Abstract
Using robots with industrial welding operations is common, but far from being a solved technological process. The problems are with the robots (still in early stages, difficult to use and program by regular operators), the welding process (complex and not really well known) and the Human-Machine interfaces (non natural, really not working). In this paper we discuss these problems and present a system designed with the double objective of serving our R&D efforts on welding applications, but also our need to assist industrial partners working with welding setups. The system is explained in some detail, and demonstrated using two test cases which reproduce two situations very common in industry: multi-layer butt welding (used on big structures requiring very strong welds) and multi-point fillet welding (used for example on structural pieces for the construction industry).

I - Introduction
If we flash remember the long history of robotics there are a few things we can learn in order to understand our present situation. Robotics can be traced back until 270 BC, in the ancient Greece, to the water clocks of the Civil Engineer Ctecibius. His work had followers like Phylo of Byzantium author of the book “Mechanical Collection” (200 BC), but also Hero of Alexandria (85 BC) and Marcus Vitruvius (25 BC). In the 12 century, the Arabian Badías-zaman al-Jazari (1150-1220) recollected some of the Greek developments in the book “The Science of the Ingenious Devices” [1], and that is how they reached our time. In those early times the problem was about mechanics, about how to generate and transmit motion. So it was mainly about mechanisms, ingenious mechanical devices [1][2].

Then in the XV century, Leonardo da Vinci showed indirectly that the problem at the time was mainly the lack of precision and of a permanent power source. He designed a lot of mechanisms to generate and transmit motion, and even some
ways to store small amounts of mechanical energy [3]. But he didn’t have means to build those mechanisms with enough precision and there was no permanent power source available (pneumatic, hydraulic or electric). Maybe that was why he didn’t finish his robot project [1][2], a XV century knight robot intended to be placed in the “Salle delle Asse” of the Sforza family castle (Milan, Italy). It wasn’t good enough. Or it was a so revolutionary idea for the time that he thought that maybe it was better to make it disappear [1][2].

And then there was the contribution of Nicola Tesla in the turn of the 19th century. He thought of using Henrich Hertz discovery of the radio waves (following the work of James Clerk Maxwell about electromagnetic phenomena) to command an automata. He built one to demonstrate his ideas and presented it in the Madison Square Garden (New York, USA) in 1905 [1][4]. The problem there was that machine intelligence was missing. Robots should be able to do pre-programmed operations, and show some degree of autonomy in order to perform the desired tasks. When that became available, robots developed rapidly and the first industrial one appeared in the beginning of the seventies and became a multi-million dollars business. Since then, evolution was not as fantastic as it could have been, since there was a lot to do and the available machines were sufficiently powerful to handle the requested jobs. Manufacturers were busy making money and more or less happy with their robots, and consequently industrial robots remained position controlled, difficult to program by regular operators, and really not especially exciting machines. Features currently common on research laboratories didn’t reach industry yet because of some lack of interest from industry.

Nevertheless, the situation is changing since actual market conditions are only compatible with small/medium batch manufacturing, due to strong competition and dynamical behavior of the market. In those conditions, robotic production setups exhibit the best “cost per unit“ performance if we compare with manual work and with hard automated setups (fig. 1) [5]. Consequently, near future requires powerful and more flexible machines in order to handle requests from small businesses, which need more remote interfaces, powerful programming languages, force control, powerful Advanced Programming Interfaces (APIs) for high level programming, etc. That means exposing to the user the flexibility stored inside the machines, as a result of several decades of engineering, which is currently barely used.
What makes robotics so interesting is that it is a science of ingenious devices, constructed with precision, powered by a permanent power source, and flexible in the programming point of view. That does not mean necessarily open source, but instead the availability of powerful APIs, and de facto standards both for hardware and software, enabling access to system potentialities without limitations. This is particularly necessary on research environments, where a good access to resources is needed in a way to implement and test new ideas. If that is available, then a system integrator (or even a researcher) will not require open source software at least for the traditional fields of robotics (industrial robot manipulators and mobile robots). In fact, that could also be very difficult to achieve since those fields of robotics have decades of engineering efforts, achieving very good results and reliable machines, which are not easy to match. That open source issue is nevertheless very important for the emerging robotics research (like humanoid robotics, space robotics, robots for medical use, etc) as a way to spread and accelerate development (fig. 2).
Industrial Robotic Welding is by far the most popular application of robotics worldwide [6]. In fact, a huge amount of products require welding operations in their assembly processes. The car industry is probably the most important example, with the spot and MIG/MAG welding operations in the Car Body Workshops of the assembly lines. Nevertheless, there is an increasing number of smaller businesses, client oriented, manufacturing small series or unique products designed for each client. These users require a good and highly automated welding process in a way to respond to client needs in time and with high quality. It is for these companies that the concepts of Agile Production [7][8] apply the most, obviously supported by flexible manufacturing setups. Despite all this interest, industrial robotic welding evolved slightly and is far from being a solved technological process, at least in a general way. The welding process is very complex, very difficult to parameterize and to effectively monitor and control [9]-[11]. In fact, most of the welding techniques are not fully understood, namely the effects on the welding joints, and are used based on empirical models obtained by experience under specific conditions. The effects of the welding process on the welded surfaces are currently not fully known. Welding can in most cases (i.e. MIG/MAG welding) impose...
extremely high temperatures concentrated in small zones. Physically, that makes the material experience extremely high and localized thermal expansion and contraction cycles, which introduce changes in the materials that may affect its mechanical behavior along with plastic deformation [9][11]. Those changes must be well known in order to minimize the effects.

Using robots with welding tasks is not straightforward and has been a subject of various R&D efforts [12]-[16]. And that is so because the modern world produces a huge variety of products that use welding to assemble some of their parts. If the percentage of welding connections incorporated in the product is big enough, then some kind of automation should be used to perform the welding task. This should lead to cheaper products since productivity and quality can be increased, and production costs and manpower can be decreased [17]. Nevertheless, when a robot is added to a welding setup the problems increase in number and in complexity. Robots are still difficult to use and program by regular operators, have limited remote facilities and programming environments, and are controlled using closed systems and limited software interfaces [18]-[22].

In this paper, we address some of these problems just by presenting a system built with the main objective of being a test bed for welding experiments. Our experience with the system shows that it has potentialities for industrial utilization, and in fact that idea is explored in the paper. For that purpose we selected mainly industrial equipment in designing the system, as a way to facilitate its industrial exploitation. Finally, the paper addresses aspects of system programming and welding parameterization, which constitute the main contribution of the paper.

After this introduction, the paper is organized as follows: Section II presents shortly the MIG/MAG welding process and suggests how to integrate available data into a full automatic parameter selection system. Section III introduces the technological aspects of the system and an overview of the software architecture. Section IV presents two test cases made with this system: multi-layer welding and a complex welding sequence, using data from CAD software to program the robot. Finally, some conclusions are drawn in section V, along with some ideas for future work.

II – Welding technology
MIG/MAG welding process, also known as gas metal arc welding process, uses the heat of the electric arc to melt the electrode wire and the metallic components to be welded. Figure 3 illustrates the welding principle. The fusion is carried out under
the protection of a gas, or mixture of gases, in order to prevent the pernicious contamination with some gases of the atmosphere (oxygen, nitrogen and hydrogen). Difference between MIG (Metal Inert Gas) and MAG (Metal Active Gas) processes is based in the type of shielding mixture used: inert gases (argon, helium) in the first case and active mixtures (containing CO₂ or O₂) in the second. MIG process is used chiefly in the welding of stainless steels, aluminum alloys and titanium alloys while MAG is in use in carbon steels.

![Welding diagram](image)

Fig. 3 – MIG/MAG welding process principle.

These processes are very popular in the automotive industry, in naval construction, in the boilers and pressure vessel industry as well in the light metalworking all over the world. The popularity of the process is a consequence of its high flexibility and high deposition rate (up to 15-20 kg/h of weld metal), consequently high cost-efficiency rate, associated to a short training period even in manual operation, and good aptitude for automation and robotics.

Proper selection of welding parameters, in combination with the suitable welding gas and the correct welding equipment and filler material, lead to the development of new welding procedures derived from MIG/MAG process. Examples of this are the Rapid Arc and Rapid Melt processes, developed at AGA AB [23] and TIME process of Fronius [24]. These processes allow increased welding speed and greater deposition rate than conventional ones.
The stability of the welding process is very sensitive to the main welding parameters, especially current, voltage, welding speed, stick-out (length of wire out of the contact tube), shielding gas and arc length [25]. A small change in the distance between the welding torch and the component being welded may produce a considerable variation on current and voltage. Current, voltage and shielding gas influence the transfer mode of melted filler wire to the component being welded, affecting the quality of the welds [26]. If the electric arc is unstable, defects like bad penetration profile, undercut or excessive spatter may occur.

As the weld bead shape may be closely related with the welding parameters, databases for MIG/MAG welding process have been developed, such as the one of the Welding Institute – UK [27]. In these databases the input data is generally the type of weld (butt weld or fillet weld), the welding position (flat, horizontal, vertical or overhead), wire diameter and the plate thickness or eventually the leg length in the case of fillet welds. The output data is usually the welding parameters (current, voltage, welding speed and number of weld beads/layers). With databases of this type in the computer the selection of the welding parameters may be made automatically. Even the selection of the wire diameter may be carried out automatically as a function of the thickness of the components or stay for free selection being an input parameter.

It would be expectable that with this information in the computer, having a CAD model of the component to be welded, the system could be able to select the welding data for each weld and send these data to the robotic welding system. Though it seems easy to achieve this goal in the case of single welds, some data is missing in the database for the case of welds with multiple layers. In fact in this case the position of the torch in each layer needs to be indicated to the robot (this is one of the test cases presented in the paper).

Since that for the majority of the companies that produce multi-layer welds there is only a small number of distinct welds, then it is not hard to fill up the database for their particular case.

With this method it is easy to carry out the off-line programming of the components to be welded, being only necessary to adjust the coordinates of the process points in the first specimen to be welded.
III – A Robotic Welding System

But why use robots with welding tasks? Mainly because of their flexibility, but also because they can perform very precisely human-like tasks. And welding is indubitably a human-like task requiring precise motions [17]. So, in some way robots are desirable. But actual robot state of the art is a problem. Robots are essentially position-controlled devices that can receive a trajectory and run it continuously. In fact, that is practically the only thing they can do [4]. With welding applications we need to start from a trajectory, given for example from a CAD model of the working piece, and have means to correct it in real time, as function of the observed results of the welding process. For that we need visual systems for guidance and inspection, the possibility to correct in real time the position of the robot and the welding parameters, and a computational platform suitable for developing the software to handle all these monitoring and control tasks [18]-[22], [28]-[29]. And these features aren’t usually available due to the following reasons:

1. Actual robot manipulators have closed controllers, not allowing real time position correction [18], [28]-[29].
2. Actual robot controllers do not allow remote control from an external computer.
3. It is very difficult to attach guidance sensors with good performance, because robot controllers are not prepared to it [9].
4. Robot programming environments are not powerful enough to handle tasks requiring complex control techniques (learning, supervisory, adaptive, etc).

With the system proposed here we reduced several of these limitations:

a) We use a robot control system that allows position correction commanded by a remote computer. That is not a standard feature, but was added by us to our system using features available from the controller [18]-[22].
b) We use a distributed software architecture based on personal computers running Win32 operating systems that enable remote control using Ethernet networks [18]-[22].
c) The guidance sensors are attached to the computer that controls the robot and not to the robot controller itself.
d) We use a personal computer as programming environment, taking advantage of the huge amount of programming and analysis tools available on those platforms.
As with any complex technological process, control a welding robotic process and monitor its quality means acting on the three phases of the process (before, in between and after):

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. This kind of approach is needed for research projects where users need to develop procedures to simplify and typify the "setup Phase", but also to identify a minimum set of parameters to be monitored during the "welding phase" with the objective of obtaining welds with high and constant quality (inside certain limits). Nowadays, the industrial welding systems are typified for certain traditional welding processes, using in some more demanding cases optical guidance systems. Nevertheless, usually no quality monitoring and control is performed by the system [28]-[29]. In consequence, those systems are unable to perform other welding tasks not previously adopted, and generally are not adaptable to those new situations. That is why there is some frustration with the actual robotic welding systems [30]. The system presented here (fig. 4) is able to act on the 3 phases of the process. Besides that, its operation is programmable and can be adapted to new situations. We finally want to make clear that our objective was to build a laboratorial system to be used with research and development tasks leading to the identification of suitable welding models applicable to automatic welding with a robot.

The proposed system (fig. 4) is composed by an industrial robot and its controller, a CCD camera for inspection, a laser 3D camera for guidance, the welding source and a workstation that will manage all the data acquisition and control of the robot position, along with the selection of the welding parameters in accordance with a data base that describes the welding process.

The software architecture used in this work, presented in detail elsewhere [18][20], is distributed using a client-server model, based on software components developed to handle equipment functionality. Briefly, when we want to use some kind of equipment from a computer we need to write code and define data structures to handle all its functionality. We can then pack the software into libraries, which are not very easy to distribute being language dependant, or build a software control
using one of the several standard architectures available (preferably ActiveX [31]-[32] or JAVA [33]). But other technologies could be used; the purpose here is on components and on integration with the environment chosen for operation, not in discussing the possibilities of each technology. Since we use win32 operating systems, mainly Windows NT and 2000, which are accepted standards in industry, ActiveX is somehow privileged because it was specially built for those environments and is based on DCOM like the operating system. Using a software control means implementing methods and data structures that hide from the user all the tricky parts about how to have things done with some equipment, focusing only on using its functions in a easy way. Beside that, those components are easily integrated into new projects built with programming tools that can act as containers of that type of software controls, i.e., they can be added to new projects in a "visual" way. We built several ActiveX software components to use with this project. Those controls expose to the programmer the basic functionalities of the equipment used (one ABB IRB1400 S4 industrial robot, a SIEMENS VS710 CCD camera and a M-Spot Laser camera). The welding power source (MIG power source ESAB A350) is controlled from the robot controller using the welding sequence presented in fig.5, and a client-server programming strategy [18]-[22].

**Setup phase** - where the user sets up all the parameters and trajectories.

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

**Analysis phase** - detail the welding seam and quality analysis.
Fig. 4 - Welding system diagram and aspects of its utilization in the laboratory.

Fig. 5 - Welding Sequence implemented by the robot controller (all the timings are programmable by the user).

The robot controller software works as a server, exposing to the client a collection of services that constitute its basic functionality. The robot can start the welding procedure, or terminate it, can be commanded to follow complex trajectories, to
simulate the entire process completely or step-by-step, etc, just by answering to remote commands issued from a PC connected to the robot by Ethernet. Basically, the user sends to the robot a complete definition of the welding task including: points, welding parameters (velocity, voltage and intensity), type of trajectory between and positioning precision, etc. All this information is stored in the robot controller and can be used to simulate the welding process and enable any adjustment necessary, or to start/stop the welding process.

Fig. 6 – Robot working as a server.

The user may start from a CAD model, preferably, and/or a copy of the working piece, and a definition of where the welds must be done. He should come up with an optimized program to perform the welding operation on every piece of the same type. From the CAD model the user can extract the points he wants to pass over and weld. Then he needs tools to adjust those points, add extra points, add approach and fly away trajectories, adjust welding parameters, test and simulate the all thing until the operation is as desired. For that, the following collection of tools was designed:
**WeldPanel**

With this tool (Fig. 7) the user can manipulate the welding points that may be obtained initially from a CAD model of the working piece. Those points may be changed or adjusted, and extra ones may be added in a way to avoid collisions, optimize trajectories, etc, and achieve best performance. All points are always referred to the welding torch Tool Center Point (TCP) and to a Work Object frame define in the table holding the working object (fig. 7). The user may adjust points just by moving the robot to the desired position. That can be done from the PC or passing control to the robot teach pendant, which is generally handier.

This tool also receives events from the robot, like status changes, actual state of the welding power source and related IO signals, etc. The status of the program running on the robot controller and of the network connection is constantly monitored, just to avoid damaging materials and persons by preventing system commands on error situations. Events are RPC calls made by the robot controller to an RPC server running on the PC as a service [35]-[37].

Fig. 7 – Shell of the WeldPanel tool.

**WeldAdjust**

This tool is used to adjust points on-line and to acquire points in any robot configuration and any program state. Basically it is a jogging application that enables the user to position the robot from the PC, using Cartesian XYZ commands or absolute joint commands.
Fig. 8 – Shell of the WeldAdjust tool.

**File Explorer**

With this tool the user can exchange files with the robot controller, facilitating the process of transferring programs, modules, etc, to and from the robot controller. It works like the windows file explorer, having the available robots as extra “disks”. The user can access the robot internal disk and also the external floppy disk.
Robot Control Panel and RPC server to receive events

Used to change the robot controller state and to load and unload modules from the robot controller. The RPC server is used to receive events from the robot controller. As already mentioned events are RPC calls [35]-[37] made by the robot controller and fired when pre-programmed actions actually occur. Actions include: IO change, system state change, program variable change, etc. All actions are programmable [18].
IV – Test Cases

Test case 1 – Multi-layer welding.

In this example we show how to perform a simple multi-layer weld using the definitions presented in fig.11 and tables I-II. The number of layers and the placement of each one of them are obtained empirically using charts from The Welding Institute and our own experience. The process is performed step by step, and any adjustment is introduced in the welding sequence being programmed. Those adjustments can be: position adjustments, welding parameter adjustments and introduction or removal of layers. Since the program is stored in a file, it can be used lately to weld other similar pieces. We think that the obtained procedure is easy to use and very useful for industrial use because the programmer can easily setup a multi-layer welding procedure controlling and observing the effect of each layer, and acting when necessary. Fig. 11 shows the working piece composed by two 20 mm-thick plates, separated by 2mm, with a V-groove joint preparation and the welding sequence (layers necessary to finish the weld). The position of the torch in each layer is indicated in Table I. The origin of the reference axis system is centered in the bottom of the V-groove. This information is generally not available in the welding databases. The welding data used in this case are indicated in table II.
a) Work Piece for multi-pass weld test case (two 20 mm thick plates, 2 mm apart from each other, with a 60º V-groove joint preparation).

b) Layers necessary (welding sequence) to finish the weld and obtained weld.

Fig. 11 – Aspect of the working object, welding sequence and obtained weld.

Table I – Position of the welding torch for each layer

<table>
<thead>
<tr>
<th>Layer</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>X (mm)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-5</td>
<td>+5</td>
<td>-6</td>
<td>+6</td>
<td>-6</td>
<td>+6</td>
</tr>
<tr>
<td>Y (mm)</td>
<td>20</td>
<td>24</td>
<td>28</td>
<td>32</td>
<td>32</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
</tr>
</tbody>
</table>

Table II – Welding data extracted from a database [27].

<table>
<thead>
<tr>
<th>Layer</th>
<th>Current (A)</th>
<th>Voltage (V)</th>
<th>Speed (mm/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>200</td>
<td>22</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>250</td>
<td>28</td>
<td>5</td>
</tr>
<tr>
<td>3-9</td>
<td>300</td>
<td>34</td>
<td>5</td>
</tr>
</tbody>
</table>

Distance from the welding torch to the working piece: 17 mm.

Test Case 2 – Multiple welding paths
In this example we show how to perform a multipoint weld, very common on companies that manufacture metal structures for the construction industry. The
idea is to extract points from the CAD model of the piece to be welded. It is usually very simple to build a routine within the CAD software, enabling the user to extract points from the working piece and defining the type of trajectories between those points. This may be the initial procedure, very handy for companies having CAD models of their products. After having the sketch of the definition file, the user must work with it using the **WeldPanel** and **WeldAdjust** tools. The working cycle should result in a properly tuned file for the purpose. An example of the definition file is presented in fig. 13. The welding parameters may be again obtained from a database or using well known charts. The process can then be simulated for trajectory and welding parameters adjustment, and tested until desirable performance is achieved, including acceptable welding quality.

b) Fillet weld preparation.

Fig. 12 – Aspect of the working pieces.

b) Working table in the laboratory.
Simple Example

4  Total number of Points
Ponto 1  Definition of Point
Origem  Name of the Point
1  Type of Point (0-welding, 1-flying)

Robtarget structure [35]
-446.816010  315.087006  436.980011
0.017820  -0.752200  0.658510
0.015250  1  

-1  0
8999999488  8999999488
8999999488  8999999488
8999999488  8999999488
8999999488
0.00  Current (A)
0.00  Voltage (V)
100  Velocity (mm/s)
0  Precision (mm)
0  Move type (0-MoveL, 1-MoveJ, 2-MoveC) [35]

Points 2 & 3 suppressed

Ponto 4
Fim
1

-288.989014  270.083008
372.670990  0.209150
0.673230  -0.678060
0.207970  0

-1  -1  0
8999999488  8999999488
8999999488  8999999488
8999999488  8999999488
8999999488
0.00  0.00
100  0
0
0

Fig. 13 – Simple 4-point definition file.

Section V – Conclusion and Future Work

In this paper a system designed to assist and simplify industrial welding procedures was presented and demonstrated. A brief overview of actual state of the art about robotic welding technology was presented, along with author’s point of views about near future evolution. A brief overview of MIG/MAG welding principles was also presented. In the process the most important limitations of actual robotic systems were summarized, showing how the presented system overcomes some of them. The software architecture used in the system was also briefly presented and
Two test cases were presented just to show the usefulness of the obtained system. In fact, a simpler version of the system is being explored by an industrial partner, which is also a good source of information namely on operational and practical problems. That allows us to refine the system in a way to handle those problems, which in most cases are related with Human Machine Interfaces.

For research, the system is currently being used in our laboratory to experiment various welding conditions and materials. The idea is to obtain an automatic procedure for welding parameter selection and on-line correction conducting to better quality welding (that is the subject of a research contract with the Portuguese Foundation for Science and Technology). For that purpose the system uses visual information that we expect to include in the very near future. Seam tracking is obtained just by using a laser 3D camera (M-Spot 90) connected to the PC using a proprietary interface.

References

[23] AGA AB, “Facts about rapid ArcTM. Rapid MeltTM. High-Productivity MIG/MAG welding”, AGA AB.


About the authors:

Corresponding author

J. Norberto Pires is Auxiliary Professor of the Mechanical Engineering Department of the University of Coimbra (Portugal). His background is on Physics Engineering (1991, University of Coimbra, Physics Department), with a Master degree in Technological Physics (1994, University of Coimbra, Physics Department) and a Ph.D. degree in Robotics and Automation (1999, University of Coimbra, Mechanical Engineering Department). He is editor-in-chief of the journal “Robótica”, the only Portuguese Journal on Robotics and Automation (RA), and organized several workshops and seminars on RA. His research includes force control, HMI, robotic welding systems and object oriented and distributed software for Flexible Manufacturing Cells.
E-mail: norberto@robotics.dem.uc.pt
Web: http://robotics.dem.uc.pt/norberto/

Altino Loureiro is Auxiliary Professor of the Mechanical Engineering Department of the University of Coimbra (Portugal). His background is on Mechanical Engineering with a Ph.D. degree in Mechanical Engineering (University of Coimbra, Mechanical Engineering Department). His research is since several years focused on welding problems, having an extensive experience in the subject namely with industrial cooperation, and recently became interested in robotic welding systems.
E-mail: altino.loureiro@dem.uc.pt
Web: http://www.dem.uc.pt/

P. Ferreira, T. Godinho, B. Fernando and J. Morgado are master students of the Mechanical Engineering 2000/2002 Master Program. Their collaboration in this subject was part of their project for the Master Discipline of “Industrial Robotics” http://robotics.dem.uc.pt/norberto/robind/
E-mail: pedro.ferreira@dem.uc.pt, t.godinho@robotics.dem.uc.pt, bruno@robotics.dem.uc.pt and joel@robotics.dem.uc.pt, respectively.