Robot Architecture

Gerrit Glaser
15.05.2014

Contents

- Robot Architecture
- Hierarchical Paradigm
- Reactive Paradigm
- Hybrid Architectures
- List of References

Robot Architecture

- Software architecture
  - Structure of algorithms
- Overall philosophy
  - Language
  - Tools
- Mainly two paradigms
  - Model-based deliberate control
  - Reactive control

Contents
Hierarchical Paradigm

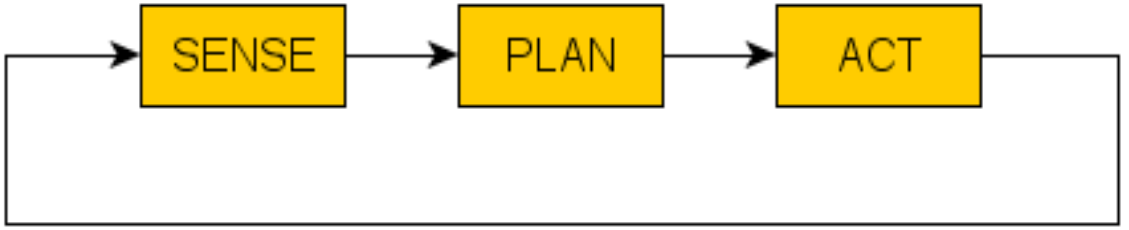
- Hierarchical Paradigm
  - SPA Cycle
  - World Model
  - Strips Algorithm
  - Representing Architectures
    - Nested Hierarchical Controller

Hierarchical Paradigm

- Oldest approach of organizing intelligence
- Brought up in 1967 dominating up to late 1980s
- Model-based deliberate control
- Sequential execution
- Monolithic sensing

Hierarchical Paradigm: SPA Cycle

- SENSE-PLAN-ACT cycle
  - Sense
    - Senses the world, constructs a world model
  - Plan
    - Plans all directives from current state to goal
  - Act
    - Executes first directive and goes to sense again



Hierarchical Paradigm: World Model

- World model
  - Updated in SENSE step
  - Three parts
    - Provided representation of the world
    - Sensing information
    - Additional cognitive knowledge

Hierarchical Paradigm: Strips Algorithm

- Algorithm for planning how to accomplish goals
- Means-end analysis
  - gradually reduces distance between current state and goal
  - Uses difference tables
    - Complex, many exceptions

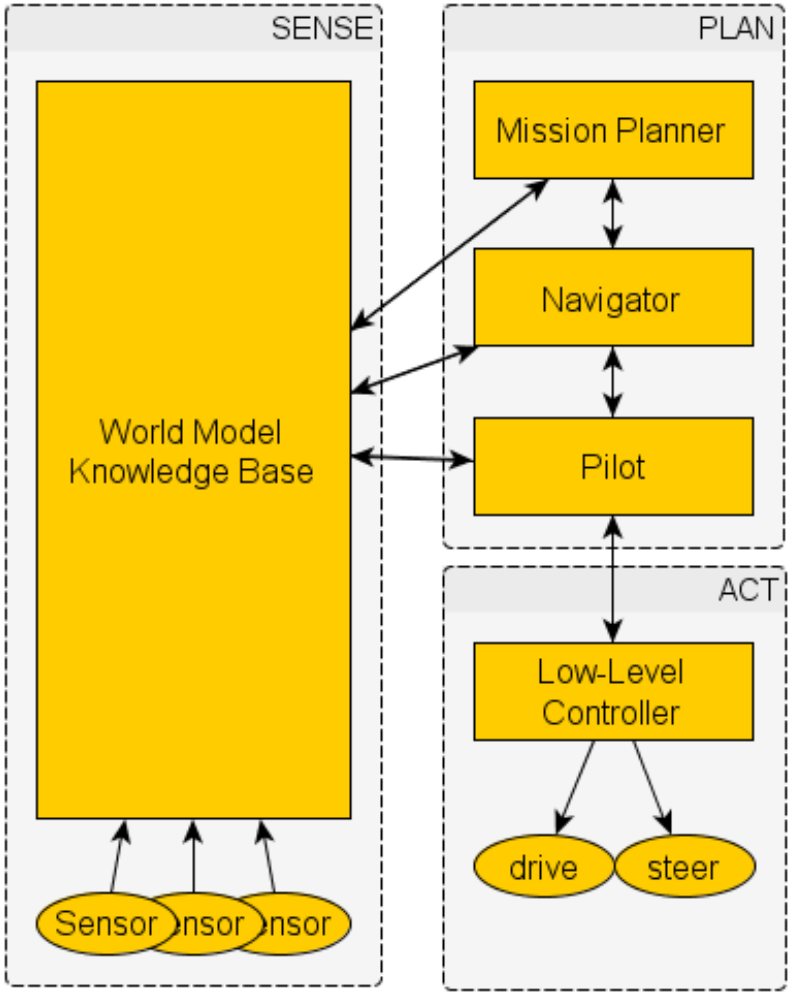
distance	operator	preconditions
d<1000	walk	
d>=1000	take_bus	at bus station
	take_car	at parking lot

Hierarchical Paradigm: Representing Architectures

- Nested Hierarchical Controller (NHC)
  - splitting of planning in three subsystems
- National Institute of Standards and Technology Real-time Control System (NIST RCS)
  - Standard for intelligent industrial robots
  - Sensor preprocessing

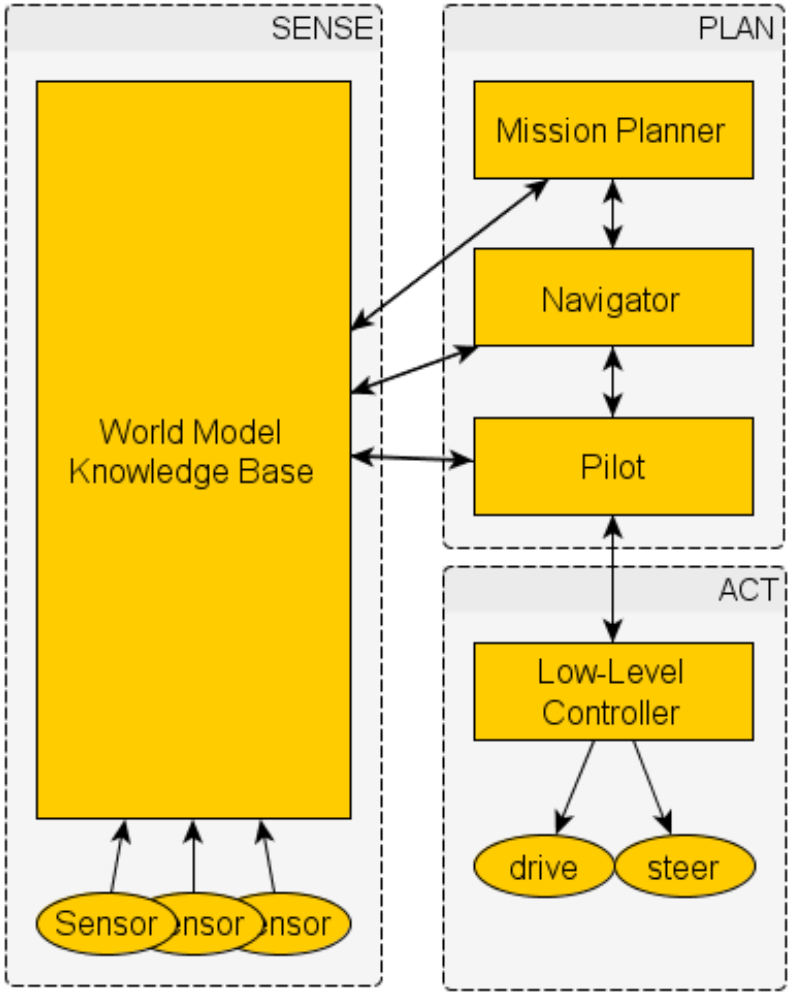
HP: Nested Hierarchical Controller

- Mission Planner
  - High level of abstraction
  - Receive mission from person
  - Get current position and target from model
  - Invokes navigation



HP: Nested Hierarchical Controller

- Navigator
  - Calculates path from current location to task location
- Pilot
  - Follows path, controls motors, informs navigator when done
  - Checks world model and adjusts path accordingly



Contents
Reactive Paradigm

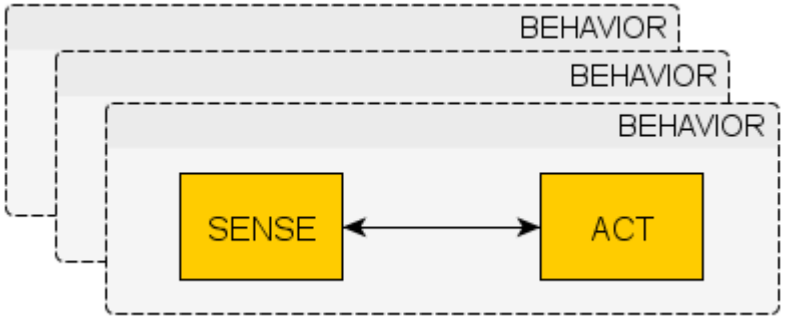
- Reactive Paradigm
  - SENSE-ACT
  - Characteristics
  - Architectures
  - Subsumption
    - Aspects
    - Evaluation

Reactive Paradigm

- Developed in the late 1980s due to the hierarchical paradigm's lack of expandability
- Vertically layered
  - Decompose functionality into behaviors
  - Tight coupling between sensing and acting → fast
- Imitates behavior of insects/animals
- No need for a world model

Reactive Paradigm: SENSE-ACT

- Threw away the PLAN phase
  - Mapping sensor inputs to actions
  - Sensor fusion for more complex behaviors



Reactive Paradigm: Characteristics

- Situated Agent
  - Robot is part of the world, can change the environment
- Behaviors as basic building blocks
- Ego-Centric
  - No global model is created, only sensable things are known
- Follows good software design principles
  - Modularity, decomposition of task into component behaviors
- Animal models cited as basis
  - Nowadays acceptable, in the beginning not

Reactive Paradigm: Architectures

- Architecture must provide mechanisms for
  - Triggering behaviors
  - What happens when multiple behaviors are active
- Dominant architectures
  - Subsumption architecture
  - Potential Field based architectures
- Other architectures
  - Rule encoding, fuzzy methods, winner-take-all voting

Reactive Paradigm: Subsumption

- Invented by Rodney Brooks
  - Most influential purely reactive system
  - Shoe-box sized insects with six legs
  - Directly embedded into hardware (on-board computing)
  - Elimination of move-think pauses
  - Behavior network of sensing and acting modules that accomplish a task

Reactive Paradigm: Subsumption Aspects

- Layers of competence
  - Reflect a hierarchy of intelligence/competence
  - Higher layers are more goal oriented
- Layers can subsume lower layers
  - Top to bottom override or subsumption (winner-take-all)
- No internal state
- Taskable
  - Activate lower layers

Reactive Paradigm: Subsumption Evaluation

Advantages

- Modular, easy to test
- Easily expandable
- Real-time reaction
- Inexpensive
- Low coupling, high cohesion

Disadvantages

- Not easily taskable
- Not useful for reaching global targets

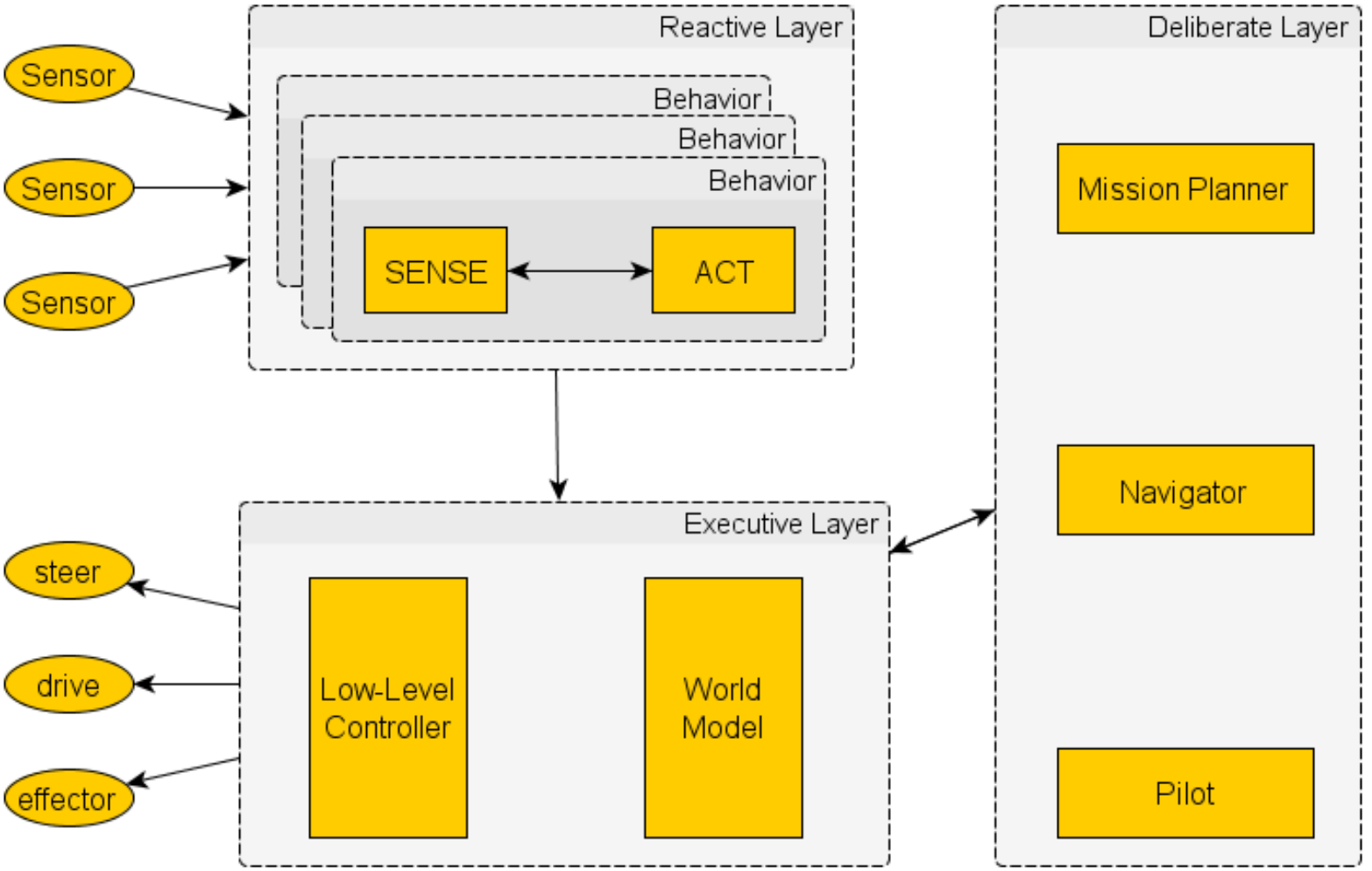
Contents
Hybrid Architectures

- Three Layer Architecture
  - Evaluation
- Other Possible Layers

Hybrid Architectures: Three Layer Architecture

- Combines reaction with deliberation
  - Reactive layer
    - Low level control, tight sensor-action coupling
  - Executive layer
    - „Glue“ between reactive and deliberate layer
    - Integrates sensor information into the world model
    - Sequences directives by the deliberate layer for the reactive layer
  - Deliberate layer
    - Generate global solutions to complex tasks by planning
    - Works on models (pre-supplied or learned)

Hybrid Architectures: Three Layer Architecture



Hybrid Architectures: Three Layer Evaluation

Advantages

- Combines advantages from hierarchical and reactive paradigm
- Reacts to unexpected circumstances
- World model to plan global solutions

Disadvantages

- Expensive regarding performance

Hybrid Architectures: Other Possible Layers

- Supervisor layer
- Sensor integration layer
- Sensor interpretation layer
- User interface layers
  - To control the interaction with persons
- Layers for coordination with other robots

List of References

- Robin R. Murphy
  - Introduction to AI Robotics
  - ISBN: 978-0-262-13383-8
- Stuart Russell & Peter Norvig
  - Artificial Intelligence: A modern Approach Second Edition
  - ISBN: 0-13-080302-2