Architecture and interaction

ST5 Autonomous robotics

Francis Colas

2022-10-07

Introduction

Path-planning and navigation

- based on perception
- go to a given place, while avoiding obstacles
- exploration

Introduction

Path-planning and navigation

- based on perception
- go to a given place, while avoiding obstacles
- exploration

Remaining questions

- where to go?
- what to do?
- how to execute?

Introduction

Path-planning and navigation

- based on perception
- go to a given place, while avoiding obstacles
- exploration

Remaining questions

- where to go?
- what to do?
- how to execute?

Decision

- higher-level planning
- decide on a list of simpler actions

Introduction

Human-robot interaction

- specific challenges
- perception, control
- communication

Introduction

Human-robot interaction

- specific challenges
- perception, control
- communication

Execution

- coordination of actions
- control architecture

Introduction

Human-robot interaction

- specific challenges
- perception, control
- communication

Execution

- coordination of actions
- control architecture

Aim of the session

- task planning
- control architecture
- human-robot interaction

01

Task planning

Task planning

Main issue

- difficulty of programming complex tasks

Task planning

Main issue

- difficulty of programming complex tasks

General idea

- decompose complex tasks into simpler actions
- handle dependencies and prerequisites
- find valid list of actions

Task planning

Main issue

- difficulty of programming complex tasks

General idea

- decompose complex tasks into simpler actions
- handle dependencies and prerequisites
- find valid list of actions

Approach

- formal description of world and actions (domain)
- specification of current state and goal (problem)
- graph search

Knowledge representation

Formal description

- robot and its evolution
- objects and environment and their evolution
- robot actions

Knowledge representation

Formal description

- robot and its evolution
- objects and environment and their evolution
- robot actions

Formal languages

- different choices
- based on logic (predicates, propositional, order-1...)

Planning Domain Definition Language

Take a ball – definitions

```
(define (domain gripper-strips)
    (:predicates (room ?r)
                 (ball ?b)
                 (gripper ?g)
                 (at-robby ?r)
                 (at ?b ?r)
                 (free ?g)
                 (carry ?o ?g))
    ...
)
```

PDDL example

Take a ball – actions

```
(define (domain gripper-strips)
    ...
    (:action move
     :parameters (?from ?to)
     :precondition (and (room ?from) (room ?to)
                        (at-robby ?from))
     :effect (and (at-robby ?to)
                  (not (at-robby ?from))))
    (:action pick
     :parameters (?obj ?room ?gripper)
     :precondition (and (ball ?obj) (room ?room) (at ?obj, ?room)
                        (gripper ?gripper) (at-robby ?room)
                        (free ?gripper))
     :effect (and (carry ?obj ?gripper) (not (at ?obj ?room))
                  (not (free ?gripper))))
    ...
)
```

PDDL example

Take a ball – problem

```
(define (problem strips-gripper1)
    (:domain gripper-strips)
    (:objects room1 room2 ball1 ball2 hand)
    (:init (room room1)
           (room room2)
           (ball ball1)
           (ball ball2)
           (gripper hand)
           (at-robby room1)
           (free hand)
           (at ball1 room1)
           (at ball2 room1))
    (:goal (at ball1 room2)))
```

High-level planning

Solving principles

- forward chaining:
  - generate deduction graph starting from initial state

High-level planning

Solving principles

- forward chaining:
  - generate deduction graph starting from initial state
- backward chaining:
  - same, but starting from goal
- mixed forward and backward approaches

High-level planning

Solving principles

- forward chaining:
  - generate deduction graph starting from initial state
- backward chaining:
  - same, but starting from goal
- mixed forward and backward approaches

Take a ball?

High-level planning

Solving principles

- forward chaining:
  - generate deduction graph starting from initial state
- backward chaining:
  - same, but starting from goal
- mixed forward and backward approaches

Take a ball

```
((pick ball1 room1 hand)
 (move room1 room2)
 (drop ball1 room2 hand))
```

Conclusion on high-level planning

Knowledge representation

- abstract description
- formal language
- *symbol grounding* problem

Conclusion on high-level planning

Knowledge representation

- abstract description
- formal language
- *symbol grounding* problem

Planning

- definition of a problem in a domain
- planning as inference

Conclusion on high-level planning

Knowledge representation

- abstract description
- formal language
- *symbol grounding* problem

Planning

- definition of a problem in a domain
- planning as inference

Probabilistic planning

- handle uncertainty
- planning as probabilistic inference
- Markov Decision Processes (MDPs)
- Partially-Observable Markov Decision Processes (POMDPs)

02

Control architecture

Control architecture

Need

- integration of perception, decision, and action

Control architecture

Need

- integration of perception, decision, and action
- integration of several levels
  - task planning
  - motion planning
  - obstacle avoidance
  - motor control...

Control architecture

Need

- integration of perception, decision, and action
- integration of several levels
  - task planning
  - motion planning
  - obstacle avoidance
  - motor control...
- various time scales
  - real time
  - millisecond
  - second
  - a few minutes...

Structural principles

Structural principles

- modularity
  - complexity reduction
  - specialized algorithms for particular roles

Structural principles

Structural principles

- modularity
  - complexity reduction
  - specialized algorithms for particular roles
- hierarchy
  - behavior layers more and more complex
  - difficult to specify

Structural principles

Structural principles

- modularity
  - complexity reduction
  - specialized algorithms for particular roles
- hierarchy
  - behavior layers more and more complex
  - difficult to specify
- concurrence
  - perception and action in parallel
  - high-level planning and reaction unexpected events

Structural principles

Structural principles

- modularity
  - complexity reduction
  - specialized algorithms for particular roles
- hierarchy
  - behavior layers more and more complex
  - difficult to specify
- concurrence
  - perception and action in parallel
  - high-level planning and reaction unexpected events
- communication
  - synchronous or asynchronous
  - message-passing, remote procedure call, shared memory

Deliberative

Sense-plan-act

- architecture of Shakey (60's)
- sensors used for perception
- world modeling for planning
- plan for execution
- sense-plan-act-repeat

Deliberative

Sense-plan-act

- architecture of Shakey (60's)
- sensors used for perception
- world modeling for planning
- plan for execution
- sense-plan-act-repeat

Limitations

- monolithic loop
- open-loop control
- execution of an obsolete plan

Behavior-based

Subsumption architecture

- subsumption: concept inclusion/generalization
- set of behaviors
- organized in layers
- upper layers can inhibit lower layers

Behavior-based

Subsumption architecture

- subsumption: concept inclusion/generalization
- set of behaviors
- organized in layers
- upper layers can inhibit lower layers

Behaviors

- finite-state automata
- between sensors and actuators

Behavior-based

Subsumption architecture

- subsumption: concept inclusion/generalization
- set of behaviors
- organized in layers
- upper layers can inhibit lower layers

Behaviors

- finite-state automata
- between sensors and actuators

Limitations

- no memory nor representation
- complicated system of behavior inhibition

Subsumption architecture (1/2)

Brooks's example (1986)

0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

1 wander

0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

2 explore

1 wander

0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

3 build maps
2 explore
1 wander
0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

4 monitor changes

3 build maps

2 explore

1 wander

0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

5 identify objects

4 monitor changes

3 build maps

2 explore

1 wander

0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

6 plan changes to the world
5 identify objects
4 monitor changes
3 build maps
2 explore
1 wander
0 avoid objects

Subsumption architecture (1/2)

Brooks's example (1986)

7 reason about behavior of objects

6 plan changes to the world

5 identify objects

4 monitor changes

3 build maps

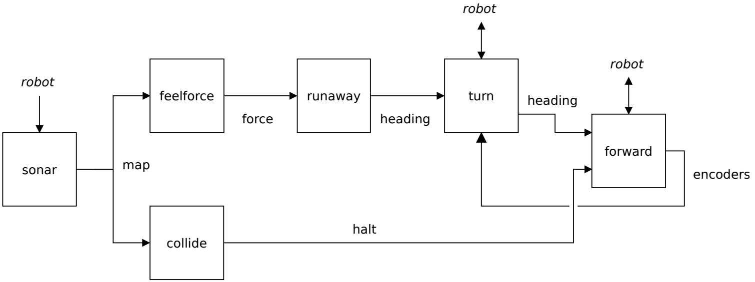
2 explore

1 wander

0 avoid objects

Subsumption architecture (2/2)

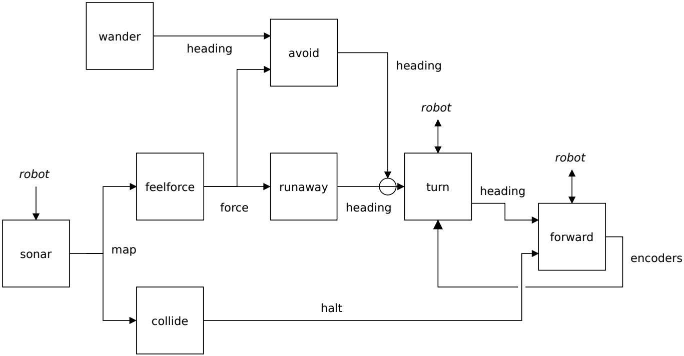
Brooks' example (1986)



Level 0: avoid objects

Subsumption architecture (2/2)

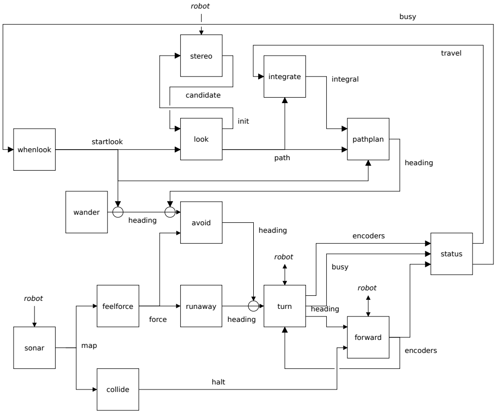
Brooks' example (1986)



Levels 0 and 1: wander

Subsumption architecture (2/2)

Brooks' example (1986)



Levels 0, 1, and 2: explore

Layered architectures

Layers

- behaviors and representations
- behavior
  - goal from upper layer
  - commands to lower layer
- representation
  - monitoring of lower layer
  - abstract data for upper layer
- upper layers at lower frequencies

Layered architectures

Three-tier architecture

- planning
  - handle high-level goals
  - maintains an abstract representation
- executive
  - task decomposition
  - monitoring and synchronization of tasks
  - manage resources
  - instantaneous memory
- behaviors
  - reactive or limited states
  - interaction between sensors and actuators

Conclusion on control architecture

Sense-Plan-Act

- deliberative
- slow and rigid
- top-down approach

Conclusion on control architecture

Sense-Plan-Act

- deliberative
- slow and rigid
- top-down approach

Subsumption

- behavior-based
- complex design and no (or limited) memory
- bottom-up approach

Conclusion on control architecture

Sense-Plan-Act

- deliberative
- slow and rigid
- top-down approach

Subsumption

- behavior-based
- complex design and no (or limited) memory
- bottom-up approach

Layered architectures

- behaviors and representations
- reactive and deliberative

Conclusion on control architecture

Typology

- deliberative: think, then act
- reactive: don't think, (re)act
- behavior-based: think the way you act
- hybrid: think and act concurrently

03

Human-Robot Interaction (HRI)

Human-robot interaction

Human-Robot Interaction (HRI)

- security
- physical interaction
- communication
- acceptability

Security

Space sharing

- industrial robots in cages
- some tasks require space sharing
- needs for intrinsic security

Security

Space sharing

- industrial robots in cages
- some tasks require space sharing
- needs for intrinsic security

Collision

- reduce gravity of impact
  - lower speed, mass, and inertia
  - *compliance*

Security

Space sharing

- industrial robots in cages
- some tasks require space sharing
- needs for intrinsic security

Collision

- reduce gravity of impact
  - lower speed, mass, and inertia
  - *compliance*
- collision detection
  - estimate external torque
  - torque error

Security

Space sharing

- industrial robots in cages
- some tasks require space sharing
- needs for intrinsic security

Collision

- reduce gravity of impact
  - lower speed, mass, and inertia
  - *compliance*
- collision detection
  - estimate external torque
  - torque error
- reaction strategies
  - stop the robot
  - gravity compensation
  - admittance control

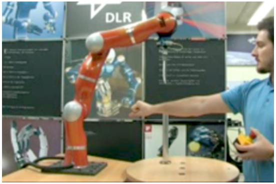
Security

Space sharing

- industrial robots in cages
- some tasks require space sharing
- needs for intrinsic security

Collision

- reduce gravity of impact
  - lower speed, mass, and inertia
  - *compliance*
- collision detection
  - estimate external torque
  - torque error
- reaction strategies
  - stop the robot
  - gravity compensation
  - admittance control



Sami Haddadin
https://youtu.be/dnUwqngH0bM

Physical interaction

Physical interaction

- more than simply space sharing
- joint task achievement: assembly

Physical interaction

Physical interaction

- more than simply space sharing
- joint task achievement: assembly

Approaches

- intelligent assistance systems:
  - crane or gantry with interaction forces
- force amplification
  - exoskeleton
  - industry or rehabilitation
- cobots (collaborative robots)
  - robot restriction in a sub part of workspace
  - fine human control

Comau AURA

Physical interaction

Physical interaction

- more than simply space sharing
- joint task achievement: assembly

Approaches

- intelligent assistance systems:
  - crane or gantry with interaction forces
- force amplification
  - exoskeleton
  - industry or rehabilitation
- cobots (collaborative robots)
  - robot restriction in a sub part of workspace
  - fine human control

Comau AURA

Estimation of what the human is doing.

Human-robot communication

Human-robot communication

- robot for non-experts
- natural communication

Human-robot communication

Human-robot communication

- robot for non-experts
- natural communication

Multimodal communication

- verbal
- non-verbal: gestures, expressions, posture...

Human-robot communication

Human-robot communication

- robot for non-experts
- natural communication

Multimodal communication

- verbal
- non-verbal: gestures, expressions, posture...

Complementary communication

- regulate conversation
- show one's state: pointing, attention, emotions...
- illustrate

Human-robot communication

Human-robot communication

- robot for non-experts
- natural communication

Multimodal communication

- verbal
- non-verbal: gestures, expressions, posture...

Complementary communication

- regulate conversation
- show one's state: pointing, attention, emotions...
- illustrate

Leonardo robot

Human-robot communication

Human-robot communication

- robot for non-experts
- natural communication

Multimodal communication

- verbal
- non-verbal: gestures, expressions, posture...

Complementary communication

- regulate conversation
- show one's state: pointing, attention, emotions...
- illustrate

Leonardo robot

Estimation of what the human is communicating.

Inria informatics mathematics

Acceptability

Acceptability

- robots can help
- only when accepted
- user studies
- worries
  - aspect
  - understanding
  - societal impacts...

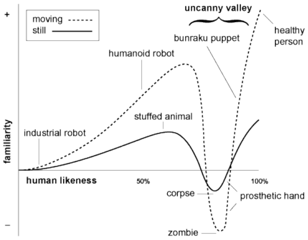
Acceptability

Acceptability

- robots can help
- only when accepted
- user studies
- worries
  - aspect
  - understanding
  - societal impacts...

Uncanny valley

- not totally human-looking
- negative emotional response
- loss of empathy
- stronger when in motion



uncanny valley (Mori)

04

Conclusion

Conclusion

High-level function

- need a high-level representation of actions and states
- reasoning at this level
- integration with low-level
- control architecture

Conclusion

High-level function

- need a high-level representation of actions and states
- reasoning at this level
- integration with low-level
- control architecture

Human-robot interaction

- security
- physical interaction
- communication
- acceptability
- human estimation

Conclusion

High-level function

- need a high-level representation of actions and states
- reasoning at this level
- integration with low-level
- control architecture

Human-robot interaction

- security
- physical interaction
- communication
- acceptability
- human estimation

Bibliography

STRIPS and PDDL

- Fikes and Nilsson, *STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving*, AI, 1971.
- McDermott *et al.*, *PDDL – The Planning Domain Definition Language*, Yale CVC, 1998.
- Fox and Long, *PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains*, JAIR, 2003.

Subsumption

- Brooks, *A robust layered control system for a mobile robot*, RA, 1986.

informatics mathematics

Bibliography

Intrinsic security

- Haddadin *et al.*, *Collision detection and reaction: a contribution to safe physical human-robot interaction*, IROS 2008.

Social robotics

- Dautenhahn, *Socially intelligent robots: dimensions of human-robot interaction*, Phil. Trans. R. Soc. B, 2007.

Books

- Siciliano *et al.*, *Springer Handbook of Robotics*, 2nd ed., Springer, 2016.

Thanks for your attention
Questions?