Robotic Welding
System Issues

J. Norberto Pires
Mechanical Engineering Department
University of Coimbra – Portugal

norberto@robotics.dem.uc.pt
robotics.dem.uc.pt/norberto/
Outline

1. Robot Technology State of the Art
2. Robot environment and Driving Forces to Automation
3. Difficulties in Automation
4. Improvements needed in the near future.
5. Welding Application - Overview
6. Software architecture
7. Remote Services
8. Adding equipment
9. Conclusions
Brief History

Robotics was part of the thoughts of many of the great thinkers of common history. From all of them we select:

**Ctesibius** (270 BC) ⇒ the **Greeks** and the **Arabians**

**Leonardo da Vinci** (1452-1519)

**Nicola Tesla** (1845-1943)
Ctesibius the Greeks and the Arabians

Briefly, the first works on robotics may be traced back until 270 BC, in the ancient Greece, to the water clocks with mobile figures designed by the Civil Engineer Ctesibius. His work was followed by Phylo of Byzantium (author of the marvellous book “Mechanical Collection”, 200 BC), Hero of Alexandria (85 BC) and Marcus Vitruvius (25 BC). Several hundred years later, the Arabians documented (the three Banu Musa working for the Kalifa of Baghdad, 786-833 AC) and developed (Badíás-Zaman Isma’Il bin ar-Razzaz al-Jazari in the book “The science of the Ingenious Devices”, 1150-1220 AC) the Greek designs to be used on their own creations.
Leonardo Da Vinci also spent some time on robotics, when he was working for the Sforza family. By the same time he painted “The last supper”, he was also involved with building the “Salle delle Asse” of the Sforza Castle, where he planned to put a human-like robot in the form of a XV century knight. Somehow, the plans and drawings were never found, although some pages of his famous book “Codex Atlanticus” are missing precisely in the point where it seems that he was preparing the robot project.
Nicola Tesla
did another outstanding contribution to robotics, in the turn to our century. He was thinking about automatons and how he could command them or “embody” intelligence on them. At the time, there was a German scientist (Hertz) claiming that an electromagnetic excitation generates radiation of the same type that can be detected far from the excitation. Tesla thought about using this to command an automaton: the term “tele-automatics” appeared. In its own words:

“... But this element I could easily embody in it by conveying to it my own intelligence, my own understanding. So this invention was evolved, and so a new art came into existence, for which the name “teleautomatics” has been suggested, which means the art of controlling movements and operations of distant automatons.”

---

These are the industrial robots of our days... How will they evolve?
Technical Milestones

- '74: Electrical Drive Train
- '74: Microprocessor Control
- '82: Cartesian Interpolation
- '82: Computer Communication
- '82: Joy-stick
- '82: Menu Programming
- '84: Vision Guidance
- '86: Digital Control Loops
- '86: AC Drives
- '90: Networking
- '91: Digital Torque Control
- '94: Full Dynamic Model
- '94: Windows Interface
- '94: Virtual Robot
- '94: Fieldbus I/O
- '96: Co-operating Robots
- '98: Collision Detection
- '98: Load Identification
- '98: Fast Pick & Place

Source: ABB Robotics
Robotic State of Art

1. Position and Motion controllers ⇒ Basically that’s all they can do.
2. PLC capabilities for IO control.
3. Ethernet, fieldbuses, serial connections.
4. Programming language or script for accessing resources.

<table>
<thead>
<tr>
<th>Repeatability</th>
<th>up to 0.03 mm (0.1 mm is common)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Velocity</strong></td>
<td>up to 5 m/s</td>
</tr>
<tr>
<td><strong>Acceleration</strong></td>
<td>up to around 25 m/s2</td>
</tr>
<tr>
<td><strong>Payload</strong></td>
<td>from 2-3 kg up to ~350 kg</td>
</tr>
<tr>
<td><strong>Weight/payload</strong></td>
<td>around 30-40</td>
</tr>
<tr>
<td><strong>Axis</strong></td>
<td>6</td>
</tr>
<tr>
<td><strong>Communications</strong></td>
<td>Profibus, Can, Ethernet and serial channels (RS 232, 485)</td>
</tr>
<tr>
<td><strong>IO capabilities</strong></td>
<td>PLC like capabilities to handle digital and analog IO.</td>
</tr>
</tbody>
</table>
Driving Forces

- Global Market
- Products defined in part by the customers.
- Products are technological complex and dense.
- Small batch manufacturing.
- More quality at lower prices.

Flexible and agile production

- Organization.
- Standards
- Programmable equipments.
- Decision support and information systems.

Programmable Automation.
- Robotics.
- Industrial computation.

Near Future

- Lower price
- Higher performance
- More sophisticated sensor control
- Better on-line control from prod. control systems
- Better simulation and off-line programming tools
- Better remote services
- Better quality
- Better diagnostics
- Lower energy consumption
- Support for new production processes
- **Force control** ⇒ As a regular feature
- Evolve to lighter structures

Better SOFTWARE ...

---

Near future

**Capacity**: better *performance*, lighter structures ⇒ better accelerations and velocities, better position control, force control ...
Actually dynamical models used by robots have 7-10% of error.

**Usability**: Programming environments, easier to use, some ways to access the original system and change its standard way of working ⇒ *object oriented software and programming*.

**Connectivity**: Standard links and protocols: nets, protocols, net services (telnet, ftp, disk mounting), RPC, RMI (Java), OPC.

We need robots very easy to command, in a way that they can follow complex requests done with “high level” interfaces.
What are the problems or difficulties of current robot control units?

We could summarize the main difficulties as follows, considering a researcher point of view and also a system integrator point of view:

**End-user programming**: It is still too complex, too platform dependent, non standard and consequently only for specialists;

**System programming**: APIs are very limited or non-existing, making a simple job of adding a new resource (a board, a sensor, a new feature, etc) almost impossible;

**Human Machine Interface (HMI)**: It is really not working. Systems require too many training and customization. There is some need for other approaches like skill learning so that robots could be easily customized and get into efficient production;

**Connectivity**: lacking of inter-connectable the facto standards;
Software architecture

Remote call to service_i of equipment_j
Synchronous answer from equipment_j to the remote call to service_i

Asynchronous answer "spontaneous message"

Software architecture

The implementation of this requires the use of three programming models:

**Client/server:** We require to have server code running on the equipment that deals with receiving calls from remote computers (clients), execute them and return the results.

**Remote Procedure Calls (RPC):** This is the usual way to implement communications between the client and the server of a distributed application. The client makes what looks like a procedure call, although the resource is not local. The RPC mechanism in use translates that into network communications. The server receives the request, executes accordingly and returns the results.

**Data Sharing:** We want to have services that share files, programs, databases, etc. The data sharing services will be built on top of RPC, which provides the means for transferring data.

**Object Technology:** OMG CORBA, COM/OLE/ActiveX, Java beans ...
Software architecture
Joystick Application

Request
Answer
Event = async. answer

Master/Slave
Tele-robotics

RFC server

Matlab example

Industrial Robot Example

Matlab 5.3

DDE

RPC Server

Robot

ABB IRB1400 S4 Controller

RPC Sync. Call

RPC Sync. Answer

RPC Async. Call

EmailWare
A tool for e-Manufacturing.

E-mail alerts
Logs and errors
System Status
Information
1. **Setup phase** - where the user sets up all the parameters and trajectories.

2. **Welding phase** - the system should monitor the welding process and correct on-line.

3. **Analysis phase** - detail the welding seam and quality analysis.
Conclusions

- State of the art of robotics technology was presented.
- Brief analysis of driving forces, difficulties and near to come innovations.
- Software needs.
- Our way of developing FMS solutions: software architecture.
- Several examples: academic & industrial & research.
- Application to welding was discussed.

Software presented can be downloaded from: http://robotics.dem.uc.pt/norberto/

Publications also on the site.
Some References


Final message:

Robotics as a technology will evolve enormously with the cooperation between academia & industry. In that point of view, it is very important to have many fields of application, like robotic welding.

In Europe there is a network that addresses Robotics research and development called EURON. Within EURON there is the interest in “industrial or applied thinking” ...

Check EURON in:

http://www.cas.kth.se/euron/

EURON is coordinated by Dr. Henrik Christensen (KTH, Sweden)

Check EURON “Industrial Robotics & Manufacturing Interest Group” in:

http://www.control.lth.se/~robot/euron/mig.html

This Group is coordinated by Dr. Klas Nilsson (Lund, Sweden), Dr. Gordon Petersen (Odense, Denmark) & Dr. J. Norberto Pires (Coimbra, Portugal)