Sensor guided robot welding in shipbuilding

Gunnar S. Bolmsjö and Mikael Fridenfalk

Division of Robotics, Dept. of Mechanical Engineering
Lund University, Box 118
SE-221 00 Lund, Sweden

Abstract

This paper presents the design and validation of a novel and universal 6D seam tracking system that reduces the need of accurate robot trajectory programming. Such sensor driven motion control together with adaptive control of the welding process is the foundation for increased flexibility and autonomous behavior of robotic and manufacturing systems. The system is able to follow any 3D spline seam in space with moderate radius of curvature by the real-time correction of the position and orientation of the welding torch, using the through-arc sensing method.

The 6D seam tracking system was developed from results of the ROWER-2 aiming at advancing the automatic welding and assembly of ship sections. The validation experiments showed that this system was both robust and reliable and is able to manage curvatures and abrupt changes in the weld joint in a stable way.

1 Introduction

A challenging application for sensor guidance intelligent systems is seam tracking in robotic welding. Seam tracking is among others used for the manufacturing of large ships such as cruisers and oil tankers, where relatively high tolerances in the sheet material are allowed to minimize the manufacturing costs [3, 12]. Seam tracking is today typically only performed in 2D, by a constant motion in $x$ and compensation of the errors in $y$ and $z$ directions, see Fig. 1. There exist solutions for higher autonomy that include seam tracking in 3D where compensation is also performed in $o$ direction or around an orientation axis. These limitations make however seam tracking only possible for workpieces with relatively simple geometrical shapes.

The next step in the evolution of sensor systems in robotic welding is the introduction of a full 6D sensor guided control system for seam tracking that is able to correct the TCP in $x$, $y$, $z$ and around roll, pitch and yaw. Such ultimate system is per definition able to follow any continuous 3D seam with moderate curvature. This system has many similarities with an airplane control system, where a change in either position or orientation effects all other degrees of freedom.

Though seam tracking has been performed in 2D for at least 40 years, the hypothetical design of 6D seam tracking systems was probably first addressed in the beginning of the last decade by suggestions of models based on force sensors [5, 6] and laser
Figure 1: Left: Definition of Tool Center Point (TCP) and the orthonormal coordinate system \( \{n, o, a\} \). Weaving, if any, is performed in \( n \) direction in arc sensing, \( o \) is opposite to the direction of welding and perpendicular to the plane \( \Omega \), and \( a \) is the direction of approach. \( \{x, y, z\} \) is a local coordinate system, defined for the workpiece. Right: The optimal position for seam tracking in arc sensing.

scanners [17, 18]. It is however difficult to evaluate these systems, since no experimental results are presented, neither any explicit control systems for creating balance between the subsystems.

The major contribution of this paper is the invention of a universal 6D seam tracking system for robotic welding, validated by simulation experiments based on physical experiments, that proved that 6D seam tracking is possible and even very robust for arc sensing [13]. On a more detailed level, the general contributions of this paper is the introduction of: (1) the concept of 6D seam tracking based on arc sensing, (2) the trajectory tangent vector in arc sensing, (3) pitch control in arc sensing, and (4) new method replacing quaternion interpolation and thereby increasing the computational efficiency significantly.

The authors have found no references from the literature that describe 6D seam tracking based on arc sensing, but there exist one similarity between this work and previous work, which is the introduction of a trajectory tangent vector by curve fitting that was made in laser scanning [17, 18]. There exist however some differences between how such vector was used and implemented. The differences consist of (1) curve fitting by 2nd instead of 3rd degree polynomials, for faster computation and still high control stability, (2) an analytic solver for curve fitting of 2nd degree polynomials developed and implemented for this system, increasing the calculation speed further and (3) using differential vector interpolation instead of direct use of the trajectory tangent vector, which showed to be essential for maintaining control stability.

A 6D seam tracking system increases the intelligence and flexibility in manufacturing systems based on robotic welding using laser scanning. This reduces the need for accurate robot trajectory programming and geometrical databases. The simulation results (based on physical experiments) showed that the initial objective to design a 6D seam tracking system was reached that could manage a radius of curvature down to 200 mm and a robust tracking during stepwise changes in the weld joint.
2 Materials and methods

2.1 Sensors guided robot control

Many systems have been developed during the last decades for seam tracking at arc welding. The patents and scientific publications within this application area show that laser scanners and arc sensors today replace older systems. Laser scanners have in general high accuracy but are expensive and have to be mounted on the torch, thereby decreasing the workspace of the robot. This problem is partly solved by introduction of an additional motion that rotates the scanner around the torch to keep its relative orientation constant with respect to the seam profile. Arc sensors are inexpensive and compared with other sensors for seam tracking, relatively inaccurate due to the stochastic nature of arc welding. Since welding of large structures in general requires a relatively low accuracy however, arc sensors are a competitive alternative to laser scanners.

In addition, it should be mentioned that there exist solutions for digital image processing of the workpiece before welding. The problems are however, that small changes in the surrounding light may disturb analysis. Further on, 3D spline curve requires the camera to follow the seam all the way to be able to capture segments that otherwise would have remained hidden. This applies also to very long seams. In practical applications today, solutions based on pure image processing by a CCD camera are successfully used to find the initial point where 2D seam tracking, based on laser scanning or arc sensing should start [12].

2.1.1 Arc sensing

![Image of arc sensing](image_url)

**Figure 2:** Arc sensing. Left: Cross section, perpendicular to the approach vector \( a \). The weaving is added to the interpolated motion between \( P_{N-2} \) and \( P_{N-1} \), including changes in position and orientation. Right: Cross section, plane \( \Omega \). By the measurement of the current from start to end, the profile of the workpiece is approximately determined.

In “through-arc” sensing or simply arc sensing [7], the arc current is sampled while a weaving motion is added to the TCP in \( n \) direction, see Fig. 1. Weaving is useful in arc welding due to process related factors and was used already before the introduction of arc sensing in the beginning of the 1980s. In arc sensing, by keeping the voltage constant, it is possible to calculate the distance from TCP to the nearest seam wall.
at each current sample, and thereby during a weaving cycle obtain the profile of the seam. According to experimental results [7] the approximate relationship between arc voltage $V$, arc current $I$ and the nearest distance between the electrode and the workpiece $l$ for an electrode extension ranging between 5-15 mm, may be expressed by the relation:

$$V = \beta_1 I + \beta_2 + \frac{\beta_3}{I} + \beta_4 l$$

(1)

where the constants $\beta_1 - \beta_4$ are dependent on factors such as wire, shielding gas and power-source. In practice the voltage and current readings of the arc contain much noise, why the signal has to be low-pass filtered before it is used for feedback control. An example of such filtered signal, sampled in the ROWER-2 project is presented in Fig. 3.

![Filtered arc signal](image)

**Figure 3:** Example of a filtered arc signal at seam tracking, sampled in the ROWER-2 project. Though anomalies are common at the ignition of the arc, only minor deviations from this pattern showed to occur during continuous seam tracking. The unit of the horizontal axis is ticks, which equals 20 ms. The current values were filtered by an analogue active 4:th order Bessel low-pass filter with $f_0 = 10$ Hz before sampling.

In automatic control, error is defined as the difference between desired (input) value $u$ and current (output) value $y$. The control algorithms in arc sensing are based on methods such as template matching or differential control [7]. In template matching the errors may for instance be calculated by integrating the difference of the template $u$ and the arc signal $y$, where $A$ denotes the weaving amplitude:

$$E_a = \epsilon_a = \int_{-A}^{A} (u(n) - y(n))dn$$

(2)

$$E_n = \epsilon_n = \int_{-A}^{0} (u(n) - y(n))dn - \int_{0}^{A} (u(n) - y(n))dn$$

(3)

Another example includes using integrated difference squared errors. The equation above use $n$ as variable to simplify the notation. Actually, more specifically, the integration between $-A$ and $A$ is performed in negative $\psi$ direction, using an auxiliary variable $\psi$ denoting the state of weaving in $n$ direction, integrated over $[\psi_{-A}, \psi_A]$, where $\psi_{-A}$ and $\psi_A$ denote the turning points of weaving. In differential control, which has proven to be quite reliable for control in $a$ and $n$ directions, current sampling is
performed at the turning points of the weaving. Here, the errors in $a$ and $n$ directions are obtained by:

$$\epsilon_a = I_{ref} - \frac{i_{+A} + i_{-A}}{2}$$  \hspace{1cm} (4)$$

$$E_a = K_a \epsilon_a$$  \hspace{1cm} (5)$$

$$\epsilon_n = i_{+A} - i_{-A}$$  \hspace{1cm} (6)$$

$$E_n = K_n \epsilon_n$$  \hspace{1cm} (7)$$

where $I_{ref}$ is a reference current value and $i_{+A}$ and $i_{-A}$ are the average measured current values at a pair of adjacent extreme points. The parameters $I_{ref}$, $K_a$ and $K_n$ are dependent on the weld joint geometry and other process parameters such as shielding gas and wire feed rate. Since these parameters will be known in advance, $K_a$ and $K_n$ may be defined for any welding application. An example of how $E_a$ and $E_n$ are used for position control follows below:

$$p_N = p_{N-1} + E_n n - r_w o + E_a a$$  \hspace{1cm} (8)$$

where $p_{N-1}$ and $p_N$ denote current and next positions and $r_w$ is nominal displacement during a weaving cycle, also in Fig. 1, which is equal to welding speed (without consideration to the weaving motion) divided by weaving frequency.

### 2.1.2 Virtual sensors

The development of sensor guided control systems may be accelerated using virtual sensors in the simulation control loop. Since a system’s state is often easily obtained in simulation, assuming that the simulation model is sufficiently accurate compared to the real system, this provides for a better foundation for process analysis than in general is possible by physical experiments. Further on, insertion of a sensor in a control loop may cause damage to expensive equipment, unless the behavior of the entire sensor guided control system is precisely known, which is rarely the case at the development stage.

Virtual sensors are used in many application areas, such as robotics, aerospace and marine technologies [4, 9, 16, 19, 21]. Development of new robot systems, such as for seam tracking may be accelerated by application of simulation [14, 15]. In general, the design methodology of virtual sensors may vary due to their specific characteristics. If a sensor model is not analytically definable, artificial neural networks or statistical methods may be used to find appropriate models [2, 20].

### 2.2 Robot system

The design of the 6D seam tracking system was based on experience and results from the European ROWER-2 project [1]. The objective of this project was to automate
the welding process in shipbuilding by the design of a robotic system, specifically for joining double-hull sections in super-cruisers and oil tankers. Figure 4 shows the robot system developed in the ROWER-2 project and the corresponding simulation model in FUSE (described below). The implementation of the 3D seam tracking system was performed in C/C++, running on a QNX-based embedded controller (QNX is a real-time operating system for embedded controllers). The robot was manufactured partly in aluminum alloy to decrease the weight for easy transportation and mounted on a mobile platform with 1 degree of freedom for increased workspace. The robot system was integrated and successfully validated on-site at a shipyard.

Figure 4: The design of the robot system for 3D seam tracking based on arc sensing in the ROWER-2 project (left) was based on simulations in FUSE (right). The 6D seam tracking system presented in this paper was developed based on experience and results gained from the ROWER-2 project.

2.3 Simulation system

The development and implementation of the 6D seam tracking system was initially performed in the Flexible Unified Simulation Environment (FUSE) [11]. FUSE is an integrated software system based on Envision (robot simulator, Delmia Inc.) and Matlab (MathWorks Inc.), including Matlab Robot Toolbox [8]. The integration was performed by compiling these software applications and their libraries into one single software application, running under IRIX, SGI. The methodology developed for the design of the 6D seam tracking system was based on the integration of mechanical virtual prototyping and software prototyping. The 6D seam tracking simulations in FUSE were performed by using the robot developed in the ROWER-2 project and a ABB S4 IRB2400.16 robot, see Figs. 4 and 5. After the design and successful validation of the 6D seam tracking system for arc sensing, the virtual prototyping and software models, partly written in C-code and partly in Matlab code, were translated to pure Matlab code. The translation was necessary, since it saved development time for the implementation of the evaluations system, for the accurate calculation of position and orientation errors during simulation.

2.4 Calculation of position and orientation errors

Error is defined as the difference between desired and current pose \( P \). The position errors were calculated by projection of \( \epsilon_a \) and \( \epsilon_n \) on a plane \( \Omega_t \) perpendicular to the
Figure 5: Examples of 6D seam tracking experiments performed in FUSE, Envision, using a model of the ROWER-2 robot.

desired trajectory while intersecting current TCP, \( P_t \). Thereby, per definition \( \epsilon_o = 0 \). The orientation error (the orientation part of \( P_\delta \)) is calculated by:

\[
P_\delta = P_t^{-1} \cdot P_{\Omega t}
\]

where \( P_t \) is the current TCP and \( P_{\Omega t} \) is the desired TCP for any sample point \( t \). The errors around \( n \), \( o \) and \( a \) are calculated by the subsequent conversion of the resulting transformation matrix to roll, pitch and yaw, here defined as rotation around, in turn, \( a \), \( o \) and \( n \). Rotation in the positive were defined as counterclockwise rotation from the perspective of a viewer when the axis points towards the viewer.

3 Modeling

The principal control scheme of this system is presented in Fig. 6. The control system is based on three main components, position, trajectory tangent vector and pitch control. Pitch denotes rotation around \( o \). The main input of this control system is the \( 4 \times 4 \) transformation matrix \( P_{N-1} \), denoting current TCP position and orientation. The output is next TCP, \( P_N \). \( N \) is the number of samples used for orientation control. The input parameters are \( N \), the proportional control constants \( K_1 \), \( K_2 \) and \( K_3 \), and the generic sensor input values denoting \( \xi = [\epsilon_a \ \epsilon_n] \) and \( \zeta = [k_1 \ k_2] \) in Figs. 2. The proportional constants \( K_1 \), \( K_2 \) and \( K_3 \) denote the control response to errors in position, trajectory tangent vector and pitch control. \( N \) is also used in trajectory tangent vector control and is a measure of the memory size of the algorithm.

3.1 Kinematics singularities

In sensor guided control, the manipulator may involuntarily move into areas in which no inverse kinematics solutions exist, generally called kinematics singularities. It is however possible to minimize inner singularity problems caused by robotic wrists. A novel method is suggested in [10] which was designed for the 6D seam tracking system, but works basically for any industrial robot with 6 degrees of freedom. In this method, the stronger motors, often at the base of the robot, assist the weaker wrist motors to compensate for any position error. This method allows also for smooth transitions between different configurations.
Figure 6: The principal scheme of the 6D seam tracking control system. $\mathbf{P}_{N-1}$ and $\mathbf{P}_N$ denote current and next TCP. Next TCP is calculated at the beginning of each weaving cycle, and the motion between $\mathbf{P}_{N-1}$ and $\mathbf{P}_N$ is interpolated.

3.2 Initial and final conditions

In the experiments, seam tracking was initiated from a proper position already from start. The start position for seam tracking is in general found either by image analysis using a camera or by seam tracking itself. Since previous positions are unknown at start, an estimation is made of previous $N - 1$ positions, by extrapolation of the trajectory in the direction of $o$. This is the reason why in some experiments, higher orientation errors occurred at the beginning of seam tracking. The condition to stop seam tracking may be based on the position of the TCP. The stop signal may be triggered by definition of volumes outside which no seam tracking is allowed.

4 Experimental results

4.1 Workpiece samples

The position and orientation errors at seam tracking were calculated based on simulation experiments of basically 8 different workpieces, see Figs. 7-9. The workpieces in Matlab were approximated by lines orthogonal to the direction of the weld and the resolution was defined as the nominal distance between these lines. These resolutions were set to 0.5 mm.

The workpieces are categorized into two groups, continuous and discrete. In the experiments presented in this paper, the range was $0 - 180^\circ$ for continuous, and 120 mm for discrete workpieces. In seam tracking of a continuous workpiece, a sudden change is interpreted as a curve with a small radius of curvature, see Fig. 7, while in seam tracking of a discrete workpiece a sudden change is regarded as a disturbance, see Figs. 8-9. The settings of the control parameters decide alone the mode of the control system: continuous or discrete. For seam tracking of objects with a large radius of curvature, the discrete mode works well for both continuous and discrete workpieces.

The control parameters $K_1, K_2, K_3$ and $N$ used in the presented experiments showed to work well for each category, continuous and discrete. These settings were found by more than 100 arc sensing simulation experiments in Matlab, and large deviation...
Figure 7: Seam tracking, arc sensing, with addition of random noise to the measured distance. Pipe intersection (top) and workpiece for isolating control around a (bottom). K_1 = 1.0, K_2 = 0.5, K_3 = 0.25, N = 10. Error denotes the difference between desired and current pose.

Figure 8: Seam tracking, arc sensing, with addition of random noise to the measured distance. The step is introduced at x = 0 mm. K_1 = 0.5, K_2 = 0.2, K_3 = 0.1, N = 20.
Figure 9: Seam tracking, arc sensing, with addition of random noise to the measured distance. In this figure, the case in the bottom denotes a step of $-30^\circ$ performed around $o$. $K_1 = 0.5, K_2 = 0.2, K_3 = 0.1, N = 20$.

from these parameters showed to create instability in the system, making the control components working against each other instead of in synergy, resulting in divergence from the seam and ultimately failure at seam tracking.

As mentioned previously, the seam tracking system is theoretically able to handle any angle $\alpha$ (in Figs. 1 and 2) between 0 and $180^\circ$. In practice however, a small angle may cause collision between torch and workpiece. For simplicity, $\alpha$ was set to $90^\circ$ for all experiments except for pipe intersections. Further, $d$ in Fig. 1 was set to 1 mm.

4.2 Arc sensing

According to [22], the accuracy of an arc sensor in seam tracking is 0.5 mm. In simulation, the workpiece line resolution was set to 0.5 mm. To increase the errors in the system, a random signal with a range of 1 mm was added to the measured distance. In addition, no integration of the current measurement values was made near the turning points at weaving. On the contrary, one measurement was considered as enough for each turning point. The signal was prefiltered by an active 4:th order Bessel low-pass filter in the ROWER-2 project. In the simulation experiments presented in this paper no filtering was performed. Thereby the accuracy had been decreased to less than half of the accuracy suggested in [22]. In practical applications however, the arc signal should be filtered by a method that does not add any phase shift to the signal, such as a proper digital filter.

The arc sensing simulations presented in this paper were performed with a constant welding speed of 10 mm/s, weaving amplitude and frequency of 5 mm and 3 Hz, and 64 current samplings per weaving cycle (or 192 samplings per second). The algorithm worked well also for lower sampling rates, such as 50 samples per second, which was
used in the ROWER-2 project. In general, higher sampling rates showed to improve the control ability of the system, for rates higher than perhaps 10 samples per weaving cycle, though rather slightly than proportionally. The simulation models proved to predict the estimations derived from physical experiments in the ROWER-2 project for 6D motion. According to those experiments, it was possible to perform differential seam tracking in $y$ and $z$ directions in Fig. 1, up to 20 mm for a seam of the length 1 m. At a speed of 9 mm per second, weaving amplitude and frequency of 3 mm and 3 Hz, this gives a redirection capability of 1.15° per weaving cycle which in the worst case is equivalent to a minimum radius of 150 mm. Using a power-source without any intrinsic current control would further give up to 3 times better results according to the experiments in the ROWER-2 project (current control was only applied by the power source to limit the current). Using a high performing industrial robot and a weaving amplitude of 5 mm instead of 3, this would finally bring down the curvature to 25-50 mm, which actually was the amount found by simulation.

5 Results and discussion

A 6D seam tracking system was designed and validated by simulation that is able to follow any continuous 3D seam with moderate curvature, using an arc sensor, and is able to continuously correct both position and orientation of the tool-tip of the welding torch along the seam. Thereby the need for robot trajectory programming or geometrical databases were eliminated for workpieces with moderate radius of curvatures, increasing intelligence in robotic arc welding.

Simulations based on physical experiments in the ROWER-2 project showed however that arc sensing in real-time welding was theoretically possible for a radius of curvature below 50 mm. This is much less than the initial objective to design a 6D seam tracking system that could manage a radius of curvature below 200 mm, which is still considered as small for instance in shipbuilding.

6 Future work

Though the 6D seam tracking system showed to be very robust, optimization of the algorithm is still possible. Presently, pitch control in arc sensing is based on data measurements during a half weaving cycle. By extending the measurements to one full cycle, pitch control may be optimized. Further on, though the control system worked well using proportional constants alone, if needed, for instance PID or adaptive control may be included to enhance the performance. One of many examples is for instance to define $K_2$ as a function of the “radius of curvature” of the 2nd degree polynomial used for calculation of the trajectory tangent vector, thereby enable the system to automatically change between normal operation, used for maximum control stability, and extraordinary operation with fast response, used for management of small radius of curvatures.

References


