots que realicen tareas de manera autónoma, se deben abordar varias tareas:
  - Modelamiento de mecanismos del robot
    - Kinematics, Dynamics
  - Seleccion de los sensores del robot
    - Active and passive proximity sensors
  - Controladores de bajo nivel para los actuadores
    - Closed-loop control
  - Arquitectura de control
    - Traditional planning architectures
    - Behavior-based control architectures
    - Hybrid architectures

Modeling the Robot Mechanism

- Cinematica (Forward kinematics) describe como la configuraciones de los ángulos de las articulaciones de los robot se traducen en ubicaciones en el ambiente

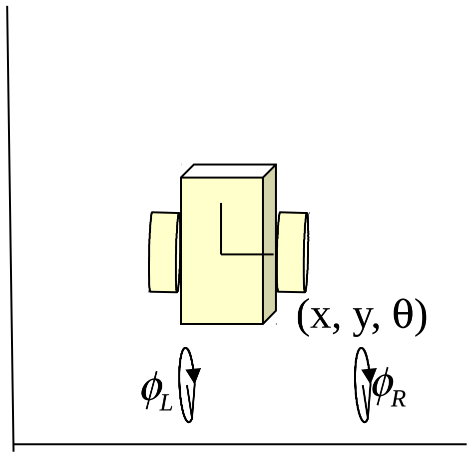


- Cinematica inversa calcula las configuraciones de los angulos de las articulaciones necesarios para alcanzar in punto en particular del espacio.
- El calculo jacobiano, como la velocidad y configuracion de los actuadores se traduce en la velocidad del robot

Mobile Robot Odometry

- En los robots mobiles la misma configuración de angulos en las articulaciones no identifican una unica locación.
  - To keep track of the robot it is necessary to incrementally update the location (this process is called odometry or dead reckoning)

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}^{t+\Delta t} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}^{t} + \begin{bmatrix} v_x \\ v_y \\ \varpi \end{bmatrix} \Delta t$$



- Example: A differential drive robot

$$v_x = \cos(\theta) \frac{r(\dot{\varphi}_L + \dot{\varphi}_R)}{2}, v_y = \sin(\theta) \frac{r(\dot{\varphi}_L + \dot{\varphi}_R)}{2}$$

$$\varpi = \frac{r}{d} \left(\dot{\varphi}_L - \dot{\varphi}_R \right)$$

Actuator Control

- Para obtener que un actuador particular de un robot vaya a una posición particular es importante aplicar la correcta cantidad de fuerza y torque.
  - Requiere conocimeinto de la dinamica del robot
    - Mass, inertia, friction
    - For a simplistic mobile robot: $F = m\,a + B\,v$
  - Frequently actuators are treated as if they were independent (i.e. as if moving one joint would not affect any of the other joints).
  - El control más común es el PD-control (Control proprocional, diferencial)
    - For the simplistic mobile robot moving in the x direction:

$$F = K_P\left(x_{desired} - x_{actual}\right) + K_D\left(v_{desired} - v_{actual}\right)$$

Robot Navigation

La planificación de la ruta aborda la tarea de calcular una trayectoria para el robot de manera que alcance la meta deseada sin chocar con obstáculos

- Optimal paths are hard to compute in particular for robots that can not move in arbitrary directions (i.e. nonholonomic robots)
- Shortest distance paths can be dangerous since they always graze obstacles
- Paths for robot arms have to take into account the entire robot (not only the endeffector)

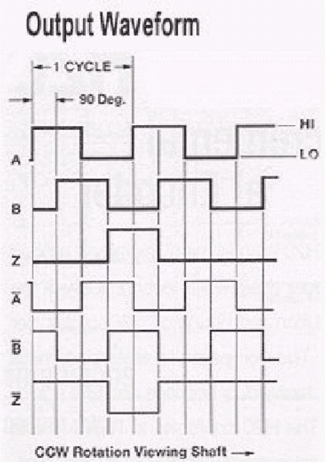
Sensor-Driven Robot Control

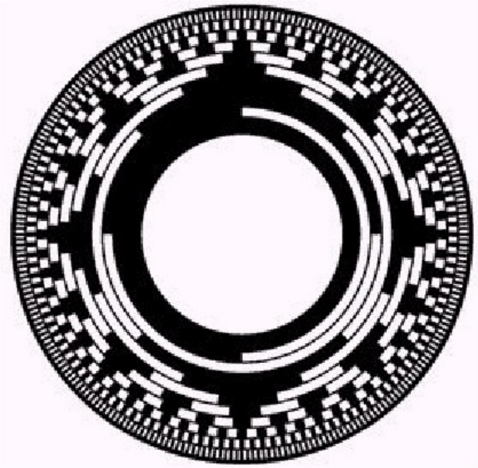
Para lograr con precisión una tarea en un entorno inteligente, un robot tiene que ser capaz de reaccionar dinámicamente a los cambios de su entorno.

- Robots need sensors to perceive the environment
- La mayoría de los robots utiliza un conjunto de diferentes sensores.
  - Different sensors serve different purposes
- Information from sensors has to be integrated into the control of the robot

Robot Sensors

- Sensores internos para conocer la configuracion del robot
  - Encoders measure the rotation angle of a joint





  - Limit switches detect when the joint has reached the limit

Robot Sensors

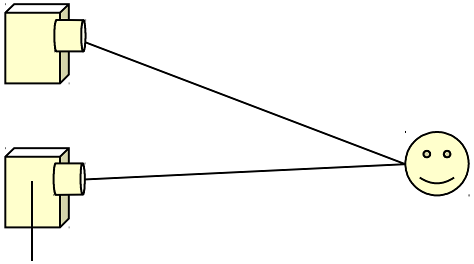
- Proximity sensors are used to measure the distance or location of objects in the environment. This can then be used to determine the location of the robot.
  - Infrared sensors determine the distance to an object by measuring the amount of infrared light the object reflects back to the robot

  - Ultrasonic sensors (sonars) measure the tin ultrasonic signal takes until it returns to the

  - Laser range finders determine distance by measuring either the time it takes for a lase beam to be reflected back to the robot or b measuring where the laser hits the object

Robot Sensors

- Vision artificial entrega al robot la capacidad de la observación pasiva del ambiente
  - Stereo vision systems provide complete location information using triangulation

  - Sin embargo, la visión artificial en muy compleja de lograr
    - Correspondence problem makes stereo vision even more difficult

Uncertainty in Robot Systems

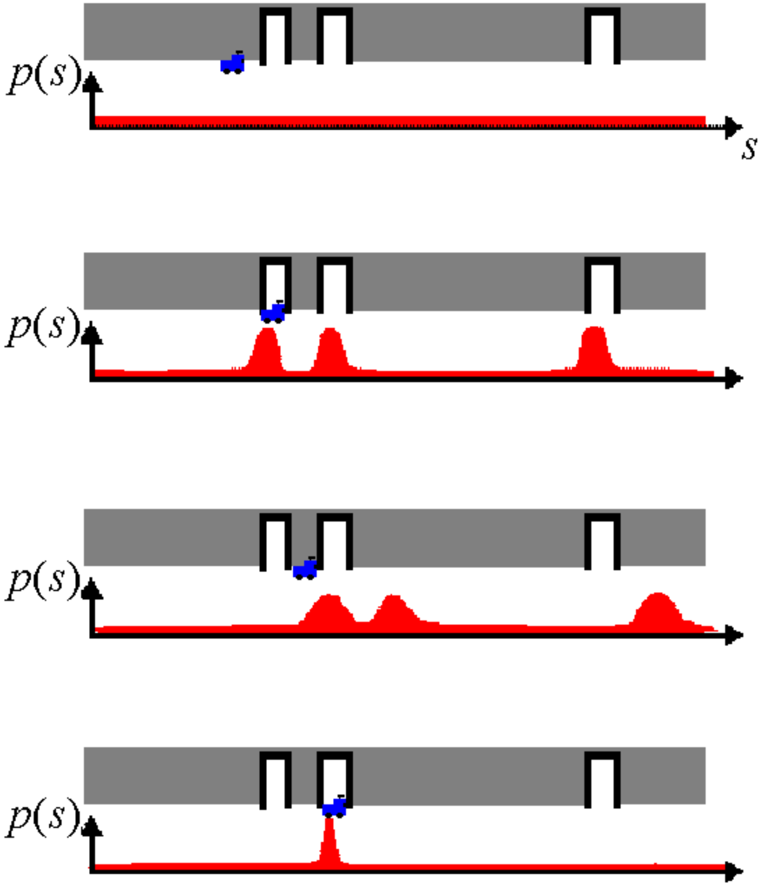
- Los sistemas roboticos en ambientes inteligentes debe lidiar con imprecisiones de los sensores (ruido) e incertidumbre
  - Incertidumbre de sensores
    - Sensor readings are imprecise and unreliable
  - Ambiente no completamente observable
    - Various aspects of the environment can not be observed
    - The environment is initially unknown
  - Incertidumbre de las acciones
    - Actions can fail
    - Actions have nondeterministic outcomes

Probabilistic Robot Localization

- Explicit reasoning about Uncertainty using Bayes filters:

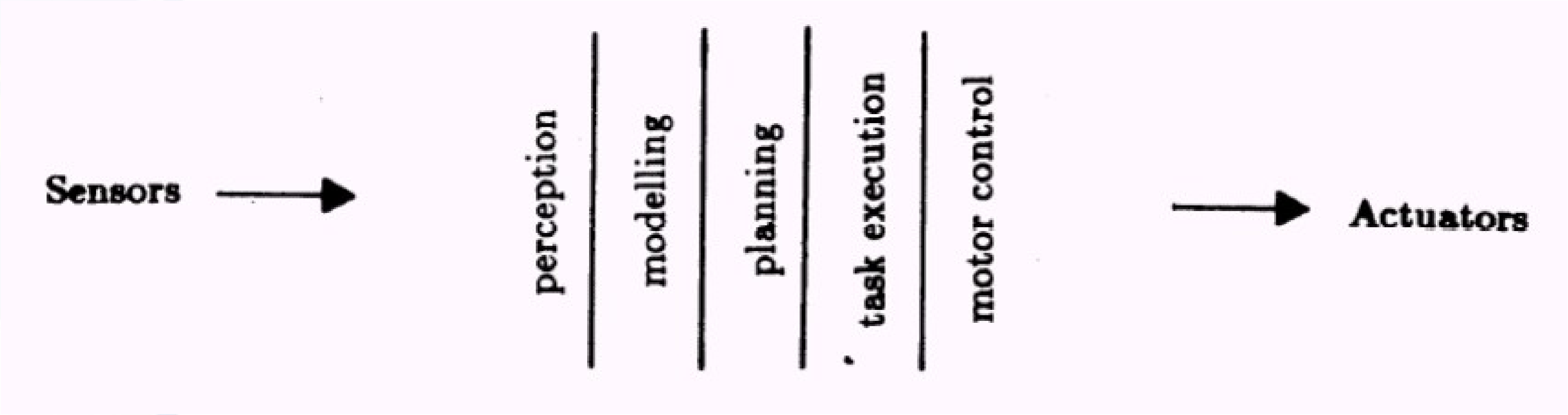
$$b(x_t) = \eta \; p(o_t \mid x_t) \int p(x_t \mid x_{t-1}, a_{t-1}) \, b(x_{t-1}) \, dx_{t-1}$$

- Used for:
  - Localization
  - Mapping
  - Model building



Deliberative Robot Control Architectures

- En una arquitectura de control deliberativo, el robot primero planifica la solución para una tarea razonando sobre el resultado de sus acciones y luego la ejecuta



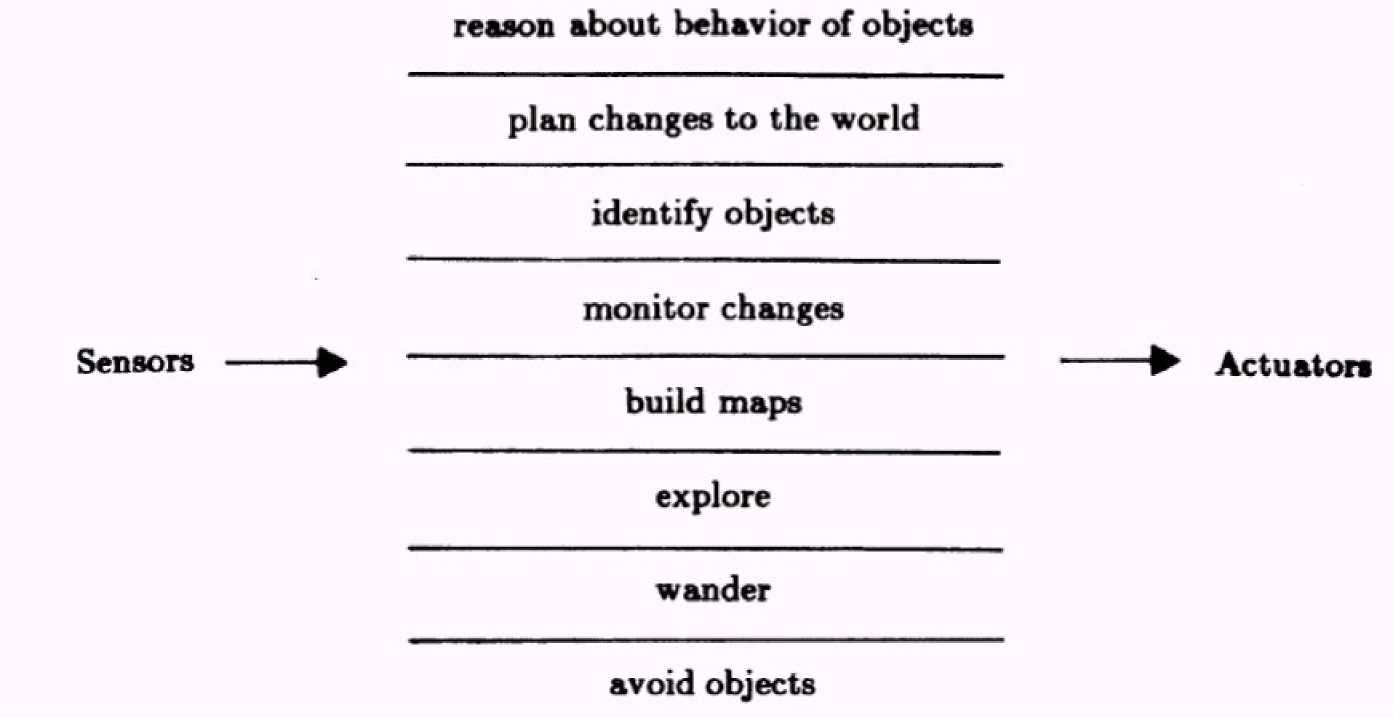
  - Control process goes through a sequence of sencing, model update, and planning steps

Deliberative Control Architectures

- Ventajas
  - Reasons about contingencies
  - Computes solutions to the given task
  - Goal-directed strategies
- Problemas
  - Solutions tend to be fragile in the presence of uncertainty
  - Requires frequent replanning
  - Reacts relatively slowly to changes and unexpected occurrences

Behavior-Based Robot Control Architectures

En una arquitectura de control basada en el comportamiento, las acciones del robot están determinadas por un conjunto de comportamientos reactivos paralelos que mapean la entrada sensorial y el estado a las acciones.



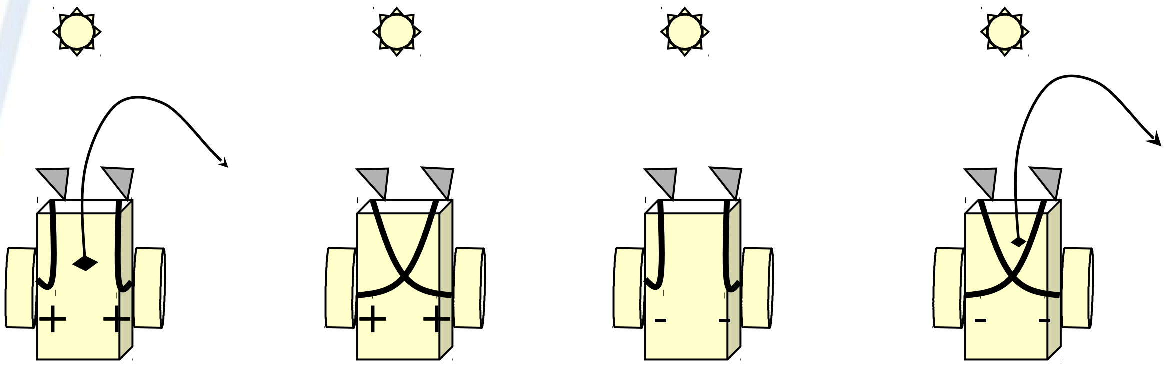
Behavior-Based Robot Control Architectures

El control reactivo, basado en el comportamiento, combina comportamientos relativamente simples, cada uno de los cuales logra una subtarea en particular, para lograr la tarea general.

- Robot can react fast to changes
- System does not depend on complete knowledge of the environment
- Emergent behavior (resulting from combining initial behaviors) can make it difficult to predict exact behavior
- Difficult to assure that the overall task is achieved

Complex Behavior from Simple Elements: Braitenberg Vehicles

- Comportameintos complejos se pueden lograr usando un control de mecanismos muy simple
  - Braitenberg vehicles: differential drive mobile robots with two light sensors

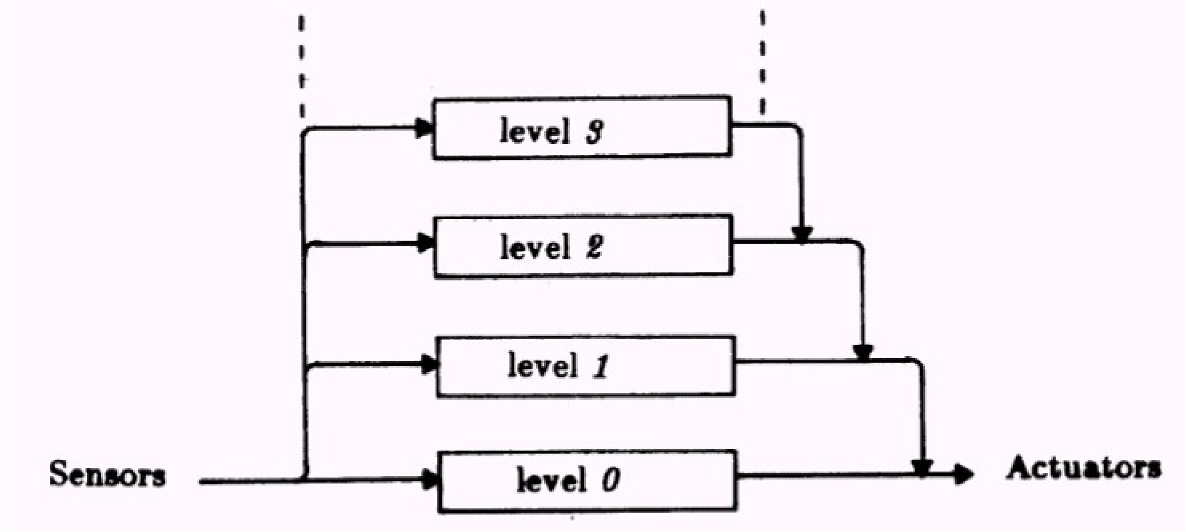


"Coward" "Aggressive" "Love" "Explore"

- Complex external behavior does not necessarily require a complex reasoning mechanism

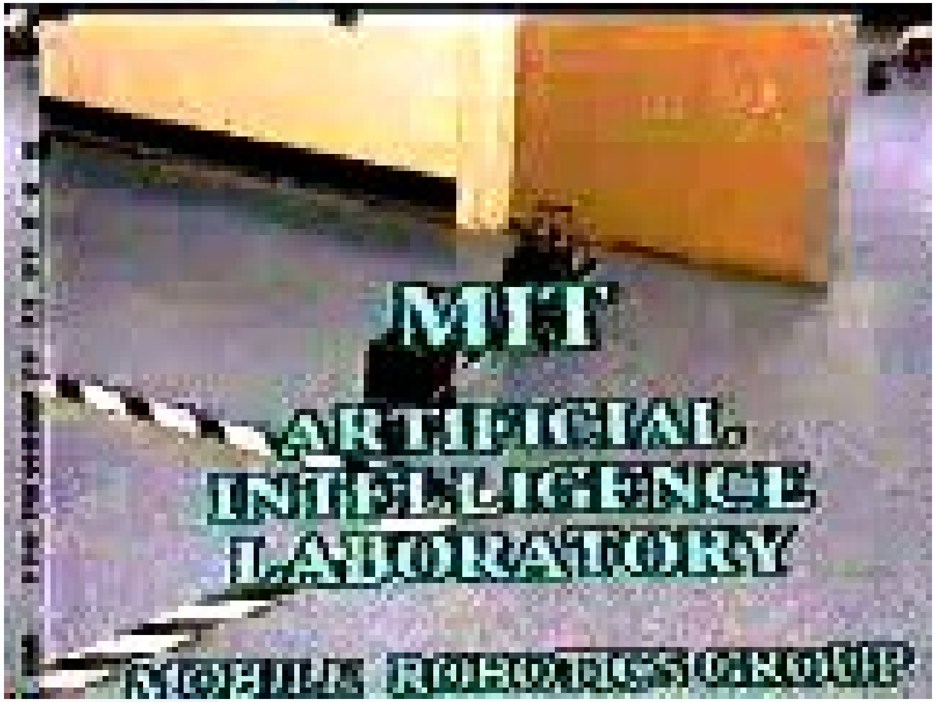
Behavior-Based Architectures: Subsumption Example

- La arquitectura de Subsumption es una de las primeras basada en el comportamiento
  - Behaviors are arranged in a strict priority order where higher priority behaviors subsume lower priority ones as long as they are not inhibited.



Subsumption Example

- A variety of tasks can be robustly performed from a small number of behavioral elements



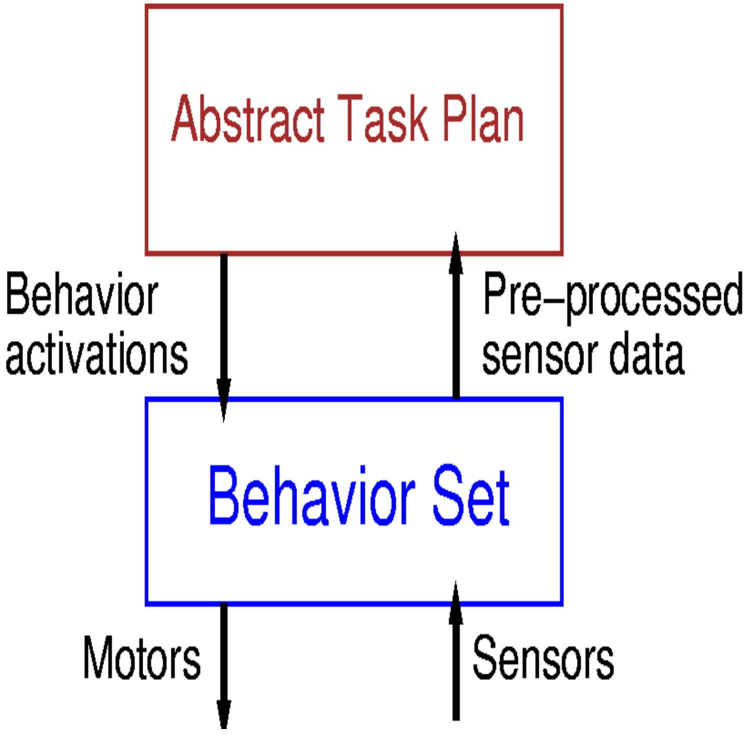
http://www-robotics.usc.edu/~maja/robot-video.mpg

Reactive, Behavior-Based Control Architectures

- Ventajas
  - Reacts fast to changes
  - Does not rely on accurate models
    - “The world is its own best model”
  - No need for replanning
- Problemas
  - Difficult to anticipate what effect combinations of behaviors will have
  - Difficult to construct strategies that will achieve complex, novel tasks
  - Requires redesign of control system for new tasks

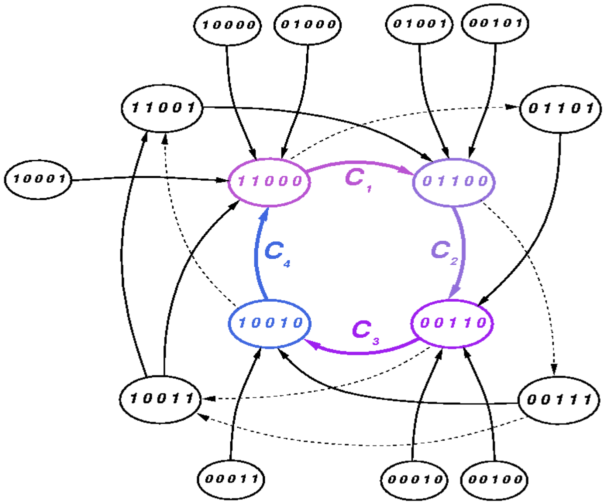
Hybrid Control Architectures

- Las arquitecturas hibridas combinan el control reactivo con la planificación abstracta de tareas
  - Abstract task planning layer
    - Deliberative decisions
    - Plans goal directed policies
  - Reactive behavior layer
    - Provides reactive actions
    - Handles sensors and actuators

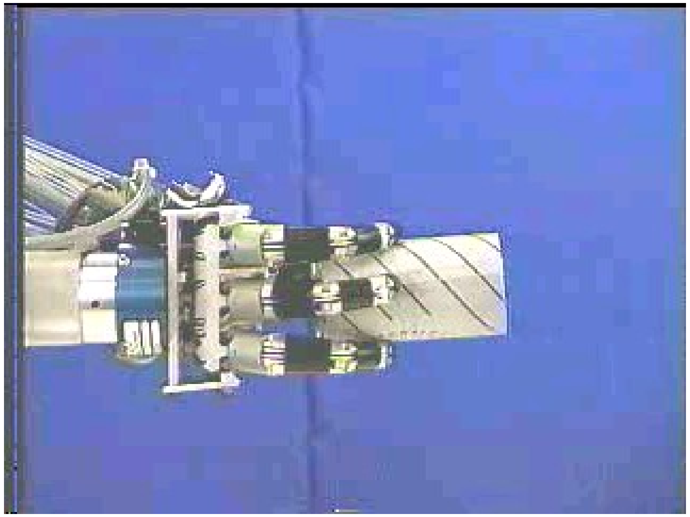


Hybrid Control Policies

Task Plan:



Behavioral Strategy:



Example Task:Changing a Light Bulb

Hybrid Control Architectures

- Advantages
  - Permits goal-based strategies
  - Ensures fast reactions to unexpected changes
  - Reduces complexity of planning
- Problems
  - Choice of behaviors limits range of possible tasks
  - Behavior interactions have to be well modeled to be able to form plans

Traditional Human-Robot Interface: Teleoperation

- Teleoperacion Remota: Operacion directa del robot por el usuario
  - El usuario usa un joystick 3-D o un exoesqueleto para manejar el robot
    - Simple to install
    - Removes user from dangerous areas
  - Problemas:
    - Requires insight into the mechanism
    - Can be exhaustive
    - Easily leads to operation errors

Human-Robot Interaction in Intelligent Environments

- Robot de servicio personal
  - Controlado y usado por usuarios no entrenados
    - Intuitive, easy to use interface
    - Interface has to “filter” user input
      - Eliminate dangerous instructions
      - Find closest possible action
  - Recive solo instrucciones intermitentes
    - Robot requires autonomous capabilities
    - User commands can be at various levels of complexity
    - Control system merges instructions and autonomous operation
  - Interactua con una variedad de humanos
    - Humans have to feel “comfortable” around robots
    - Robots have to communicate intentions in a natural way

Example: Minerva the Tour Guide Robot (CMU/Bonn)

http://www.cs.cmu.edu/~thrun/movies/minerva.mpg

Intuitive Robot Interfaces: Command Input

- Graphical programming interfaces
  - Users construct policies form elemental blocks
  - Problems:
    - Requires substantial understanding of the robot
- Deictic (pointing) interfaces
  - Humans point at desired targets in the world or
  - Target specification on a computer screen
  - Problems:
    - How to interpret human gestures ?
- Voice recognition
  - Humans instruct the robot verbally
  - Problems:
    - Speech recognition is very difficult
    - Robot actions corresponding to words has to be defined

Ejemplo

Intuitive Robot Interfaces: Robot-Human Interaction

- He robot has to be able to communicate its intentions to the human
  - Output has to be easy to understand by humans
  - Robot has to be able to encode its intention
  - Interface has to keep human's attention without annoying her
- Robot communication devices:
  - Easy to understand computer screens
  - Speech synthesis
  - Robot "gestures"

Example: The Nursebot Project

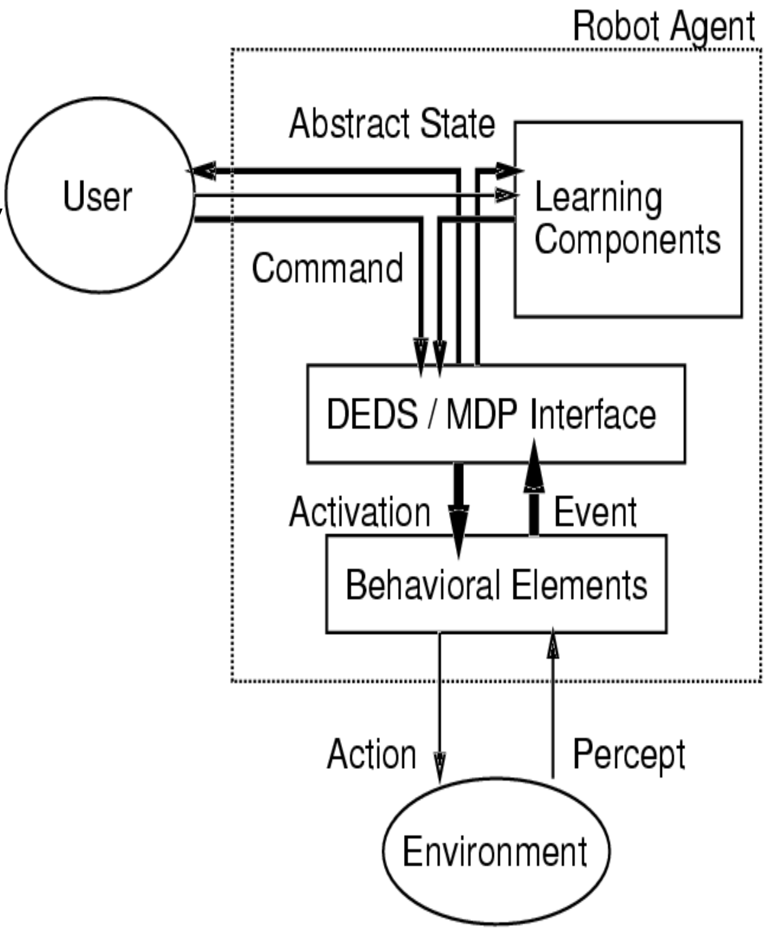
http://www/cs/cmu.edu/~thrun/movies/pearl_assist.mpg

Human-Robot Interfaces

- Existing technologies
  - Simple voice recognition and speech synthesis
  - Gesture recognition systems
  - On-screen, text-based interaction
- Research challenges
  - How to convey robot intentions ?
  - How to infer user intent from visual observation (how can a robot imitate a human) ?
  - How to keep the attention of a human on the robot ?
  - How to integrate human input with autonomous operation ?

Integration of Commands and Autonomous Operation

- Adjustable Autonomy
  - The robot can operate at varying levels of autonomy
  - Operational modes:
    - Autonomous operation
    - User operation / teleoperation
    - Behavioral programming
    - Following user instructions
    - Imitation
  - Types of user commands:
    - Continuous, low-level instructions (teleoperation)
    - Goal specifications
    - Task demonstrations



Example System

"Social" Robot Interactions

- To make robots acceptable to average users they should appear and behave “natural”
  - "Attentional" Robots
    - Robot focuses on the user or the task
    - Attention forms the first step to imitation
  - "Emotional" Robots
    - Robot exhibits “emotional” responses
    - Robot follows human social norms for behavior
      - Better acceptance by the user (users are more forgiving)
      - Human-machine interaction appears more “natural”
      - Robot can influence how the human reacts

"Social" Robot Example: Kismet

Kismet

Regulating Interaction Intensity:
Face stimulus (human)

Cynthia Breazeal (Ferrell)
Brian Scassellati

MIT Artificial Intelligence Lab

http://www.ai.mit.edu/projects/cog/Video/kismet/kismet_face_30fps.mpg

"Social" Robot Interactions

- Advantages:
  - Robots that look human and that show “emotions” can make interactions more “natural”
    - Humans tend to focus more attention on people than on objects
    - Humans tend to be more forgiving when a mistake is made if it looks “human”
  - Robots showing “emotions” can modify the way in which humans interact with them
- Problems:
  - How can robots determine the right emotion ?
  - How can “emotions” be expressed by a robot ?

Human-Robot Interfaces for Intelligent Environments

- Robot Interfaces have to be easy to use
  - Robots have to be controllable by untrained users
  - Robots have to be able to interact not only with their owner but also with other people
- Robot interfaces have to be usable at the human’s discretion
  - Human-robot interaction occurs on an irregular basis
    - Frequently the robot has to operate autonomously
    - Whenever user input is provided the robot has to react to it
- Interfaces have to be designed human-centric
  - The role of the robot is it to make the human’s life easier and more comfortable (it is not just a tech toy)

Adaptation and Learning for Robots in Smart Homes

- Intelligent Environments are non-stationary and change frequently, requiring robots to adapt
  - Adaptation to changes in the environment
  - Learning to address changes in inhabitant preferences
- Robots in intelligent environments can frequently not be pre-programmed
  - The environment is unknown
  - The list of tasks that the robot should perform might not be known beforehand
    - No proliferation of robots in the home
  - Different users have different preferences

Adaptation and Learning In Autonomous Robots

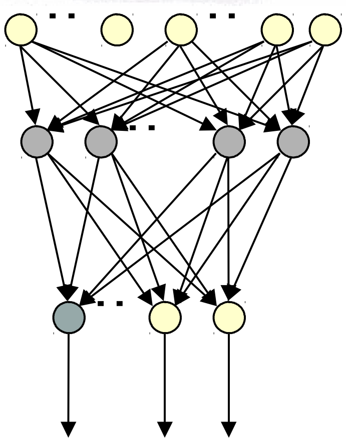
- Learning to interpret sensor information
  - Recognizing objects in the environment is difficult
  - Sensors provide prohibitively large amounts of data
  - Programming of all required objects is generally not possible
- Learning new strategies and tasks
  - New tasks have to be learned on-line in the home
  - Different inhabitants require new strategies even for existing tasks
- Adaptation of existing control policies
  - User preferences can change dynamically
  - Changes in the environment have to be reflected

Learning Approaches for Robot Systems

- Supervised learning by teaching
  - Robots can learn from direct feedback from the user that indicates the correct strategy
    - The robot learns the exact strategy provided by the user
- Learning from demonstration (Imitation)
  - Robots learn by observing a human or a robot perform the required task
    - The robot has to be able to “understand” what it observes and map it onto its own capabilities
- Learning by exploration
  - Robots can learn autonomously by trying different actions and observing their results
    - The robot learns a strategy that optimizes reward

Learning Sensory Patterns

- **Learning to Identify Objects**
  - How can a particular object be recognized ?
    - Programming recognition strategies is difficult because we do not fully understand how we perform recognition
    - Learning techniques permit the robot system to form its own recognition strategy
  - Supervised learning can be used by giving the robot a set of pictures and the corresponding classification
    - Neural networks
    - Decision trees



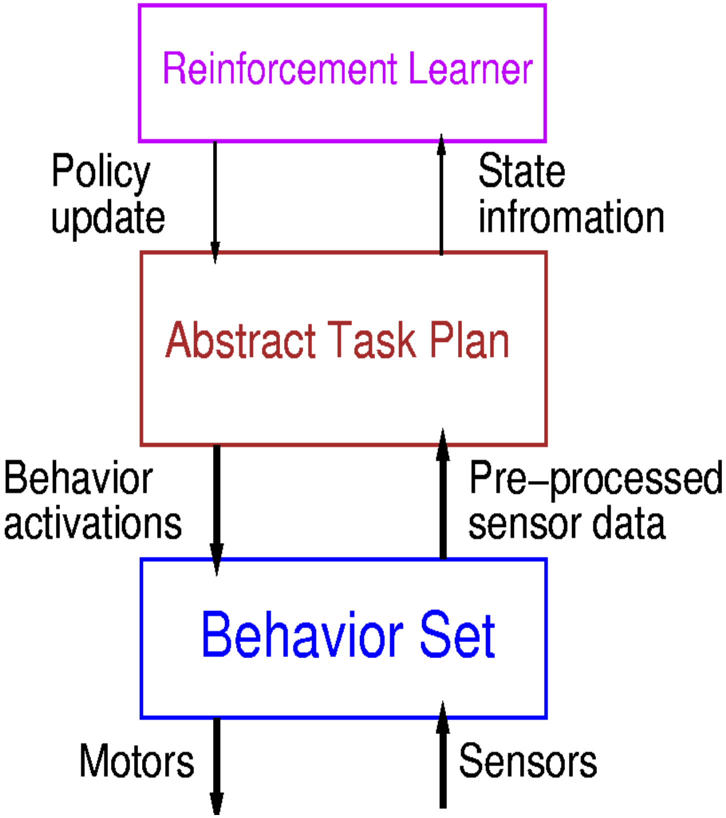
Chair

Learning Task Strategies by Experimentation

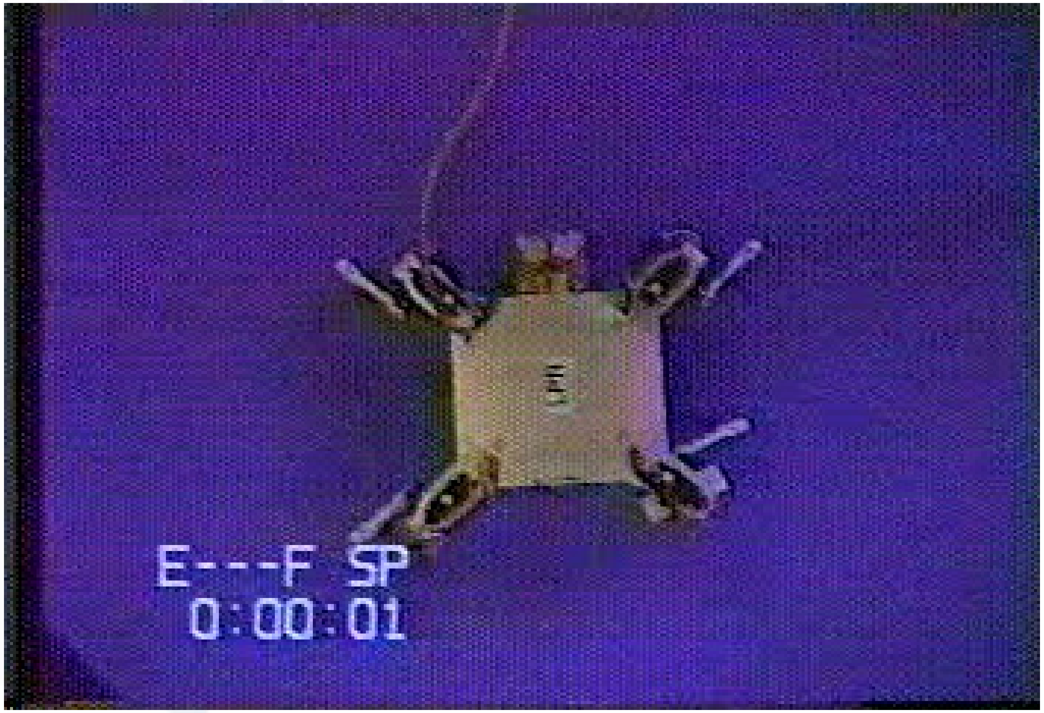
- Autonomous robots have to be able to learn new tasks even without input from the user
  - Learning to perform a task in order to optimize the reward the robot obtains (Reinforcement Learning)
    - Reward has to be provided either by the user or the environment
      - Intermittent user feedback
      - Generic rewards indicating unsafe or inconvenient actions or occurrences
    - The robot has to explore its actions to determine what their effects are
      - Actions change the state of the environment
      - Actions achieve different amounts of reward
    - During learning the robot has to maintain a level of safety

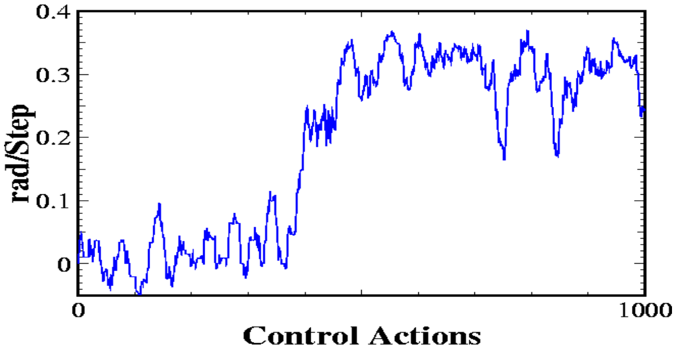
Example: Reinforcement Learning in a Hybrid Architecture

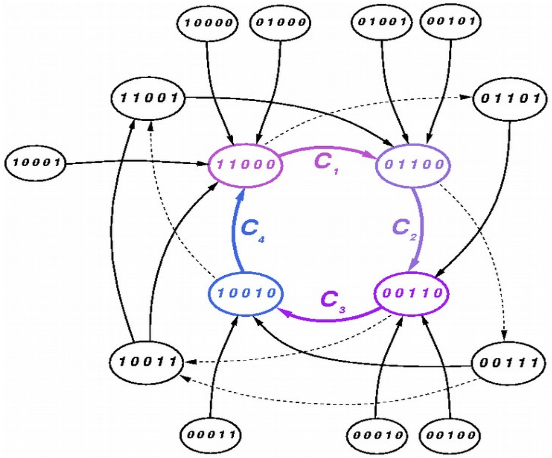
- Policy Acquisition Layer
  - Learning tasks without supervision
- Abstract Plan Layer
  - Learning a system model
  - Basic state space compression
- Reactive Behavior Layer
  - Initial competence and reactivity



Example Task: Learning to Walk







Scaling Up: Learning Complex Tasks from Simpler Tasks

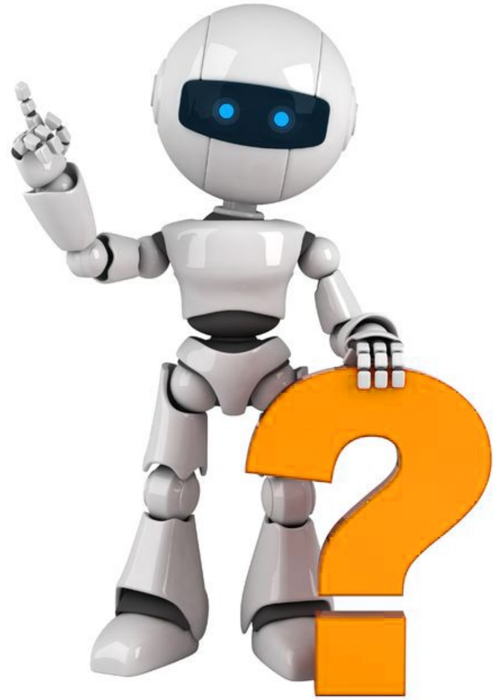
- Complex tasks are hard to learn since they involve long sequences of actions that have to be correct in order for reward to be obtained
- Complex tasks can be learned as shorter sequences of simpler tasks
  - Control strategies that are expressed in terms of subgoals are more compact and simpler
  - Fewer conditions have to be considered if simpler tasks are already solved
  - New tasks can be learned faster
  - Hierarchical Reinforcement Learning
    - Learning with abstract actions
    - Acquisition of abstract task knowledge

Example: Learning to Walk

Last example

Conclusions

- Robots are an important component in Intelligent Environments
  - Automate devices
  - Provide physical services
- Robot Systems in these environments need particular capabilities
  - Autonomous control systems
  - Simple and natural human-robot interface
  - Adaptive and learning capabilities
  - Robots have to maintain safety during operation
- While a number of techniques to address these requirements exist, no functional, satisfactory solutions have yet been developed
  - Only very simple robots for single tasks in intelligent environments exist



Demos