17

Performance Evaluation of Autonomous Contour Following Algorithms for Industrial Robot

Anton Satria Prabuwono*, Samsi Md. Said**,
M.A. Burhanuddin*** and Riza Sulaiman*
*Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia
**Faculty of Electrical and Automation Technology, TATi University College
***Faculty of Information & Communication Tech., Universiti Teknikal Malaysia Melaka
Malaysia

1. Introduction

The current trend towards industrial robotics requires the development of simple programming techniques. Programming by demonstration has emerged as one of the most promising solutions for effective programming of robot tasks (Ikeuchi & Suehiro, 1994; Zollner et al., 2002). Contour following is one of basic task in industrial robot manipulation. In this task, the robot is holding a tool to follow the contour of an object whose shape and pose are often unknown (Mi & Jia, 2004). These applications include part polishing, inspection, sealing, painting, cleaning, modeling, etc. During the following process, the tool is constrained on the surface to maintain contact force while moving along some tangential direction.

In order to use robotics for such application two sequences of step need to be considered, the programming phase and the playback phase. In the programming phase, teaching a group of points is required while for playback phase, the robot Tool Centre Point (TCP) will follow the taught points recorded previously. This programming phase especially for contour following application is quite tedious and time consuming. For example, in order to track an arc, the robot programmer needs to manually use teaching box or teaching pendant to jog (powered motion) the robot Tool Centre Point to three points that enclosed an arc. For a complex contour, several series of three points must be taught, besides finding the optimum process parameter (voltage, current and electrode speed for arc welding application) related to those points. Next, the motion instruction, speed and type of termination that describes the closeness of zoning radial distance to the taught points needs also to be defined. The programmer must iterate the points and process parameters several times until the optimum combination are achieved. In comparison to assembly operation where the programmer just need to teach few points such as approach, insert and depart points, contour following for painting, arc welding and sealing application requires a large number of points recorded and at the best location. After all the best program and process

parameters are achieved for one sample part, the same quality is expected for the subsequent parts in a batch. This expectation alone poses difficult challenges to the industry since parts do vary dimensionally due to inaccuracy in manufacturing and joining operation. For example, in welding job such as gas-metal arc welding (GMAW) process the part expands dimensionally since the volume of molten material in the weld bead is proportional to the heat input (Tomizuka et. al., 1980). Furthermore, the current Flexible Manufacturing System (FMS) concept requires different kind of parts variations for one production run. This means that a great number of robot programming is required to cater parts variations and uncertainties per production run compared to the old days of batch or mass production concept.
This work describes a study on developing several types of simple algorithms in order to automate the manual programming process. The main objective of the study is to present the performance evaluation of those algorithms for autonomous contour following task. The algorithms have been developed and tested using Adept Selective Compliant Assembly Robot Arm (SCARA). These algorithms include adapting gradient method, staircase method, and sweeping radius method .

2. Automation and Enhancement of the Robot Teaching Process

Certain robot application requires complex and tedious teaching procedures where skilled programmer needs to iterate the teaching process several times until the disposal of opposing parameter are achieved for optimum robot contour tracking program such as in the arc welding, windshield glass sealing and painting application. Normally it takes days to teach several optimal arc welding points especially in a flexible, just-in-time, and CAD customized production approach (Samsi & Nazim, 2005).
A lot of efforts have been done in automating and enhancing the teaching process of the discussed applications above. Yuehong et al. (2004) explained that for the application of robot to complex tasks, which often requires its end-effector to come into contact with the unknown surface, it is often necessary to control not only the position but also the force exerted by the end-effector on an object, otherwise the arising contact forces may damage the object and the robot system. Active tracking of unknown surface is also a real problem in the industrial world. Force sensing and control, similar to vision, is a fundamental part of robot. It essentially simulates human tactility. Over 70% information can be obtained by human vision from external environments. However, over 2/3 of human brain and neural system are used to manage the tactility. The senses of force, contact, press, and slide all belong to human tactility. Compared with the others, force sensing is the most synthetic and complex means, because it consist of touching the object with the robotic system while maintaining optimum path. Therefore, active tracking of unknown surface using force sensing and control technique has become one of the most important studies in robotics. But, the strategy of force sensing and control has not been effectively solved because of the acute contradiction between the strict requirement of the force exerted by robot on external environment and the stiffness of position-servo and mechanism of robot in the free space. Cartesian space force control is achieved by controlling every actuators torque in joint space. The simplest torque control is an armature current control loop, which is often called an inner current loop.

Robotics welding process is quite challenging for robot programmers since different parts are to be welded demanding an intelligent robot-welding concept because of the individual part dimension deviations due to heat and low tolerance manufacturing process. The realization of such ambitious goals leads to the use of sensors which provide the robot with the necessary information, so that it can interact within their environment (Hewit, 1996). Further requirements are that the robot must hold the electrode at the correct orientation and distance to the seam and move at a constant velocity so that a constant of material flow into the joint. This problem becomes too complex for three dimensional objects than on flat plates, and often requires geometric modeling to plan the robot motion. Several types of arc welding sensors are under investigations (Prinze & Gunnarson, 1984). The most promising are preview sensing, through the arc sensing, and direct-arc sensing. Variants of these are available commercially (Hanright, 1984). The majority of research into seam tracking focuses on non-contact sensors. The most promising sensing method is optical triangulation, where a rotating mirror scans a thin beam of light across the objects to be welded (Oomen & Verbeck, 1983). Preferentially, the robot should autonomously find and precisely weld metal joining paths in order to fulfill some given manufacturing tasks. This solution can be expanded further to surface following task in painting application and the data measured can be used for machining purpose as in coordinate mesuring machine (CMM).

Adolfo et. al. (2001) used predictive look ahead sensor guided technique using CCD camera to capture contour line and approximate them with a polynomial of certain degree to smooth out the curve and neglecting any experimental error. It was shown how a smart tool integrated with sensor that can look ahead and plan the trajectory can be attached and reattached in a flexible manner to robotic manipulators. This solution can automate industrial programming processes in an intelligent manner. Those are objects of intense research efforts in the field of artificial intelligence (AI): to build machines that consider the information captured from the surrounding environment in a proper (intelligent) manner. With the support of sensors the working trajectory of the robot can be obtained within a certain sensor field which will be used here as the minimization of the tracking error. The main contribution of the work was an on-line tracking optimization scheme for sensor guided robotic manipulators by associating sensor information, manipulator dynamics and a path generator model. Experimental results on implementation of a CCD-camera guided hydraulic robot and a welding robot demonstrates the proposed approach.

Intelligence and flexibility are two essential features in a smart mechatronics product. In his research work several problems were addressed such as sensor integration, real-world modeling, trajectory path planning, task-level planning and execution, and the control of the robotic system as a whole. Moreover, building autonomous smart tool that enhances the welding process provides a stringent test bed for new concepts and approaches in both hardware and software which is very near to commercialization. The concentration of the work was on the design and development of an autonomous platform, using mechatronics to implement intelligent behaviors, with the help of a industrial robot controller interfaced with sensors (Gopalakrishnan et. al., 2004).

In the automation of arc welding, it is necessary to find the starting point for the welding and control the welding torch precisely along the welding line. However, in the actual welding, there are a lot of external variations including the change of the weld gap, the movement of work piece due to local thermal expansion, and the existence of obstacles and weld tacks. They are mainly to detect a groove location with high precision by the slit

lighting method (Awahara & Taki, 1979), gap by an edge detection operator (Inoue, 1979), and to detect a weld line on a thin plate by considering lighting condition (Suga et al., 1992). Rasol et al. (2001) successfully created a prototype system that automates the teaching of spot welding process by building a prototype knowledge based expert system software. Two main areas were considered such as automatic setting of spot welding parameters and automated placing of spot welds using a robot fitted with a welding head. The prototype system reduces the teaching time, improves positioning of spot welds location, reducing unnecessary spot weld points and increases the flexibility level of manufacturing system.
Andersson & Johansson (2000) developed method to implement a control strategy for wood carving operations. A control method that improves robot control and supports simpler programming, based on a wrist mounted force/torque-sensor, is proposed for the wood carving process. Their work describes the structure and control algorithm of the system and how different machining parameters affect the cutting forces. The evaluation of the system showed that it is possible to control the cutting depth at speeds up to 7.5 mm/s by adjusting the rake angle of the tool to obtain a nominal force.
All the previous works reviewed require force sensor, vision system, laser sensor and CAD data. These entire devices are quite expensive relatively and the algorithm that support them also quite complex. Majority of the works require contact sensing between end-effector and work piece which is undesirables in term of tool reliability and work piece quality in certain process. These research vacuums justify the unique scope of undertaking research to develop and design a cheap and non contact simpler algorithm program which is being tested using real industrial machines. So, a research that employs a simpler solution system which uses only one cheap discrete sensor with simple algorithm, and taking advantage of common industrial robot controller, is justified.

3. Contour Following Methods

3.1 Adapting Gradient

This method requires a sampling distance at time T where the robot TCP traversing a distance dX defined by user. The sampling distance dX is the critical parameter that defines the contour resolution measured just like sweeping segment radius r as in the method discussed earlier. The longer the distance dX, the coarser the curve modeled, and the smaller the r value the finer the curve is modeled (Prabuwono & Samsi, 2007). Fig. 1 shows the details of adapting gradient formulation.

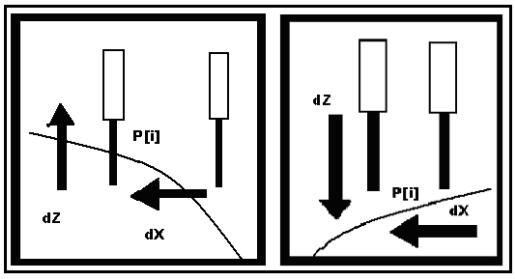


Fig. 1. Adapting gradient formulation.

After the initial sampling distance *X*, then the sensor on tool tip will sense if the contour is within the sensor sensing range. If the sensor reading is on, that is mean that the tool is inside the contour and need to be brought upward. Fig. 2 shows the adapting gradient algorithm.

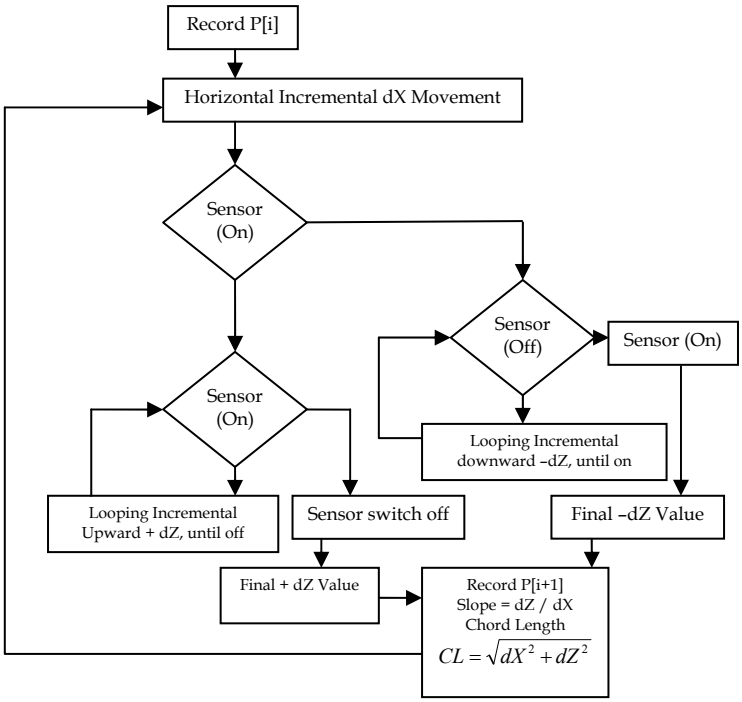


Fig. 2. Adapting gradient algorithm.

On the other hand if the sensor is off then that means the tool is too well upward above the contour and needs to be brought downward incrementally. Incremental downward or upward movement is then employed depending on the sensor status whether on or off . At the end of each cycle the position is stored. The process is being repeated until the whole complex contour is measured. The incremental distances *dX* and *dZ* variable are very important in measuring the chord segment of this method. Unlike sweeping chord algorithm that maintains constant chord segment length while measuring and recording the α angle, the adapting gradient method only maintaining constant sampling distance *dX* while having variable *dZ* measured and recorded.

The chord length and slope of the ratio of *dZ/dX* are functions of *dX* and *dZ*. The upward adapting motion will approximate incrementally the positive gradient along contour while downward adapting motion will approximate the negative gradient along contour. Using this method the position *p[i]* after each measuring cycle is stored in the position database and to be used repeatedly in the playback mode. In this way the whole contour is being approximated and the positions stored (refer to Fig. 1 and Fig. 2). The positions can be used for $P_{i+1} = P_iD(i+1)$. The drive function that summarizes all these can be represented as:

$$D(i+1)=\begin{bmatrix} C(\theta) & 0 & S(\theta) & n_A \bullet dX \\ 0 & 1 & 0 & s_A \bullet dX \\ -S(\theta) & 0 & C(\theta) & a_A \bullet dZ \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

Total trajectory point generated at point *N* as follows:

$$P_N = \prod_{i=1}^{N-1} P_i D(i+1) \tag{2}$$

$$\delta_{XZ} slope = \frac{a_A \bullet dZ}{n_A \bullet dX} \tag{3}$$

The advantage about this method is that the numbers of points measured and stored are constant throughout horizontal contour length regardless of contour complexity (only chord length differs). Anyway, for short sensing range and at high gradient complex contour this method is quite dangerous which can cause collision because it penetrates into the contour solid areas. The decision to measure variable distance *dZ* is not consistent, from the solid to empty contour edge at positive gradient and from inside the empty contour to the edge of contour, the condition is not uniform. Several precautions must be taken care proactively such as the sampling distance *dX* and speed $\dot{dX}$ dynamic overshoot for high gradient and short sensing distance.

3.2 Staircase

Staircase method is introduced whereby it requires the robot TCP climbing a constant distance *LZ* and the sampling distance *LX*. Then the tool moves downward incrementally until the contour is within the sensor sensing range (Prabuwono et al., 2008). Then it repeats the previous process of climbing upward a constant distance *LZ* and traversing horizontally a distance *LX*. At this moment the new position of point *P(i+1)* is being recorded and the difference *dZ(i)* of *LZ(i+1)* and *dZ(i)* is calculated. This method is maintained constantly *dX* and *dZ*. The *dZ(n)* is the critical parameter that defines the contour resolution measured just like sweeping segment radius *r*. Fig. 3 shows the staircase method formulation in several sequences.

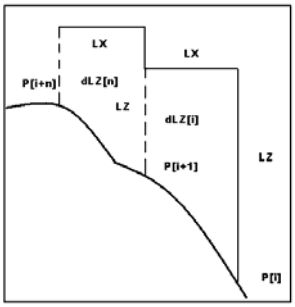


Fig. 3. Staircase method formulation in several sequences.

The longer the distance *LX*, the coarser the curve modeled, and the smaller the *LX* values the finer the curve being modeled. In this way the whole contour is being approximated and the positions stored. The positions can be used for $P_{i+1} = P_i D(i+1)$.
The drive function that summarizes all these can be represented as:

$$D(i+1) = \begin{bmatrix} C(\theta) & 0 & S(\theta) & n_A \bullet (LX[i]) \\ 0 & 1 & 0 & s_A \bullet (LX[i]) \\ -S(\theta) & 0 & C(\theta) & a_A \bullet (LZ[i] - ABS[dLZ[i]) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\delta_{XZ} slope = \frac{a_A \bullet (LZ[i] - ABS[dLZ[i]])}{n_A \bullet (LX[i])} \tag{5}$$

Total trajectory point generated at point *N* as follows:

$$P_N = \prod_{i=1}^{N-1} P_i D(i+1) \tag{6}$$

Unlike sweeping chord algorithm that maintains constant chord segment length and while measuring and recording the α angle, the offset staircase method only maintains constant sampling distance *LX* and *LZ* while having variable *dZ* measured and recorded. The chord length and gradient slope of the ratio of *dZ/LX* are a function of *LX*, *LZ* and *dZ(n)*. The upward staircase (*dZ* is positive value) will approximate incrementally the positive gradient along contour while downward staircase (*dZ* is negative value) will approximate the negative gradient along contour. Using this method the contour information is stored in the position database to be used repeatedly in the playback mode. The sampling distance *dX* is constant throughout the horizontal X axis distance along the complex contour regardless of gradient slope measured.
The measurement variable is only distance *dZ* where through some manipulations the slope *dZ/dX* at constant segment $CL = \sqrt{dX^2 + dZ^2}$ is being derived and stored at program database. The problem with this method is the nature of uniform constant horizontal segment regardless of variable slope gradient .The number of positions stored can be calculated depending upon the horizontal contour distance and sampling distance *dX*. On the other hand the chord length *CL* will vary according to the measured *dZ* and constant sampling distance *dX*. Since *dX* is constant the slope will be decided by the *dZ* length, the higher the *dZ* value, the higher the length of chord segment. The advantage of this method is that the numbers of points measured and stored are constant throughout horizontal contour length regardless of contour complexity. Fig. 4 shows staircase algorithm.

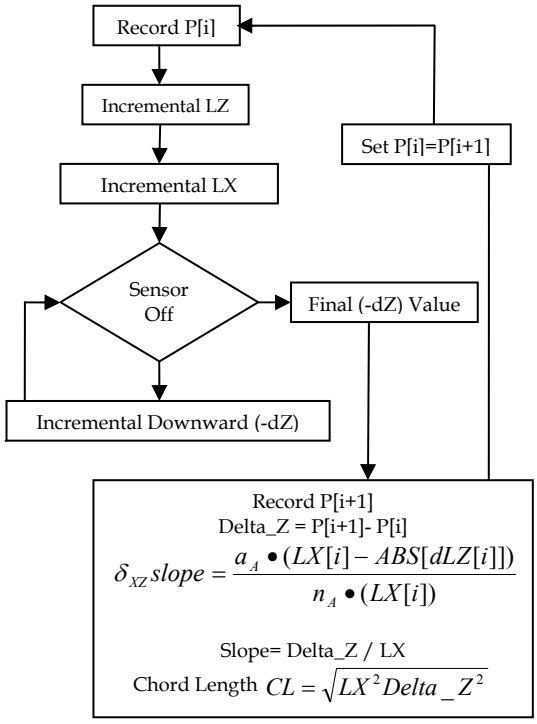


Fig. 4. Staircase algorithm.

3.3 Sweeping Radius

This method is to automate the incremental measuring motion utilizing the gross output of total positions and yaw orientation angles from task planning algorithm. Another important point is the slope gradient measurement at any knot points for correcting the optical sensor reflectance correction factor along the contour positive and negative slope gradient. The complex contour of any different gradient is being approximated by segment of chord distance *r*. The smaller the *r* value, the higher the accuracy of contour shapes being measured but at a higher computation cost (refer to Fig. 6 and Fig. 7). The first part of robot program is to measure the incremental position and slope along the contour gradient and store the positions recorded in the database. The stored locations will be used repeatedly for playback purpose in subsequent passes (running a production part program). A low cost digital optical sensor is being fitted into a tool holder as shown in Fig. 5.

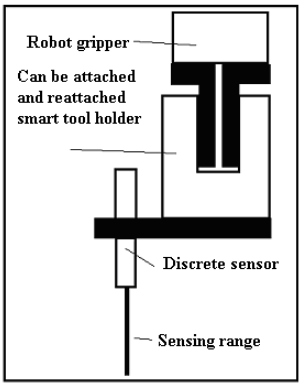


Fig. 5. Smart tool holder with sensor.

This sensor has a sensing range of about 25 millimeter, which is important to avoid tool colliding with the contour surface. The robot TCP started by moving upward in *Z* axis a distance *r* and suddenly sweeping downward in radius *r* from angular step $\alpha = 0$ to $\alpha = 180$ in one degree step. The sweeping motion is terminated when the contour shape is within the digital optical sensing range. In this way, at certain angle α and segment chord length *r*, the new relative point and slope is measured and recorded.

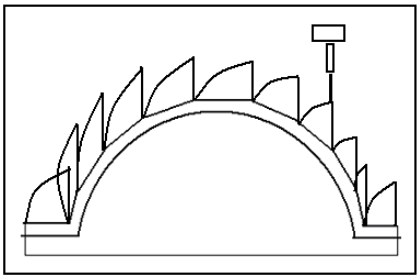


Fig. 6. Semi-circle sweeping radius method.

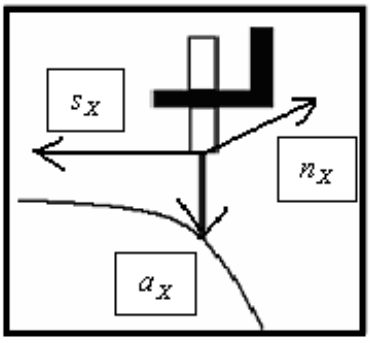


Fig. 7. Tool axes.

Measuring slope is also important for correcting the optical digital sensor reflectance factor. In this way the whole contour is being approximated and the positions stored. The positions can be used for $P_{i+1} = P_iD(i+1)$. The drive function that summarizes all these can be represented as:

$$D(i+1) = \begin{bmatrix} C(\theta) & 0 & S(\theta) & n_A \bullet r(\sin\alpha) \\ 0 & 1 & 0 & s_A \bullet r(\sin\alpha) \\ -S(\theta) & 0 & C(\theta) & a_A \bullet r(\cos\alpha) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Total trajectory point generated at point N for playback purpose in subsequent passes is:

$$P_N = \prod_{i=1}^{N-1} P_i D(i+1) \tag{8}$$

Constant chord segment will follow complex contour regardless of gradient slope measured. The measurement variable is only angle α where through some manipulation the slope at constant segment will be derived and stored at program database. Fig. 8 shows sweeping radius algorithm.

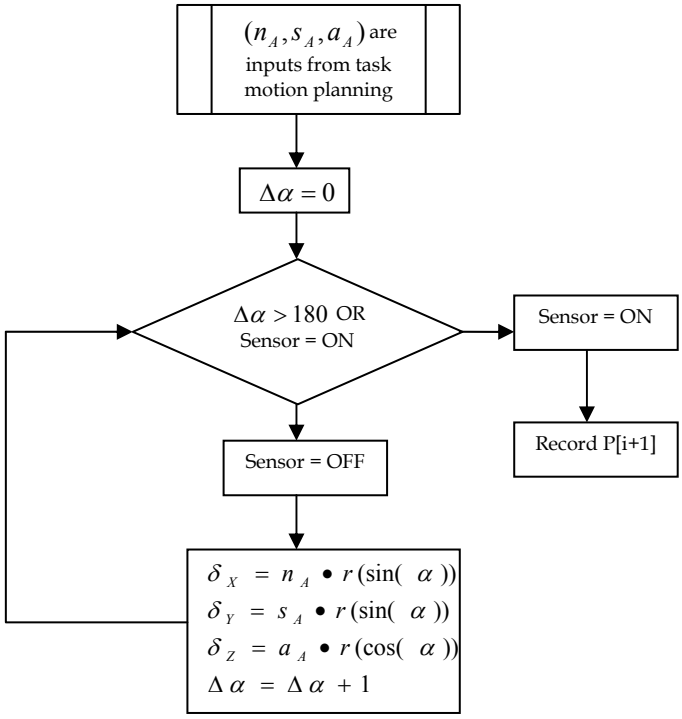


Fig. 8. Sweeping radius algorithm.

The problem with this method is the nature of constant segment will measure the flat surface with the same intensity as high slope gradient (the measurement should vary according to gradient slope which should be less in flat surface contour and a bit high on higher slope gradient contour).

4. Task Planning Formulation

4.1 Cartesian Trajectory Planning

In the path planning process, two choices were given either keying several points of X-Y Cartesian coordinate or manually jog the robot tool centre point (TCP) to the desired initial and final location defining one line segment. By jogging, user will bring the robot end effector TCP using powered motion via a teach pendant (refer to Fig. 9 and Fig. 10). This process will be repeated for multi segment lines defining a closed curve or an open curve. The information from these two initial and final points will be used to adjust the end effector TCP yaw orientation rotation angle from initial point heading to final point.

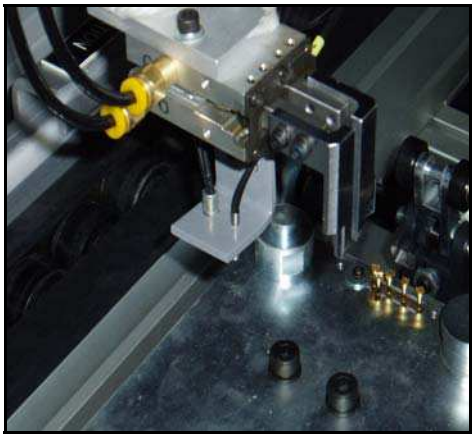


Fig. 9. Initial line top view position teaching (point recording).

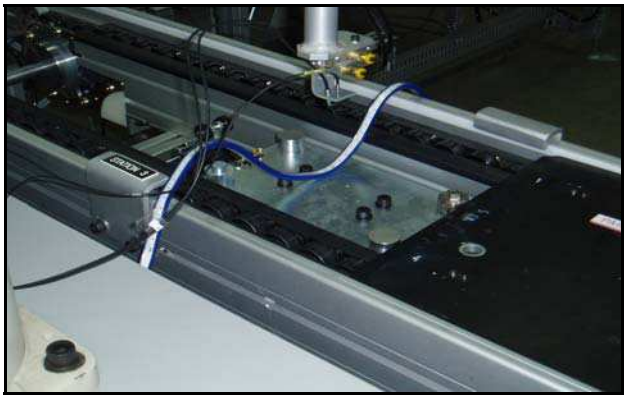


Fig. 10. Final line top view position teaching (point recording).

The initial and final location of robot TCP taught locations are actually homogeneous transformation matrices describing both orientation and position of the TCP with respect to robot base world coordinate system. The homogeneous transformation matrices are shown below which role is to describe the initial and final location of taught points.

$$P_{initial} = B = \begin{bmatrix} n_A & s_A & a_A & p_A \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x{}^A & s_x{}^A & a_x{}^A & p^A{}_x \\ n_y{}^A & s_y{}^A & a_y{}^A & p_y{}^A \\ n_z{}^A & s_z{}^A & a_z{}^A & p_z{}^A \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

$$P_{final} = A = \begin{bmatrix} n_B & s_B & a_B & p_B \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x{}^B & s_x{}^B & a_x{}^B & p^B{}_x \\ n_y{}^B & s_y{}^B & a_y{}^B & p_y{}^B \\ n_z{}^B & s_z{}^B & a_z{}^B & p_z{}^B \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

The final point is related to initial point through the Gross X-Y plane task planning matrix as follows:

$$P_{final} = P_{initial} T(N) \tag{11}$$

Multiplying by the inverse of $P_{initial}$ for both sides, the gross motion X-Y plane task planning matrices are derived as follows (● indicates dot product of two vectors):

$$T(N) = \begin{bmatrix} n_{A.} \bullet n_B & n_A \bullet s_B & n_A \bullet a_B & n_A \bullet (p_B - p_A) \\ s_{A.} \bullet n_B & s_A \bullet s_B & s_A \bullet a_B & s_A \bullet (p_B - p_A) \\ a_A \bullet n_B & a_A \bullet s_B & a_A \bullet a_B & a_A \bullet (p_B - p_A) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

Information from gross X-Y plane task planning matrices can be used to describe the tool position component P_x, P_y, P_z in Cartesian coordinate, the rectilinear distance L and orientation yaw angle β_z as described below:

$$Px = n_A \bullet (p_B - p_A) \tag{13}$$

$$Py = s_A \bullet (p_B - p_A) \tag{14}$$

$$Pz = a_A \bullet (p_B - p_A) \tag{15}$$

$$L = (P_x^{\,2} + P_y^{\,2} + P_z^{\,2})^{\frac{1}{2}} \tag{16}$$

$$\beta_Z = \tan^{-1}\left\{\frac{\left[(n_A \bullet a_B).^2 + (s_A \bullet a_B)^2\right]^{\frac{1}{2}}}{a_A \bullet a_B}\right\} \quad 0 \le \beta_Z \le \pi \tag{17}$$

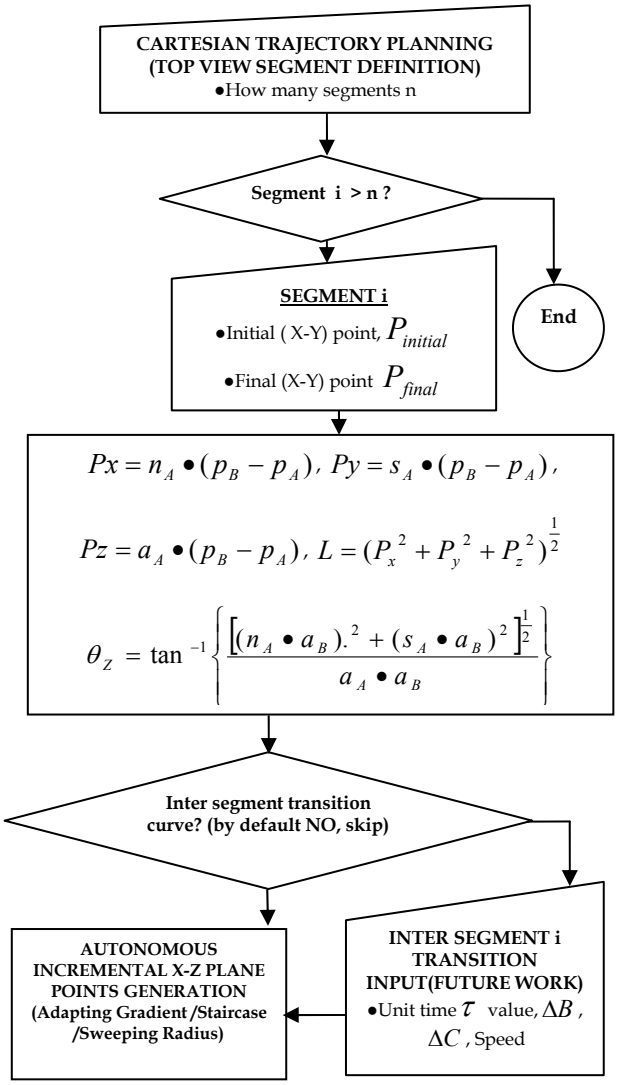


Fig. 11. Hybrid three dimensional path planning.

Fig. 11 shows the hybrid three dimensional path planning. In this diagram, the selected algorithm will replace autonomous incremental X-Z plane points generation. The n_A, S_A, α_A and θ_z will become input into incremental X-Z plane segment differential chord planning which is the major problem to be solved in this work.

4.2 Drive Transform Model

Smart sensor feedback and programming algorithm will guide the TCP to approximate the curve with a straight line segments that knot from points to points in three dimensional Cartesian X-Y-Z plane. The measured knot points and segment slope at any points will be stored in the database and will be used repeatedly in robot part program playback mode. The objectives of automating tedious and time consuming contour tracking programming process will be achieved. Adapting gradient algorithm will further explain the incremental position of δ_X, δ_Y and δ_Z of general incremental drive transform described in Equation 19. For start, in order to simplify the mathematical formulation, four degree of freedom Adept SCARA robot is used. In future research, a six degree of freedom robot can be used to test the robustness and applicability of the algorithms. Utilizing drive transform equation for four degree of freedom SCARA robot will simplify a lot of things (Paul, 1972; Paul, 1979). For example, only one yaw orientation angle exist. Then, the chord segment relative path transformation drive transform is being decomposed only into one rotation matrix to orientate tool about Z axis and one straight line translation matrix also along tool axis. In order to achieve the motion between two consecutive Cartesian knot points, the derivation of segment drive transform is very useful since motion from i to $i + 1$ is related to drive transform as:

$$T_4(i+1) = C_{workobject} P_i D(i+1) ({}^{tool}T_{i+1})^{-1} \tag{18}$$

$T_4\ (1 + 1)$ is the transformation stored to the database and contain both tool position and orientation at any points which also becomes input to the inverse kinematics routine in order to get local coordinate of individual robot joint angles (another joint level cubic polynomial trajectory planning or differential Jacobian method which is not discussed here). After some mathematical operation, the position of consecutive knot points at beginning from i to end of segment $i + 1$ is a function of drive transform as $P_{i+1} = P_i D(i+1)$.
The general transformation matrices drive transform that summarizes all these can be represented as:

$$D(i+1) = \begin{bmatrix} C(\theta) & 0 & S(\theta) & \delta_x \\ 0 & 1 & 0 & \delta_y \\ -S(\theta) & 0 & C(\theta) & \delta_z \\ 0 & 0 & 1 & 1 \end{bmatrix} \tag{19}$$

The yaw orientation angle θ is actually an input from gross motion task planning that was discussed previously. The detail of incremental position in Cartesian X-Y-Z three dimensional plane δ_X, δ_Y and δ_Z explained in several individual alternate algorithm developments in the previous section just to automate this differential relative motion. Incremental drive transform will describe the final position at any point N is generated as:

$$P_N = \prod_{i=1}^{N-1} P_i \, D(i+1) \tag{20}$$

The related transformation at any point *N* which became input to inverse kinematics routine for joints space trajectory planning as follows:

$$T_4(N) = \prod_{i=1}^{N} C_{worobject} P_i \, D(i+1)({}^{tool}T_{i+1})^{-1} \tag{21}$$

5. Experiment and Results

5.1 Experiment

The V+ programs from those algorithms will be written and will be tested using an Adept SCARA robot. The contours traced by robot TCP on the semicircle object which has radius value of 40 millimeter along the *X* axis are plotted according to the algorithm used. All actual contour coordinate traced at any point will be compared to the known geometry contour shape equation in order to find tracking error at any point *i*. The error between actual coordinated traced by robot TCP and known coordinate value of semicircle geometry at any time *i* is:

$$\varepsilon_i = w_i - z_i \tag{22}$$

Where w_i is actual geometry and z_i is the contour traced.
Two graphs are plotted for individual algorithm such as the actual contour traced versus semicircle geometry, tracking error versus contour geometry (all along *X* axis).The mean of tracking error will be used to measure the performance index of proposed algorithm for all point *N* captured as follows:

$$\bar{\varepsilon} = \frac{1}{n}\sum_{i=1}^{N} \varepsilon_i^2 \tag{23}$$

Another criterion to measure the error distribution is by employing standard deviation as follows:

$$s = \left[\frac{1}{n}\sum (\varepsilon_i - \bar{\varepsilon_i})^2\right]^{\frac{1}{2}} \tag{24}$$

Semicircle shape was chosen because it provides an ideal test bed and it contains all ranges of slope gradient that are available in real world. It exhibits infinity value at the very beginning point and progressing down with a finite very high positive slope. The slope decreasing into zero value in the middle of the contour and finally reaches very high negative slope at the other end along the *X* axis. At the very end of the semicircle contour the infinity slope reappear again. These phenomena cause high reading of Cartesian vector *Z* for any minute vector *X* displacement value. These infinity region problems will be

avoided by introducing a safety margin ranging from 0.1-2.5 millimeter at the both ends of the semicircle geometry. The numbers of sampling measurement points depend on the method employed so the sampling points of every method do vary depend on the method employed. It is anticipated that the tracking error value will be quite high in certain slope region of contour gradient (Prabuwono et al., 2009). Fig. 12 shows the four degrees of freedom SCARA robot that used in this study.

Fig. 12. The four degrees of freedom SCARA robot.

5.2 Results

The actual contour traced and the tracking error along contour, matching the semicircle geometry of radius 40 millimeter is plotted. For adapting gradient method, the enlargement of mean of tracking error with the value of - 0.3773 millimeter and the standard deviation of tracking error with the value of 2.3085 millimeter are shown in Fig. 13 and Fig. 14 respectively. The safety margin of 0.1 to 1 millimeter is allowed at the beginning and near to the end of semicircle object in order to avoid measuring the very high slope at those regions. The adapting gradient measuring advance parameter of 1 millimeter is chosen for this contour following experiment. The total sample of good 79 points was collected over 80 millimeter horizontal measuring distance.

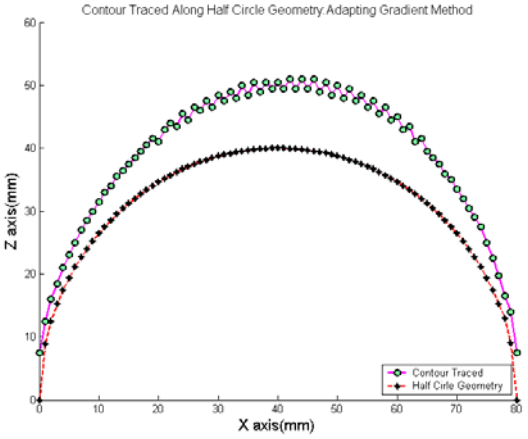


Fig. 13. Contour traced along half circle geometry with adapting gradient method.

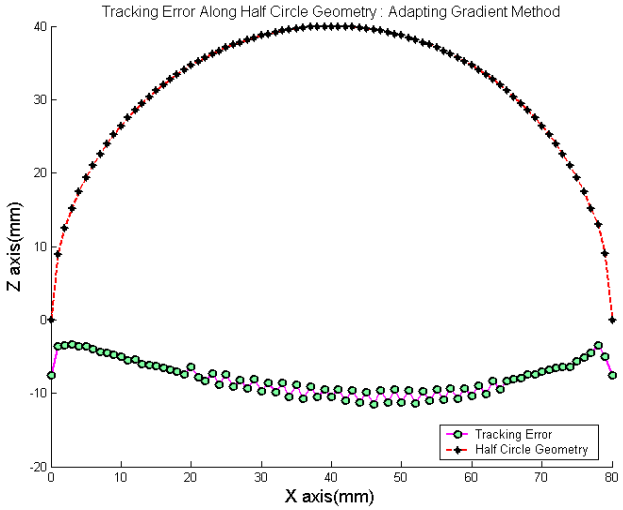


Fig. 14. Tracking error along half circle geometry with adapting gradient method.

For staircase method, the enlargement of mean of tracking error with the value of 3.4011 millimeter and the standard deviation of tracking error with the value of 1.8412 millimeter are shown in Fig. 15 and Fig. 16 respectively. The safety margin of 0.1 to 1 millimeter is allowed at the beginning and near to the end of semicircle object in order to avoid measuring the very high slope at those regions. The staircase measuring advance parameter of 1 millimeter is chosen for this contour tracking experiment.The total good sample of 78 points was collected over 80 millimeter horizontal measuring distance.

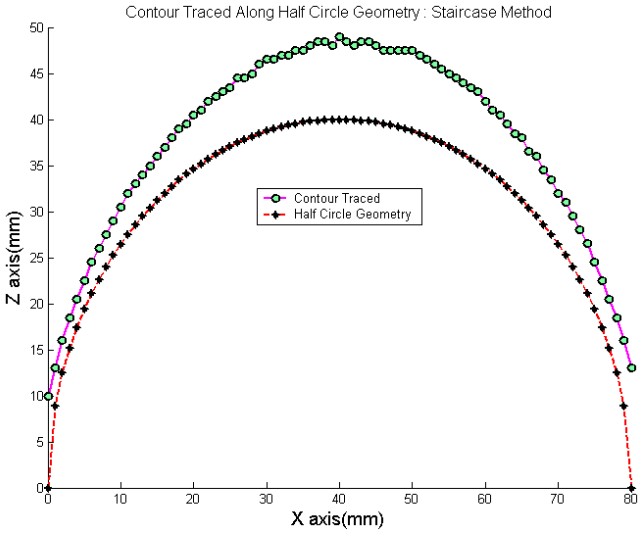


Fig. 15. Contour traced along half circle geometry with staircase method.

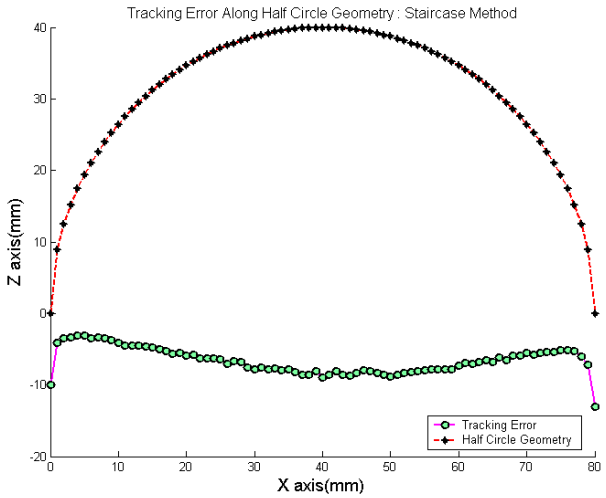


Fig. 16. Tracking error along half circle geometry with staircase method.

For sweeping radius method, the enlargement of mean of tracking error with the value of 0.2101 millimeter and the standard deviation of tracking error with the value of 3.2663 millimeter are shown in Fig. 17 and Fig. 18 respectively. The safety margin of 0.1 to 1 millimeter is allowed at the beginning and near to the end of the semicircle object in order to avoid measuring the very high slope at those regions. The sweeping radius parameter of 1 millimeter is chosen for this contour tracking experiment. The total sample of 67 points was collected over 80 millimeter horizontal measuring distance.

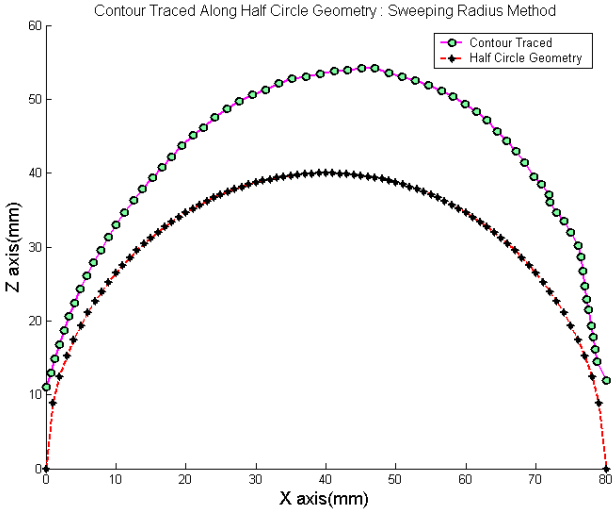


Fig. 17. Contour traced along half circle geometry with sweeping radius method.

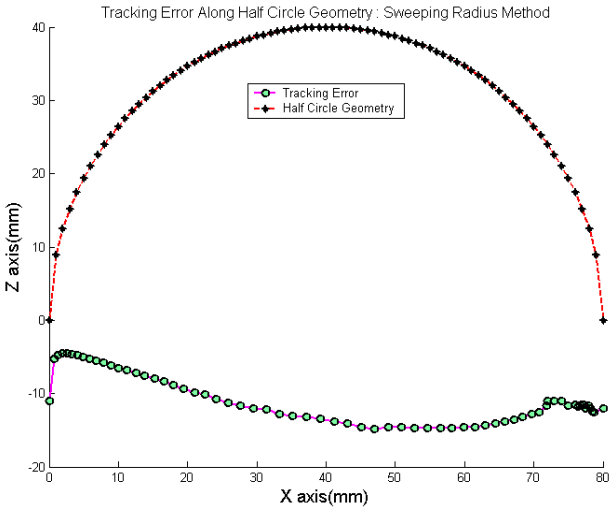


Fig. 18. Tracking error along half circle geometry with sweeping radius metohd.

6. Performance Evaluation

Fig. 19 summarizes all different methods for path traveling in order to evaluate their efficiency among all algorithms or methods implemented previously. The efficiency is measured with regard to the least tracking error standard deviation value and the shortest distance traveled. The best is assumed to be the least tracking error standard deviation value with the shortest sampling distance. In Fig. 19, the adapting gradient method follows path 1A to 2A, while the sweeping radius method starts from path 1B to 2B. The staircase method is the path that started from 1B to 4D.

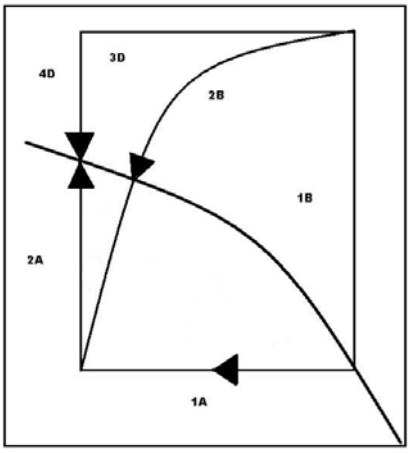


Fig. 19. Path comparison among three different contour following methods.

It is clearly seen in that the staircase method has the longest path followed by the adapting gradient method. The shortest distance is done by the sweeping radius method. With the same speed, it seems that the staircase method takes the longest time while sweeping radius is the fastest of all methods.
All the results are tabulated in Table 1. The adapting gradient method consumes medium teaching time at standard deviation value of 2.3085 millimeter, while the staircase method consumes the longest teaching time at standard deviation value of 1.8412 millimeter. The sweeping radius method is very efficient in term of shortest teaching path but its standard deviation value of 3.2663 is a bit high.

Criteria	Adapting Gradient	Staircase	Sweeping Radius
Mean of Error	-0.3773	3.4011	0.2101
Standard Deviation	2.3085	1.8412	3.2663
Path Length	Medium	Long	Shortest

Table. 1. Summaries of the results for three different contour following methods.

7. Conclusion

In this study, the performance evaluations of autonomous contour following task with three different algorithms have been performed for Adept SCARA robot. A prototype of smart tool integrated with sensor has been designed. It can be attached and reattached into robot gripper and interfaced through I/O pins of Adept robot controller for automated robot teaching operation. The algorithms developed were tested on a semicircle object of 40 millimeter radius. The semicircle object was selected because it exhibits the stringent test bed which provides the changing gradient gradually from steepest positive slope into zero slope of flat curve in the middle and finally to steepest negative slope. The adapting gradient method consumes medium teaching time at reasonable accuracy of standard deviation value of 2.3085 millimeter, while the staircase method consumes the longest teaching time at standard deviation value of 1.8412 millimeter. The sweeping radius method is very efficient in term of shortest teaching path but its standard deviation value of 3.2663 is a bit high. It can be concluded that the staircase method is the most accurate method, while the sweeping radius method has the shortest teaching path.
These tests exhibit the performance of algorithms used which prove its possibility to be applied in the real world application. For the future, automatic curve radius determination between straight line segments can be improved by integrating vision system for the automation of top view (X-Y coordinate) edge finding and path planning. The integration of vision system with the present study will improve the automation level of the project from two to three dimensional capabilities.

8. References

Adolfo, B.; Sadek, C.A.A. & Leszek, A.D. (2001). Predictive sensor guided robotics manipulators in automated welding cells. *Journal of Materials Processing Technology*, Vol. 109, No. 1-2, February 2001, 13-19, ISSN 0924-0136

Andersson, J.E. & Johansson, G. (2000). Robot control for wood carving operations. *Mechatronics*, Vol. 11, No. 4, June 2001, 475-490, ISSN 0957-4158

Awahara, M. & Taki, K. (1979). Tracking control for guiding electrodes along joints by pattern detection of welding groove. *Transactions of the Society of Instrument and Control Engineers*, Vol. 15, 492

Gopalakrishnan, B.; Tirunellayi, S. & Todkar, R. (2004). Design and development of an autonomous mobile smart vehicle: A mechatronics application. *Mechatronics*, Vol. 14, No. 5, 491-514, ISSN 0957-4158

Hanright, J. (1984). Selecting your first arc welding robot – a guide to equipment and features. *Welding Journal*, Vol. 1, 41-45

Hewit, J. (1996). Mechatronics design – the key to performance enhancement. *Robotics and Autonomous Systems*, 135–142, ISSN 0921-8890

Ikeuchi, K. & Suehiro, T. (1994). Towards an assembly plan from observation, Part I: Task recognition with polyhedral objects. *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 3, 368-385, ISSN 1042-296X

Inoue, K. (1979). Image processing for on-line detection of welding process (report 1): simple binary image processor and its application (welding physics, processes & instruments). *Transactions of JWRI*, Vol. 8, No. 2, 169-174

Mi, L. & Jia, Y.B. (2004). High precision contour tracking with joystick sensor. *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04)*, Vol. 1, 804-809, Sendai, Japan, September-October 2004

Oomen, G.L. & Verbeck, W.J.P.A. (1983). A real-time optical profile sensor for robot arc welding. *Proceedings of the 3rd International Conference on Robot Vision and Sensory Controls*, 659-668, Cambridge, USA, November 1983

Paul, R. (1979). Manipulator Cartesian path control. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 9, No. 11, 702-711, ISSN 0018-9472

Paul, R.P.C. (1972). *Modeling, trajectory calculation and servoing of a computer controlled arm*. Ph.D. Dissertation, Stanford University, CA., USA

Prabuwono, A.S.; Burhanuddin, M.A. & Samsi, M.S. (2008). Autonomous contour tracking using staircase method for industrial robot. *Proceeding of the 10th IEEE International Conference on Control, Automation, Robotics and Vision (ICARCV'08)*, 2272-2276, Hanoi, Vietnam, December 2008

Prabuwono, A.S. & Samsi, M.S. (2007). Development of adapting gradient method for contour tracking in industrial robot application. *Proceeding of the 10th IASTED International Conference on Intelligent Systems and Control (ISC'07)*, 592-068, Cambridge, USA, November 2007

Prabuwono, A.S.; Samsi, M.S.; Sulaiman, R. & Sundararajan, E. (2009). Contour following task with dual sensor logic algorithm for Adept Selective Compliant Assembly Robot arm robot. *Journal of Computer Science*, Vol. 5, No. 8, 557-563, ISSN 1549-3636

Prinze, F.B. & Gunnarson, K.T. (1984). Robotics seam tracking. *Interim Report*, CMU-RI-TR-84-10, Carnegie-Mellon University, Pittsburgh, USA

Rasol, Z.; Sanders, D.A. & Tewkesbury, G.E. (2001). New prototype knowledge based system to automate a robotics spot welding process. *Elektrika*, Vol. 4, 28-32

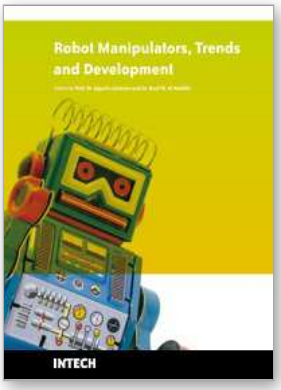
Samsi, M.S. & Nazim, M. (2005). Autonomous and intelligent contour tracking industrial robot. *Proceedings of International Conference on Mechatronics*, 78-86, Kuala Lumpur, Malaysia, May 2005

Suga, Y.; Takahara, K. & Ikeda, M. (1992). Recognition of weld line and automatic weld line tracking by welding robot with visual and arc voltage sensing system. *Journal of the Japan Society for Precision Engineering*, 1060-1065

Tomizuka, M.; Dornfield, D. & Purcelli, M. (1980). Applications of microcomputer to automatic weld quality control. *ASME Journal of Dynamics Systems, Measurement and Control*, 62-68

Yuehong, Y.; Hui, H. & Yanchun, X. (2004). Active tracking of unknown surface using force sensing and control technique for robot. *Sensors and Actuators: A Physical*, Vol. 112, No. 2-3, 313-319, ISSN 0924-4247

Zollner, R.; Rogalla, O.; Dillmann, R. & Zollner, M. (2002). Understanding users intention: programming fine manipulation tasks by demonstration. *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, 1114-1119, Laussane, Switzerland, September-October 2002



Robot Manipulators Trends and Development

Edited by Agustin Jimenez and Basil M Al Hadithi

ISBN 978-953-307-073-5
Hard cover, 666 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

This book presents the most recent research advances in robot manipulators. It offers a complete survey to the kinematic and dynamic modelling, simulation, computer vision, software engineering, optimization and design of control algorithms applied for robotic systems. It is devoted for a large scale of applications, such as manufacturing, manipulation, medicine and automation. Several control methods are included such as optimal, adaptive, robust, force, fuzzy and neural network control strategies. The trajectory planning is discussed in details for point-to-point and path motions control. The results in obtained in this book are expected to be of great interest for researchers, engineers, scientists and students, in engineering studies and industrial sectors related to robot modelling, design, control, and application. The book also details theoretical, mathematical and practical requirements for mathematicians and control engineers. It surveys recent techniques in modelling, computer simulation and implementation of advanced and intelligent controllers.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821