Online ISSN 1848-3380, Print ISSN 0005-1144
ATKAFF 54(1), 19–27(2013)

Edin Golubovic, Eray A. Baran, Asif Sabanovic

Contouring Controller for Precise Motion Control Systems

DOI 10.7305/automatika.54-1.304
UDK 681.511.4.037.09-53; 519.254
IFAC 4.7; 1.1

Original scientific paper

This paper discusses the trajectory generation algorithm, contour error construction method and finally the contour controller design. In the trajectory generation algorithm combination of elliptical Fourier descriptors (EFD) and time based spline approximation (TBSA) is used to generate position, velocity and acceleration references. Contour error is constructed using transformation of trajectory tracking errors. Transformation is computationally efficient and requires only reference velocity information. Contour controller is designed using sliding mode control. Experiments are performed on planar linear motion stage and significant contour error reduction is observed.

Key words: Contouring Controller, Precise Motion Control Systems, Time Based Spline Interpolation, Sliding Mode Control, Elliptical Fourier descriptors, Acceleration controller

Konturni regulator za precizne slijedne sustave. U članku se raspravlja o algoritmu za generiranje trajektorija, metodi za konstrukciju pogreške konture te o sintezi konturnog regulatora. U algoritum za generiranje trajektorija, korištena je kombinacija eliptičnih Fourierovih odrednika (EFD) i vremenske aproksimacije splajnovima (TBSA) za određivanje referentnih vrijednosti položaja, brzine i ubrzanja. Pogreška konture je konstruirana korištenjem transformirane pogreške slijeđenja trajektorije. Transformacija je računski efikasna i potrebna joj je samo informacija o referentnoj brzini. Konturni regulator je projektiran koristeći upravljanje u kliznim režimima. Provedeni su eksperimenti na linearnom slijednom sustavu i primijećena su znatna smanjenja pogreške konture.

Ključne riječi: konturni regulator, precizni slijedni sustavi, vremenska interpolacija splajnovima, upravljanje u kliznim režimima, eliptični Fourierovi odrednici, upravljanje akceleracijom

1 INTRODUCTION

Recently the miniaturization of parts and devices has become prevailing trend in many modern technologies. With the miniaturization, the machining requirements of the parts are becoming more strict and their contribution to the final product quality more important. Although there are many sources of error that affect the quality of machining, such as geometric errors of the structural elements and thermally induced errors, one of the main evaluation criteria for the part quality is it's dimensional accuracy [1]. When a multi-axis motion control system has a task of following the desired path, contouring errors arise. These errors are usually in the acceptable range in low speed applications, however they become significant in the high speed, high accuracy applications. Contouring errors have poor effect on the dimensional accuracy of the machined part, hence the need for implementation of proper contouring controller arises.

This topic has been intensively investigated by researchers and authors. Generally the research has taken two main paths on the investigation of methods to reduce the contouring error. First path is concerned with the reduction of the tracking errors in a single axis which indirectly reduces the contouring errors. The second path taken by the researchers is concerned with the estimation of the contouring errors and contouring controller design based upon them. Extensive literature review in this research area has been provided in [2], so following literature survey will reference the works that are not included in this article or that are published afterwards. There are many methods for improving tracking accuracy in individual axes for multi axis systems, among them work in [3] has gained the most attention because the method of designing of feedforward controller assures the zero phase error in a sense of frequency response. In general this method doesn't guaranty the reduction of the contouring errors. One of the earliest works towards the increasing of contouring performance, namely cross coupled controller (CCC), has been published in [4]. Many consecutive works explored the advantages of CCC with additional robust [5], self-tuning [6],

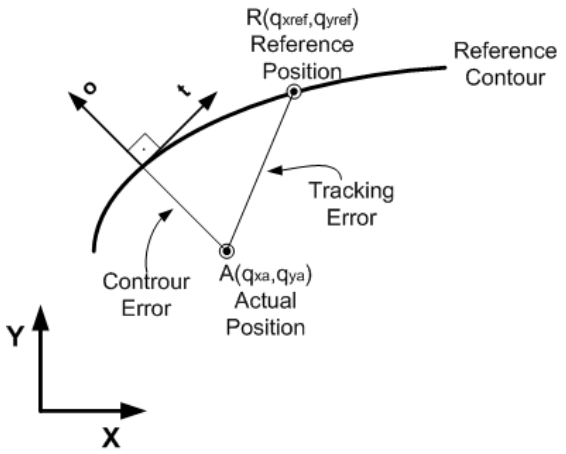


Fig. 1. Graphical description of controur error

position based loop [7], and fuzzy-logic [8] control methods. Besides CCC, some other control architectures were developed involving coordinate transformation. Namely tracking errors in individual axes are mapped to the task space frame attached to the desired contour. Among the works involving coordinate transformation there are three distinct approaches. First approach is proposed in [9]. To achieve good contouring performance, tracking error is decomposed into normal and tangential component and separate control laws are applied to dynamically decouple system for independent tracking and feed rate control. In [10] authors formulate the contour tracking problem in a task coordinate frame that is attached to the desired contour. The coordinate transformation requires the knowledge of the feedrate, velocity direction, and instantaneous curvature of the desired contour for proper operation. Control law assigns different dynamics to the motion in normal and tangential directions relative to the desired contour. Polar coordinate transformation is presented in [11] and the dynamics of radial orientation is derived. Contour error is approximated as the difference between true radius and the desired radius. Based on this linear relationship between the contour error and radial position, the dynamics of the contour error is obtained.

In this paper the contouring controller is designed using similar approach to the coordinate transformation. Transformation (Jacobian) matrix is formed using facts that velocity in the direction of the gradient of the constraint contour in the current operation point must be zero and the tangential velocity must be equal to some predefined velocity profile. Position and velocity tracking errors in the individual axes are mapped to the contour space through the transformation matrix. In order to form a transformation matrix no knowledge of explicit analytical expression of the curve is needed, rather the knowledge of the desired reference velocities in each axis are used. This information is calculated in the offline trajectory generation algorithm and details about that algorithm are also given. Acceleration controller in combination with disturbance observer is used to stabilize system and provide trajectory tracking control in the individual axes. Contouring controller, designed in sliding mode framework, is used as a corrective controller to reduce the contour errors. Proposed control strategy is verified experimentally on the planar linear actuator positioning stage. The rest of the paper is organized as follows: problem formulation is given in the next section. Section 3 contains details about trajectory generation, controller design details are given in Section 4, experimental results are given in the Section 5 and finally the conclusion and future work is given in Section 6.

2 PROBLEM FORMULATION

The problem of the design of contour error compensating controller is formulated taking into consideration the functional relationship (task formulation) for a planar motion required to follow some smooth contour with defined tangential velocity profile. Moreover the requirements of the system operation may be described by two simple functional relations

$$\phi(q_x, q_y) = 0 \tag{1}$$

$$v_{\phi\perp}(\dot{q}_x, \dot{q}_y) = v(t) \tag{2}$$

First relationship describes the contour to be tracked and the second one the requirement that velocity along the contour should be controlled to have defined time profile - usually constant. On the other hand the reference position, velocity and acceleration of the machine tool in Cartesian coordinates, necessary for velocity and position tracking, is defined using time based relations as following,

$$\mathbf{q_{ref}} = [q_x^{ref}(t) \quad q_y^{ref}(t)] \tag{3}$$

$$\dot{\mathbf{q}}_{\mathbf{ref}} = [\dot{q}_x^{ref}(t) \quad \dot{q}_y^{ref}(t)] \tag{4}$$

$$\ddot{\mathbf{q}}_{\mathbf{ref}} = [\ddot{q}_x^{ref}(t) \quad \ddot{q}_y^{ref}(t)] \tag{5}$$

where $\ddot{\mathbf{q}}_{\mathbf{ref}}$ is the reference acceleration vector, $\dot{\mathbf{q}}_{\mathbf{ref}}$ is the reference velocity vector and $\mathbf{q_{ref}}$ is the reference position vector satisfying previously defined constraint in Eq. 1.

Contour error is formally defined as the shortest distance between the reference and actual contour. From Fig.1 it can be seen that the contour error is the deviation from the reference path in gradient direction [12]. On the other hand, the deviation in the tangential direction causes the errors in the reference tangential velocity. Following these definitions, naturally, the task of the contour controller should be minimization of the gradient and tangential components of the contour error. This formulation is

valid, but it is better to look at it from slightly different way by taking the Eq.1 as a constraint in the task space then it is a fact of simple geometry to formulate that velocity in constrained direction - in the direction of the gradient of the constraint contour in the current operation point - must be zero and the tangential velocity must be equal to $v(t)$. For known reference contour $\phi(q_x, q_y) = 0$ gradient can be easily determined as:

$$\mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}^{\mathrm{T}} = \frac{1}{\|v(t)\|}\begin{bmatrix}\dfrac{\partial \phi_{ref}}{\partial q_x^{ref}} & \dfrac{\partial \phi_{ref}}{\partial q_y^{ref}}\end{bmatrix} \tag{6}$$

where $\|v(t)\|$ is the tangential velocity magnitude and formulated as

$$\|v(t)\| = \sqrt{\dot{q}_x^{ref}(t)^2 + \dot{q}_y^{ref}(t)^2} \tag{7}$$

The velocity in the constrained direction is then defined as

$$\mathbf{v}_\phi = \mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}^{\mathrm{T}} \dot{\mathbf{q}}_{\mathbf{ref}} \tag{8}$$

and the tangential velocity satisfies the dot product equation

$$\mathbf{v}_{\phi\perp} \cdot \mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\} = 0 \tag{9}$$

where

$$\mathbf{v}_{\phi\perp} = \mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}_{\perp}^{\mathrm{T}} \dot{\mathbf{q}}_{\mathbf{ref}} \tag{10}$$

Now task space contour tracking can be defined by the following expressions

$$\mathbf{v}_\phi = \mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}^{\mathrm{T}} \dot{\mathbf{q}}_{\mathbf{ref}} = 0 \tag{11}$$

$$\mathbf{v}_{\phi\perp} = \mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}_{\perp}^{\mathrm{T}} \dot{\mathbf{q}}_{\mathbf{ref}} = v(t) \tag{12}$$

or in simpler form

$$\begin{bmatrix}\mathbf{v}_\phi \\ \mathbf{v}_{\phi\perp}\end{bmatrix} = \mathbf{J}\begin{bmatrix}\dot{q}_x^{ref}(t) \\ \dot{q}_y^{ref}(t)\end{bmatrix} = \begin{bmatrix}0 \\ v(t)\end{bmatrix} \tag{13}$$

The Jacobian matrix is defined by the gradient of the constraint contour.

$$\mathbf{J} = \begin{bmatrix}\mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}^{\mathrm{T}} \\ \mathbf{grad}\{\phi_{\mathbf{ref}}(\mathbf{q_x^{ref}}, \mathbf{q_y^{ref}})\}_{\perp}^{\mathrm{T}}\end{bmatrix} \tag{14}$$

In this paper the Jacobian is obtained directly using $\dot{\mathbf{q}}_{\mathbf{ref}}$ vector which is readily available from trajectory generation. Defining the Jacobian matrix in this way avoids any complex calculations of gradient and no information about the geometry of the contour is needed except the already available one. Using these facts Jacobian matrix that is later used in the controller design is written in the following way

$$\mathbf{J} = \frac{1}{\|v(t)\|}\begin{bmatrix}\dot{q}_y^{ref}(t) & -\dot{q}_x^{ref}(t) \\ \dot{q}_x^{ref}(t) & \dot{q}_y^{ref}(t)\end{bmatrix} \tag{15}$$

3 TRAJECTORY GENERATION

In most of the industrial production processes, the pieces machined by robotic manipulators usually consist of geometries that can be represented by closed curves. In that sense, generation of the desired trajectories require an algorithm which can give the manipulator parametric representation of the closed contour to be tracked. Moreover, in order to have the ability to reproduce any arbitrary geometric shape it is decided to generate trajectories out of images of shapes provided to the overall system. This flexibility could be provided by using an algorithm based on computer vision. So, the overall goal of the trajectory generation algorithm was defined as generating the parametric representation from the image of a closed curve.

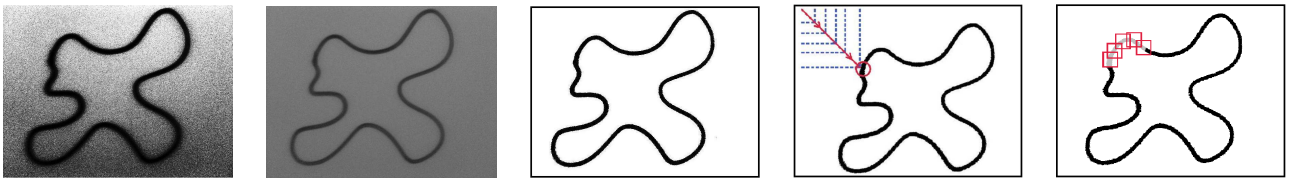
The trajectory generation algorithm mentioned above consisted of two main segments, namely, trajectory acquisition and trajectory parametrization segments. The details related to those segments as well as time based spline approximation algorithm are described in the following subsections.

3.1 Trajectory Acquisition

The acquisition of the reference trajectory supplied from an image is done by using a combination of several computer vision algorithms in series. The main target in that segment of the algorithm is to acquire the rough content of image in terms of locations of ordered data-points that can be used to regenerate the image. In that sense, a series of operations are carried out. First, the given image is smoothed with a Gaussian Kernel and transformed into binary domain. Following the smoothing, a fast scanning algorithm is executed to find the data point that first coincides with the bottom-right corner of an evolving square starting from the top-left corner of the image. Once the very first data-point of the image is reached, a tracking window is created. The tracking window is enforced to follow the boundaries of the existing shape via an optical flow algorithm. As the window moves along the trajectory, the data points are recorded to finalize the trajectory acquisition segment of the algorithm.

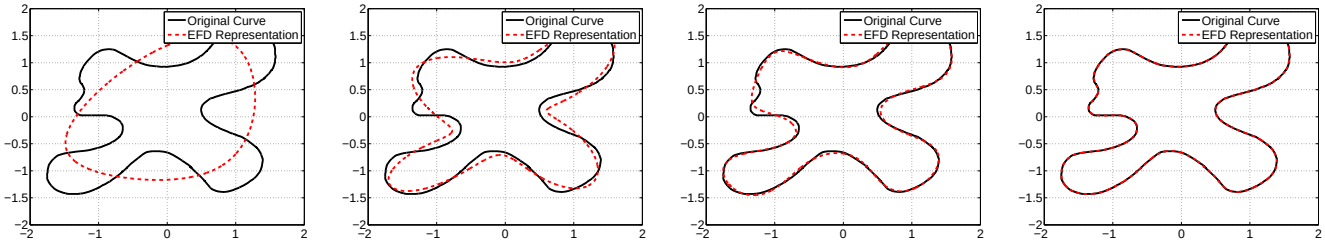
It is important to note here that the algorithms implemented in that section is not very sophisticated. The steps are just a combination of simple and well known computer vision algorithms organized in a functional way to bring more flexibility in the application level. Further studies on the robustness of this section is beyond the scope of this work. The overall procedure is summarized as follows while a depiction of these steps are given in Fig. 2;

- Image Smoothing & Binary Thresholding
- Initialization of First Point for Tracking
- Optical Flow & Coarse Trajectory Acquisition



(a) (b) (c) (d) (e)

Fig. 2. Trajectory Acquisition; (a) Original Image, (b) Smoothed Image, (c) Thresholded Binary Image, (d) Identification of Initial Point, (e) Optical Flow Tracking and Acquisition of Data Points



(a) (b) (c) (d)

Fig. 3. EFD representation, number of harmonics; (a) 2, (b) 5, (c) 10, (d) 30

3.2 Trajectory Curve Fitting

As mentioned in the preceding sections, the second segment of the trajectory generation algorithm is fitting a smooth and parametric curve to the closed contour obtained via the algorithm described in Trajectory Acquisition section. In the context of this study, we adopted the use of Elliptic Fourier Descriptors (EFDs) for the representation of given closed curvature.

In recent years, EFDs have been popularized and studied widely for the parametric representation of 2D closed curves [13], [14] with further extensions on 3D geometries [15]. The major ease brought by EFDs is the ability to represent the closed curve with a finite set of parameters which are obtained via an ordered combination of sinusoidal frequencies. Moreover, the advantage that the shape information is kept in the low frequency components makes EFDs further feasible for application.

The n-harmonic Elliptic Fourier Descriptor representation of any 2D curve can be given as,

$$\begin{aligned} x(t) &= a_0 + \sum_{k=1}^{n} \{a_k cos(kt) + b_k sin(kt)\} \\ y(t) &= c_0 + \sum_{k=1}^{n} \{c_k cos(kt) + d_k sin(kt)\} \end{aligned} \quad (16)$$

where, a_0 and c_0 is the center location of the curve and a_k, b_k, c_k and d_k $(k = 1, ..., n)$ are the Elliptic Fourier coefficients of the 2D curve up to n^{th} harmonics.

Given a set of M points from an image (i.e. locations of M points that lie on the curve boundary), one can make use of $2M$ number of data to calculate the $4n + 2$ coefficients that represent the corresponding curve. Usually in practice one cannot guarantee the equivalence of number of data points and the coefficients (i.e. usually we have $M \neq 2n + 1$. Hence, for practical purposes, least squares approximation to minimize a quadratic error between the estimated points and the actual points is considered which is what is done in the context of this study. The results obtained from EFD representation is depicted in Fig. 3 with different number of harmonics. Axes in Fig. 3 are given in millimeters. This figure is created only for demonstration purposes, however it is easy to change size of the feature by changing pixel scaling factor. This decision should be given based upon the production requirements.

3.3 Time Based Spline Approximation

To perform machining process of the geometrical features of the part to be machined have to be designed and technical drawing or image containing the part's geometrical description must be available. In this work we use the EFD method to obtain 2D representation of the curve as described in Eq. 16. In order to assure perfect replication of the part specified in technical drawing or image and achieve high quality machining, trajectory generation algorithm results should guaranty smooth position, velocity and acceleration transitions. Jerky motion and position overshoot at the corner points should be minimized or completely avoided in order to guaranty high quality ma-

chining. Trajectory generation algorithm satisfying these requirements is determined to be time based spline approximation [16], [17], [18]. Time based spline approximation is incorporating both time and geometrical position together resulting in motion that shows the smooth transitions in position, velocity and acceleration. As described in [16] the procedure for performing of the time based spline approximation is as follows;

- using the geometry points, the displacement to be covered in one sampling interval is determined for a given feedrate
- the x-y trajectory is divided into new coordinate points - separation distance is determined in the previous step
- time based spline approximation is executed using the coordinates of the divided points.

After the time based spline approximation is applied the position, velocity and acceleration references at a given sampling time can be found using following equations [16]:

$$\begin{aligned} q_{xn}^{ref}(t_n) &= \alpha_n t_n^3 + \beta_n t_n^2 + \gamma_n t_n + \delta_n \\ \dot{q}_{xn}^{ref}(t_n) &= 3\alpha_n t_n^2 + 2\beta_n t_n + \gamma_n \\ \ddot{q}_{xn}^{ref}(t_n) &= 6\alpha_n t_n + 2\beta_n \\ n &= 1...k \end{aligned} \tag{17}$$

where k is the total number of coordinate points. α_n, β_n, γ_n and δ_n are calculated polynomial coefficients and t_n is the sampling instant. The similar equations can be written for position, velocity and acceleration in y direction. When these references are supplied to the controller position command is supplied at a current sampling time, velocity reference command is supplied one sampling time ahead and acceleration reference command is supplied two sampling steps ahead. Time based spline approximation allows for the desired tangential velocity profile generation. In this work we use constant tangential velocity.

4 CONTROLLER DESIGN

This section contains the details about the controller design for positioning mechanism under consideration, the two axes x-y linear motion stage. The control strategy for motion stage has to be developed such that the fast motion is provided while the contouring error is minimized. Here we decided to combine an acceleration controller with disturbance observer as a basic trajectory tracking control in each of the axes and sliding mode control is used in the design of contour controller.

In order to design controller for trajectory tracking, plant model has to be developed first. For a two axes linear positioning stage driven by the brushless direct drive linear servomotors via current controlled amplifiers, the dynamic equation of motion can be described by the following equation

$$\mathbf{M_n}\ddot{\mathbf{q}} + \mathbf{F_l}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{K_{tn}}\mathbf{i} \tag{18}$$

where $\mathbf{M_n}$ denotes the the nominal mass matrix of the load, $\mathbf{K_{tn}}$ is the nominal motor force constant matrix, both of these matrices are square and diagonal, $\mathbf{F_l}(\mathbf{q}, \dot{\mathbf{q}}, t)$ represents the vector of the total disturbance forces acting on the system, $\mathbf{i}$ is the reference current vector supplied to the plant, $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$ are stages position, velocity and acceleration vectors respectively.

In the design of acceleration control the desired acceleration has to be defined and system must be disturbance compensated. Firstly the errors are constructed as following

$$\varepsilon_{\mathbf{pxy}} = \begin{bmatrix} \varepsilon_{px} \\ \varepsilon_{py} \end{bmatrix} = \begin{bmatrix} q_x^a(t) - q_x^{ref}(t) \\ q_y^a(t) - q_y^{ref}(t) \end{bmatrix} \tag{19}$$

$$\varepsilon_{\mathbf{vxy}} = \begin{bmatrix} \varepsilon_{vx} \\ \varepsilon_{vy} \end{bmatrix} = \begin{bmatrix} \dot{q}_x^a(t) - \dot{q}_x^{ref}(t) \\ \dot{q}_y^a(t) - \dot{q}_y^{ref}(t) \end{bmatrix} \tag{20}$$

The desired acceleration is now found to be

$$\ddot{\mathbf{q}}_{\mathbf{des}} = \ddot{\mathbf{q}}_{\mathbf{ref}} + \mathbf{K_v}\varepsilon_{\mathbf{pxy}} + \mathbf{K_p}\varepsilon_{\mathbf{pxy}} \tag{21}$$

where $\mathbf{K_p}$ and $\mathbf{K_v}$ are acceleration controller gain matrices, that are square and diagonal with positive diagonal entries, $\ddot{\mathbf{q}}_{\mathbf{ref}}$, $\dot{\mathbf{q}}_{\mathbf{ref}}$, $\mathbf{x}_{\mathbf{ref}}$ are reference acceleration, velocity and position vectors respectively, $\dot{\mathbf{q}}_{\mathbf{a}}$, $\mathbf{q}_{\mathbf{a}}$ are the measured or observed velocity and measured position vectors of the stage respectively. As noted previously, desired acceleration defined in the Eq. 21 could be achieved only if the disturbances acting on the plant are compensated. Since it is hard to model the disturbance forces, the estimation should be done. The estimation of the disturbance force can be done using first order low pass filter and reference input currents information and velocity feedback [19].

$$\widehat{\mathbf{i}}_{\mathbf{dis}} = \frac{g}{s+g}(\mathbf{i_{tot}} - \mathbf{M_n}\mathbf{K_{tn}^{-1}}s\dot{\mathbf{q}}_{\mathbf{a}}) \tag{22}$$

where g is the positive coefficient determining the cut-off frequency of the low pass filter and $\mathbf{i_{tot}}$ is the total supplied reference current vector. The resulting control action supplied by an acceleration controller can be calculated as,

$$\mathbf{i_a} = \mathbf{M_n}\mathbf{K_{tn}^{-1}}\ddot{\mathbf{q}}_{\mathbf{des}} + \widehat{\mathbf{i}}_{\mathbf{dis}} \tag{23}$$

Contour error compensating controller is designed based on the errors transformed through the expression in Eq.15.

$$\varepsilon_{\mathbf{p}\phi\phi\perp} = \begin{bmatrix} \varepsilon_{p\phi} \\ \varepsilon_{p\phi\perp} \end{bmatrix} = \mathbf{J}\varepsilon_{\mathbf{pxy}} \tag{24}$$

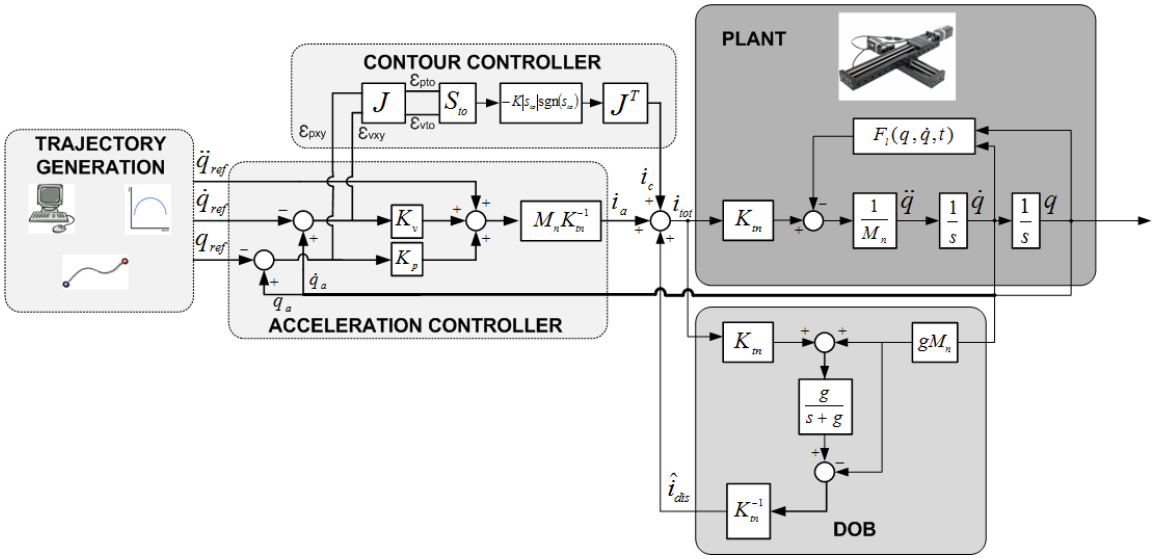


Fig. 4. Controller Block Diagram

Following the similar procedure, the projection of the errors in velocity can be obtained as

$$\varepsilon_{\mathbf{v}\phi\phi\perp} = \begin{bmatrix} \varepsilon_{v\phi} \\ \varepsilon_{v\phi\perp} \end{bmatrix} = \mathbf{J}\varepsilon_{\mathbf{vxy}} \tag{25}$$

Contour compensating controller is designed in sliding mode control framework. Firstly the sliding manifold that ensures the desired closed loop dynamics is chosen.

$$\mathbf{s}_{\phi\phi\perp} = \begin{bmatrix} s_{\phi} \\ s_{\phi\perp} \end{bmatrix} = \mathbf{C}\varepsilon_{\mathbf{p}\phi\phi\perp} + \varepsilon_{\mathbf{v}\phi\phi\perp} \tag{26}$$

where $\mathbf{C}$ is the controller gain matrix, that is square and diagonal with positive diagonal entries. Further the control is chosen such that Lyapunov stability criteria is satisfied [20].

$$\begin{bmatrix} i_{c\phi} \\ i_{c\phi\perp} \end{bmatrix} = -\begin{bmatrix} k_1 & k_2 \end{bmatrix} \begin{bmatrix} |s_{\phi}|\, sgn(s_{\phi}) \\ |s_{\phi\perp}|\, sgn(s_{\phi\perp}) \end{bmatrix} \tag{27}$$

Now the current components of this vector to be applied to the individual axes are found using the back transformation to the (x,y) coordinate frame as

$$\mathbf{i_c} = \begin{bmatrix} i_{cx} \\ i_{cy} \end{bmatrix} = \mathbf{J^T} \begin{bmatrix} i_{c\phi} \\ i_{c\phi\perp} \end{bmatrix} \tag{28}$$

Finally, combining the Eq. 23 and Eq. 28 the total current reference applied to the current amplifier that drives the motor can be written as:

$$\mathbf{i_{tot}} = \mathbf{i_a} + \mathbf{i_c} \tag{29}$$

Block diagram of the controller is shown in the Fig. 4

5 EXPERIMENTAL VERIFICATION

In order to test the performance of the controller and to justify the reliability of the trajectory generation algorithm series of experiments for different reference contours were performed. In this section only the most important results are presented. Experiment results clearly show the reduction of contour error when contour controller is used together with acceleration controller and disturbance observer. The experimental setup used is shown in the Fig. 4. It consists of the positioning stage, control system and personal computer. The positioning mechanism used to perform the experiments is MX80L series linear x-y stage by Parker Hannifin Corporation. This stage has incorporated optical position measurement which was used to obtain the necessary position feedback. Resolution of the measurement device is 0.1 μm. As a control system modular Dspace control system DS1005 was used. This system features personal computer running on Windows XP and control hardware featuring, a PowerPC 750GX processor running at $1GHz$, DA/AD cards, encoder signals processing cards and digital I/O cards. The control loop frequency was set to $10KHz$.

The following two experiments were performed in the following manner; Shapes are drawn using technical drawing software and then image is generated. Drawings are further analyzed and dimensions of the part to be processed are extracted. After the trajectory acquisition is done and EDF, an algorithm described in Section 3, is applied the time based spline approximation is performed in order to obtain the reference cartesian position, velocity and acceleration for both axes. References are transferred to the control system and controlled motion sequences are executed.

In both experiments the difference between the position responses with and without additional corrective contour controller are shown. Information about the control and plant parameters are shown in Table 1.

5.1 Experiment 1 - Circle

First experiment relates the performance of the control system with and without contour control error. Reference is circle of $150\mu m$ in radius, obtained using previously proposed trajectory generation scheme. The tangential velocity, for this experiment is $15\frac{mm}{s}$. The position response is shown in Fig. 5 for the case when only acceleration controller is used and when contour controller is used in addition with acceleration controller. In Fig. 6 the magnified plot of the section $A1$ is shown. Positioning response of system with added contour controller is the evidence of increased performance. Contour errors in gradient direction (ε_o) and tangential directions(ε_t) are shown in Fig. 7. Besides the graphical data, the statistical data related to the experiments also provide insight about the improvement of performance when proposed contour controller is used in the control loop. Mean absolute values calculated for the errors from Fig. 7 are $2.511\mu m$ and $1.335\mu m$ in gradient and tangential directions respectively when acceleration controller alone is used, and when contour controller is added this values are $0.272\mu m$ and $0.202\mu m$.

5.2 Experiment 2 - Arbitrary Shape

Second experiment is very similar to previous experiment only here the performance of the contour controller is shown using different reference contour. Reference arbitrary shape is of relatively bigger size. The tangential velocity for this experiment is $20\frac{mm}{s}$. The position response of the controller is shown in Fig. 8 for both cases with and without the corrective contour controller. In Fig. 9 part of the previous figure marked with $A2$ ellipse is magnified and shown. From this graph we can clearly see the response of system is increased with additional contour controller. Figure 10 shows the contour errors in gradient direction(ε_o) and tangential direction (ε_t). Graphically the difference is evident and statistical data is supplied for additional analysis; mean absolute values calculated for the errors from Fig. 10 are $1.896\mu m$ and $0.710\mu m$ in gradient and tangential directions respectively when acceleration controller alone is used, and when contour controller is added this values are $0.191\mu m$ and $0.234\mu m$.

6 CONCLUSION AND FUTURE WORK

Reduction of the contour error in an efficient way and with minimum computational requirement needed by controller is still an research issue open for discussion. In this work, contour controller is designed as the corrective

Table 1. Experiment Parameters

Name($Unit$)	Symbol	Value
Nominal Motor Constant (N/A)	K_{tnx}	4.3
Nominal Motor Constant (N/A)	K_{tny}	4.3
Nominal Stage Mass (kg)	M_{nx}	0.3
Nominal Stage Mass (kg)	M_{ny}	0.9
Control gain	K_{px}	600
Control gain	K_{vx}	500
Control gain	K_{py}	600
Control gain	K_{vy}	500
Control gain	C_{11}	700
Control gain	C_{22}	700
Control gain	k_1	10
Control gain	k_2	10
DOB filter gain	g	1000

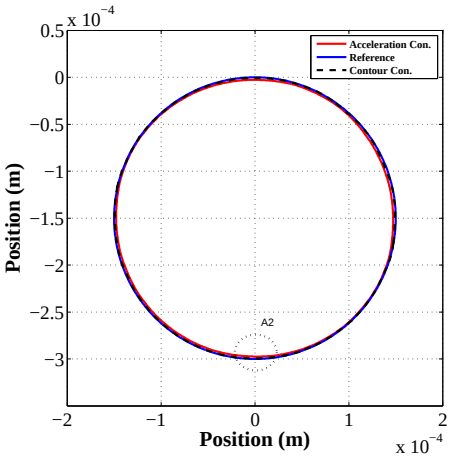


Fig. 5. Circular Contour Position Response

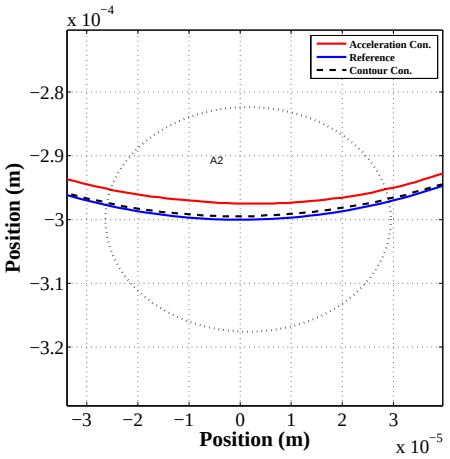


Fig. 6. Magnified plot of A2

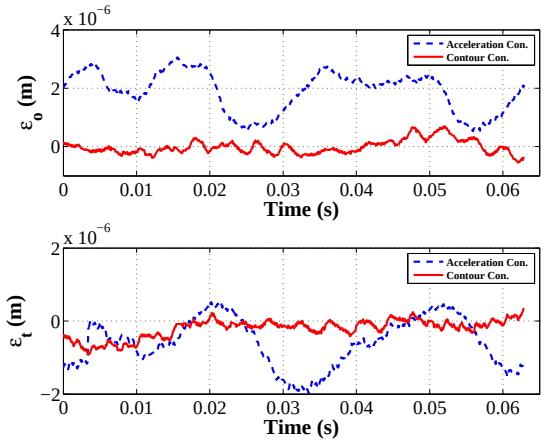


Fig. 7. Orthogonal and Tangential errors

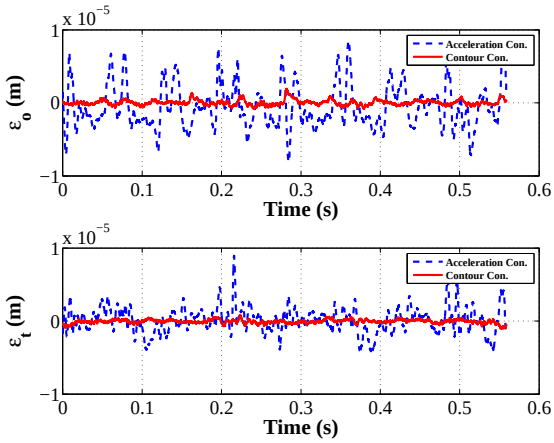


Fig. 10. Orthogonal and Tangential errors

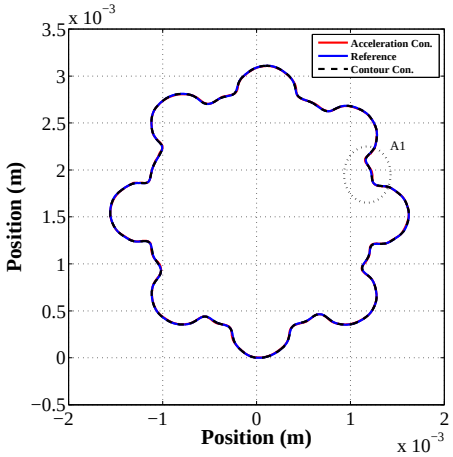


Fig. 8. Arbitrary Contour Position Response

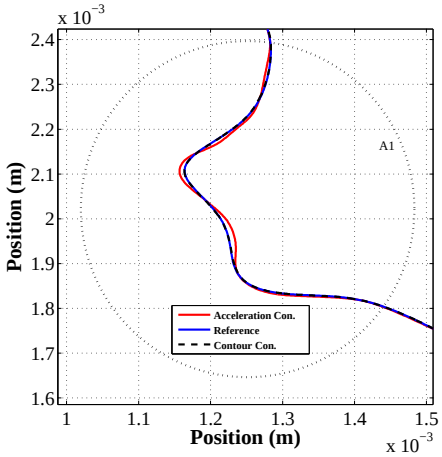


Fig. 9. Magnified plot of A1

controller to be used together with trajectory tracking controller. The benefits of the presented scheme lay mostly in two facts; construction of the contour error is done in a simple way using already available information from trajectory generation which proved to be computationally very efficient method and the design of the controller is fairly efficient because it guaranies the stability and robustness of the system. The presented method is verified experimentally for circular reference of $300\mu m$ in diameter and relatively larger other arbitrary reference. Experimentally it has been proved that the reduction of contour error is achieved. Future work will be based on application of this method to the manipulators with three dimensional workspace.

ACKNOWLEDGMENT

Authors acknowledge TÜBITAK Project number 111-M-359 and Yousef Jameel Scholarship Fund for partial financial support.

REFERENCES

[1] M. Donmez, D. Blomquist, R. Hocken, C. Liu, and M. Barash, "A general methodology for machine tool accuracy enhancement by error compensation," *Precision Engineering*, vol. 8, no. 4, pp. 187 – 196, 1986.

[2] R. Ramesh, M. Mannan, and A. Poo, "Tracking and contour error control in cnc servo systems," *International Journal of Machine Tools and Manufacture*, vol. 45, no. 3, pp. 301 – 326, 2005.

[3] M. Tomizuka, "Zero phase error tracking algorithm for digital control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 109, no. 1, pp. 65–68, 1987.

[4] Y. Koren, "Cross-coupled biaxial computer control for manufacturing systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 102, no. 4, pp. 265–272, 1980.

[5] S.-S. Yeh and P.-L. Hsu, "Theory and applications of the robust cross-coupled control design," *Journal of Dynamic Systems, Measurement, and Control*, vol. 121, no. 3, pp. 524–530, 1999.

[6] M.-T. Yan, M.-H. Lee, and P.-L. Yen, "Theory and application of a combined self-tuning adaptive control and cross-coupling control in a retrofit milling machine," *Mechatronics*, vol. 15, no. 2, pp. 193 – 211, 2005.

[7] J. Yang, D. Zhang, and Z. Li, "Position loop-based cross-coupled control for high-speed machining," *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, pp. 4285 –4290, june 2008.

[8] J.-H. Chin, Y.-M. Cheng, and J.-H. Lin, "Improving contour accuracy by fuzzy-logic enhanced cross-coupled precompensation method," *Robotics and Computer-Integrated Manufacturing*, vol. 20, no. 1, pp. 65 – 76, 2004.

[9] J.-Y. Yen, H.-C. Ho, and S.-S. Lu, "A decoupled path-following control algorithm based upon the decomposed trajectory error," *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, vol. 3, pp. 3189 –3194 vol.3, 1998.

[10] G.-C. Chiu and M. Tomizuka, "Contouring control of machine tool feed drive systems: a task coordinate frame approach," *Control Systems Technology, IEEE Transactions on*, vol. 9, pp. 130 –139, jan 2001.

[11] S. L. Chen, H. L. Liu, and S. C. Ting, "Contouring control of biaxial systems based on polar coordinates," *Mechatronics, IEEE/ASME Transactions on*, vol. 7, pp. 329 – 345, sep 2002.

[12] Y. Koren, *Computer Control of Manufacturing Systems*. New York, USA: McGraw-Hill, Inc.., 1983.

[13] F. P. Kuhl and C. R. Giardina, "Elliptic fourier features of a closed contour," *Computer Graphics and Image Processing*, vol. 18, no. 3, pp. 236 – 258, 1982.

[14] C.-S. Lin and C.-L. Hwang, "New forms of shape invariants from elliptic fourier descriptors," *Pattern Recognition*, vol. 20, no. 5, pp. 535 – 545, 1987.

[15] M.-F. Wu and H.-T. Sheu, "Representation of 3d surfaces by two-variable fourier descriptors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, pp. 858 –863, aug 1998.

[16] T. Inoue, M. Morimoto, and K. Ohnishi, "A preview controller with time based spline approximation for multi-axis manipulator," *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on*, pp. 247 – 251, nov. 2006.

[17] J.Miyata, T.Murakami, and K. Ohnishi, "Trajectory tracking control of mobile robot by time-based spline approach," *Electrical Engineering in Japan*, vol. 151, no. 4, pp. 65–71, 2005.

[18] J. Miyata and T. Murakami, "Trajectory tracking control of mobile robot by fluid model," *IEEJ Transactions on Industry Applications*, vol. 127, no. 3, pp. 315 – 321, 2007.

[19] A. Sabanovic and K. Ohnishi, *Motion Control Systems*. Singapore: John Wiley and Sons, 2011.

[20] J. G. Vadim Ivanovich Utkin and J. Shi, *Sliding Mode Control in Electromechanical Systems*. USA: Taylor and Francis, 1999.

Edin Golubovic received the B.S. degree in Electrical-Electronics engineering from Fatih University, Istanbul, Turkey, in 2009, and the M.S. degree in Mechatronics from Sabanci University, Istanbul, Turkey in 2011. He is currently working toward the Ph.D. degree in Mechatronics at Sabanci University, Istanbul, Turkey. His research interests include high precision motion control systems, power electronics and electrical machines control and mechatronics systems design and integration.

Eray A. Baran received the B.S. and M.Sc. degrees in Mechatronics Engineering from Sabanci University, Istanbul, Turkey, in 2008 and 2010 respectively. He is currently working toward the Ph.D. degree in Mechatronics Engineering program of Sabanci University. His research interests include motion control systems, bilateral control, telemanipulation and control of redundant multibody systems.

Asif Sabanovic received B.S '70, M.S. '75, and Dr. Sci. '79 degrees in Electrical Engineering all from University of Sarajevo, Bosnia and Herzegovina. He is with Sabanci University, Istanbul, Turkey. Previously he had been with University of Sarajevo; Visiting Professor at Caltech, USA, Keio University, Japan and Yamaguchi University, Japan and Head of CAD/CAM and Robotics Department at Tubitak - MAM, Turkey. His fields of interest include power electronics, sliding mode control, motion control and mechatronics.

AUTHORS' ADDRESSES

Edin Golubovic
Eray A. Baran
Asif Sabanovic, Ph.D.
Sabanci University
Istanbul, Turkey
email: edin@sabanciuniv.edu, eraybaran@sabanciuniv.edu, asif@sabanciuniv.edu

Received: 2012-06-26
Accepted: 2012-10-15