

Using Object-Oriented and Distributed Software to Assist Industrial Robotic Welding Applications



Welding Robots

By J. NORBERTO PIRES, ALTINO LOUREIRO, T. GODINHO
P. FERREIRA, B. FERNANDO, and J. MORGADO

Using robots in industrial welding operations is common but far from being a streamlined technological process. The problems are with the robots, still in their early design stages and difficult to use and program by regular operators; the welding process, which is complex and not really well known; and the human-machine interfaces, which are nonnatural and not really working. In this article, these problems are discussed, and a system designed with the double objective of serving R&D efforts on welding applications and to assist industrial partners working with welding setups is presented. The system is explained in some detail and demonstrated using two test cases that reproduce two situations common in industry: multilayer butt welding, used on big structures requiring very strong welds, and multipoint fillet welding, used, for example, on structural pieces in the construction industry.

Background

Actual market conditions are compatible with small/medium batch manufacturing, due to strong competition and dynamic market behavior. In these conditions, robotic production setups exhibit the best “cost per unit” performance if compared with manual work and hard automated setups (Figure 1) [1]. Consequently, in the near future, powerful and more flexible machines will be required 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. This means exposing the user to the flexibility stored (and barely used) inside the machines as a result of several decades of engineering.

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 from the programming point of view. This does not necessarily mean open source but,

instead, the availability of powerful APIs and de facto standards for both hardware and software, enabling access to system potentialities without limitations. This is particularly necessary in research environments, where good access to resources is needed to implement and test new ideas. If good access is available, 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, an open source would also be very difficult to achieve since those fields of robotics have decades of engineering efforts, achieving very good results and reliable machines, that are not easy to match. The open source issue is, nevertheless, very important for emerging robotics research (like humanoid robotics, space robotics, robots for medical use, etc) as a way to spread and accelerate development (Figure 2).

Industrial robotic welding is by far the most popular application of robotics worldwide [2]. In fact, a large number of products require welding operations in their assembly processes. The automobile industry is probably the most significant example, with the spot and metal-inert-gas/metal-active-gas (MIG/MAG) welding operations in automotive assembly lines. An increasing number of small, client oriented businesses manufacture small series or unique products designed for each client. Their users require a good and highly automated welding process to respond to client needs in time and with high quality. It is for these companies that the concepts of Agile Production [3], [4] apply the most, obviously supported by flexible manufacturing setups. Despite all this interest, industrial robotic welding has evolved only slightly and is far from being a trouble-free technological process, at least in a general way. The welding process is complex and difficult to parameterize and effectively monitor and control [5]–[7]. In fact, most welding techniques are not fully understood, namely the effects on the welding joints, and are used based on empirical models

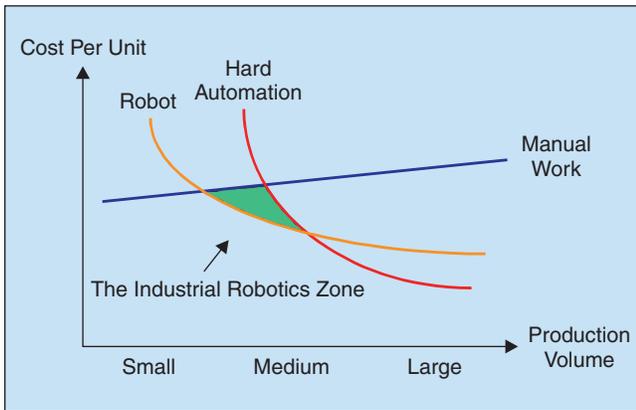


Figure 1. The industrial robotics zone [1].

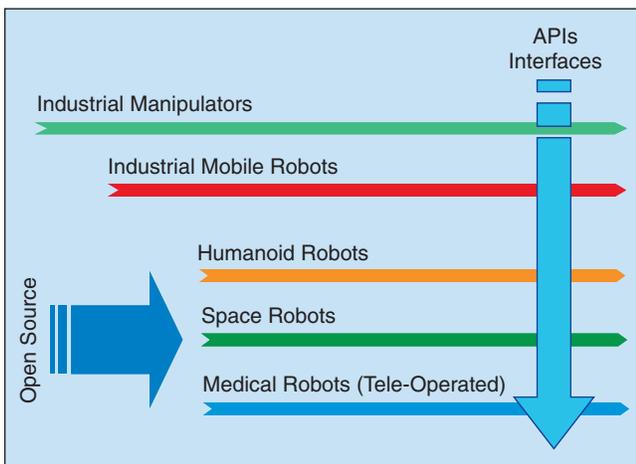


Figure 2. Traditional robotics and requirements for the new research.

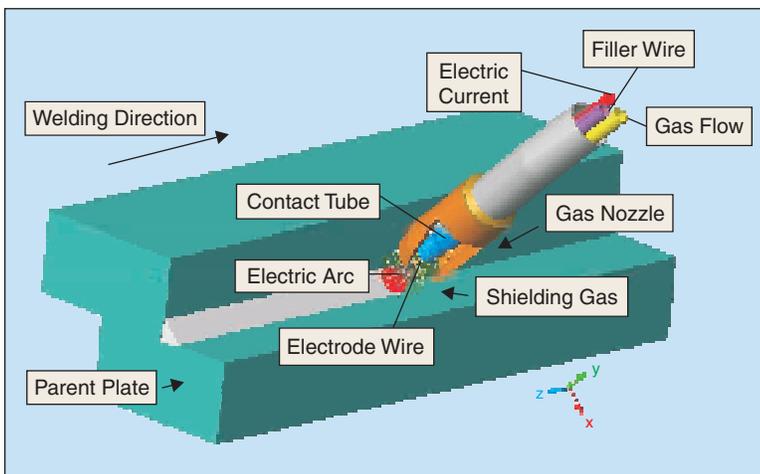


Figure 3. The MIG/MAG welding process principle.

obtained by experience under specific conditions. The effects of the welding process on welded surfaces are currently not fully known. Welding in most cases (i.e., MIG/MAG welding) imposes extremely high temperatures concentrated in small zones. Physically, that makes the material experience extremely high and localized thermal expansion and contraction cycles. This introduces changes in the material that may affect mechanical behavior along with plastic deformation [5], [7]. These changes must be well known in order to minimize the effects.

Using robots with welding tasks is not straightforward and has been the subject of various R&D efforts [8]–[12]. This is because the modern world produces a large variety of products that use welding to assemble some of their parts. If the percentage of welding connections incorporated in the product is large enough, 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 [13]. Nevertheless, when a robot is added to a welding setup, the problems increase in number and 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 [14]–[18].

In this article, some of these problems are addressed 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 the potential for industrial utilization. For this purpose, mainly industrial equipment was selected in designing the system as a way to facilitate its industrial exploitation. Finally, the article addresses aspects of system programming and welding parameterization.

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 pernicious contamination with atmospheric gases (oxygen, nitrogen, and hydrogen).

The stability of the welding process is sensitive to basic welding parameters, especially current, voltage, welding speed, stick-out (length of wire out of the contact tube), shielding gas, and arc length [19], [20]. 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 [21]. 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, MIG/MAG welding process databases have been developed, such as the one at the Welding Institute in the United Kingdom [22]. In these databases, the input data is generally the type of weld (butt or fillet), the welding position (flat, horizontal, vertical, or overhead), wire diameter, and 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 remain an input parameter.

It would be expected that, having a CAD model of the component to be welded, the system would be able to select the welding data for each weld and send their 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 in the case of welds with multiple layers. In this case, in fact, the position of the torch in each layer needs to be indicated to the robot.

For the majority of the companies that produce multilayer welds, there is only a small number of distinct welds, so it is not hard to fill up the database for their particular case.

With this method it is easy to carry out the offline programming of the components to be welded; it is only necessary to adjust the coordinates of the process points in the first specimen to be welded.

A Robotic Welding System

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 [2]. With welding applications, it is necessary to start from a trajectory, given, for example, from a CAD model of the working piece, and have the means to correct it in real time as a function of the observed results of the welding process. For that, systems for guidance and inspection, the possibility of real-time correction of 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 are needed [14]-[18], [23], [24]. These features aren't usually available due to the following reasons:

- ◆ Actual robot manipulators have closed controllers, not allowing real time position correction [14], [23], [24].
- ◆ Actual robot controllers do not allow remote control from an external computer.
- ◆ It is very difficult to attach guidance sensors with good performance because they are not compatible with robot controllers [5].

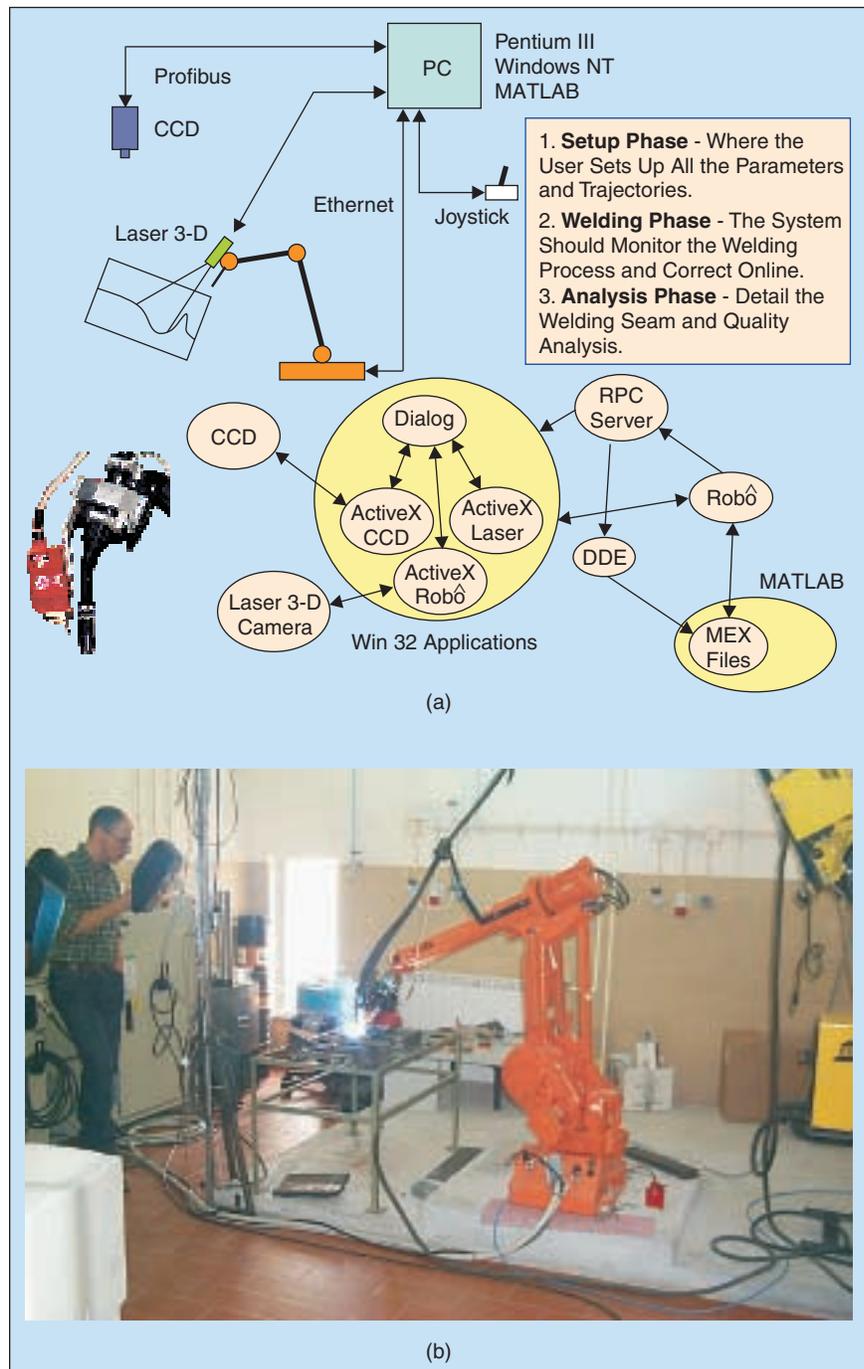


Figure 4. The welding system diagram and aspects of its utilization in the laboratory.

- ◆ Robot programming environments are not powerful enough to handle tasks requiring complex control techniques (learning, supervisory, adaptive, etc).

With the system proposed here, several of these limitations are reduced as follows:

- ◆ Use a robot control system that allows position correction commanded by a remote computer. This is not a standard feature but was added to the system by using features available from the controller [14]-[18].
- ◆ Use a distributed software architecture, based on personal computers running Win32 operating systems, that enable remote control using Ethernet networks [14]-[18].

- ◆ Attach the guidance sensors to the computer that controls the robot and not to the robot controller itself.
- ◆ Use a personal computer as programming environment, taking advantage of the huge amount of programming and analysis tools available on those platforms.

The proposed system (Figure 4) is composed by an industrial robot and its controller, a CCD digital video camera for inspection, a laser 3-D 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 database that describes the welding process.

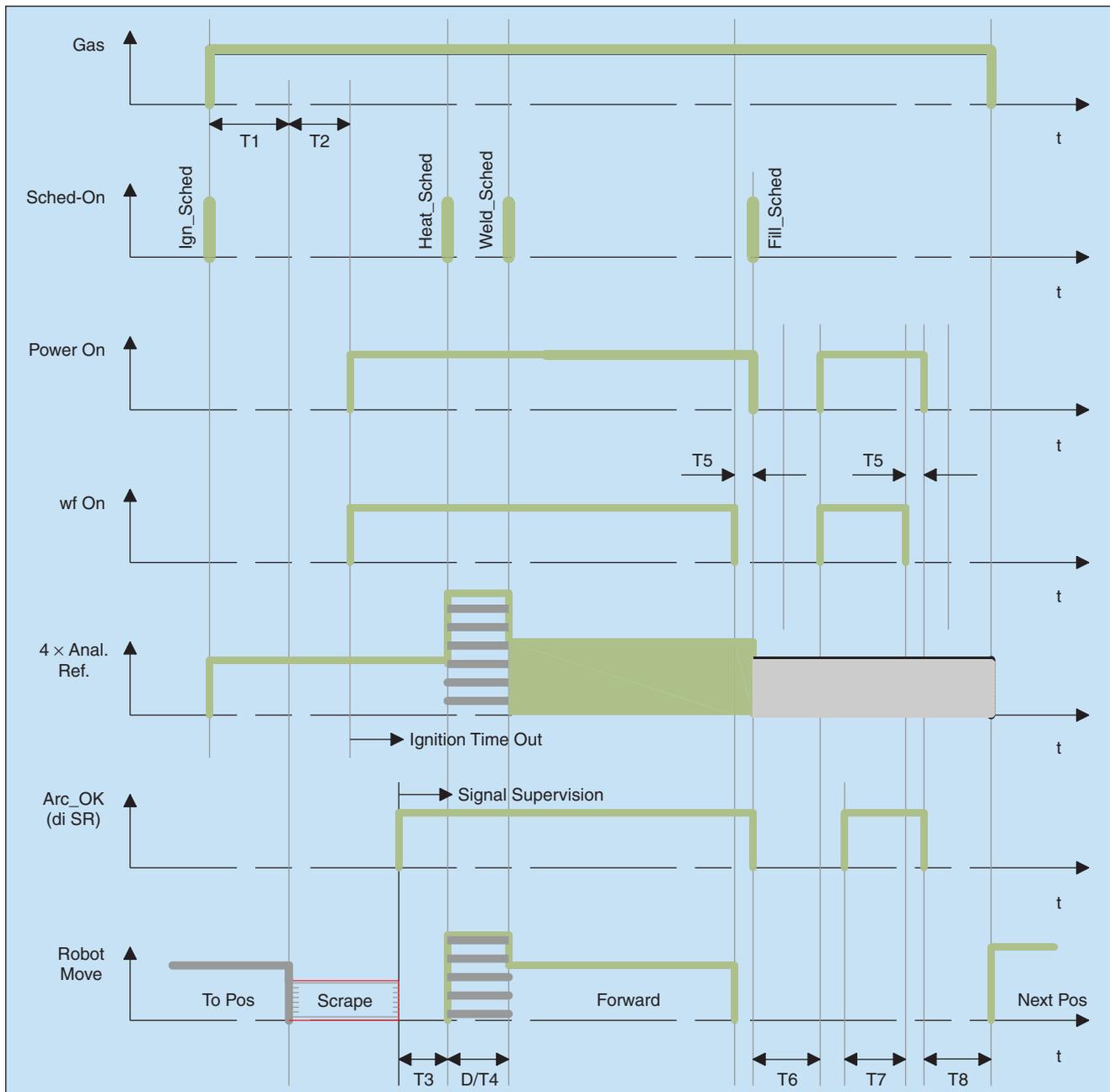


Figure 5. The welding sequence implemented by the robot controller (all the timings are programmable by the user).

Achieving automatic parameter selection also means using a sensing device not only for guiding the welding torch (joint tracking), but also for real-time acquisition of the welding geometry. Simple torch guidance and joint finding can be obtained using mechanical, electrical, or optical sensors, or current sensors to sense the arc current during torch weaving, laser beams, etc. Monitoring the welding geometry requires more sophisticated sensors, like laser 3-D cameras, available, for example, from Meta Systems in the United Kingdom [25] and Servo-Robot in Canada [26]. By joint geometry, we mean information about the gap, cross-section area, mismatch, and type of joint (fillet, corner, lap, V-groove, butt, etc). With that information and with the appropriate software and welding models, it is possible to perform several adaptations in real time that go beyond the simple joint finding and tracking tasks (here considered as background tasks). Weaving width, welding speed, wire feed speed, arc voltage, arc current, etc., are some examples of welding parameters that should be adapted if a better quality weld is required. In this article, we present a system able of incorporating the above mentioned adaptation capabilities just by using the appropriate sensor (a laser 3-D camera).

The software architecture used in this work, presented in detail elsewhere [14], [16], is distributed using a client-server model based on software components developed to handle equipment functionality. Currently, when we want to use some kind of equipment via 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 dependent), or build a software control using one of the several standard architectures available (preferably ActiveX or JAVA). But other technologies could be used; the focus here is on components and 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 that are accepted standards in industry, ActiveX is somehow privileged because it was specially built for those environments and is based on Microsoft's Distributed Component Object Model (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. Besides that, those components are easily integrated into new projects built with programming tools that can act as containers of that type of software control, i.e., they can be added to new projects in a "visual" way. We built

several ActiveX software components to use with this project. These controls expose to the programmer the basic functionalities of the equipment used (one ABB IRB1400 S4 industrial robot, a Siemens VS710 CCD camera, and an 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 Figure 5 and a client-server programming strategy [14]-[18].

The robot controller software works as a server (Figure 6), exposing the client to a collection of services that constitute its

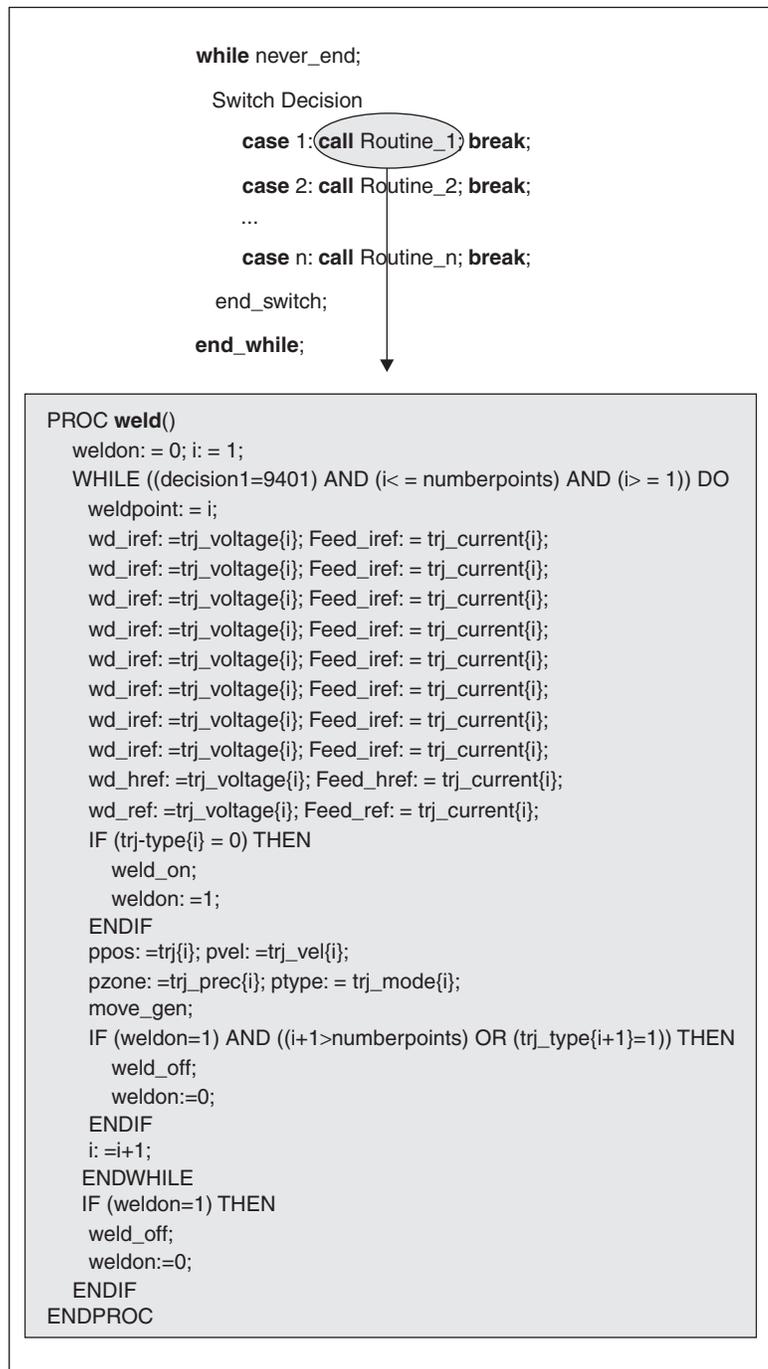


Figure 6. A robot working as a server.

Industrial robotic welding is by far the most popular application of robotics worldwide.

basic functionality. The robot can start the welding procedure or terminate it, and 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 the robot a complete definition of the welding task, including: points, welding parameters (velocity, voltage, and intensity), type of trajectory, and positioning precision, etc. All this information is stored in the robot controller and can be

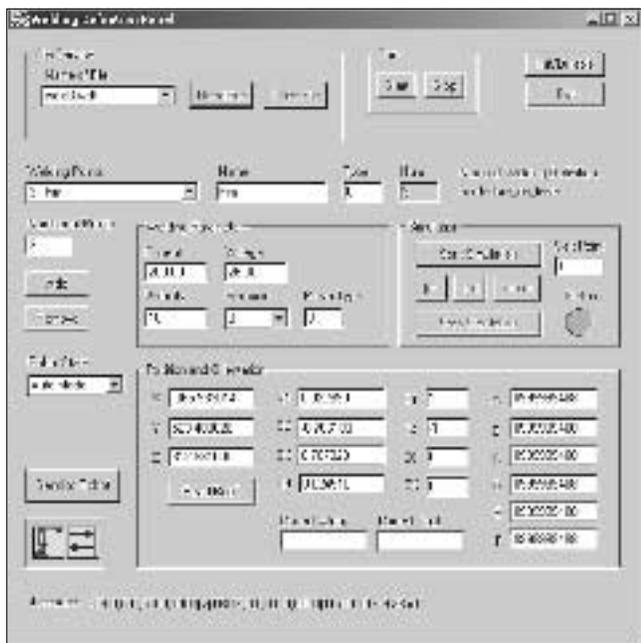


Figure 7. A shell of the WeldPanel tool.



Figure 8. A shell of the WeldAdjust tool.

used to simulate the welding process and enable any adjustment necessary or to start/stop the welding process.

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. The user 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 to pass over and weld. Then the user needs tools to adjust those points, add extra points, add approach and fly-away trajectories, adjust welding parameters, test, and simulate the entire process until the operation is as desired. For that, the following collection of tools was designed:

WeldPanel

With this tool (Figure 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 defined in the table holding the working object (Figure 7). The user may adjust points just by moving the robot to the desired position. This 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 input/output (I/O) signals, etc. The status of the program running on the robot controller and of the network connection is constantly monitored to avoid damaging materials and persons by preventing system commands on error situations. Events are remote procedure calls (RPCs) made by the robot controller to an RPC server running on the PC as a service [27]–[29].

WeldAdjust

This tool (Figure 8) is used to adjust points online 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.

File Explorer

With this tool (Figure 9), 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 Windows 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

This tool is used to change the robot controller state and to load and unload modules from the robot controller (Figure 10). The RPC server is used to receive events from the robot controller (Figure 10). As already mentioned, events are RPC calls [27]–[29] made by the robot controller and fired when preprogrammed ac-

Table 1. Position of the welding torch for each layer.									
Layer	1	2	3	4	5	6	7	8	9
X (mm)	0	0	0	-5	+5	-6	+6	-6	+6
Y (mm)	20	24	28	32	32	36	36	36	36

Table 2. Welding data extracted from a database [24].			
Layer	Current (A)	Voltage (V)	Speed (mm/s)
1	200	22	5
2	250	28	5
3-9	300	34	5

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

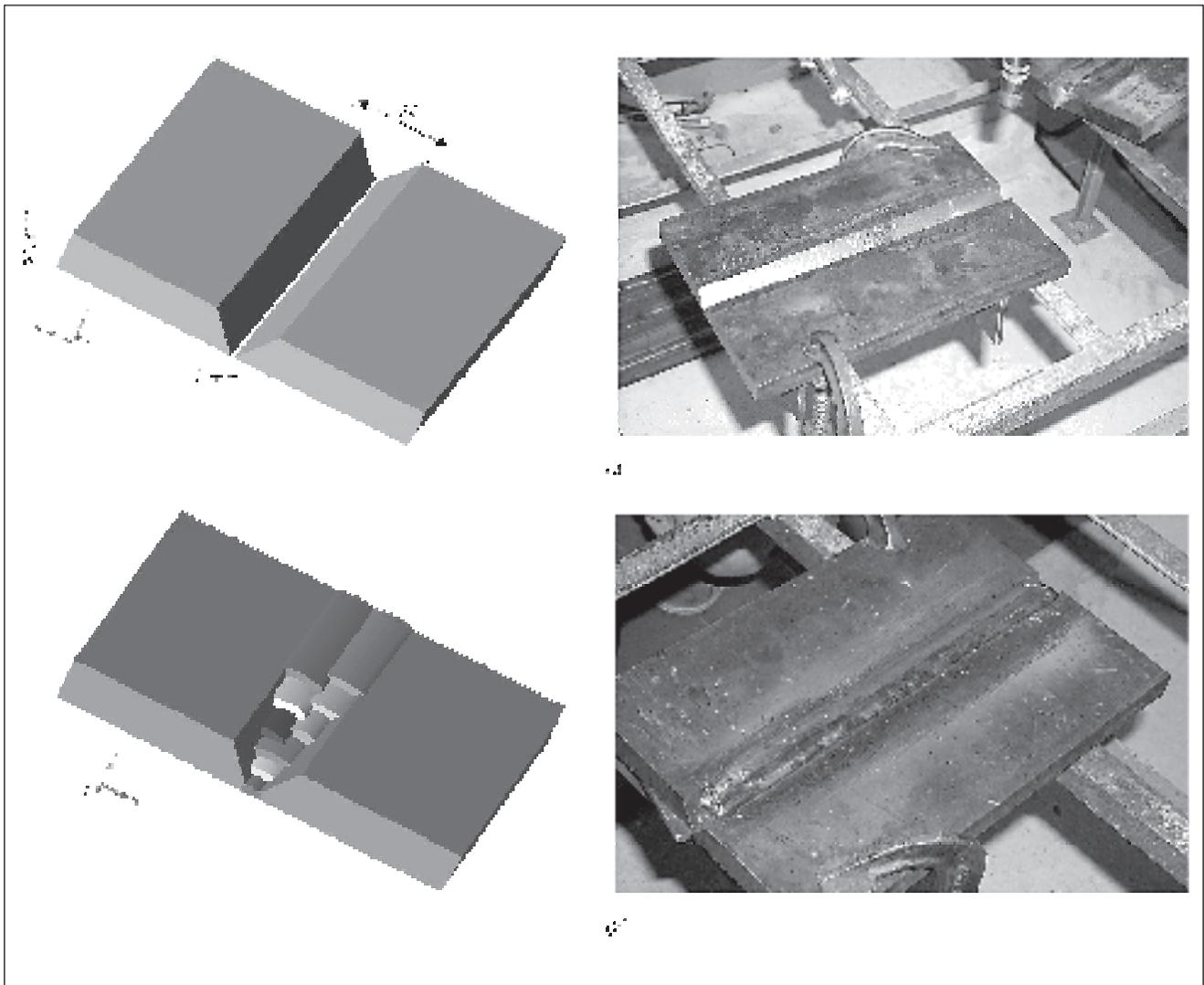


Figure 11. An aspect of the working object, welding sequence, and obtained weld. (a) Work Piece for multipass weld test case (two 20-mm-thick plates, 2 mm apart from each other, with a 60° V-groove joint preparation). (b) Layers necessary (welding sequence) to finish the weld and obtained weld.

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 Figure 13. Again, the welding parameters may be obtained from a database. The process can then be simulated

for trajectory and welding parameters adjustment and tested until desirable performance is achieved, including acceptable welding quality.

The online remote control of the weld quality is carried out using a vision system focused behind the weld torch. With

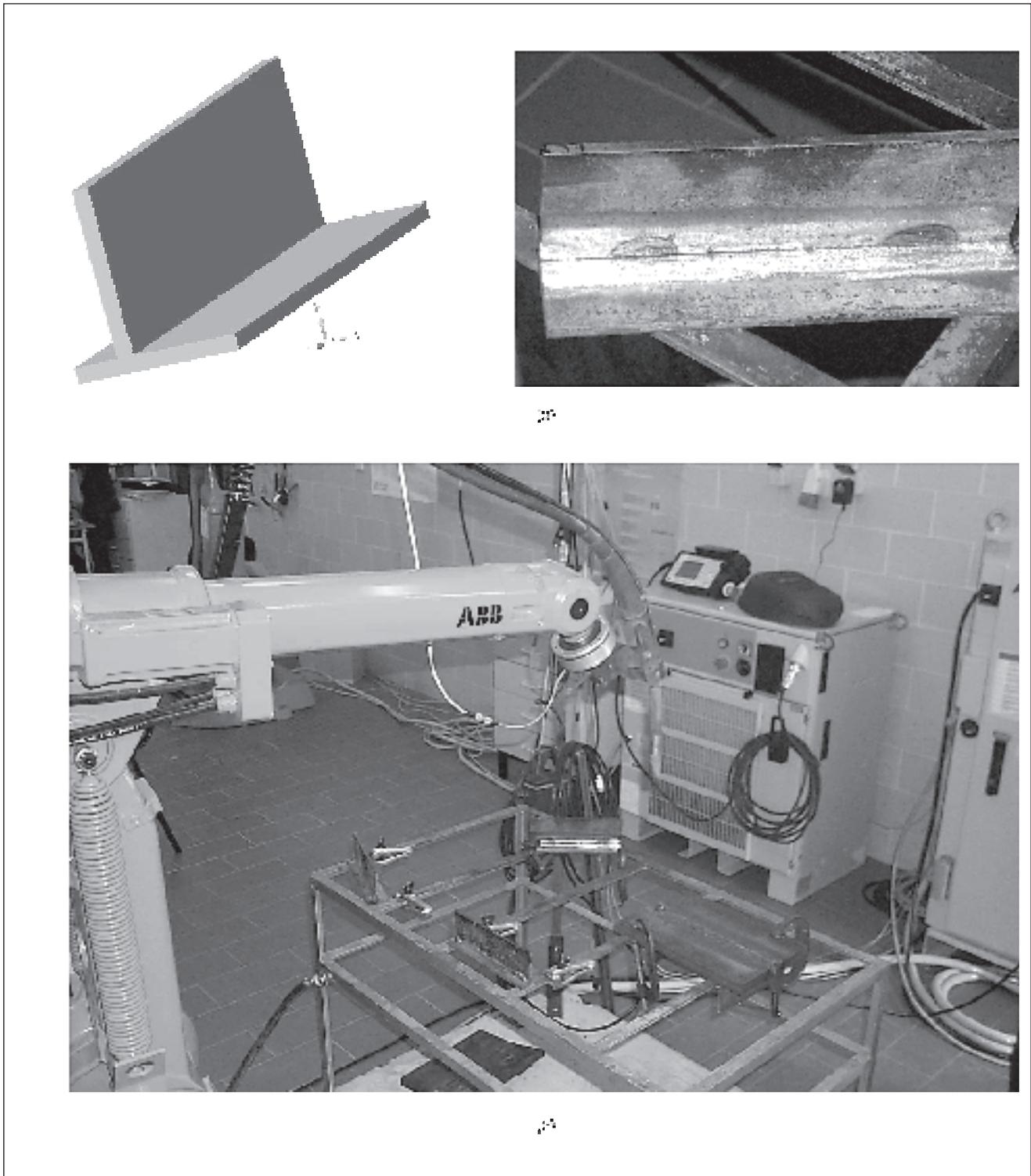


Figure 12. An aspect of the working pieces. (a) Fillet weld preparation. (b) Working table in the laboratory.

this method, only surface irregularities of the weld, such as undercuts, underfill, or variations in weld height, may be detected. Other defects, like porosity, inclusions, or underbead cracks, are not detectable using this technique.

The main advantages of the system are the remote control of the welding process and online remote control of the weld quality.

Summary

In this article, 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 points of view 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 referenced. Two test cases were presented just to show the usefulness of the obtained system.

Future Work

A simpler version of the system presented in this article is being explored by an industrial partner, which is also a good source of information, namely on operational and practical problems. This allows refining the system in a way to handle those problems that, in most cases, are related to 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 online 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 by using a laser 3-D camera connected to the PC using a proprietary interface.

Keywords

Industrial robotics, robotic welding, distributed software.

References

- [1] M. Myhr, "Industrial new trends: ABB view of the future," in *Proc. Int. Workshop Industrial Robotics, New Trends and Perspectives*, Lisbon, Portugal, 1999, pp. 59-87. [Online]. Available: <http://robotics.dem.uc.pt/ir99/>
- [2] United Nations and International Federation of Robots, *World Industrial Robots 1996: Statistics and Forecasts*, ONU/IFR, New York, 2000.
- [3] A. Kusiak, *Modelling and Design of Flexible Manufacturing Systems*. New York: Elsevier, 1986.
- [4] A. Kusiak, *Computational Intelligence*. New York: Wiley, 2000.

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)
-446.816010                     Robtarget structure [35]
315.087006
436.980011
0.017820
- 0.752200
0.658510
0.015250
1
-1
0
0
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
0.00
0.00
100
0
0

```

Figure 13. Simple four-point definition file.

- [5] G. Bolmsjö, "Process and quality monitoring and control," *J. Robótica*, vol. 42, no. 1, pp. 67-83, Jan. 2001.
- [6] G. Bolmsjö, M. Olsson, G. Nikoleris, and K. Brink, "Task programming of welding robots," in *Proc. Int. Conf. Joining Materials, JOM-7*, 1995, pp. 573-585.
- [7] A. Loureiro, M. Velindro, and F. Neves, "The influence of heat input and the torch weaving movement on orbotized MIG weld shape," *Int. J. Joining of Material*, vol. 10, no. 3/4, pp. 86-91, 1998.
- [8] B. Agren, "Sensor integration for robotic arc welding," Ph.D. dissertation, Lund University, Lund, Sweden, 1995.
- [9] R. W. Richardson, "Robotic weld joint tracking systems—Theory and implementation methods," *Welding J.*, vol. 65, no. 11, pp. 43-51, Nov. 1986.
- [10] A. Hirai, Y. Kaneko, T. Hosoda, S. Yamane, and K. Oshima, "Sensing and control of weld pool by fuzzy-neural network in robotic welding system," in *Proc. 27th Annu. Conf. IEEE Industrial Electronics Society*, 2001, vol. 1, pp. 238-242.
- [11] P. Drews and G. Starke, "Development approaches for advanced adaptive control in automated arc welding," Mechatronics Dept., Univ. of Aachen, Germany, Internal Report XII-970, 1986.
- [12] Y. Yada et al., "Development of real time welding control system by using image processing," Mechatronics Dept., University of Aachen, Germany, Internal Report XII-998, 1986.
- [13] P. Dahlén and G. Bolmsjö, "Human factors in the justification of advanced manufacturing systems," *Int. J. Human Factors in Manufacturing*, vol. 6, no. 2, pp. 147-162, 1996.
- [14] J.N. Pires and J.M.G. Sá da Costa, "Object oriented and distributed approach for programming robotic manufacturing cells," *IFAC J. Rob. Comput. Integrated Manufacturing*, vol. 16, no. 1, pp. 29-42, Mar. 2000.
- [15] J.N. Pires, P. Monteiro, and V. Schölkze, "Using robot manipulators on high efficient wrapping machines for paper industry," in *Proc. Int. Symp. Robotics*, Seoul, Korea, 2001.
- [16] J.N. Pires, "Using Matlab to interface industrial robotic & automation equipment," *IEEE Robot. Automat. Mag.*, vol. 7, pp. 32-41, Sept. 2000.
- [17] J.M.G. Sá da Costa and J.N. Pires, "Future welding robot developments," *J. Robótica*, no. 41, pp. 1-23, Jan. 2001.
- [18] J.N. Pires, "Programming industrial robotic and automation equipment," *Industrial Robot, Int. J.*, vol. 27, no. 4, pp. 279-287, July 2000.
- [19] S.K. Choi, C.D. Yoo, and Y.S. Kim, "Dynamic simulation of metal transfer in GMAW. Part 1: Globular and spray transfer mode welding," *Welding J.*, vol. 77, no. 1, pp. 38-44, 1998.
- [20] S.K. Choi, C.D. Yoo, and Y.S. Kim, "Dynamic simulation of metal transfer in GMAW. Part 2: Short circuit transfer mode," *Welding J.*, vol. 77, no. 1, pp. 45-51, 1998.
- [21] L.A. Jones, T.W. Edgar, and J.H. Lang, "Images of steel electrode in Ar-20% shielding during constant current gas metal arc welding," *Welding J.*, vol. 77, no. 1, pp. 135-141, 1998.
- [22] *Standard Data for Arc Welding*. Abington, UK: The Welding Inst., 1975.
- [23] S. Adolfsson, A. Bahrami, G. Bolmsjö, and I. Claesson, "On-line quality monitoring in short-circuit gas metal arc welding," *Welding J.*, vol. 75, pp. 59s-73s, Feb. 1999.
- [24] G. Bolmsjö, "Knowledge based systems in orbotized arc welding. Knowledge based systems—Advanced concepts," in *Techniques and Applications*, S. Tzafestas, Ed. Singapore: World Scientific, 1997, ch. 17, pp. 465-495.
- [25] Meta Systems, U.K. Available: <http://www.weta.co.uk>
- [26] Servo-Robot, Canada. Available: <http://www.servorobot.com>
- [27] *ABB IRB1400 Users Manual*. Sweden: ABB Flexible Automation, 1997.
- [28] *RAP; Service Protocol Definition*. Sweden: ABB Flexible Automation, 1996.
- [32] J. Bloomer, *Power Programming with RPC*. New York: O'Reilly & Associates, Inc., 1992.

J. Norberto Pires is an auxiliary professor of the Mechanical Engineering Department of the University of Coimbra (Portugal). His background is in physics engineering (1991, University of Coimbra, Physics Department), with an M.S. in technological physics (1994, University of Coimbra, Physics Department) and

a Ph.D. 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, human-machine interface, robotic welding systems, and object-oriented and distributed software for flexible manufacturing cells.

Altino Loureiro received his licentiate and Ph.D. from the University of Coimbra and the M.Sc. from Porto University, Portugal, in 1976, 1991, and 1987 respectively. He is also a welding engineer, and, between 1978-1983, was head of the Quality Control Department in a Portuguese steam boilers company. He joined the Department of Mechanical Engineering at the University of Coimbra, Portugal, in 1978. He has been a professor since 1991 and is currently conducting research in the area of welding technology and automation of welding processes.

P. Ferreira is currently an assistant lecturer at the Mechanical Engineering Department of the Coimbra Polytechnic Institute. He is a student in the University of Coimbra's Mechanical Engineering 2000/2002 Master Program. His background is in mechanical engineering (University of Coimbra), and his interests are mainly in robotic welding and industrial robotics applications.

T. Godinho is currently an application engineer at San Jose SA, a Spanish company operating in Portugal. He is a master student at the University of Coimbra's Mechanical Engineering 2000/2002 Master Program. His background is in mechatronics (Transportation and Communications Institute, Entrocamento, Portugal), and his interests are mainly on force control, industrial robotics applications, and distributed software.

B. Fernando is a student in the University of Coimbra's Mechanical Engineering 2000/2002 Master Program. His background is in mechanical engineering (University of Coimbra), and his interests are mainly in robotic welding distributed software applied to industrial robotics.

J. Morgado is currently a research assistant with the Mechanical Engineering Department of the University of Coimbra. He is student in the University of Coimbra's Mechanical Engineering 2000/2002 Master Program. His background is in mechanical engineering (University of Coimbra), and his interests are industrial robotics applications.

Note: The master students collaborated in the research presented in this article as part of their project for the Master Discipline of "Industrial Robotics" (<http://robotics.dem.uc.pt/norberto/robind/>).

Address for Correspondence: J. Norberto Pires, Mechanical Engineering Department, University of Coimbra, Polo II Campus, P-3030 Coimbra, Portugal. E-mail: norberto@robotics.dem.uc.pt. URL: <http://robotics.dem.uc.pt/norberto/>.