EUROP european robotics technology platform

Appendix to:

“Robotic Visions to 2020 and beyond – The Strategic Research Agenda for robotics in Europe, 07/2009”

Technology roadmaps

Version: 1

Date: 07.07.2009

“Robotic Visions to 2020 and beyond – The Strategic Research Agenda for robotics in Europe, 07/2009” can be obtained from:

EUROP Secretariat
C/o EUnited Robotics
Diamant Building, Bd. A. Reyers 80
1030 Brussels, Belgium

Phone: +32 2706-8222
Fax: +32 2706-8223
Email: info@robotics-platform.eu
Internet: www.robotics-platform.eu

Supported by the “Coordination Action for Robotics in Europe”, FP6-IST-045058

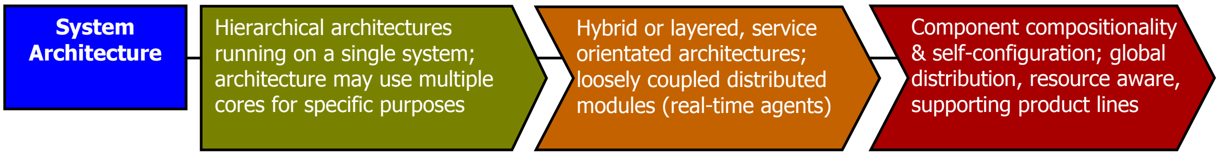
CARE
Coordination Action for Robotics in Europe

Content

Content ... 2
1. System Architecture ... 3
2. System engineering & deployment tools ... 5
3. Cooperating robots and ambient intelligence ... 8
4. (Real-time) communication ... 10
5. Human-machine interface (HMI) ... 11
6. Safety ... 13
7. Actuation ... 15
8. End effector for handling purposes ... 17
9. Locomotion ... 19
10. Materials ... 21
11. Navigation ... 22
12. Planning ... 25
13. Power management ... 27
14. Control ... 29
15. Learning ... 31
16. Modelling ... 33
17. Sensors ... 36
18. Sensing & Perception ... 39

1. System Architecture

1.1. High level document



1.2. Definition of terminology

An architecture defines the structure of system components, their interrelationships, and the principles and guidelines governing their design and evolution over time. It thereby provides a plan from which (sub-)systems can be procured or developed. Components (HW and SW) are separately distributable and deployable entities, of which systems can be composed.

Close relations with Control and with Communications

1.3. Drivers of technology

Most aspects of system architecture are driven by other disciplines, but the following aspects are mainly driven by robotics:

- Robotics is the science of integration; a system architecture of a robot system describes how various sensors, actuators and processing modules are organised and integrated
- Mechatronic system design for robots
    - Building blocks, self configuration (plug & play of electrical and mechanical aspects) during Runtime of the system
- Brounded cognitive architectures are robot specific
    - Architectures for autonomy and learning
- Architectures for robots and robot systems/application
- Robot system architectures should be flexible to allow for configuration, expansion etc. by the system and component developer
- Robot architectures often need more interfacing capabilities to allow the robot to be used in different ways
- Needs to support developers, ISPs & end users
- Architecture integration in ISP business models & environments
- Potential for standardisation / modularisation within domains / application areas

1.3.1 Driven by others:

- The architecture of a whole factory is not driven by robotics, the robot has to fit into it → users will provide requirements for robots which the system has to satisfy.

1.4. European strengths and weaknesses

Core technologies for both components and systems and competencies are available with the EU, but both scalable platforms and integration in a reusable way are lacking.

1.4.1 Strengths:

- OSGI, TOPCASED, European initiatives in telecomm and aerospace, not directly robotics projects but are closely related and could be employed in robotics

- Genom project, not well known outside LAAS CNRS robotics lab but employs very well tested and accepted practices for SW verification/validation and partially MDE based approach in robotics.
- Several (from a performance point of view) leading components and research platforms are from labs with the EU:
    - Public: OROCOS, LAAS/GenoM, etc.
    - Internal: DLR multi-arm and hand real-time control, etc.
    - Commercial: Robosoft, etc.
- European motion-control systems and components are state of the art.
    - Industrial arm/machine control: ABB, KUKA, Siemens, etc.
    - Industrial modules : Technosoft Motion, Beckhoff, etc.
- Industrial experience of complex robotics system architectures

1.4.2 Weaknesses:

- Current popular frameworks and integration platforms for implementation of robot system architectures originate from US developers:
    - Public: Player/Stage, Claraty (partly Internal)
    - Internal: Claraty, JAUS (partly commercial)
    - Commercial:Microsoft Robotics Studio (integration platform)
    - Lego (Mindstorms)
    - Orocos project, proposes a structured approach for robot control software with well defined component model and respective component interfaces
- Fragmentation / Lack of communication
    - “REUSE” of technology and products (mostly SW related) not best practice in Europe. “State of the Art analysis” at the onset of projects would be a first step

1.5. Short term development (~2010)

- Resources include implicit (not given an explicit identifier in the software) embedded resources in terms of memory, processing power, energy consumption, and IO bandwidth.
- Composition of embedded/real-time software today requires extensive hand-on engineering since neither classes/objects/libraries/frameworks nor components are resource aware, and hence cannot be configured automatically.
- Client server architectures, distributed but coupled
- Distributed database-supported blackboard architecture on high (non real-time) levels
- Multi-core CPUs for low-cost high performance, for engineered components
- Single system architectures (scalability problems, limited or engineered performance, less complex applications, runs on a single system) Not state of art!
- Hierarchical architectures (central processing unit, dependence on central master module, shorter life cycle)
- Layered architectures (special form of hierarchical architectures) (exchangeable layers, longer life cycle, improved configuration management, lower complexity, potential loss of performance due to abstraction)
- Vertical integration based on a combination of top-down (coming from CAD/CAM, cell/line-control, etc., going towards devices) and bottom up (coming from field-buses and coordinated control of devices, going towards machine and cell control) approaches. This is driven by vendors extending their businesses, which unfortunately leads to coupled systems and vendor lock-ins.

1.6. Mid term development (~2015)

Here the main focus lies on the engineering effort.

- Distributed multi-agent architecture (mulit-robot) ; peer-to-peer active modules
- Modular, multi-vendor sub-system integration , including for example

  - 100% separate memory between processes running untrusted binaries
  - Strict sandboxing by safe languages
- Resource management built into the (deployment and runtime) platform
- Distributed service oriented architecture, with
  - Interoperability (but no real-time) with web-services on enterprise level
- Plug-and-Play functionality for devices, including real-time support
- Hybrid architectures with self-describing data and interfaces, and:
  - Horizontal integration based on existing standards, ranging from field-buses on low-level to web-services on high level
- (Benchmark – Emergence of markets)

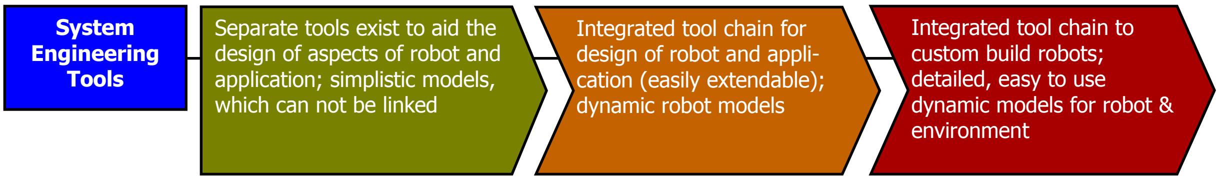
1.7. Long-term development (~2020+)

Here the main focus moves more strongly towards the end user and the ease of use of robotics systems.

- Cognitive architectures (behaviour based) (makes use of ontology and semantic web technologies)
- World-wide distributed architectures (GRID technologies) – multi-vendor, multi-service, multi-application
- Self configuring architectures
- While mid-term development is fully occupied by engineering of systems based on current engineering tools/know-how and application needs, the long term goal must be to bring compositionality into software.
- While mid-term research aims at solving the most challenging problem with available/primitive instruments/tools, long-term development needs to focus on the simplest possible problems that capture the fundamental difficulties.
- On the system level compositionality of meta-information and knowledge need to be well integrated with the actual software platforms/tools. That is to better support semi-automatic configuration, to formally capture/reuse engineering experiences, to easier embed learning into systems, and to optimize quality-of-service. The use of declarative descriptions is fundamental.
  - Control algorithms, grid computing, cognitive capabilities, etc., are topics of for components and thereby relevant for architecture, but those developments do not comprise the architectural issue. Software for product lines does.

2. System engineering & deployment tools

2.1. High level document



2.2. Definition of terminology

Set of tools for engineering a robot system, including designing a robot and its mechatronic components, developing robot system software and applications (specification and simulation of instantaneous (possibly sensor-based) robot actions; specification and simulation of action coordination between several robots, or between several tasks of the same robot) and simulating kinematical and dynamical properties of the robot system (hardware and software) and its environment and deployment of a robot in its environment throughout the entire life-cycle of the product. Engineering tools could also be used for engineering support for customers (as well as product support).

2.3. Technology driver

2.3.1 *Driven by robotics:*

- Modelling of robot HW.
  - Computation (functional algorithms basically),
  - Communication (middleware for message passing and event handling),
  - Coordination (to synchronize the activities of several robots and/or several skills on one single robot).
- Modelling of robot systems / dynamics / environment
- Modelling for motion planning
- Online object modelling robotics tends to be in the lead
- Tools for the deployment of a robot system in its task. These will be application-specific (logistics, painting, welding, drilling, …)
- Tools for the deployment of robot systems in its tasks (High priority)

2.3.2 *Driven by others:*

- Engineering tools are mainly driven by manufacturing system end-users
- Offline Object modelling: driven by Engineering and games industry
- Simulation and model description software / technologies
- Dynamic Environmental modelling driven by defence, and also the games industry
- Simulation of complex dynamic effects (fine evaluation of friction; non-regular mechanics for multi-contacts; non-isotropic material; cables and highly flexible components; over-constraint mechanisms) is driven by key players in general mechanics simulation (Adams, Ansys, etc).
- Hardware-in-the-loop simulation (electro-mechanics + control) is driven by automotive industry needs (dSpace, Matlab/Simulink, etc)

2.4. European strengths and weaknesses

2.4.1 *Strengths:*

- Good Academic basic training in key sciences and technologies in most European countries (math-based modelling; artificial vision; mechanical engineering and control)
- A few key players with international recognition (KUKA, ABB, Bosch, etc)
- A strength in this could give Europe an advantage:
  - Easy access to new markets; e.g. agriculture (stronger than in Japan and no cheap human labour like in the USA)
- Modelling and simulation
  - Joint with the US e.g. walking machine
  - Software like CATIA, PLM-System

2.4.2 *Weaknesses:*

- Rapid prototyping of robot systems
  - Japan and US are big here
  - Weakness due to lack of large integrated companies or research centres in this area
- Big gap between academia and industry (e.g. in modelling). When it exists, the connection between academia and industry remains mostly mono-country (KUKA and German Universities; ABB and Swedish Universities)
- European robotics industry mostly focused on traditional sectors (heavy industry, mainly). US has moved to Military and Japan to next generation humanoids.
- Lots of developing capacity in the US, but also in Europe; US has strong brand names

- In the US there is loads of work going on in the open source part of engineering tools while a lot of the developments around embedded system development on top of Eclipse are European (e.g. the Topcased and OpenEmbedd projects).

2.5. Short term development (~2010)

- Independent, special purpose tools exist to engineer a robot system and its application:
  - Designing of a robot and its mechatronic components
    - CAD programs and special design programs
  - Developing robot system software and applications
    - Special purpose tools
      - For system software development such as Workbench (for VxWorks), Visual Studio
      - For application development, e.g., by means of
        - Editors for native robot code (text-based programming (any formal specification is in text)); well defined semantics, domain specific constructs; GUI based programming (Visual programming, more intuitive) based on flow chart and similar or icon-based is slowly becoming available
        - Teach-in with manual guidance
        - Offline programming tools (allowing for simulation and programming in a very limited number of domains (e.g. ship welding)
  - Simulating kinematical and dynamical properties of the robot system (hardware and software) and its environment
    - With special purpose tools such as Dymola + Modellica
    - For better designing mechatronic components
    - Better inter-connection between high-end mechanics simulation (including non linear effects) and advanced controllers simulation
  - Deployment of a robot in its environment
    - Very limited in their task variety
- Product properties database
- Information design (requires consistent information)
- Structuring of data (difficult to structure information in a meaningful and consistent way)
- Product data management (PDM)
- Tools for designing robot applications: focusing on the product developer/ system integrator

2.6. Mid term development (~2015)

Tools to support _system-level_ design on the basis of components. This should go hand in hand with an improvement in the education and training of the engineers that are supposed to work with such tools, since currently few of this is already in the engineering curricula.

- A more integrated tool chain exists
- To engineer a robot system and its application:
  - Designing of a robot and its mechatronic components
  - Developing robot system software and applications
  - Including non-classical robots (PKM – Cable driven robots – Mobile manipulators – etc)
- The system engineer can easily integrate special purpose tools into the tool chain
  - e.g., system software development tools such as Workbench (for VxWorks), Visual Studio
  - e.g., application development tools (see above)

- Simulating kinematical and dynamical properties of the robot system (hardware and software) and its environment is much easier because more models and computational algorithms have become available to be used by the tool chain
- The software developed at the simulation/designing stage can be directly used in the real environment (direct code compatibility)
- Deployment of a robot in its environment
    - Besides static robot manipulators also mobile robots and even mobile manipulators can be deployed for a larger variety of tasks
- Customization of robot applications
- Integration of elements, one complete system as result
- Combination of several known elements
- Integration of heterogeneous data
- Fast integration of new components
- Non-complex standards
- Teach in with pointing device
- Tools for designing robot applications: focusing on the ISP, design of specific behaviours

2.7. Long-term development (~2020+)

- A unified framework exists that helps robotics engineers and application developers alike to develop customized robot systems and applications for them
- Special purpose tools will be seamlessly integrated into the tool chain (i.e. will not be recognized as former stand alone tools any more)
- Various environmental, kinematical and dynamical models exist and parameters can be easily modified by the end user (robot developer, system integrator, end user)
- Customisation of robot systems
- Large support for auto coding of robot systems, i.e. robot programs are automatically derived from a high-level task description (e.g. Rinas system using pre-existing macros)
- Tools for designing robot applications: focusing on the end user, integration of applications, integration with other systems / web based resources

3. Cooperating robots and ambient intelligence

3.1. High level document

Cooperating robots and Ambient Intelligence

Team of robots; centralised control and communication; tasks specified for each individual robot; use of common map

Distributed control; inter-agent communication; task specified for group; games & swarm theories are applied

Cooperation without explicit representation of action; skill-/learning-/knowledge-based automation

3.2. Definition of terminology

Approach to the coordination of multi-robot systems which consist of numbers of physical robots and possibly external sensors that can co-operate intentionally or, such that a desired collective behaviour emerges from the robot-robot interactions and the interactions of the robots with the environment. Here both collective behaviour of similar partners and behaviour of dedicated partners are involved.

3.3. Drivers of technology

3.3.1 Driven by robotics:

- Collective behaviour
- Middleware for physical multi agent systems
- Bioinspired studies (insect studies, ethology, cell population studies) and social sciences

- Modularity in the sense that price level is critical as the swarm is big and such a way might be that the members can be disposed.
- Modular self-configurable robots
- Embedded systems design
- Energy scavenging of mobile devices
- Fault tolerant functioning of the system
- Robot (mobile sensor) networks

3.3.2 Driven by others:

- Internet of things (i.e. internet with physical objects)
- Microsystems technology (sensors/actuators)
- Sensor networks
- Wireless comms

3.4. European strengths and weaknesses

Somewhere in the middle in global comparison.

3.4.1 Strengths:

- Strong cooperative robotics groups and activity in Europe (EURON SIG)
- European projects:
  - I-SWARM (recently finished, IP EU FET FP6, Prof. Wörn, University of Karlsruhe, estana@ira.uka.de, szymanski@ira.uka.de, hamann@ira.uka.de)
  - Swarm-Bots (finished 2005, STREP EU FET FP6, Prof. M. Dorigo)
  - Comets MultiUAVs (AICIA/University of Seville),
  - Aware (AICIA/University of Seville),
  - SYMBRION (IP EU FET FP7),
  - SWARM (finished 2006, Helsinki University of Stuttgart),
  - REPLICATOR (IP EU ICT FP7, University of StuttgartTechnology,
  - continues in connection with DAMOCLES and national level).
  - “SwarmBots” (M. Dorigo, ULBruxelles),
  - “URUS” (A. Sanfeliu, UPCatalonia, Spain)
- Biomimetics / swarms, e.g., Dario Floreano (Lausanne)

3.4.2 Weaknesses:

- Military applications in the US, military swarms are weak in Europe

3.5. Short term development (~2010)

- Planning: generation of plans, maintenance of plans, plans for individual robots
- Centralised control, work area coverage through GPS
- Agents (bidding systems)
- Geometrical approaches (data origin from map, calculate direction of detection)
- Multi-SLAM navigation and mapping
- Cooperative perception and navigation
- Practical planning and decision-making under uncertainty

3.6. Mid term development (~2015)

- Distributed planning for multiple robots, formation of the swarm
- Regeneration of work tasks failed to realise by one member of the swarm
- Setting up own communication network between members automatically

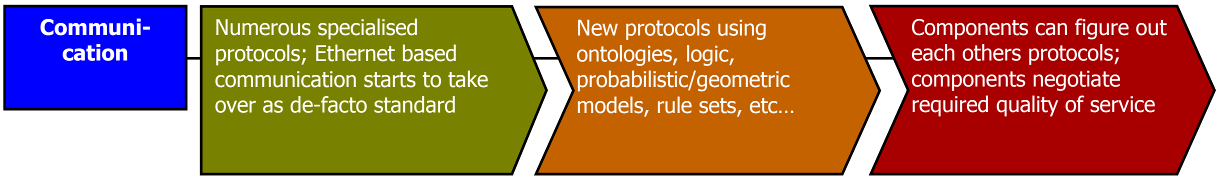
- Distributed control
- Knowledge based learning
- Games theory, failed tasks with new task generation
- Swarm theory beginning to penetrate
- Education and entertainment robotics
- Including humans to form human-robot teams
- Decentralised planning and decision-making under uncertainty

3.7. Long-term development (~2020+)

- Emerging co-operative behaviour without explicit representation of action
- Learning: skill based execution of tasks
- Reasoning: knowledge based, skill based execution of tasks
- Emerging capacity distribution between members of the swarm
- Space and deep-sea applications (surveillance, terraforming, mining, defence)
- Search and Rescue
- Surveillance
- Solving energy autonomy of swarming robots

4. (Real-time) communication

4.1. High level document



4.2. Definition of terminology

Hardware and software communicating in the given time constraint of the system.

4.3. Drivers of technology

4.3.1 Driven by robotics:

- Real-time communication means are mostly driven by other industries (e.g., the aerospace or consumer electronics industries)
- Real-time operating systems which are the basis for real-time communication are also driven by other industries
- A robot system needs real-time communication means for internal communication purposes (e.g. for the communication between the drives and the controller) and external communication purposes (e.g., between peripheral devices such as grippers / welding guns and the controller) or for communicating with external (supervisory) controllers of the plant in which the robotic system is embedded
- Communication is closely related to the communication framework needed to support a certain system architecture
- As a robot system consist of many different components (SW and HW) with a variety of communication requirements a unifying framework (middleware like FDT/DTM or DriveServer) for robotics could be beneficial for future developments

4.3.2 Driven by others

- (e.g. aerospace, automotive or consumer electronics):

- Real time operating systems

4.4. European strengths and weaknesses

4.4.1 Strengths:

- Developing field busses for process automation (e.g. Siemens Profinet, Beckhoff Ethercat, Powerlink etc.) and middleware (e.g. DriveServer etc.) to simplify the support of numerous bus systems.

4.4.2 Weaknesses:

- Developing generalised communication frameworks that are used by today's robot systems, such as Player, ORCA, CORBA, etc.

4.5. Short term development (~2010)

- Ethernet-based communication (e.g., EtherCat, Profinet, both real time) becomes the de-facto standard in process automation
- A large family of specialised protocols (SercosII, Profibus, CanBus, Firewire…) still exists which makes it difficult for robot manufacturers to serve them all
- USB
- Bluetooth (not in real time)
- Laser / fibre communication
- Infrared
- Wireless communication

4.6. Mid term development (~2015)

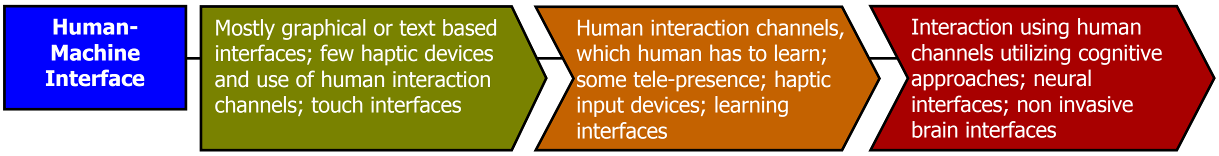
- Protocols change over time: knowledge representation through ontologies, geometric models, sharp logic (non-fuzzy), probabilistic models, rule sets, objects, frame constraints
- ZigBee (replacing Bluetooth, slower but consuming less energy)
- Microwave

4.7. Long-term development (~2020+)

- Protocols: systems can figure out each others protocol and adjust to it.
- Negotiation regarding required quality of service etc…

5. Human-machine interface (HMI)

5.1. High level document



Human-Machine Interface

Mostly graphical or text based interfaces; few haptic devices and use of human interaction channels; touch interfaces

Human interaction channels, which human has to learn; some tele-presence; haptic input devices; learning interfaces

Interaction using human channels utilizing cognitive approaches; neural interfaces; non invasive brain interfaces

5.2. Definition of terminology

IO-System that enables humans and robots to communicate with each other, allows humans to command the system and enables the robot to give information to (and in some applications, command) the human. This includes a variety of input and output channels. Efficient human machine interaction depends on the design of the interface.

5.3. Drivers of technology

5.3.1 *Driven by robotics:*

- Aspects of human-robot interfaces (HRI):
  - An HRI is an HMI able to control something in three dimensions
  - The interaction with a physical entity is often multimodal
  - Robots form parts of HRI when used as haptic input devices
  - A particular type of a HRI is a neural interface for prosthesis (invasive or non invasive)
  - HRI includes physical interaction between the user and the device
- Integrating the devices required for HMI for robotics is robotics driven
- Physical simulation with 3 dimensional animation (Virtual Reality) during the planning/programming stage to test and optimise the robot control

5.3.2 *Driven by others:*

- Robotics is using devices which are driven by other industries (PDAs, Touch interfaces, etc.)
- Loads of work on some aspects in other areas (e.g. human-computer interaction, mouse GUI)
  - Speech recognition (Speaker-independent or dependent)
  - Text to speech
- Multimodal interfaces merging visual, gesture, haptic and speech channels
- Augmented Reality (AR) overlaying process-integrated information

5.4. European strengths and weaknesses

5.4.1 *Strengths:*

- Industrial robots, traditional interfaces (Teach pendant, I/O panel, Display…)
- When taking a broader view on IO channels Europe has competencies, too. Especially within robotics a more technical interaction is necessary, and less about typical physical interaction
- Strength within decision making process of type of interaction, when and how interaction takes place between human and robot and the type and depth of interaction necessary
- Interfaces based on neuroscience / Neural interfaces

5.4.2 *Weaknesses:*

- US having a finger on the pulse of people in respect to design concepts
  - Concepts that concentrate strongly on design and physical interaction like Apple iPhone™ operation (touch technologies like TouchFlo™)
- Europe has a deficit of companies that mass produce high tech consumer products
- Games (HW/SW) are coming from Japan
- Level of involvement in standardisation

5.5. Short term development (~2010)

- Strongly physical driven with clear differentiation between interaction levels
- Physical interaction by touch (see Apple „touch“ products)
- Visual displays: text based output, Graphical User Interfaces
- Non-haptic devices (keyboard, space mouse, joystick)
- Voice (speech synthesis, speech understanding) utilising platform independent programming interfaces (e.g. web-services like the Voice Control API)

5.6. Mid term development (~2015)

- Visual displays: Augmented Reality

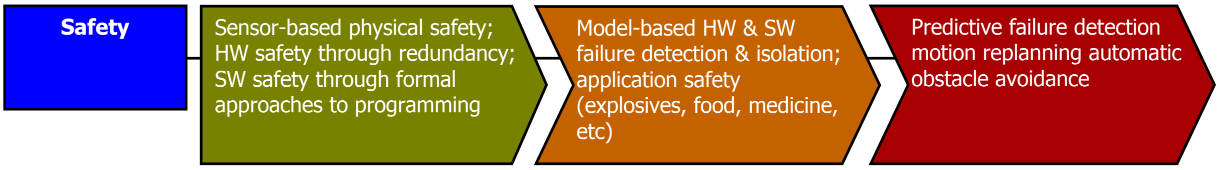
- Movements:
    - Gesture recognition
    - Uses visual perception (not only gesture, face, …)
    - More possibilities through stronger sensor/ sensing
    - Recognition of movement to specify the task to be done
- Cognitive approaches:
    - Gestures (learned and known)
    - Interaction through recognition of expression of emotions via sound or visual channel
- Techniques for learning the user’s intentions by experience, e.g. robots in domestic/ personal applications being able to understand the user’s intention by learning his/her preferences and habits over time
- (Non)cognitive approaches:
    - Multimodal input/output (including smell)
- Tele-presence beginning to show influence in HMI
- Haptic devices:
    - Haptic feedback of visual displays, force feedback sensors, joysticks, artificial skin, exoskeletons (non necessary haptic dev.)
- Non-haptic devices detecting and tracking eye movement, programming using eye movement recognition
- Direct physical interaction with the robot, e.g. robot that can physically help a person, especially elderly people, interaction in this scenario will be direct and physical: the robot will have to detect the forces exerted by the person and to provide forces as well to give support.
- Visual displays: Virtual Reality

5.7. Long-term development (~2020+)

- Neural system interfaces (muscle / nerve ending interfaces, user moves and the system saves all the neural information, brain interface)
- Non invasive brain interfaces (brain imaging techniques applied to detect user’s intentions)
- Cognitive approaches
    - More advanced human motion interpretation of unknown or unlearned gestures
    - Recognition of emotions
    - Recognition of user behaviour and intent interpretation
- Interaction through recognition of expression of emotions via sound or visual channel
- Voice: natural language, translating (in different noise situation)
- “Natural” interfaces that can read non explicit signals from the user
- Closed loop of the interaction/ emotion feeling, e.g. toys, personal companion

6. Safety

6.1. High level document



6.2. Definition of terminology

Aspects of a system designed, or measures taken, to handle unexpected and hazardous situations in a safe way. The goal is to transform the severity and likelihood of risks that are inherent in all autonomous robot activities and human-robot interactions to lower, acceptable levels. Primary aspect is the distinction between handling objects and humans. Furthermore, current standards only cover human injuries, but aspects such as robot damage and damage to the environment may also be considered relevant.

6.3. Drivers of technology

6.3.1 Driven by robotics:

- Where human and robot interact, collaborate or share space
- Robotic safety issues may result in changes in the relevant laws (e.g. liability towards the robot manufacturer or retailer); actually ISO standard 10218 is changed. Part 1 (robot systems) has been completed and published; part 2 (automation systems) is still under discussion and the consensus concerning this part seems to be rather difficult to find.
- Robotic specific safety systems (which often use devices from other areas): sensor systems have to be integrated into the developments

6.3.2 Driven by others:

- Proposed addition: Changes of national legislations, (e.g. new machinery directive) can affect the content of standards, i.e. adoptions are driven by legislation and not by robotics
- Technical development, e.g. new sensors or machine vision concepts have an impact on safety issues.
- Safety as a philosophy to make a machine safe (→ which is the basis of the current standards, e.g. general machinery directives, civil aviation standards etc.)
- General regulations and standards involved in (human) safety
- Occupational health and safety issues
- Safe design principles

6.4. European strengths and weaknesses

6.4.1 Strengths:

- Intensive use, adoption and development of industrial robot safety standards
- Safe robot controllers
- Force control for Human injury protection
- ESPE (electro sensing protective equipment)
- Innovative safety concepts, e.g. development of protective fenceless system by companies like: Sick, Pilz, etc. ; e.g. in the area of 3D PMD cameras.

6.4.2 Weaknesses:

- Not enough efforts regarding safety certifications of new products, e.g. sensor systems (too little efforts are undertaken by sensor system developers regarding certification of vision sensors (e.g. 3D PMD cameras) towards safety properties. Sensors must be certified “safe” to use them in a safe robot solution)
- Not enough European activities in standardisation panels and committees (e.g. Korea and Japan are currently more active for example in standardisation committees, in particular regarding service robotics and Humanoids)

6.5. Short term development (~2010)

- Safety compliance of software:
  - Formal semantics
  - Formal specifications
  - Formal verifications
  - Theorem proofing
  - Model checking

    - Predictive failure models for failure management
- Safety compliance of hardware:
    - Sensor based: for collision avoidance between human and robot, and for fenceless cooperation, for low impact injury
    - Intrinsically safe structures and architecture: like using 2 cores for redundancy, for fault managements
    - Majority voting: different sensors for same measurement

6.6. Mid term development (~2015)

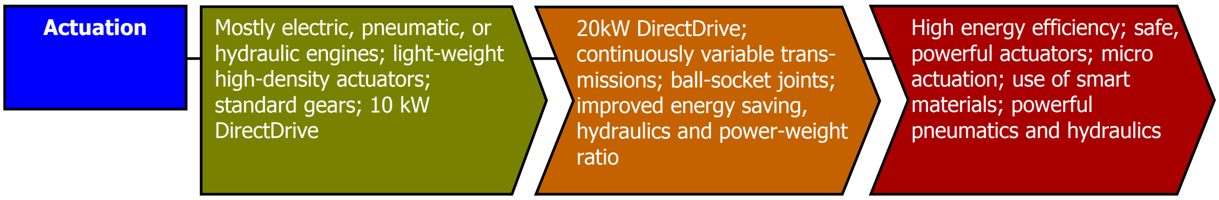
- Hardware redundancy through safety circuits
- Improved monitoring: functional monitoring, model based monitoring
- Developments in robotics controllers will reach saturation
- Failure detection and isolation of failures
- Covering of aspects in cleanness and sterilisation within food and drug industry, medical robotics, domestic applications, e.g. robot cleans toilet and then works in kitchen environment
- User intention estimation to make systems fault tolerant
- Atex: explosion protection, other military application

6.7. Long-term development (~2020+)

- Predictive failure detection
- Automatic obstacle avoidance in conjunction with automatic path replanning
- Self-repairing systems
- Further development towards dynamic robot motion replanning due to emersion of dynamic obstacles.
- Automatic obstacle avoidance
- Detection of the intention of persons in the working area of a robot (e.g. by detection of line of vision or foot/leg positioning)

7. Actuation

7.1. High level document



Actuation

Mostly electric, pneumatic, or hydraulic engines; light-weight high-density actuators; standard gears; 10 kW DirectDrive

20kW DirectDrive; continuously variable transmissions; ball-socket joints; improved energy saving, hydraulics and power-weight ratio

High energy efficiency; safe, powerful actuators; micro actuation; use of smart materials; powerful pneumatics and hydraulics

7.2. Definition of terminology

Techniques to generate forces and torques. Hardware to manage the motion of the robot, e.g. move the robot tool into a desired pose, change the kinematic configuration or shape of the robot. Important parts of a robotic drive train are (electrical) drives, motors, gears, media distribution and brakes.

Examples:

- Electric motor
- Fluid actuators
- Air muscles
- Hydraulic power transmission
- Piezo-electric actuator
- Bimetallic
- Ultra-sonic motor

- Power electronics and drives
- Gears
- Brakes
- Shape memory alloys
- Electro-active polymers

7.3. Drivers of technology

7.3.1 Driven by robotics

Most parts of actuation are not driven by robotics, but some are:

- Light-weight motors particularly designed for intermittent (non-continuous) motions, i.e. frequent changes of acceleration, deceleration and reversion of direction
- Gear box compact design; combinations with electric motors and sensors
- Artificial muscles, pressure sensitive devices
- Passive and Active compliance
- Direct drive technologies, avoidance of gears
- Robotics produces different requirement: compactness, precision, light-weight, low-power

7.3.2 Driven by others:

- Strong drivers are other industry branches, e.g. automation

7.4. European strengths and weaknesses

7.4.1 Strengths:

- Large scale actuation: ABB, Siemens, Festo, SCHUNK, Bosch-Rexroth, Indramat
- Artificial muscle (Italian Institute of Technology, Dawin Coldwell; FZI Karlsruhe, Stefan Schulz, Shadow Robot)
- AC/DC motors
- High density drives (DLR, CEA-List)
- Small scale actuation servo drives (Japan, Korea. Swiss made ones are good but too expensive for some products) (Maxon motor)
- Linear actuator designed to customer specifications

7.4.2 Weaknesses:

- Gear boxes
    - Some properties are better in Europe – precision (Spinea)
    - Harmonic drives are Japanese
    - Europeans are not rigid enough, and are too heavy and too expensive
    - Own production of gear boxes, e.g. Stäubli

7.5. Short term development (~2010)

- Gears: standard, torque, direct drive up to 10 kW, straight or angled teeth, planetary gear, harmonic drives, cycloidical gears
- Drives: power convertor (with / without control)
- Motors: electric (external rotor, internal rotor, with brushes / brushless, synchronous / asynchronous, step motor), pneumatic, hydraulic, internal combustion
- Magnetic liquid, in order to control optical lenses
- Linear electromagnetic actuators
- Direct drive actuator
- Actuator with sensor integrated

- Piezoelectric actuators

7.6. Mid term development (~2015)

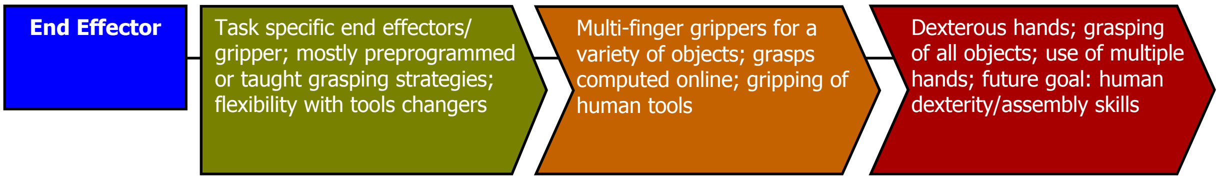
- Gears: continuous variable transmissions with learning capabilities
  - Weight reduction (progressive materials)
  - Miniaturisation
  - Reduction of weight/power ratio
- Electrical motors direct drive with power over 20 kW, higher power density, smaller size, low speed character
- Drives and motors: improved energy saving aspects, focused on start-stop applications
- Bi-metallic actuators (used in micro applications)
- Ball and socket joint
- Ongoing improvements within pneumatics, e.g. by FESTO; e.g. pneumatic actuators, air muscle
- Remarkable development within hydraulic pumps, improved motor output ratio, reduction of oil heat generation
- Linear motor technology for forces over Kilo-Newton

7.7. Long-term development (~2020+)

- Special motor concepts (chemical, nuclear, etc.)
- Smart actuator (SMA, EAP) smart plastics for muscles, new design shapes
- Energy buffering, saving load potential energy generated by the robot
- Safety aspects with back-drivable actuators with extensive memory handling
- Micro actuators

8. End effector for handling purposes

8.1. High level document



End Effector

Task specific end effectors/ gripper; mostly preprogrammed or taught grasping strategies; flexibility with tools changers

Multi-finger grippers for a variety of objects; grasps computed online; gripping of human tools

Dexterous hands; grasping of all objects; use of multiple hands; future goal: human dexterity/assembly skills

8.2. Definition of terminology

Device to enable a robot to interact with and change its environment, e.g., by grasping, manipulating and processing objects.

8.3. Drivers of technology

8.3.1 Driven by Robotics:

- End effectors such as grippers and hands are mostly being developed for robotic applications
- Tool changers allowing for an automatic change of end effectors

8.3.2 Driven by other industries:

- Prostheses (rehabilitation, assistive surgery, health care)
- Pharmaceutical, biology
- Agriculture/food
- Nuclear, military

8.4. European strengths and weaknesses

8.4.1 Strengths:

- Grippers (SCHUNK, FESTO; Schmalz)
- Dexterous hands (products: Shadow, SCHUNK; research: DLR, SSSA, Univ. Bologna, IPA)
- Tool changers for industrial (static) robotics
- Grasp planning / grasping strategies
- Hand prostheses (Otto Bock, Touch Bionics)
- Flexible hands in production

8.4.2 Weaknesses:

- Tools changers for mobile robotics
- Lack of common software interfaces
- Missing standards on software/hardware interfaces
- Flexibility on a low cost level
- Grippers require the development of special tools and attachments.
- Hands have only the approximate power of a human hand. This of course can be "amplified" by the tools normally available to humans.

8.5. Short term development (~2010)

- End effectors are chosen depending on the application and required grasping tasks; end effectors can be selected mostly among different two- or three finger grippers; products of reconfigurable dexterous hands available (Barret, 3-Finger-Shadow, SDH, ...) but still too expensive for industrial use.
- Pre-programmed grasping strategies are straight forward: end-effector configurations are taught together with the configurations of the robot arm / wrist
- Different grasps are pre-programmed, and a programmer can select a suitable grasp from a small repertoire of grasps.
- Grasping flexibility originates mostly from tool changers that allow the pre-programmed exchange of grippers depending on the task
- Grippers are becoming more flexible with respect to objects and tasks, i.e., flexible grippers with three and more fingers with multiple joints are available
- Mostly rigid bodies are manipulated and industrial equipment is available for rigid bodies. Where non rigid bodies are manipulated, touch sensors are required to prevent damage during manipulation.
- Grasping is mostly done with no controlled compliance (position controlled) as most manipulated objects are rigid
- Current technology only gives about one and a half hours use in a typical environment with a back pack. We really need a breakthrough in battery or small nuclear power sources to provide a viable prosthesis.

8.6. Mid term development (~2015)

- Grippers are becoming more flexible with respect to objects and tasks, i.e., flexible grippers with three and more fingers with multiple joints are becoming available
    - Grasping strategies for these flexible grippers will become available so that pre-programming of grasping actions become obsolete;
    - Grasping will be computed online depending on the object position and orientation and the task;
    - The recognition of object poses may still be limited, so that these poses may convey to the grasping system in a different way

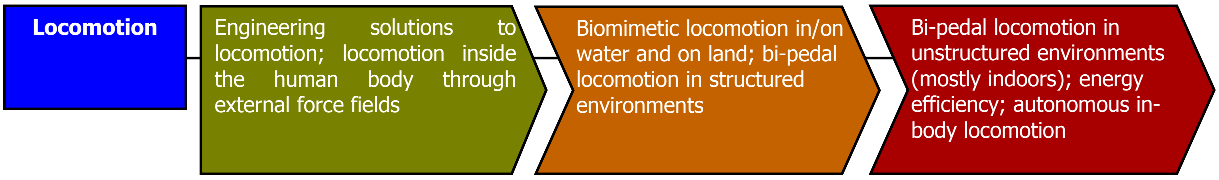
  - The number of object positions, orientations and shapes which is manageable by a gripper will still be limited as well.
  - Single handed operations prevail, but a clear trend is visible towards dual-handed operation to mimic human operation, e.g. in assembly tasks.
- Tool changers are gradually becoming flexible grippers with the ability to pick and use (human) tools to work on and handle objects
- Service robotic applications are deploying dexterous manipulators for tele-operation applications
- Industrial handling systems are able to perform simple manipulation tasks of limp materials like leather, technical textiles

8.7. Long-term development (~2020+)

- Traditional industrial two and three-finger grippers are still in use because of their robustness and simplicity, however, more sophisticated object handling tasks (especially in service and space robotics) will need dexterous hands with multiple fingers that will be robust as well
- Dexterous hands may be used as prosthesis, fully integrated with the neural signals of the user.
- Grasping strategies extend from using single hands to multiple hands;
- Grasps and dexterous operations are computed online depending on the object pose and task (→ homework for the planning people); which are perceived through sensors in-hand and sensors looking at the object and hand from the outside.
- Many different objects can be managed; human-like assembly becomes possible, but does still not match human robustness and tolerance levels
- Industrial handling systems able to perform complex manipulation tasks of limp materials in general including any kind of technical textiles and fabrics.
- Multi-hand cooperation
- New technologies for micro-grasping and micro-handling (new physical principles, complex tasks)

9. Locomotion

9.1. High level document



Locomotion

Engineering solutions to locomotion; locomotion inside the human body through external force fields

Biomimetic locomotion in/on water and on land; bi-pedal locomotion in structured environments

Bi-pedal locomotion in unstructured environments (mostly indoors); energy efficiency; autonomous in-body locomotion

9.2. Definition of terminology

Ability to move the robot from place to place. Locomotion allows a robot to move to a specified location on the ground, in the air, in space, on or under water, or inside a living body. Locomotion includes techniques like propulsion, propelling, legs, wheels, snake like, etc.

9.3. Drivers of technology

9.3.1 Driven by robotics:

- Innovative locomotion principles are developed by studying and replicating the natural locomotion mechanisms of animals and humans (biomimetics)
- Locomotion control is robotics specific, especially for autonomous vehicles (except for autonomous car, autopilot for planes and ships)

9.3.2 Driven by others:

- Robotics uses often locomotion devices and technologies that originate from other branches (automotive, aerospace, ship)

9.4. European strengths and weaknesses

9.4.1 Strengths:

- Varying according to environment in which locomotion is required
- Some competencies regarding air locomotion in Europe
- Europe has good competencies in bio-inspired locomotion (multi-legged, snakes, fish robot)
- In Europe, we are good in adaptability from control point of view.
- Existing competencies within Europe regarding under sea robotics (ifremer, Network: freesubnet, seabyte, ECA, Southampton Oceanographic Institute, Norwegian: Kongsberg Maritime; Sintef, DFKI Robotics Lab Bremen, Germany, Heriot-Watt Universtiy Edinburgh UK, Scuola Superiore St'Anna (SSSA) Pisa Italy, Centre for Biorobotics Tallinn Estonia, DFKI Robotics Lab, Sintef)
- Existing competencies within Europe regarding bipedal robotics (Aldebaran, INRIA, CNRS LAAS)

9.4.2 Weaknesses:

- Europe has little competencies regarding adaptability according to environment.
- Bipedal locomotion mainly driven through Asia and their humanoid robots but expertise and industry exists in Europe

9.5. Short term development (~2010)

- Under water: wheels, tracks, propeller, water jet
- On water surface: propeller, water jet
- On land (earth or planetary exploration): wheels (standard, omni-directional), legs, legs-wheels coupled, magnetic tracks and suction for climbing flat walls.
- In air: jet engine, propeller, fixed wings, rotary wings, flapping wings, balloon
- Inside the human body: locomotion with the help of external magnetic field
- In space: wheels

9.6. Mid term development (~2015)

- Under water:
  - Fish mimicking, undulating swimming (snake), biomimetics
  - Under water: swimming with fins
  - Using swimming fin rather than propeller (energy efficiency)
- On land:
  - Snake movement and improvement of legs and legs-wheels coupled solutions in structured environment (stairs, slope,…)
  - Fast locomotion in unstructured ground: important issue
  - Using grapple fixtures
  - Light-weight robots use biomimetic principles for climbing (lizard feet)
- In space:
  - Grapple fixtures, chemical, electric
  - Legged locomotion for exploration of difficult terrain
- Miniaturisation of locomotion
- Artificial muscles
- Flapping wings

9.7. Long-term development (~2020+)

- On land: locomotion biped or multi-legged in unstructured ground (in mountain lane)
- Adhesion forces for climbing walls (Gecko lizard like)
- On land: robots climb vertical walls (unspecified wall).
- For planetary exploration
    - Wheels and jet propulsion (hoppers principle)
    - Legged robots
- Anti-gravity device
- In space: combination of wheeled locomotion and propulsion (hoppers)

10. Materials

10.1. High level document

10.2. Definition of terminology

The elements or substances of which something is composed or can be made of. Materials are chosen to facilitate and implement certain requirements of the system through their characteristics.

10.3. Drivers of the technology

10.3.1 Driven by others:

- Materials research and development is mostly driven by other branches.
    - When there is a need to realise certain system behaviour, e.g. a washable robot surface or a light-weight structure, robotics developers are using the materials that are available.
    - However, more intelligent materials have the potential to contribute more extensively in the medium and long term development to other robotics technologies like manipulation, grasping, movement, locomotion.
- Funding bodies, e.g. the EU can also be considered drivers of materials: part of programme covers Nanomaterials.
- Material recycling aspects driven through regulatory bodies, will play more important role, but starting to become relevant only in the mid term.

10.4. European strengths and weaknesses

10.4.1 Strengths:

- Europe is quite strong in materials science and materials engineering which are interdisciplinary fields involving the properties of matter and its applications to various areas of science and engineering. Fields included are applied physics and chemistry, as well as chemical, mechanical, civil and electrical engineering. Recently, nanoscience and nanotechnology became popular. Examples for European leadership in materials science and engineering are
    - Piezo-based sensors and actuators, e.g. for adaptronics (e.g. active damping)
    - Magneto rheological fluids (e.g. for drive systems)

  - Composites (fibres, ceramics…) and light metal foams (e.g. as link material for robot manipulators)
  - Functional integrated material (sensor and actuators integrated, conducting light and current)
  - Soft materials and fabrics for bumpers and other protective measures
  - In the microtechnology area: “Micromanufabrication / Materials are bigger in Europe, but not robotics”
  - In the nanotechnology area Europe is strong with Carbon Nano Tubes developments, but in general Europe is lacking behind

10.5. Short term development (~2010)

- Shape memory alloys (applied to micro-robots actuation)
- Electro-active polymers (applied to micro-robots actuation), conductive polymers
- Optical-electrical conductive materials
- Polymer surface finishing
- Composites: drop in production cost
- Carbon fibre: drop in production cost
- Bragg fibres (strain measurements)
- Ceramics (TBC)
- Metal foams: simplified production cycle, improvements for larger parts

10.6. Mid term development (~2015)

- Shape memory alloys (applied to robot reconfiguring)
- Electroactive polymers (applied to robot reconfiguring)
- Biomimetic materials
- Sensing composites
- Functional composites
- Metal foams: decrease in production cost expected, driven through construction sector
- Nanomaterials, e.g. carbon nanotubes: wider use in robotics sectors for some robotics parts because of parallel reduction in production cost
- Biodegrading (environment friendly) materials, driven by regulations

10.7. Long-term development (~2020+)

- Nanomaterials: wide use in all robot parts (self-cleaning surfaces, distributed “skin” sensing, ultra-light structures, smart displays, reconfiguring components, others to come)
- Small structures with distributed sensing capabilities, integrating mechanics and electronics, combining sensing and actuation at microlevel or even nanolevel
- Bio-engineered tissue materials

11. Navigation

11.1. High level document

Navigation

Navigation expensive (computation & sensors); localisation and mapping in controlled environments solved

Some perception based localisation; SLAM for challenging environments; planners consider dynamic objects

SLAM in unconstrained environments; planning with dynamic, non-co-operative obstacles through perception

11.2. Definition of terminology

Process of controlling the movement of a system from one place to another reliably. A navigation system usually relies on the following sub-systems:

- Localistion: determination of the systems position and attitude with respect to a defined coordinate frame based on the robot's perception
- Path planning: determination of a path from target to goal, including the input of localisation; The trajectory should be adopted to avoid dynamic obstacles (re-planning)
- Path execution: translates a path into a sequence of commands for the robot's actuator; this can potentially include the ability to locally modify the path computed by the planner while travelling
- Mapping: computes a map suitable for localisation and path planning from a sequence of sensor readings

(SLAM is a methodology to carry out localisation and mapping at the same time)

11.3. Drivers of the technology

11.3.1 Driven by robotics:

- The combination of localisation and mapping using sensors is robotics driven.
- Motion planning (trajectories of platforms and robotic arms)
- Obstacle avoidance and path re-planning during trajectory execution
- Localisation of ground based vehicles, underwater robots and small aerial vehicles: Robotics uses existing systems to achieve this or uses existing sensors to do it. The development of localisation and mapping algorithms using such sensors is robotics driven. (Especially in the field of Aerial Navigation for aircrafts and long-range military drones there is a lot of ongoing research as well. The used techniques are to some extent comparable – Kalman-Filter, Particle-Filters, ... – to some extent specific – SAR, Terrain Reference Systems (Radar/Vision), ... - for that field)
- Autonomous Mapping of indoor environments is mainly robotics

11.3.2 Driven by others:

- Sensors e.g., laser scanners, radars, vision systems
- Localisation systems for navigation purposes, e.g., GPSs, inertial navigation systems, wireless sensor network

11.4. European strengths and weaknesses

11.4.1 Strengths:

- Simultaneous localisation and mapping (SLAM)
- Localisation and motion planning
- Controlled trajectories for autonomous vehicles
- Obstacle avoidance

11.4.2 Weaknesses:

- Navigation in unstructured outdoor environments

11.5. Short term development (~2010)

- Localisation:

  - Localisation problem is well understood (global localisation, tracking and kidnapped robot problem)
  - 2D localisation solutions available for outdoor/indoor (structured environment) and in well defined areas: using external localisation means (e.g., beacons, guidelines, laser scanner and reflecting markers; GPS outdoors); applied computational methods: Kalman filter, no simultaneous map building; maps are largely topological
  - Approaches used in research: probabilistic localisation, particle filter, grid based maps, landmark based maps and Kalman filter.
- Mapping:
  - Mapping problem is well understood: limited applications in very well defined areas, feature maps, grid maps
  - Maps are usually manually built or derived from a building's CAD data
- Motion planning:
  - Motion planning with limited scope of plan: waypoint calculation; avoid known and relatively slow and collaborative dynamic obstacles, taking into account physical properties of the own vehicle
- Sensors:
  - Consideration of safety aspects of navigation (e.g. in AGV-Systems, but comparably slow motion in contrast to cars)
  - Navigation techniques are based on using single or multiple sensors and multi-sensor fusion (GPS, Sonars, Lasers, Giro, IMU, PM, Odometry…)
  - 2D Sensors available with adequate properties and price
  - Applied in research: 3D sensors in indoor environments (e.g., CSEM Swiss Ranger) and in outdoor environments (e.g., Velodyne LIDAR, rather expensive for outdoor), as well as visual Systems (high-res cameras)

11.6. Mid term development (~2015)

- General:
  - Consideration of safety aspects of navigation with faster motion.
  - Strengthen competencies in navigation for unstructured outdoor environments.
- Localisation:
  - Localisation based on the perception of general environmental features moves from lab status to product status (increased robustness and accuracy) in limited environments
  - Active localisation: improvements expected as more resources are provided for self-localisation
  - Vision based approaches (optical flow, visual odometry, ...) might gain importance as "low-cost" additions improving performance and stability of traditional approaches
  - Topological ultra-low-cost approaches for domestic applications (vacuum cleaners, lawn-mowers)
- Mapping:
  - "2.5D" Mapping: efficient, accurate and multiple representations for large (a few kilometres) environments with some dynamic objects, and terrain classification.
  - Research: 3D-Mapping: efficient and accurate maps for dense, small-scale (a few meters, mainly for manipulation tasks) or rough large-scale (a few kilometres) environments
  - SLAM is improved for unmodified and dynamic environments, VSLAM gains further importance
  - Development of user interfaces which allow a user to influence the mapping process to guarantee a consistent map in 100% of the cases
  - Development of systems which can reliably operate with cheaper sensors (i.e. cameras and structured light emitters) in limited domains

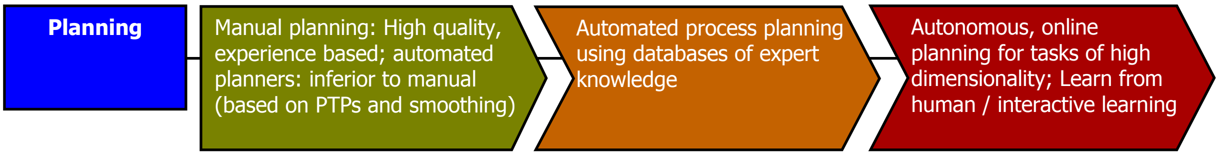
- First "life long" mappers
- Motion planning
    - Motion planning for larger maps, more efficient plans, more dynamic environments in real products
    - Cognitive approaches in planning
    - Planning considers obstacle avoidance: improvements and solutions from the technological point of view; further support through improvements within the regulatory framework
    - Collision avoidance through geometrical reasoning on static obstacles and objects
    - Need to consider interaction with other entities: moving in populated areas needs to consider social rules / standards / common reactions… (e.g. moving down a corridor on the right hand side, not the left)
- Sensors:
    - Especially 3D sensors will become more affordable, in general improvement in performance, reduction of cost and size, increased reliability with respect to environmental interferences, improvements with hard to detect obstacles
    - Vision based systems need to become available as only viable solution for many situations / markets

11.7. Long-term development (~2020+)

- General:
    - Further increase in the reliability of the system for certification and airworthiness purposes.
    - Safe and reliable navigation in unstructured outdoor environments.
- Localisation:
    - Low-Cost localisation through visual approaches
    - For high-performance industrial systems visual localisation may be used to bridge regions of lower importance (travel of AGV's between several working areas) significantly reducing the installation costs or increasing the operating area of the systems
- Mapping
    - 3-D mapping & visual mapping
    - Object representations with symbolic interpretation
    - SLAM in large and dynamic environments
- Motion planning
    - Planning considers obstacle avoidance of dynamic obstacles and objects, cooperative obstacles and non-cooperative obstacles, through more sophisticated sensing
- Sensors:
    - Sensors are largely benefiting from results coming from MEMS and optical developments
    - Smart sensors: sensors with embedded algorithms

12. Planning

12.1. High level document



Planning

Manual planning: High quality, experience based; automated planners: inferior to manual (based on PTPs and smoothing)

Automated process planning using databases of expert knowledge

Autonomous, online planning for tasks of high dimensionality; Learn from human / interactive learning

12.2. Definition of terminology

Planning is the selection of path, actions, tasks, policies, and procedures to achieve the mission of a robot. Task or mission planning may comprise motion planning, path / trajectory planning, grasp planning, manipulation planning, route planning, constraint management and resource coordination (scheduling).

12.3. Drivers of technologies

12.3.1 Driven by robotics:

- Symbolic planning
- Task planning
- Grasp planning
- Manipulation planning
- Path planning / trajectory planning (especially depending on process parameters)
- Motion planning

12.3.2 Driven by others:

- Robotics is not the only driver of above mentioned aspects (path planning→ manufacturing, aeronautics)
- Planning theory (e.g., optimisation)
- Scheduling
- Simulated plan validation
- Structural and computational biology / molecular biophysics (e.g. protein folding)

12.4. European strengths and weaknesses

12.4.1 Strengths:

- Motion planning for arms and platforms (on academic and product level)
- Task planning (more AI)
- Interactive learning / programming by demonstration
- Trajectory / motion planning on product level (but only one company: Kineo)
- Situation awareness (for domestic robots)

12.4.2 Weaknesses:

- Small companies in Europe
- Academic research on mobile robotics / space robotics is heavily driven by the military and space organization in USA
- Fragmentation of interested organizations
- Situation awareness (Low TRL i.e. laboratory level in the majority of areas)

12.5. Short term development (~2010)

- Path and mission planning are performed manually/ interactively with the aid of dedicated tools in many of the application areas. Collision free motion planning is done automatically.
- Replanning is performed manually/ interactively as well.

12.6. Mid term development (~2015)

- Replanning is performed automatically in most domain and planning from scratch with little user interaction

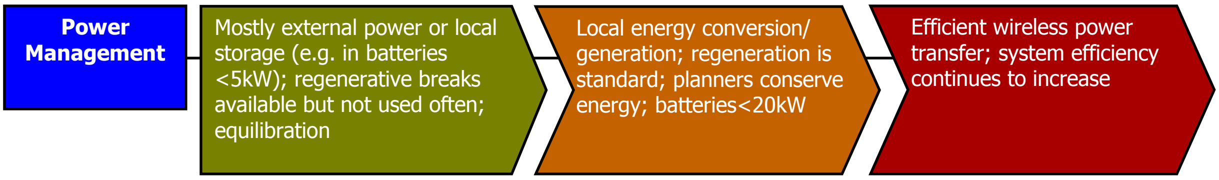
- Planning uses extendable knowledge bases / databases offering the end user and system integrator a way of inputting their process knowledge
- Offline realisation of time optimal planning where the planning complexity is high

12.7. Long-term development (~2020+)

- Autonomous planning for tasks of high dimensionality.
- More complex robot systems, such as mobile manipulators, are planning for themselves including analysing the proper sequence of steps, making adjustments to parameters to avoid problems and to plan and re-plan both path and task in an optimal/sub-optimal way under time constraints
- Interleaved planning and execution with environmental model produced online.
- Cognitive approaches for planning available with tradeoffs between possibilities: search algorithms
- Robot can learn or be instructed by human to take previously unknown constraint into account (interactive learning)
- Artificial intelligence: interesting for final planning evolution, understanding human instructions for the process planning as a first step, autonomous decision making

13. Power management

13.1. High level document



Power Management

Mostly external power or local storage (e.g. in batteries <5kW); regenerative breaks available but not used often; equilibration

Local energy conversion/ generation; regeneration is standard; planners conserve energy; batteries<20kW

Efficient wireless power transfer; system efficiency continues to increase

13.2. Definition of terminology

Set of technologies generating, storing, providing and conditioning power to the system and to ensure that the system makes the most efficient use of the power available to it at any given time while performing its task and minimizing emissions.

13.3. Drivers of the technology

13.3.1 Driven by robotics

- sensor power management typical for robotics

13.3.2 Driven by others

- Plane manufacturers (UAVs, sensors, power management)
- Car manufacturers (Electrical vehicles)
- Consumer electronics (mobile devices: mobile phones, laptops, music players…)

13.4. European strengths and weaknesses

13.4.1 Strengths:

- Electrical power management, power harvesting
- Fuel cell technology
- Energy efficient electric drives

- Alternative/ non electrical:
  - Bio digestion: Sludge cells
  - Compressed air
  - Hydraulic
    - Digital hydraulics
    - Water
    - Other fluids
  - Combustion: Diesel, Automotive industry
  - Solar energy and other renewable sources

13.4.2 Weaknesses:

- Japan: strength in batteries
- Wireless, especially important for nano-robots

13.5. Short term development (~2010)

- Storage systems:
  - Kinetic energy (e.g. flywheel)
  - Batteries up to 5 KWh
  - Capacitive storage
  - Chemical energy: gas, liquid, liquefied gas, liquid hydrogen, solids, coal, metals, e.g. zinc, aluminium, potassium, sodium
  - Refuelling
- Power harvesting, sensor power generation from robot movement /solar
- Local energy conversion: wireless power transmission: inductive systems
- Power regeneration:
  - Generative brakes
  - Heat

13.6. Mid term development (~2015)

- Local energy conversion:
  - Fuel and solar cells
  - Potential energy (glider)
  - Wired power transmission (nuclear, satellites, submarines)
  - Harvesting from surroundings (wave energy, wind energy, bio ingredients for fuel regeneration)
  - Chemical, combustion, electric, heat, pneumatic and hydraulic engines.
- Energy management:
  - More efficient devices/drives
  - Intelligent use of devices/drives to conserve energy
  - By advanced control and path planning
  - Switching off unused components (task dependent, usage dependent, ACPI approach from laptops / mobiles, user-friendly)
- Storage systems:
  - Batteries up to 20 kWh

13.7. Long-term development (~2020+)

- Energy management:
  - Spring systems
- Local energy conversion:

  - Powering flying robots through laser beams
  - Special purpose motors(small motors, light weight motors, low internal inertia motors, dynamic use of motors (start-stop)
  - Microbiological power generation
  - Wireless power transmission: infrared and microwave
- Storage systems:
  - Increased battery energy density

14. Control

14.1. High level document


Control

Control through cascades, state-space controller, sliding mode, feedback linearisation

Predictive , distributed, self calibrating, self tuning

Fault tolerance, automatic reconfiguration

14.2. Definition of terminology

The algorithms and mathematics to manage, command, direct or regulate the behaviour of devices or systems (not hardware).

14.3. Drivers of technology

14.3.1 Driven by robotics:

- High level control (of robot systems and automation processes)
- Advanced kinematics and dynamics control
- Force control
- Non-linear control of articulated systems
- Advanced control for locomotion
- Safety for physical HRI at the planning/control level
- Control of variable-stiffness joints
- Control based on robot dynamics

14.3.2 Driven by others:

- Control theory
- Image processing
- Dependability in the controller?
- Lightweight/flexible materials

14.4. European strengths and weaknesses

14.4.1 Strengths:

- Dynamic walking
- Vehicle control (including aerial and underwater vehicles)
- Robot arm control
- Hand control
- Visual servoing
- Force control
- Control of process and motion of industrial robots
- Mobile base (wheeled and legged) control

14.4.2 Weaknesses:

- Humanoids (exception Aldebaran, SSSA)
- Service robotics
- Field robotics

14.5. Short term development (~2010)

General engineering principles for shaping the feedback and feed forward compensators:

- Cascades
- State-space controller
- Coordinated control
- Sliding mode (e.g. variable structure control)
- Feedback linearisation which is equivalent to ‘computed-torque control’
- Feed forward decoupling
- Structured and more widely used model-based control, with more explicit and reusable dynamic models.
- Automatic code generation only for specific control functions.

14.6. Mid term development (~2015)

- Distributed control (not only in swarms)
- Predictive control
- H-infinity (linear controller with variable order)
- Internal model control (IMC)
- Self configuration, self calibration, self tuning
- Robust controllers (e.g. linear and nonlinear H-infinity, possibly with variable order)
- Loosely coupled control
- Optimization of highly redundant systems
- Adaptation to the changing conditions
- Adaptive control is relevant outside fault tolerance
- Bio-inspired control
- More automatic, or semi-automatic, use of the models (extended but in line with the short term development) for different purposes including:
  - Optimization of performance and robustness
  - Loosely coupled control
  - Predictive control (connection with fault detection)
  - Adaptation and reconfiguration (connection with fault tolerance)

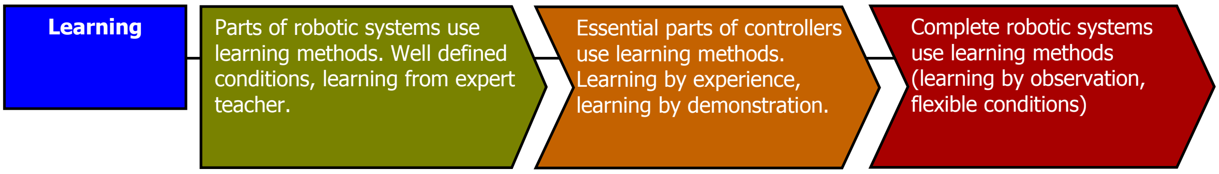
14.7. Long-term development (~2020+)

- Mostly dedicated & engineered model-based or mathematically sound control laws
  - Reconfiguration (connection with fault tolerance)
  - Geometric control (differential geometric methods in control design)
  - Adaptive control
  - Model-based control
  - The problem with model-based control in the long-term perspective (with a large number of robot devices and very short market opportunities calling for rapid engineering) will be to obtain and utilize the models in an efficient way. Continued progress will then call for:
    - Intelligent system identification and much enhanced modelling tools
    - Automatic code generation (for major parts of system)
    - System analysis tools considering robustness and failure modes
    - Self configuration, self calibration, and self tuning, for efficient (re)deployment

- Where we don’t know how to catch the behaviour or physics of the system to be controlled into equations:
    - M-Law
    - Fuzzy / Neuro (ways to control new actuators with new materials like polymers, no known “conventional” solutions)

15. Learning

15.1. High level document



Learning

Parts of robotic systems use learning methods. Well defined conditions, learning from expert teacher.

Essential parts of controllers use learning methods. Learning by experience, learning by demonstration.

Complete robotic systems use learning methods (learning by observation, flexible conditions)

15.2. Definition of terminology

Changes in the knowledge base of the robot gained through interaction with the environment (including people) that may result in a persistent change to the robots behaviour. Learning refers to the improvement through practice, experience or teaching.

15.3. Drivers of the technology

Currently robotics is one of the main drivers of this technology.

15.3.1 Driven by robotics:

- Learning about a changing environment
- Learning to interact with the environment (e.g. reinforcement learning)
- Programming by demonstration
- Basic research on machine learning is often evaluated / challenged by robotic applications (e.g. grand challenge, urban challenge, robot soccer, RoboCup@home, …)
- Developmental psychology provides insights on epigenetic approaches to robotics

15.3.2 Driven by others:

- General learning approaches often have AI / Cognitive Science drivers as well as neuroscience, e.g. mirror neurons
- Web technologies are also a strong commercial driver for learning systems, e.g. aspects of data mining on giga-sets
- Behavioural learning is also used by the games industry etc… so this is also computer science driven.

15.4. European strengths and weaknesses

15.4.1 Strengths:

- Numerous world-leading research groups in the field of learning including learning robotics in European universities and research institutes
- Selected related Networks of Excellence:
    - EURON (IST-2000-26048)
    - EUNITE (IST-2000-29207)
    - EVONET (IST-1999-14087)

  - NEuroNet1&2 (ESPRIT-4)
  - MLNet (ESPRIT-4)
  - NISIS (FP&-13569)
  - NEURO-IT-NET (IST-2001-35498)
- Selected related Projects:
  - ADAPT (IST-2001-37173)
  - ARTESIMIT (IST-2000-29689)
  - SIGNAL (IST-2000-29225),
  - BIBA (IST-2001-32115)
  - COGNIRON (FP6-IST002020)
  - ECAgents(IST-FET-1940)
  - NEUROBOTICS (FP6-IST-001917)
  - iCub
  - Pascal Network
  - COSY
  - POETICON
  - Check: The Learning and Adaptation projects in the portfolio of Cognitive Systems and Robotics projects (http://cordis.europa.eu/fp7/ict/cognition/projects_en.html)

15.4.2 Weaknesses:

- Big gap between academia and industry referring to proof-of-principle capabilities vs. customer needs
- Insufficient public support to close that gap (e.g. support of novel application fields for learning robotics)

15.5. Short term development (2010)

- Within robotics some well defined sub-tasks (e.g. object recognition, navigation, map-making, SLAM) are known to be solved using machine learning mechanisms. Nevertheless these developments are still mostly to be found within universities and research institutes. Industrial applications only seldom apply learning methods within the field of robotics. The existing solutions tackle only a small fraction of robot control, and apply only for well defined conditions.
- The learning approaches still require well defined circumstances/conditions.
- The behaviour of the robots is fixed (programmed).
- Application to very specific aspects of the task
  - Pattern recognition
  - Adaptive/learning control
  - Path planning and navigation (in part)

15.6. Mid term development (~2015)

- Big gap between academics and industry still exists.
  - Improvement in the existing learning mechanisms with respect to the specific robotics demands. Integration of the solutions developed within learning into robotic systems.
- The existing solutions are adopted for robotics for the different application areas mentioned. Selected parts (e.g. modules) of the robot control systems will be based on learning approaches.
  - Object recognition
  - Data representation, basic object ontologies
  - Navigation and map making (including 2D and 3D SLAM)
  - Learning/adaptive control (e.g. learning inverse kinematics)
  - The learning modules react flexible to changing conditions.

  - The behaviour is learned within strict pre-defined boundaries.
- Further developments are necessary within the fields of:
  - Learning by demonstration (e.g. human role model)
  - Reinforcement learning
  - Generalisation to novel/unknown situations
  - Decision making
  - Stability / Plasticity dilemma
  - Dependability and reliability

15.7. Long-term development (~2020+)

- Learning has more and more to do with systems that are able to internalise and use world realities including the observing organism in that world.
- Most robot controllers will have several modules based on learning.
  - Some robot controller structures are a result of the learning process.
- Some robot control systems will be completely based on learning:
  - The behaviour of the robot systems is a result of the (individual) learning process.
  - The learning robotic systems can adapt their behaviour to changing situations and altered requirements (training on the job).
- Make novel approaches from learning available for robotic applications.
  - Cognitive approaches
  - Life-long learning
  - Learning by observation of others (robots and humans)
  - Learning teamwork
  - Behavioural learning
  - Epigenetic developmental adaptation

16. Modelling

16.1. High level document

Modelling

Lack of standards for model descriptions, simulation not as good as real world experiments; high computation times

Standard language for model description; interchangeable models; modeling flexible and soft bodies; improved cybernetics

Real-time, dynamic modeling and interpretation allow for accurate assessment of the robot’s and the world’s state

16.2. Definition of terminology

Modelling is describing a part of the reality in a simpler way, so that it can be represented and controlled. A model is an approximation of reality, which is too complex to be mathematically described. The more precise a model is, the better the representation of the reality, and the more computation time is needed.

Examples of model usage are:

- By the engineers during the design process - through simulation of the kinematics, dynamics and mechanical properties,
- By the simulation software when setting up and tuning the control
- And by the embedded application software at run-time to enhance the control of the robot.

16.3. Drivers of the technology

16.3.1 Driven by robotics:

- Robotics may benefit from other domains in terms of elementary components but combining them to build a complex sensing/actuating system is specific of robotics
    - Robot optimisation: i.e. how to find the mechanism type and dimensions so that it fits given requirements
    - Uncertainty management: a robot is typical of a system whose modelling is affected by uncertainties although its safe behaviour may be critical (e.g. medical or space robots). To ensure that the robot behaves properly we may use mathematical methods but incorporating them to deal with all sources of uncertainties (computer, mechanism, sensing, control...) is quite specific
- Kinematical and dynamical modelling of robot arms and structures (e.g. humanoids, legged robotics)
- Kinematical and dynamical modelling of mobile platforms (e.g. AGV, AUV)
- Online (real-time) modelling of the environment, e.g. for obstacle avoidance, object recognition, reconstruction sensor modelling as prerequisite of sensor data processing
- Online (real-time) modelling of the interaction between the robotic system and its environment to improve autonomous behaviour; cybernetics and cognition
    - Calibration: both off-line and on-line

16.3.2 Driven by others:

- Most basic aspects of modelling are driven by other engineering or natural science disciplines but robotics has to integrate all these specific modelled components to build a simulation of the full mechatronic system
- Modelling for realistic physical simulation is driven, for example, by the demands of the automobile and aerospace industries
- Computer game industry (e.g. Physix)
- Modelling takes advantage of the possibilities provided by new or enhanced sensor technologies

16.4. European strengths and weaknesses

16.4.1 Strengths:

- Biomimetics / bionics (understanding nature to solve technical problems)
- Quite good in relation to the world (who ETH)
- Contrain dynamics
- Modelling force feedback (ABB, DLR)
- Cybernetics / cognitive systems (interaction between autonomous systems)
- Quite good in relation to the world
- Definitely kinematics is a strength (European school of kinematics)

16.4.2 Weaknesses:

- Human modelling
- Space
- Experimental validation of complex models
- Quite low in relation to the world
- Integration. We are able to model very specific aspects but we have difficulty to put together all the modules for a full simulation. This includes software development for such purpose.
- Robot controler: we definitely lack of an open robot controler that is flexible enough to allow to manage all sort of robots, sensors, actuators and modelling

- Not necessarily European: The big gap between industry and academia: how many robot controllers allow implementing efficiently a real-time dynamic model for control purposes?

16.5. Short term development (~2010)

- Robot kinematics / dynamics modelling
- Tools for modelling (Mathlab, Modellica)
- Lie-algebra, se(3)
- Modelling of gear elasticities
- Modelling of motor dynamics
- Interaction between robot and environment not clear enough to be modelled dynamically
- Most complex models study the behaviours of the systems without considering their interaction with the environment
- Absence of models of reflex-like dynamics
- Models are currently too rigid to face “natural” adaptation
- Biomimetics: simple imitation of a natural system (kinematics). No quantitative evaluation of the model
- Environmental modelling
- Rigid modelling, collision modelling
- Real time obstacle avoidance
- Autonomous navigation through changing environment
- Missing computation time for a realistic response of an autonomous system (particularly in case of complex embedded robotics)
- Sensor data modelling:
- Input data not precise enough
- Input data not robust towards environmental influences
- Model not sufficient to breach gap between sensor data and reality
- Modelling for sensor fusion through calibration
- Representation
- Many standards and representations (e.g. vast number CAD model standards)
- X3D, Collada, JT, first steps towards interchangeable 3D data formats that integrate other than just 3D data (e.g. dynamics). Exists, but it’s not supported by industry.

16.6. Mid term development (~2015)

- Robot kinematics / dynamics modelling
- Standard language for modelling setup (Mathlab++)
- Modelling of structure elasticity
- Deal with local uncertainties for an autonomous evolution in the real world
- Simple hybrid models increasing adaptation of the robotics systems in its environment
- Biomimetics: quantitative evaluation of a simple imitation of a natural system by an artificial one (kinematics).
- Environmental modelling
- Flexible/Soft adaptive modelling
- Standard model for flexible materials
- Unstructured environments only solvable through improved sensors
- Less complex / easier models not expected → further abstraction to high level models
- Autonomous learning and recognition of objects
- Sensor data modelling:
- Main improvements stem from sensor data quality
- Simplification of models
- Improvements in optical/ vision sensor
- Sensor fusion e.g. vision & force torque
- Representation:
- Unified model for kinematic and dynamic

- High level description of processes and tasks
- Models include besides 3D data descriptions of other object properties like product live cycle information
- Supported standards available for interchangeability
- New elements easily integrated into the model
- As design process is heavily interwoven, it is necessary to have an interactive manageable design process

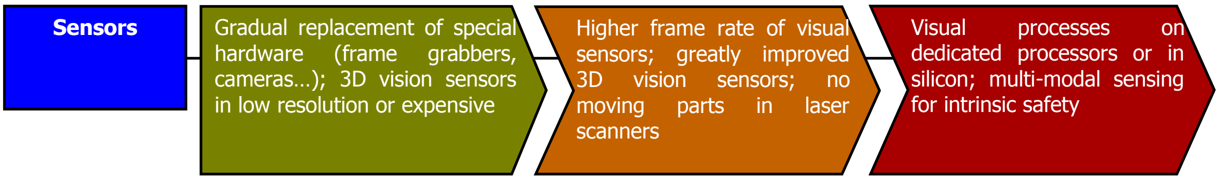
16.7. Long-term development (~2020+)

We need modelling for or simulation, design and control.

- Having a generic simulation software framework allowing to easily integrate all sort of models to be able to fully simulate the robot
- Having a generic robot controller that accepts all sort of models, control laws, sensor inputs etc...
- The management of uncertainties will be a key component of success. We may not need more sensors (we may even need less!) if we are able to take into account the uncertainties in the modelling.
- Adaptive modelling in real-time for enhanced online control
- Modelling more than just kinematics and dynamics. (Power consumption, wear) in one unified model setup
- Quasi-autonomous evolution in hostile environment
- Models of behaviours including tolerance to the real world uncertainties
- Complex hybrid models experimentally validated (including reflex-like behaviours)
- Biomimetics: quantitative evaluation of a mid-complex imitation of a natural system by an artificial one (kinematics and dynamics).
- Direct interaction between robot and human in clinical applications (motion assistance and replacement: autonomous exoskeletons and intelligent prostheses)
- Representation
- Relationships between the objects can be represented and used for reasoning
- Highly abstract models enable advanced human machine interaction and better productivity
- Full scale complete simulation system with the management of uncertainties, allowing to deal with all sort of mechatronic system and all of their components (mechanical, computers, electronic, sensing etc..)
- Robot appropriate design methodology including the management of all sources of uncertainties
- Real-time calibration method running as a background task allowing to constantly update the robot models
- Generic robot controler, highly flexible in term of inputs, modeling, outputs, control laws etc...

17. Sensors

17.1. High level document


Sensors

Gradual replacement of special hardware (frame grabbers, cameras...); 3D vision sensors in low resolution or expensive

Higher frame rate of visual sensors; greatly improved 3D vision sensors; no moving parts in laser scanners

Visual processes on dedicated processors or in silicon; multi-modal sensing for intrinsic safety

17.2. Definition of terminology

A sensor is a device which detects or measures a physical parameter and converts it into an electrical signal which can be used by the robot system. Actual sensor devices may include embedded sensing (see Section 18).

17.3. Drivers of the Technology

See also division in the timely development sections!

17.3.1 Driven by robotics:

- For some sensors we have special requirements, e.g.:
    - Skin sensors / pressure sensors
    - 3D vision systems
    - Sensors for nano robotics
    - Laser scanners (safety aspects)
    - All other sensors for which the specific constraints (energy, size, weight) imposed by robotics drive the development (for example, radar).
- Risk analyses for robotic applications often reveal that sensors have to be improved with respect to safety; sensors are not available at the required Safety Integrity Level (SIL).
- Integration of sensing processes into the sensors (smart sensors), also relevant to security.

17.3.2 Driven by others:

- Reliability by redundancy of sensors (in terms of numbers and with respect to different measurement principles of the same physical entity) is, to an extent, a robotics issue.

17.4. European strengths and weaknesses

17.4.1 Strengths:

- No sensors, which could be driven by robotics, which we could not make/develop in Europe
- 3D sensors: PMD, Swiss ranger; Metris, Leica, Faro, Zoller&Fröhlich
- Laser scanners: SICK
- Microsensors and Nanosensors
- Inertial Sensor Platform
- Camera systems: Visioniar, Perception, LMI, Matrix Vision (applications close to Automotive)
- Imaging sonars and radars

17.4.2 Weaknesses:

- Force Torque Sensors: ATI, JR3 (US Companies) but there are other suppliers. Not critical.
- Cameras: we don’t make the imaging chips, but this is not critical (we have someone for mini cameras in mobile phones; we also have advanced skills in some high level technology niche markets)

17.5. Short term development (~2010)

Many of the following technology developments will be driven by application areas outside of robotics. It is likely that their development will approximately follow Moores Law.

- Single cameras: range up to infinity, resolution degrades with distance, single shot: 12MPix, same as chemical imaging
- Industrial cameras lag behind consumer market, currently 2MPix, because robustness and integration into system takes time

- Amount of data: currently several cameras are used if necessary
- Colour depth: several million colours, industrial: 80% monochrom but this is sufficient
- Aperture: up to 90 degree or so, but lens dependant
- Communication channel: USB, firewire, GigabitEthernet (GigE) becoming standard

The development of the following technologies will be driven by Robotics:

- Specific frame grabber: especially for synchronised applications
- Panoramic cameras: range limited by resolution spread across 360 degree, resolution bad due to distortion, not good enough for obstacle avoidance, but good enough for navigation; frame rate same as single shot; colour depth same as single shot; aperture 360 degree
- Laser scanner: can distinguish between light and dark
- 1-d laser sensor as proximity sensor: sensitivity high power for dark surfaces, currently: >=2% reflection required, phase base lasers are very good in this respect, currently 500Hz for 70 degree, accuracy highest, frame rate highest, depth resolution highest
- 2-d laser sensors as proximity sensor: resolution 0.1 degree, navigation only up to 30m distance, aperture wide angle up to 270 degree for navigation and collision avoidance, wide angle up to 360 degree for navigation only, range currently up to 80m outdoor for security purposes, depth resolution can detect surface pattern on coin, mechanical properties rotating mirror
- 3-d laser sensors as proximity sensor: resolution as pivot scanner with high vertical resolution, but mechanically challenging, automotive use 4 scan levels
- Infrared sensors: range 40m for wall following and collision avoidance, available at low cost
- Ultrasound sensors: range up to 2.5m, no problems with dark surfaces, problems with sound absorbing surfaces, large detection angle, good for collision avoidance, small detection angle, not used for navigation any more, measurement of distance to surfaces, much cheaper than laser
- Multi-scale chirp based ultrasonic sensing, steerable arrays.
- PMD Sensor: vertical resolution higher than laser for object recognition, horizontal resolution lower than laser, for collision detection but not navigation, problems with edges, resolution approx. 70*70, sensitivity less dependent on reflective surfaces, more expensive than ultra sound, for low resolution cheaper than laser at round about €2000, for high resolution more expensive than laser greater than €10000 as prototypes (Other markets will drive the cost of these sensors down).
- GPS / EGNOS: Outside use only, resolution 1m, Kinematic GPS, Differential GPS RTK 1cm
- Internal GPS
- Proprioceptive sensors: Incremental encoders with odometry (navigation) or pose estimation, Position encoders, Resolvers, Rate gyroscopes for navigation, Inertial sensors as Accelerometer/ Gyrometer/ Inclinometer, Magnetometer, force sensor, torque gauge, strain gauge: may not be packaged into small enough designs for use on every joint
- In joint strain sensing, small internal packages for on link usage, self-calibrating gyros.

17.6. Mid term development (~2015)

The following technologies will develop without influence from robotics, which is likely to be a minor user in terms of sales volume shipped.

- Amount of data in visual sensors will gradually increase but no high pressure to develop
- Step changes/ improvements in frame rate of visual sensors expected
- Specific frame grabber will disappear more and more
- Cameras with high dynamic range, e.g. changes in lighting, sometimes sudden, less need for lighting systems
- Be able to handle structured light

The following will be partly driven by robotics:

- Smaller, cheaper sensor packages for inertial/force sensing

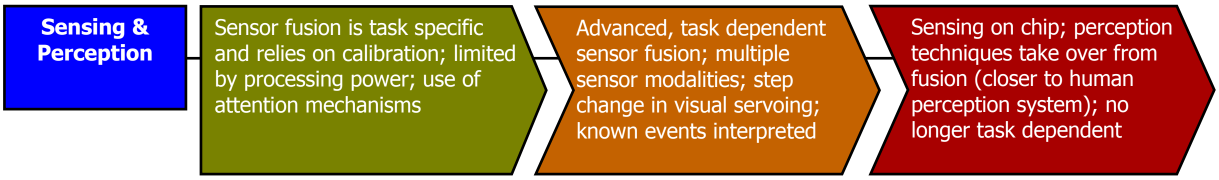
- Multiple cameras: ability to synchronise cameras, knowledge of relative position of cameras on one robot and in context of swarms
- Panoramic cameras: rectification of resolution required
- 1-d laser sensor as proximity sensor: no step changes expected in sensitivity, limited by physics
- 2-d laser sensors as proximity sensor: no improvements for navigation etc, different for quality control/surface analysis, frame rate requirement would be to have this frequency for up to 270 or more degree and higher power/longer range, mechanical properties no more moving parts
- Cost of laser sensors is expected to drop significantly
- PMD sensors resolution 1024*768, improved accuracy, no problems with edges, improved range, lower price
- Development of spatial awareness sensors based on dedicated sensor processing networks, higher reliance on parallel architectures and integrated sensor processing systems as pre-processors for more conventional sensor processing.

17.7. Long-term development (~2020+)

- Development of integrated multi-resolution sensors
- Dedicated recognition processors, silicon face recognition, hand recognition able to extract facial parameters or hand orientation pose. Body posture recognisers. Used in non contact haptic input devices. These will be used in toys, game interfaces and other entertainment devices.
- Dedicated texture and object sensors.
- Integrated multi-modal sensing for intrinsic safety of electro mechanical systems where torque and forces would currently preclude human co-operation.

18. Sensing & Perception

18.1. High level document



Sensing & Perception

Sensor fusion is task specific and relies on calibration; limited by processing power; use of attention mechanisms

Advanced, task dependent sensor fusion; multiple sensor modalities; step change in visual servoing; known events interpreted

Sensing on chip; perception techniques take over from fusion (closer to human perception system); no longer task dependent

18.2. Definition of terminology

A robot system uses sensing and perception to gather information about its own state and the surrounding environment. Sensing is the ability of a robot system to obtain information about the environment through its sensors. Perception is the ability of a robot to build representations of the physical world from sensed data. Perception may therefore involve cognitive and learning aspects while sensing is not.

18.3. Drivers of the technology

18.3.1 Driven by robotics:

- Technology developments will partly be driven by robotics, but also by other uses of sensing technology. These include current technologies which will undergo development outside of robotics, but the application of them to robot specific problems will be driven by robotic applications.
- For robotics, the constraint of real time processing must be respected, which requires the use of less sophisticated, but faster algorithms
- Sensing as a prerequisite for perception is largely driven by robotics

- As robotics is the science of integration of (or fusion of) multiple sensor/sensing technologies is an integral part of robotics science
- Power saving sensors
- Close integration of sensing into the sensing chain and system architecture, for example in the context of motion
- Developing artificial perception systems is a main motivating force for robotics scientists to make their robots more intelligent
- Sensing and perception allow to give autonomy to the robots

18.3.2 Driven by others:

- Computer vision (visual sensing) is an important discipline in its own right; robots often employ visual sensing for object detection and other environmental recognition purposes
- Human and animal perception has been largely studied by psychologist and physiologists
- The robots are designed by humans for whom the vision is the sense most developed.

18.4. European strengths and weaknesses

18.4.1 Strengths:

- SLAM (at university level)
- Bionics is very level across the world
- Very strong sensor industry, especially in smart sensors (Sensor industry is not driven by robotics)
- Force feedback sensing / haptic

18.4.2 Weaknesses:

- No products available image indexation

18.5. Short term development (~2010)

- Sensor fusion used to combine different sensors or viewpoints
- Data processing through statistics
- Data processing with optical flow
- Data processing through predictive: model based, Kalman
  - Used for visual servoing
  - Learning not used generally
  - High processing requirements but predeterminable /controllable
- Data processing through fuzzy:
  - Learning systems easily implemented
  - Processing requirements: variable
- Data processing through neuro:
  - Learning systems easily implemented
  - Processing requirements variable
- Data processing through competitive
- Data processing through Bayesian:
  - Model based
  - Used for colour analysis, segmentation
- Data processing through complementary data
- Data processing through data-mining
- Data processing through reasoning: inferential (inductive and deductive), constraint based, Bayesian, fuzzy/neuro-fuzzy, first-order logic, model-based, case-based
- No cognition here for data processing

The following are developments in the industrial sector where correlations to CAD data are important.

- Many of the mentioned problems have been solved by using different models, which approximate CAD models (such as super-quadric representations for object recognition), or considering aspects usually not described in CAD models, at least not the features used (e.g. texture). If you consider, for example, face recognition, you don't perform it by using a CAD model of the face.
- Object recognition in vision sensing: appearance based model using colour/intensity information in the model, Image indexing, learning based, with topographical matching; CAD data based system identifying artificial information with sensor data, currently learning required to link sensor data to CAD data (not object specific), weakness CAD data needed; positioning often implemented through object recognition
- Free space recognition in vision sensing
- Visual servoing in vision sensing using sequence of images, track objects, appearance based, CAD data based but models need to be generated beforehand (processing requirements are high) and processing takes 3-4 sec → much too long
- Since perception is a resource-intensive task, it is limited by the lack of onboard computing power in commercial platforms and it has to be highly task-dependent
- Artificial perception focuses on single sensor modalities, i.e. only visual or tactile or auditory sensing; multimodal perception combining multiple different sensing sources is still a matter of research
- For single modalities perception is also active, i.e. information from the sensing source is exploited to direct or focus the sensing modality to the region of interest
- A few concrete examples:
  - Speech / audio recognition for one speaker
  - Object recognition:
    - Object recognition of less than100 objects in real time
    - Object properties can be retrieved through a single sensor modality
  - Identification of humans
    - The robot recognises the human with which it interacts and understands his behaviour
    - Face recognition from the front
    - Simple emotion recognition (laugh, cry, cough, winge)
    - Gait recognition
  - Interaction with artificial systems is limited to one input channel for commercial applications
- Adjustable Autonomy to robots

18.6. Mid term development (~2015)

- Object recognition in vision sensing:
  - CAD data based system
  - No learning required anymore to link sensor data to CAD data
- Visual servoing in vision sensing:
  - Step changes through realisation of high dynamic range cameras
  - Smooth handover from visual servoing to other control/sensor mechanism
- Developments in cheaper more accurate sensors with higher resolution and sensitivity will require faster, more efficient algorithms to process the higher volumes of data produced.
- Development of audio (and other, e.g. smell, compound ultrasound, etc.) environmental data extraction providing event cues.
- Perception is still highly task-dependent, but includes now multiple sensor modalities (e.g., visual, tactile, haptic, force, auditory, chemical modalities) to classify scenes and to gain scene understanding; known events are properly recognized, classified and interpreted in the robot’s context
- A few concrete examples:
  - Multi speaker speech / audio recognition

  - Object perception
    - Relationships (between objects)
    - Object classification object recognition of less than10000 objects in real time
    - Identification of object properties
    - Identification of object affordances (cognitive context)
  - Identification of humans
    - Face recognition from different viewing angles and despite simple disguise
    - More complex emotion recognition
    - Human activity recognition
    - Safety-related classification of scenes for industrial applications
    - Distinction between humans, pets, objects
  - Interaction with avatars is multimodal
- Above items represent efforts toward cognitive approaches solution
- Vision System in Every Day Live (Real Time, without parameter to adjust)

18.7. Long-term development (~2020+)

- Development of uncertainty management techniques that might help to further improve reliability of data and perception. These are highly relevant for sensor fusion, object recognition and tracking (anchoring), and most of the sensor-based activities.
- Development of low cost high capability systems based on low cost integrated sensor/processor systems.
- Higher level sense data processing, reliable extraction of human emotional cues.
- Development of sensing systems able to track and identify 100s of targets in real time using multi-modal data.
- Perception systems are becoming more general (closer to human perception system) are not any longer only task-dependent
- A few concrete examples:
  - Object perception
    - Object recognition of more than 10000 objects in real time
  - Identification of humans:
    - Human intention recognition
    - Friend or foe differentiation
  - Interaction with robots is multimodal
- Perception system gives Full Autonomy to Robots