6th International Conference on Trends in Welding Research
15-19 April 2002
Pine Mountain, GA USA
ASM International

Intelligent Control of Modular Robotic Welding Cell

Herschel B. Smartt, Kevin L. Kenney, Charles R. Tolle
Idaho National Engineering and Environmental Laboratory
P.O. Box 1625
Idaho Falls, ID 83415-2210

Abstract

Although robotic machines are routinely used for welding, such machines do not normally incorporate intelligent capabilities. We are studying the general problem of formulating usable levels of intelligence into welding machines. From our perspective, an intelligent machine should: incorporate knowledge of the welding process, know if the process is operating correctly, know if the weld it is making is good or bad, have the ability to learn from its experience to perform welds, and be able to optimize its own performance. To this end, we are researching machine architecture, methods of knowledge representation, decision making and conflict resolution algorithms, methods of learning and optimization, human/machine interfaces, and various sensors.

This paper presents work on the machine architecture and the human/machine interface specifically for a robotic, gas metal arc welding cell. Although the machine control problem is normally approached from the perspective of having a central body of control in the machine, we present a design using distributed agents. This new design is loosely based on biological models of social insects. For example, in an ant colony each ant functions according to local rules of behavior [Hölldobler and Wilson, 1990, see chapters 8 and 9]. There is no "king" or "queen", although the latter name has been given to the reproducing ant. Following a similar approach, we present a modular machine architecture in which each machine element has local rules of behavior but no single element understands how to make a weld. A prime goal of this work is to develop an architecture for an intelligent machine that will support a modular, plug and play standard. A secondary goal of this work is to formulate a human/machine interface that treats the human as an active agent in the modular structure.

Introduction

James Albus [1991] at NIST has defined machine intelligence as "the ability of a system to act appropriately in an uncertain environment, where appropriate action is that which increases the probability of success, and success is the achievement of behavioral subgoals that supports the system's ultimate goal." Following Albus' intent, we can say that intelligent machines are those that either know or can learn everything they need to know to perform a process or task. Such machines may be able to perform a process or task autonomously (without operator intervention) or semi-autonomously (with operator intervention).

In this paper, we present a modular design of a machine using distributed knowledge represented as local rules of behavior. In the design there is no central knowledge of how to perform a weld. Our approach is inspired by biological models of social insects. Related work has been conducted, for example, by Dorigo and Colorni [1996] using ant-based local behavior of multiple agents to solve the Travelling Salesman Problem and other classical hard problems. Schatz et al. [1999] formulated a model for route learning in ants. Lambrinos et al. [2000] used a similar model for navigation of a mobile robot. Overgaard, Petersen, and Perram [1995, 1996] used local agent control of dynamic motion and path planning in multiple link robot arms.

Consider an intelligent machine in which various machine functions are carried out in a distributed manner. A schematic of such a machine for arc spot welding is shown in Figure 1. In addition to the machine hardware required (most of which is not shown) there are several "agents", see Figure 2. These agents have local control of various machine functions and are able to communicate with each other and with an operator agent. The operator agent may be a human or may be an interface to a human (or even an interface to another machine). (Although it would be possible to focus on autonomous machines, we chose not to do so; our machines interact with humans who have supervisory control authority.) The various agents incorporate knowledge of how to perform their local tasks, although there is no single agent that has knowledge of the entire process. The agents communicate via a

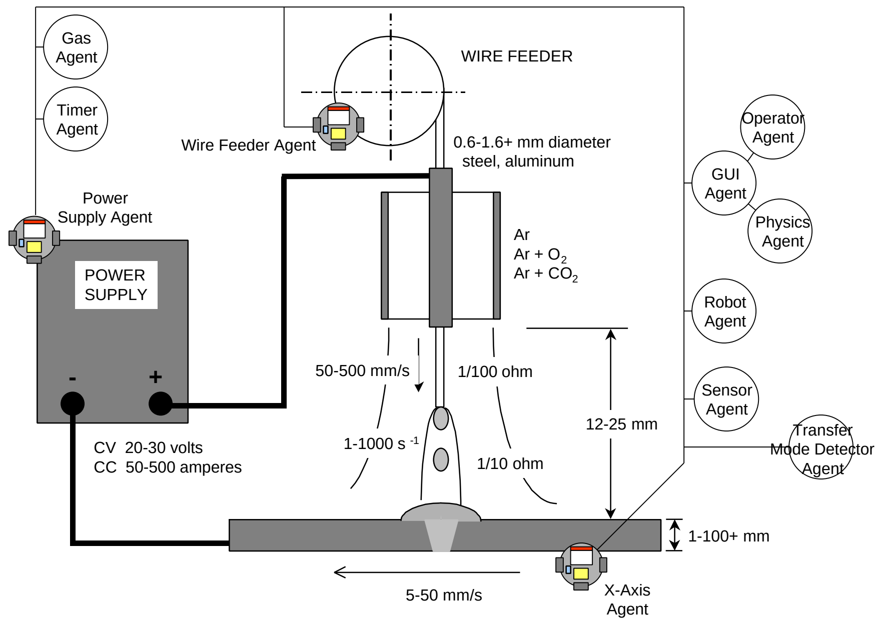

Figure 1. Gas Metal arc spot welding machine with various agents including power supply, electrode wire feeder, robot, sensor, metal transfer mode detector, process physics, operator, and user interface.

bus using a protocol similar to TCP/IP and a vocabulary incorporating both generic and process specific words. Each agent is dedicated to a specific machine hardware element or function. The individual agents incorporate specific knowledge expressed as fuzzy logic rules of behavior. The total machine is modular in that individual machine elements or functions may be removed or replaced with other elements or functions. The total machine configuration is defined in an external data file that is downloaded to the machine agents at run time. This allows the machine configuration to be changed without modifying the source code.

To formulate such a machine, we need a variety of methods. In addition to distributed learning and

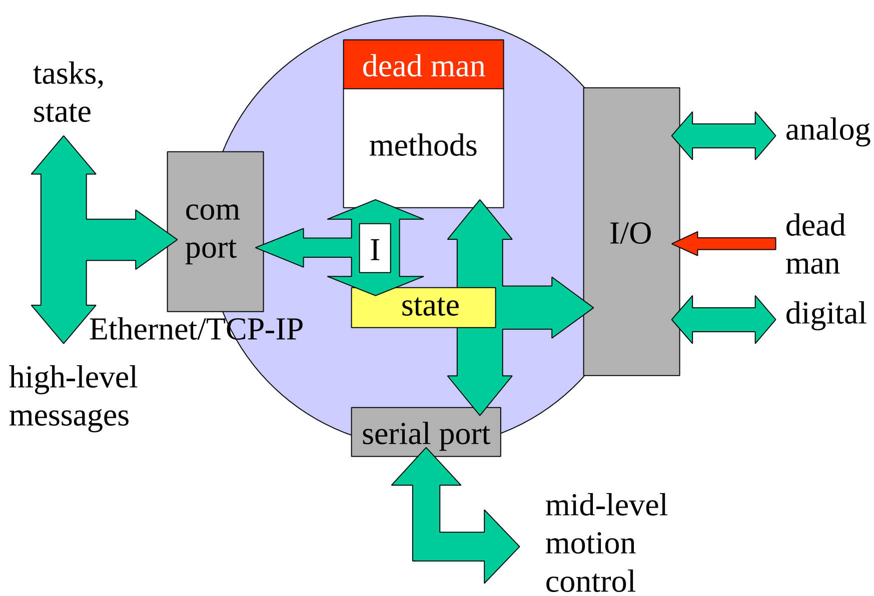

Figure 2. Agent block diagram.

control, we also chose to have our machines learn rules of behavior. This is distinct from learning control trajectories, a method frequently employed for machine learning. Our rules are embodied using a variant of fuzzy logic [Johnson and Smartt, 1995] that allows the system to learn by back propagation [Rumelhart, 1986].

Welding Application

Consider a specific welding control problem. We desire to fabricate a spot welded steel structure using gas metal arc welding (GMAW), Figure 1. Thus, steel sheet will be welded to an underlying structure by means of weld nuggets deposited into circular holes in the sheet.

In this situation, the weld torch may be moved to a suitable position over a weld site, using motion control as discussed earlier. The welding power supply contactor is activated, the power supply voltage is set, the shielding gas is turned on, and the electrode wire is fed downward. This will result in ignition of an arc with corresponding heat and mass transfer to the weldment. After a suitable time, the power supply contactor is deactivated and the electrode wire feed is stopped. A short time later the shielding gas is turned off. Although this is perhaps the simplest arc welding example we can consider, there are still important control decisions that ensure that the weld will meet its acceptance requirements

To obtain a good weld in this example, the current must be high enough but not too high and the weld time (the time the arc is on) must be equal to or greater than some critical minimum. This will ensure that adequate heat and mass have been transferred to the weldment. It is also necessary for the voltage to be above some minimum (to reduce spatter) and below some maximum (to avoid melt through and burn back).

Weld procedures are normally developed by a weld engineer based on the requirements identified for a "good" weld. A good weld in this spot weld application is one that is strong enough, does not excessively over or under fill the hole, has minimal spatter, and does not contain gross defects such as cracks or porosity that could lead to failure. To be strong enough, the weld bead must adequately penetrate the lower structure (but not excessively melt through that structure) while fusing into the upper sheet. For most applications, the cross-sectional area of the weld bead in the plane of the interface between the upper sheet and lower structure needs to be equal to or greater than some critical amount. The independent weld variables that are specified in the weld procedure include arc voltage, wire feed speed and weld time. The appropriate settings for these variables are based on several key characteristics of the weld process.

First, the heat transferred to the base metal directly from the arc and molten metal droplets is often a key consideration in procedure development. The weld heat input must be sufficient to provide the penetration and weld bead interface area required for joint strength. However, when joining some materials (e.g., advanced high strength steels) the heat input must be limited to minimize the metallurgical degradation associated with the high peak temperatures and slow cooling rates experienced during welding. The relationship between heat input and the welding process variables is given by [Rosenthal, 1946]

$$H = EIth \qquad (1)$$

where H is heat input, E is arc voltage, I is arc current, t is arc-on time and η is heat transfer efficiency from the process to the base metal. Arc voltage and time are independent variables selected by the weld engineer. Arc current, on the other hand, is a dependent variable that is function of the independent variables of wire feed speed, electrode diameter and electrode stick-out in. This relationship is given by [Smartt & Einerson, 1993]

$$I = K_0 + K_1 S + K_2 E + K_3 (CT) \qquad \mathbf{(2)}$$

where S is wire feed speed, E is arc voltage, CT is contact tip to workpiece distance and K_0, K_1, K_2 and K_3 are constants. For constant-voltage gas metal arc welding at a given CT, arc current is directly controlled by the wire feed speed.

Second, the weld nugget volume (or mass input to the weld) is an important characteristic of a good weld. The relationship between mass input (G) and the weld variables is given by

$$G = \left(\frac{pd^2}{4}\right) St \qquad (3)$$

where d is the electrode diameter, S is the wire feed speed and t is the arc-on time. For a given electrode, the values for WFS and time are selected to provide adequate fill.

A third consideration for weld procedure development is arc length. If the arc length is too short the electrode will stub into the weld pool, producing spatter which is undesirable for aesthetic reasons as well as for the potential to interfere with mating or adjacent parts. If the arc length is too long the electrode will burn back into the contact tube, the arc transfers to the contact tube, and the process becomes uncontrollable. Arc length is also a function of the weld variables and is given by [Reutzel, 1996]

$$L = C_0 E + \frac{C_1}{I} + C_2 I \tag{4}$$

where L is arc length, E is arc voltage, I is arc current and C0, C_1 and C_2 are constants.

An examination of Equations 1-4 shows heat input, mass input and arc lengths are all functions of the independent weld variables and are thus all inter-related. A change in wire feed speed affects heat input, masses input and arc length. A change in weld time affects both heat and mass input, and changes in arc voltage affect both heat input and arc length. This means that independent adjustment of heat input, mass input or arc length requires combined changes of wire feed speed and weld time in some unique ratio. For example, increasing the mass input heat input can be accomplished by increasing wire feed speed or time (it is impractical to change electrode size). However, either action also increases the heat input. Consequently, if wire feed speed is increased than the weld time must be decreased, or visa versa. Thus it is difficult to control heat and mass inputs to the weld independently. However, such independence can be obtained by solving Equations 1-4 appropriately. First, Equation 2 must be defined explicitly. An empirically derived relationship was obtained assuming a constant contact tip workpiece distance (CT). This expression for arc current (I) is given by

$$I = 0.97S + 3.25V + 5.84 \tag{5}$$

Equations 1, 3 and 5 may be solved iteratively to obtain values for wire feed speed (S) and weld time for a given heat input, mass input and voltage.

This procedure is the basis for the methodology employed by the intelligent robotic weld system for providing the human operator with advanced tools for developing the weld procedure. After all, since the values of wire feed speed, voltage and weld time are all based on the requirements of heat input, mass input and arc length, why not present the operator with controls to select the latter parameters directly. As such, the virtual weld panel of the robotic welding cell contains controls for heat input and mass input, and, since arc length varies linearly with arc voltage, a control for arc voltage is also provided. Using this methodology, the operator can dial in a heat-input value, a fill value and a voltage value directly.

It is interesting how the system adjusts the wire feed speed and weld time with independent changes of heat input, mass input or voltage. For example, an increase in heat input is accomplished by decreasing the wire feed speed while increasing the weld time to maintain a constant mass input. An increase of the mass input is accomplished by increasing the wire feed speed while decreasing the weld time to maintain a constant heat input with additional fill. When voltage is increased, wire feed speed increases and time decreases so that both heat input and fill remain unchanged as the voltage is increased.

In addition to the weld process variables, another important process characteristic is metal transfer mode. The characteristics of the constant-voltage GMAW process are best described in terms of the size and frequency of metal droplets transferred from the electrode to the work piece. The metal transfer mode is known to affect weld deposition rate, penetration, and spatter. Metal transfer mode is an important factor in out-of-position welding, and its effect on weld penetration makes it important for thin section welding where burn-through is a concern. Three distinct metal transfer modes have been identified for gas metal arc welding: spray, globular and short-circuiting (also referred to as short-arc) transfer. Spray transfer mode is characterized by consistency, good penetration, and a low degree of spatter. For applications where maximum penetration and/or minimal spatter are desired, spray transfer is often preferred. Short-arc is often preferred for welding thin sections or for out-of position welding due to the lower current levels (lower heat input) offered by the process. In addition, spray transfer may not be achievable with higher concentrations of CO2 in the shielding gas. Since metal transfer mode is an important feature, a weld mode identifier was considered an integral component of an intelligent welding system.

This task of identifying the metal transfer mode is dedicated to one of the independent agents of the intelligent weld system. This weld mode analysis agent performs a fuzzy logic based analysis of the acquired current and voltage signals [Smartt, et.al. 2001], and is referred to as a Fuzzy Logic Weld Mode Identifier (FLWMI) agent. The FLWMI agent performs a fuzzy logic comparison of the average current (I_{avg}), the minimum value of current (I_{min}), the standard deviation of current (I_{sd}), and the standard deviation of voltage (V_{sd}) and, based on this comparison, identifies the metal transfer mode as spray, globular, short-circuiting, or shorting streaming. The Fuzzy Logic Rules from which the diagnostic algorithm is based are as follows:

- If I_{sd} is low, then mode is spray
- If I_{sd} is high AND V_{sd} is high AND I_{min} is high, then mode is globular
- If I_{sd} is high AND V_{sd} is high AND I_{min} is low AND I_{avg} is low, then mode is short-circuiting
- If I_{sd} is high AND V_{sd} is high AND I_{min} is low AND I_{avg} is high, then mode is shorting streaming.

As shown in Figure 1, the robotic gas metal arc welding cell is composed of the following independent agents:

- Analog Input Agent
- Power Supply Agent
- Wire Feeder Agent
- Timer Agent
- Gas Agent
- Robot Agent
- GUI Agent
- FLWMI (Fuzzy Logic Weld Mode Identifier) Agent

Each agent is assigned a specific, independent task, requiring the agents to work together to accomplish the global objective of making a spot weld. In the process of performing a spot weld, the operator specifies the maximum allowable weld time, the mass input, the heat input and the arc voltage for the spot weld. The Physics Agent, which is embedded in the GUI Agent, then iteratively solves Equations 1, 3, and 5 for the appropriate wire feed speed setting and the weld time setting, as discussed in detail above. The Wire Feeder, Power Supply and Timer agents then retrieve their appropriate settings from the GUI Agent. Before the weld process can proceed, the operator must Arm each agent. If an agent is not armed, it will not perform any task beyond initialization. Once armed, the agents are able to perform their tasks associated with the spot welding process. The weld process is initiated by the operator and the weld operation proceeds in the sequence that follows.

1. Operator presses “Weld” button
   - Message is sent to Power Supply Agent to “Weld”
2. Power Supply Agent
   - Receives message to “Weld”
   - Check status of Deadman Switch
   - If Deadman is open
   - Then abort sequence and notify operator
   - If Deadman is closed
   - Then
     - Enable power supply contactor
     - Send message to AI Agent to “Weld” (goto 3)
     - Send message to Gas Agent to “Weld” (goto 4)
     - Send message to WF Agent to “Weld” (goto 5)
3. AI Agent
   - Receives message to “Weld”
   - Start data acquisition
4. GAS Agent
   - Receives message to “Weld”
   - Turn gas on
5. WF Agent
   - Receives message to “Weld”
   - Enable wire feeder
   - Send message to Timer Agent to Weld (goto 6)
6. Timer Agent
   - Receives message to “Weld”
   - Reset Timer
   - Monitor weld time. When weld time expires, send message to Wire Feeder Agent to “End Weld” (goto 7)
7. Wire Feeder Agent
   - Receives message to “End Weld”
   - Disable wire feeder
   - Send message to Power Supply Agent to “End Weld” (goto 8)
   - Send message to AI Agent to “End Weld”, also send time to AI Agent (goto 9)
8. Power Supply Agent
   - Receives message to “End Weld”
   - Disable Contactor
   - Send message to Gas Agent to End Weld (goto 10)
   - Send message to Timer to End Weld (goto 11)
9. AI Agent
   - Receives message to “End Weld”
   - Read waveforms
   - Compute statistics
   - Send waveforms and statistics to GUI Agent
   - Send waveforms to FLWMI Agent (goto 13)
10. Gas Agent
    - Receives message to “End Weld”
    - Turn gas off
    - Send message to Robot Agent to End Weld (goto 14)
11. Timer Agent
    - Receives message to “End Weld”
    - Request time from WF Agent

– Compares desired weld time to actual weld time, compute time offset for next weld

12. FLWMI Agent
    – Receives message to “End Weld”
    – Receives waveform data
    – Executes analysis algorithm
    – Sends weld mode to GUI Agent
13. CRS Agent
    – Receives message to “End Weld”
    – Move torch to new position

During the welding sequence, each agent independently monitors the status of the Deadman Switch. If the switch opens up before the Timer Agent initiates the “End Weld” sequence, then each agent terminates the weld sequence.

Conclusion

An approach to design of an intelligent machine has been presented based on distributed intelligence. Local agents are used to control individual machine functions and to process information needed by the machine functions. Examples of how this approach may be used to build a specific machine are presented for an arc spot welding application. A possible agent internal structure is presented that provides for local rules of behavior and safety considerations. Additional research on this approach is presented in [Smartt, et. al., 2000].

Acknowledgement

This work is supported by the Basic Energy Sciences, Office of Science, U.S. Department of Energy, under DOE Idaho Operations Office Contract DE-AC07-99ID13727.

References

Albus, James. S., “Outline for a Theory of Intelligence,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol 21, no. 3, pp. 473-509, May/June (1991).

Dorigo, Marco; Maniezzo, Vittorio; Colorni, Alberto. “Ant System: Optimization by a Colony of Cooperating Agents,” IEEE Transctions on Systems, Man, and Cybernetics-PartB: Cybernetics, v 26, n 1, pp. 29-41, Feb. 1, (1996).

Hölldobler, B. and Wilson, E. O., *the ANTS*, The Belknap Press of Harvard University Press, Cambridge, Mass., (1990).

Johnson, J. A., and H. B. Smartt, "Advantages of an Alternative Form of Fuzzy Logic," IEEE Transactions on Fuzzy Logic, 3, pp. 149-157, (1995).

Lambrinos, D., Möller, R., Labhart, T., Pfeifer, R., and Wehner, R., “A mobile robot employing insect strategies for navigation”, Robotics and Autonomous Systems, 30, pp. 39-64, (2000).

Overgaard, L., Petersen, H. G., and Perram, J. W., “A General Algorithm for Dynamic Control of Multilink Robots”, The International Journal of Robotics Research, Vol. 14, No. 3, pp. 281-294, June (1995).

Overgaard, L., Petersen, H. G., and Perram, J. W., “Reactive Motion Planning: a Multiagent Approach”, Applied Artificial Intelligence, 10:35-51, (1996).

Reutzel, E. W., Personal communication (1996).

Rosenthal, D., The theory of moving sources of heat and its application to metal treatments, Trans. ASME, pp. 849-866, (1946).

Rumelhart, D. E., G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, Vol. 1, ed. D. E. Rumelhart and J. L. McClelland, MIT Press, Cambridge, MA, pp. 318-362 (1986).

Schatz, B., Chameron, S., Beugnon, G., and Collett, T. S., “The use of path integration to guide route learning in ants”, Nature, Vol. 399, No., 24, pp. 769-772, June (1999).

Smartt, H. B., and Einerson, C. J., “A model for heat and mass input control in GMAW”, Welding Journal 72(5):217-s to 229-s (1993).

Smartt, H. B., Kenney, K. L., Johnson, J. A., Carlson, N. M., Clark, D. E., and Reutzel, E. W., “Method and Apparatus for Assessing Weld Quality”, United States Patent 6,236,017, May 22, 2001.

Smartt, H. B., Tolle, C. R., and Kenney, K. L., “Complex Intelligent Machines”, Proceedings, Eighteenth Symposium on Energy Engineering Sciences, DOE-BES, Argonne National Laboratory, May 15-16, (2000).