Newton-Based Contour Error Estimation and Robust Cross-Coupling Control for High-Precision Fast Contouring

Azad Ghaffari and A. Galip Ulsoy

Abstract— **High-performance position control algorithms for multi-axis servo-systems, due to presence of asymmetric dynamics and disturbance, do not guarantee high-precision contouring. Cross-Coupling Control (CCC) conventionally uses a static Contour Error Estimate (CEE) to reduce the shortest distance between the reference map and actual position, known as contour error. However, reliability of the static CEE deteriorates for high-speed reference feeds and also for sharp corners and deep curves. We propose a dynamic CEE using a Newton-based update law to obtain a precise CEE for fast and highly-curved contours. The Newton-based algorithm uses an estimate of the contour error curvature in order to eliminate the convergence dependence on the contour shape. The proposed CCC design includes one PID controller per axis, and combines the proposed Newton-based CEE with Integral Sliding Mode Control (ISMC) which is well-known for its capability in dealing with parameter uncertainty and external disturbances. The proposed ISMC performance is enhanced with an adaptive disturbance estimate. The proposed CCC algorithm reduces the time-averaged contour error (TACE) at least by an order of magnitude in comparison to conventional CCC algorithms. Various simulation results are presented to highlight the significant improvement achieved by the proposed algorithm.**

I. Introduction

The contour error in multidimensional contouring is defined as the shortest distance between the actual position and the reference contour. Even high-performance position control algorithms, without simultaneous treatment of the contour error, are vulnerable to asymmetric dynamics of the system and external disturbances [7], [10]. In response to this contouring problem, Koren [12] proposed the idea of the Cross-Coupling Control (CCC) for linear contours which was developed and enhanced by many other researchers [2], [3], [5], [9], [11], [13], [16], [18], [19], [23], [24].

The CCC algorithm differs from position synchronization algorithms which are used when a number of separate servo-systems (or other moving structures) are required to follow the same tasks simultaneously [1], [4], [21]. Cross-coupling control algorithms reduce the contour error in a standalone servo-system, while position synchronization algorithms maintain the same level of position precision among a group of servo-systems. A high-precision position synchronization algorithm may not guarantee high-precision contouring or vice versa.

A successful CCC algorithm maintains an appropriate balance between position control and contour error control. In other words, regardless of the designer's knowledge about system dynamics or the type of implemented position control algorithms, when a multi-axis system faces disturbances, particularly various forms of friction, and/or fast contours with complex curves, one can not guarantee high-performance contouring without compensating for the effect of contour error. Even a multivariable control algorithm does not improve contouring precision when a CCC algorithm is not used. In high-precision contouring it is essential to maintain the actual contour as close as possible to the reference contour. Position control reduces tangential position error and the CCC algorithm eliminates perpendicular position error with respect to the reference contour.

Obtaining a close estimate of the contour error is crucial prior to CCC design. The Contour Error Estimate (CEE) is a key part of each CCC algorithm [5], [6], [11], [15], [25], [26]. Current CEE algorithms rely on algebraic methods which use the reference map information and actual position and velocity in order to calculate the estimate of the contour error in a single step. However, such static CEE algorithms are not accurate enough for high feed rate reference signals or for contour maps with sharp corners and deep curves.

A static CEE uses linear or circular approximation of the contour at each time step in order to provide an estimation of the contour error. The static CEE methods are based upon the Taylor series expansion with different levels of precision. Thus, when the current position error suddenly increases or when the reference contour is highly-curved, then the reliability of the static CEE algorithm deteriorates dramatically. A dynamic CEE algorithm is presented here to overcome the limitations imposed by static CEE algorithms.

Assume that the contour map is smooth and the current position is located close to the reference contour. One can then approximate the distance between the current position and the reference contour as a quadratic function with a local minimum point associated with the contour error and use an extremum seeking algorithm to find the location and value of the contour error. Gradient-descent algorithms find the local extremum point using the actual or estimated value of the cost function gradient. However, gradient-descent depends on the cost function shape. The contour error in a Computer Numerically Controlled (CNC) system generates a wide range of cost functions with vastly different curvature which changes from one time step to the next. Newton-based extremum seeking is more sophisticated and removes the closed-loop performance dependence on the cost function shape [8], [17]. In this paper a dynamic CEE using the Newton-based update law is presented to achieve high-precision contouring for high-speed operation with highly-

The authors are with Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48107-2125, USA, aghaffar@umich.edu and ulsoy@umich.edu.

curved contour maps. Since the reference contour is known, analytically or numerically, one can reconstruct the gradient and the Hessian of the contour error at each time step.

The proposed CEE algorithm is multivariable by its nature and can be easily applied to multi-axis contouring structures without further modification. In order to implement the Newton-based CEE algorithm one needs to have access to the first and second order derivatives of the reference vector. We introduce an estimate of the Hessian of the contour error using only the first order derivative of the reference vector. The proposed Hessian estimate reduces the numerical effort of the CEE algorithm and improves its convergence time.

The sliding mode class of controllers are well-known for their ability in compensating for the effect of dynamic uncertainty and external disturbances. In addition to the proposed CEE and CCC algorithms an Integral Sliding Mode Control (ISMC) is used for position control of each axis [14], [20], [22]. A saturation actuator is used instead of the sign function in order to eliminate the chattering phenomenon. The saturation actuator degrades the tracking performance of the sliding mode control. Hence, integral action is added to the sliding surface in order to achieve full tracking. Moreover, adaptive disturbance estimation is included in the integrated design in order to enhance closed-loop performance.

The remainder of the paper is organized as follows. Section II presents the proposed CEE and CCC algorithms. Section III is dedicated to the ISMC design and adaptive disturbance estimation. Section IV provides various numerical simulations. The last section includes concluding remarks.

II. Newton-Based Contour Error Estimate and Cross-Coupling Control

Dynamic equations of axis i are given as

$$\frac{\mathrm{d}}{\mathrm{d}t}P_i = V_i \tag{1}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}V_i = -\frac{1}{\tau_i}V_i + \frac{k_i}{\tau_i}T_i + \frac{k_i}{\tau_i}D_i, \tag{2}$$

where T_i is torque command, D_i is disturbance input including load, V_i is velocity, P_i is position, k_i is DC gain, and τ_i is time constant for axis i, where $i = 1, 2, \cdots, n$. Position vector of the servo-system is defined as $P(t) = [P_1(t)\ P_2(t), \cdots,\ P_n(t)]^T$.

Assume the reference contour map is parametrized by θ

$$R(\theta) = [R_1(\theta)\ R_2(\theta)\ \cdots\ R_n(\theta)]^T, \tag{3}$$

where θ is a real number and indicates the current reference position. Current position is $P(mT_s)$, where T_s is the sampling period of the CNC and m is the time-step number. The contour error, E^*, is the shortest distance from the current position to the reference contour. Let E^* occur at θ^*, i.e., $E^* = R(\theta^*) - P(mT_s)$. Denote by $\hat{\theta}$ the estimate of θ^*. Define a cost function as

$$J(\hat{\theta}) = \frac{1}{2} \parallel E(\hat{\theta}) \parallel^2, \quad E(\hat{\theta}) = R(\hat{\theta}) - P(mT_s), \tag{4}$$

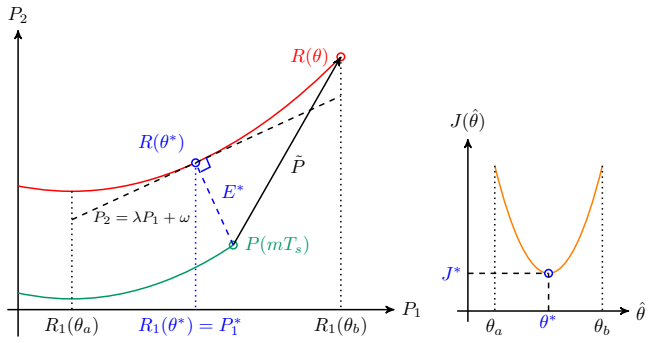

Fig. 1: (left) Contour error (right) cost function map

where $\| \cdot \|$ represents the Euclidean norm and E is distance from current position to the reference contour. The minimum value of the cost function is associated with the values E^* and θ^* that minimize $J(\hat{\theta})$. A schematic of the contour error for a 2-axis system is illustrated in Fig. 1.

Remark 1: Assume that the current position is close enough to the reference map and $R(\theta)$ has smooth curvature everywhere. Then, at each time step, one can replace the contour with its tangent estimate, $P_2 = \lambda P_1 + \omega$, where λ and ω are unknown constant parameters varying from one time step to the next. For the $P_1 - P_2$ planar contour shown in Fig. 1, the quadratic approximate of the cost function is given as

$$J' = J^* + \frac{1}{2}(1+\lambda)^2 (P_1 - P_1^*)^2 \tag{5}$$

with a minimum value of

$$J^* = \left(\frac{P_2(mT_s) - \lambda P_1(mT_s) - \omega}{\lambda^2 + 1} \right)^2 \tag{6}$$

at

$$P_1^* = \frac{P_1(mT_s) + \lambda \left(P_2(mT_s) - \omega \right)}{\lambda^2 + 1}. \tag{7}$$

Since $R_1(\theta^*) = P_1^*$, one can find θ^* from P_1^*.

Since the tangent estimate of the contour at each time step requires prior knowledge of the optimal parameter, θ^*, the explicit approximation given by (6) and (7) cannot be implemented. Static CEE algorithms use the actual position and velocity measurements and reference contour information to approximate the tangent line in order to calculate the contour error from (6) and (7) [5], [6], [11], [15], [25], [26]. However, for high speed or highly curved contouring applications static CEE algorithms do not yield precise contour error estimation.

Here a dynamic CEE is proposed to estimate θ^* and J^* in order to increase both contouring precision and speed. A plot of the cost function, $J(\hat{\theta})$, for a sufficiently smooth contour at time step m is shown in Fig. 1. The contour reference map and position measurements are available from all axes. Hence, a model-based extremum seeking algorithm, e.g., gradient-descent or the Newton-based algorithm, can be used to find J^*. Moreover, the Newton-based algorithm alleviates convergence dependence on contour shape and maintains a uniform transient over a wide range of contour shapes and

speeds. Hence, here the Newton-based extremum seeking algorithm is used:

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\theta} = -k'\frac{g}{h}, \tag{8}$$

where g is the gradient and h is the Hessian of the cost function with respect to $\hat{\theta}$. The Newton-based CEE at each time step finds θ^* and J^*. One can calculate from (4) the associated contour error using $\hat{\theta} = \theta^*$.

The Newton-based algorithm requires calculation of the gradient and the Hessian. It is possible to calculate the gradient of $J(\hat{\theta})$, analytically or numerically

$$\begin{aligned} g=&\frac{\partial}{\partial\hat{\theta}}J(\hat{\theta}) \\ =&E^T\frac{\partial}{\partial\hat{\theta}}R(\hat{\theta}). \end{aligned} \tag{9}$$

Also, the second order derivative of $J(\hat{\theta})$ with respect to θ, the Hessian, is calculated as

$$\begin{aligned} h=&\frac{\partial^2}{\partial\hat{\theta}^2}J(\hat{\theta}) \\ =&E^T\frac{\partial^2}{\partial\hat{\theta}^2}R(\hat{\theta})+\parallel\frac{\partial}{\partial\hat{\theta}}R(\hat{\theta})\parallel^2. \end{aligned} \tag{10}$$

Assume that the contour distance, E, is reasonably small and the contour map has smooth curves, then one can approximate (10) as

$$h \approx \parallel\frac{\partial}{\partial\hat{\theta}}R(\hat{\theta})\parallel^2 \tag{11}$$

which is always positive semi-definite and improves the stability margin of the Newton-based CEE. Moreover, the computational burden for (11) is less than (10).

A schematic of the proposed Newton-based CEE is shown in Fig. 2. A linear estimator is introduced to remove undesired high frequency oscillations from the parameter update law to achieve smooth transients. The estimator bandwidth, c', is designed with respect to the CNC sampling time, T_s, and reference contour speed. Large values of c' reduce the stability margin of the Newton-based CEE algorithm. The adaptation gain, k', needs to be designed such that the Newton-based CEE is sufficiently faster than the highest reference speed.

Remark 2: The estimate of θ^* is defined as $\hat{\theta}$. Denote $\tilde{\theta} = \hat{\theta} - \theta^*$ as the parameter error. Without loss of generality and for simplicity assume $P_1 = R_1(\hat{\theta})$, which its linear approximation can be written as $P_1 = P_1^* + \mu\tilde{\theta} + \mathcal{O}(\tilde{\theta}^2)$, where $\mathcal{O}(\cdot)$ stands for order of the terms. Replacing P_1 in (5) and truncating the higher order terms gives

$$J''(\hat{\theta}) = J^* + \frac{1}{2}\mu^2(1+\lambda^2)\left(\hat{\theta}-\theta^*\right)^2. \tag{12}$$

From the approximate cost function (12) the gradient equals $g = \mu^2(1+\lambda^2)\tilde{\theta}$. Also, the second order derivative of $J''(\hat{\theta})$ with respect to $\hat{\theta}$, the Hessian, equals $h = \mu^2(1+\lambda^2)$. Then the Newton-based update law (8) gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\theta} = -k'\tilde{\theta} \tag{13}$$

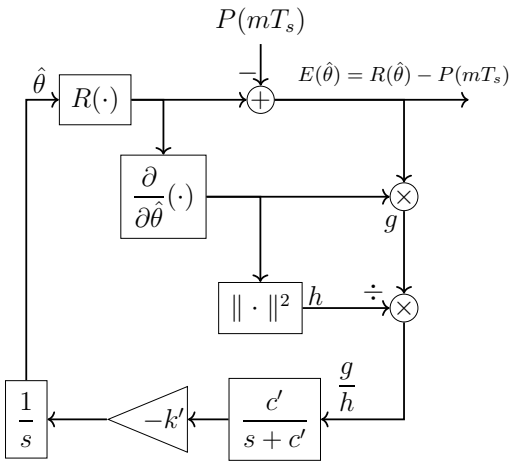

Fig. 2: Proposed Newton-based CEE

which indicates that the convergence rate and transient performance of the proposed Newton-based CEE is determined only by the feedback gain, k'.

Expansion of the stability analysis of the Newton-based CEE algorithm to multi-axis servo-systems becomes more complex. However, it is possible to investigate the stability of the algorithm intuitively. Substituting (9) and (11) into the parameter update law (8) gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\theta} = -k'\frac{E^T(\hat{\theta})U(\hat{\theta})}{\parallel\frac{\partial}{\partial\hat{\theta}}R(\hat{\theta})\parallel}, \quad U(\hat{\theta}) = \left(\frac{\frac{\partial}{\partial\hat{\theta}}R(\hat{\theta})}{\parallel\frac{\partial}{\partial\hat{\theta}}R(\hat{\theta})\parallel}\right), \tag{14}$$

where $U(\hat{\theta})$ is the unit tangent vector of the contour map at $\theta = \hat{\theta}$. Inner product of E and U equals E^TU which indicates the projection of contour distance, E, along the tangent vector. The Newton-based law (14) updates $\hat{\theta}$ in the opposite direction of E^TU until $\hat{\theta}$ reaches θ^* where $E^TU = 0$ which proves that the contour distance is aligned with the contour error, E^*.

Remark 3: The result of the Newton-based CEE is local, meaning that one cannot guarantee global convergence. In special cases, where the reference contour has sharp corners or deep curves, as shown in Fig. 3, if the actual position is located on the dotted line at the top left-hand side contour or at the center of the dotted circle on the top right-hand side contour, the cost function will have multiple extremum points. Such cases, however, only happen at certain time steps and due to the effect of the CCC algorithm the next position will be off the dotted line. The same reasoning applies to the right-hand side scenario. Moreover, at each step one can initialize $\hat{\theta}$ using the final estimate of θ^* from the last step, θ^*_{m-1}. Without loss of generality, one can assume θ is constantly increasing with time, then θ^*_{m-1} is close to θ_a. Hence, the proposed Newton-based CEE converges to a local minimum closer to θ_a which gives the actual contour error. Also, one can easily avoid these extreme scenarios by specifying appropriate contours.

Unity contour error defines the normalized impact of

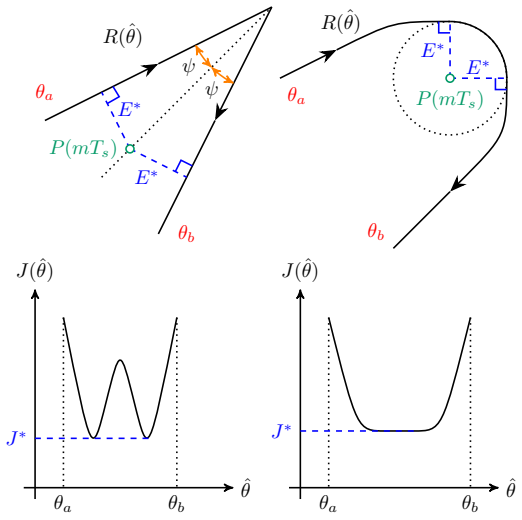

Fig. 3: Cost function with multiple extremum points for (left) sharp corners and (right) circular deep curves

contour error along each axis

$$\epsilon = \frac{E^*}{\parallel E^* \parallel}, \tag{15}$$

where $\epsilon = [\epsilon_1 \ \epsilon_2 \ \cdots \ \epsilon_n]^T$. One can drive the contour error along each axis, ϵ_i, toward zero using a PID controller along the same axis. Due to contour error normalization, all the PIDs may have the same gains.

III. Robust Position Control and Disturbance Compensation

Consider a robust position controller using Integral Sliding Mode Control (ISMC) to regulate each axis to its desired position. Define error variables as

$$\tilde{P}_i = R_i - P_i \tag{16}$$

$$\tilde{V}_i = \frac{\mathrm{d}}{\mathrm{d}t} R_i - V_i. \tag{17}$$

One can obtain actual error dynamics using (1), (2), (16), and (17)

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{P}_i = \tilde{V}_i \tag{18}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{V}_i = -\frac{\tilde{V}_i}{\tau_i} - \frac{k_i}{\tau_i} T_i + \frac{\mathrm{d}^2}{\mathrm{d}t^2} R_i + \frac{\mathrm{d}}{\tau_i \mathrm{d}t} R_i - \frac{k_i}{\tau_i} D_i. \tag{19}$$

A basic sliding mode scheme uses a hard switch to overcome the effect of uncertainty and external disturbance. However, when the switch equation is very close to zero the conventional sliding mode control (SMC) shows high frequency oscillations, known as chattering. A common way to alleviate chattering is to replace the hard switch with a smooth counterpart, e.g., saturation or sigmoid functions. When using a smooth version of the hard switch one cannot achieve perfect tracking anymore. A common way to eliminate the steady state error is by using integral action in the switching equation. The integral action is defined as

$$\frac{\mathrm{d}}{\mathrm{d}t} q_i = -b_i q_i + \mu_i \mathrm{sat}\left(\frac{\sigma_i}{\mu_i}\right), \tag{20}$$

where $\mathrm{sat}(u) = u$ for $|u| \leq 1$ and $\mathrm{sat}(u) = \mathrm{sign}(u)$ for $|u| > 1$. Parameters a_i, b_i, and μ_i are all positive real numbers and the switching equation is given as

$$\sigma_i = \tilde{V}_i + a_i \tilde{P}_i + b_i q_i. \tag{21}$$

Here the idea of equivalent control is used to design the ISMC, the detailed design and stability analysis of which can be achieved using Lyapunov methods. The main result is summarized in the following theorem.

Theorem 1: Assume a multi-axis servo-system is given as (1) and (2), and the external disturbances acting on the system are upper-bounded as $|D_i| \leq M_i$, where M_i is a positive real number. Moreover, the reference inputs, $R_i(\theta)$, and their first time derivatives, $\mathrm{d}R_i(\theta)/\mathrm{d}t$, are smooth and differentiable. Control signals of the form:

$$T_i' = \left(\frac{(a_i + b_i)\tau_i - 1}{k_i}\right)\tilde{V}_i + \left(\frac{a_i b_i \tau_i}{k_i}\right)\tilde{P}_i + \\ + \frac{\tau_i}{k_i}\frac{\mathrm{d}^2}{\mathrm{d}t^2} R_i + \frac{1}{k_i}\frac{\mathrm{d}}{\mathrm{d}t} R_i + \kappa_i \mu_i \mathrm{sat}\left(\frac{\sigma_i}{\mu_i}\right). \tag{22}$$

govern the system asymptotically to the desired reference positions for positive real numbers a_i, b_i and μ_i, and switch parameter satisfying

$$\kappa_i \geq \frac{M_i}{\mu_i} + \frac{b_i \tau_i}{k_i}. \tag{23}$$

■

The proposed control design of (22) ensures closed-loop stability and perfect position tracking for an upper-bounded disturbance signal with unknown dynamics. The disturbance dynamics is not considered. Although, one can use the estimate of the disturbance signal to compensate for the unknown dynamics of the disturbance.

One can introduce the following estimator to reconstruct a velocity derivative using velocity measurement:

$$\frac{\mathrm{d}}{\mathrm{d}t}\Omega_i = -c_i \Omega_i - c_i V_i. \tag{24}$$

One can then create an initial estimate of the disturbance as follows

$$\xi_i = \frac{\tau_i}{k_i}(\Omega_i + V_i) + \frac{V_i}{k_i} - T_i. \tag{25}$$

An additional estimation step which helps to remove high frequency oscillations from ξ_i is given as

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta = -d_i \Delta_i + d_i \xi_i. \tag{26}$$

Finally, a single-axis view of the proposed integrated CCC algorithm is shown in Fig. 4. The next section provides detailed comparisons between the proposed CEE and CCC algorithms with conventional variable gain CCC shown in Fig. 5. For a fair comparison identical ISMC and disturbance compensation is used in both algorithms.

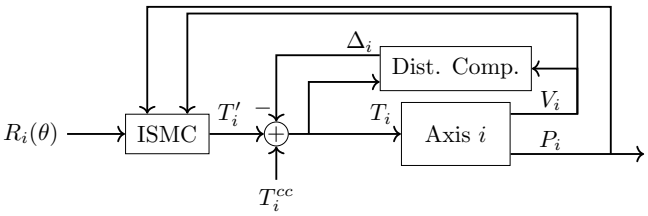

Fig. 4: A single-axis view of the integrated CCC

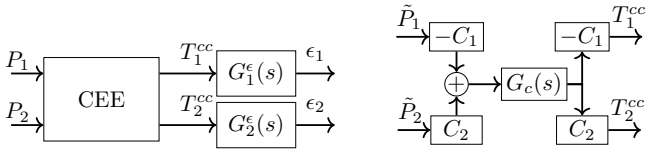

Fig. 5: (left) proposed CCC and Newton-based CEE, and (right) variable-gain CCC. Coefficients C_1 and C_2 are time-varying and depend on the reference contour and current position.

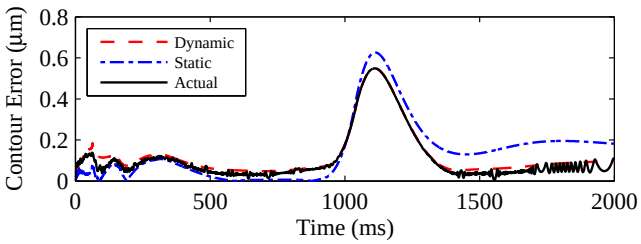

Fig. 6: Contour error for slightly-curved maps, $h_r = 50$

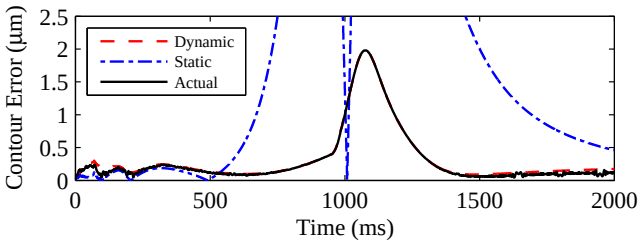

Fig. 7: Contour error for deeply-curved maps, $h_r = 100$

IV. Simulation Results

Two-axis simulation results are presented to verify the precision of the proposed CEE and CCC algorithm for high speed reference feeds. The conventional variable gain CCC shown in Fig. 5, originally proposed by Koren and Lo [11] and enhanced by other researchers, achieves accurate tracking for linear and circular contours. However, due to the static CEE the precision of the conventional cross-coupling class of algorithms deteriorates significantly for non-linear and non-circular contours, particularly at high reference feedrates. Simulations are conducted to highlight the precision of the proposed CEE and CCC algorithm for a high speed parabolic map defined as $R_1 = \theta$, $R_2 = h_r\theta^2$, where $\theta = 2a_r t/T_r - a_r$, $t \in [0, T_r]$, $\theta \in [-a_r, a_r]$, and h_r, a_r, and T_r are constant parameters. Parameters of the Newton-based CEE algorithm are $c' = 500$, $k' = 500$, and $\theta_0 = -a_r$. A PID is used for contour error control in the variable-gain CCC, $G_c(s) = 8 + 80/s + 0.6s$. In the proposed design a PID is used to control contour error along each axis. Since the unity contour error vector is used in the proposed design, the PIDs have the same structure, $G_i^{\epsilon}(s) = 1 + 10/s + 0.1s$, where $i = 1, 2$. The rest of the parameters are $k_1 = 10.3$, $k_2 = 10$, $\tau_1 = 0.040$, $\tau_2 = 0.045$, $|D_1| = |D_2| = 0.15$, $a_1 = b_1 = 12.5, a_2 = b_2 = 11.1$, $\mu_1 = \mu_2 = 1$, $\kappa_1 = 0.2985$, $\kappa_2 = 0.3$, $c_1 = d_1 = 25$, and $c_2 = d_2 = 22.2$. Maximum contour error (MCE) and time-averaged contour error (TACE) are calculated to compare the results of the proposed algorithm with the results of the variable gain CCC algorithm shown in Fig. 5.

A. Comparison of Contour Error Estimation Only

Initially, instead of ISMC it is assumed that PID and reference feedforward controllers are used for position control of each axis. No disturbance compensation is included. The variable gain CCC achieves MCE=58.7 μm and TACE=39.5 μm. The proposed CCC algorithm reduces MCE and TACE to 23.2 μm and 7 μm, respectively. In both structures, the result of the Newton-based CEE and the conventional static version closely follow the actual contour error. In this simulation $PID_1 = 8.08 + 93.2/s + 0.24s$, $PID_2 = 9.36 + 108/s + 0.3s$, $a_r = 20$ mm, $h_r = 50$, and $T_r = 2$ s.

In all the following simulation results, the actual contour error is shown using a solid line, the Newton-based CEE using a dashed line, and the static CEE using a dash-dotted line.

B. Comparison of Proposed Algorithm to Conventional Variable Gain CCC (Fig. 5)

Replacing the position control PID with ISMC and adding disturbance compensation improves the closed-loop performance of both designs. The variable gain CCC achieves MCE=15.3 μm and TACE=1.6 μm. As shown in Fig. 6, the proposed Newton-based CEE and CCC achieve MCE=0.55 μm and TACE=0.11 μm. Moreover, Fig. 6 shows that the static contour error estimate fails to accurately capture the actual contour error. Due to the use of small switching gains and careful selection of the ISMC parameters, results show that high transient peaks in the control signals do not occur.

The proposed CCC design also guarantees high precision contouring for deep curves. The contour curve is increased from $h_r = 50$ to $h_r = 100$. As shown in Fig. 7 the proposed CCC keeps the contour error below MCE=2 μm with TACE=0.31 μm, while the conventional design experiences MCE=23 μm and TACE=4.6 μm. Moreover, as shown in Fig. 7, while the proposed Newton-based CEE is almost identical to the actual contour error, the static CEE fails to capture the actual contour error.

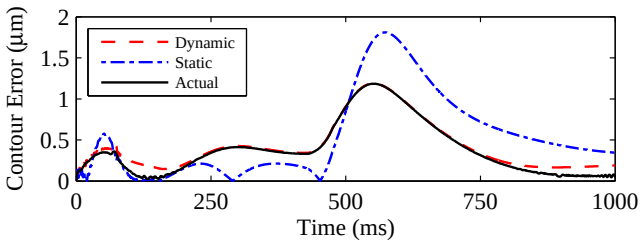

Fig. 8: Contour error for fast contouring, $h_r = 50$

C. High-Speed Complex Contours

To verify the effectiveness of the proposed algorithm in face of higher reference feedrates one can reduce the contouring time by half, i.e., $T_r = 1$ s. The performance of the variable-gain CCC deteriorates dramatically and the contour error experiences MCE=241 μm and TACE=7.6 μm. Meanwhile, as shown in Fig. 8, the proposed algorithm successfully keeps the contour error below MCE=1.2 μm with TACE=0.19 μm. Also, the Newton-based CEE closely follows the actual error during the contouring process. Again, the static contour estimate fails to capture the actual contour error, especially in the close vicinity of the deep curve.

From the conducted simulations, particularly the results shown in Fig. 6, 7, and 8, one can conclude that the static CEE is not an appropriate option for high-precision fast contouring. More importantly, for deep curves, as occurs when one increases h_r, the static CEE departs from the actual contour error. Relying on an unrealistic contour error estimate, even in presence of high-performance position control algorithms, will not guarantee reduction in the contour error.

V. Conclusions

A new dynamic contour error estimate (CEE) using a Newton-based update to achieve high-precision estimates of contour error for high reference feedrate contours with sharp corners and deep curves was presented. The Newton-based CEE and CCC in combination with Integral Sliding Mode Control (ISMC) improved contour tracking precision by at least an order of magnitude in comparison to the conventional variable gain CCC. Since the Newton-based algorithm uses an estimate of the contour map curvature to tune the convergence rate of the CEE, uniform transient response is achieved for a wide range of contour maps. The proposed CEE algorithm proved to be effective under highly-curved and fast reference feeds. The normalized contour error vector increased uniformity and stability margins of the proposed CCC. Moreover, the ISMC improved system robustness with respect to asymmetric dynamics and external disturbances. The adaptive disturbance estimate also partially enhanced closed-loop performance of the integrated control.

References

[1] Y. Altintas, K. Erkorkmaz, and W.-H. Zhu, "Sliding mode controller design for high speed feed drives," *CIRP Annals - Manufacturing Technology*, vol. 49, pp. 265–270, 2000.

[2] B. Chen, D. Tilbury, and A. Ulsoy, "Modular control for machine tools: Cross-coupling control with friction compensation," in *Proc. ASME IMECE*, 1998.

[3] C.-S. Chen and L.-Y. Chen, "Cross-coupling position command shaping control in a multi-axis motion system," *Mechatronics*, vol. 21, pp. 625–632, 2011.

[4] C. Chen, M. Jang, and K. Lin, "Modeling and high-precision control of a ball-screw-driven stage," *Precision Engineering*, vol. 28, pp. 483–495, 2004.

[5] M.-Y. Cheng, K.-H. Su, and S.-F. Wang, "Contour error reduction for free-form contour following tasks of biaxial motion control systems," *Robotics and Computer-Integrated Manufacturing*, vol. 25, pp. 323–333, 2009.

[6] H.-Y. Chuang and C.-H. Liu, "Cross-coupled adaptive feedrate control for multiaxis machine tools," *Journal of Dynamic Systems, Measurement, and Control*, vol. 113, pp. 451–457, 1991.

[7] K. Fujimoto, K. Sakurama, and T. Sugie, "Trajectory tracking control of port-controlled Hamiltonian systems via generalized canonical transformations," *Automatica*, vol. 39, pp. 2059–2069, 2003.

[8] A. Ghaffari, M. Krstić, and D. Nešić, "Multivariable Newton-based extremum seeking," *Automatica*, vol. 48, pp. 1759–1767, 2012.

[9] S.-J. Huang and C.-C. Chen, "Application of self-tuning feed-forward and cross-coupling control in a retrofitted milling machine," *International Journal of Machine Tools and Manufacture*, vol. 35, pp. 577–591, 1995.

[10] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal of Robotics and Automation*, vol. RA-3, pp. 43–53, 1987.

[11] Y. Koren and C.-C. Lo, "Variable-gain cross-coupling controller for contouring," *CIRP Annals - Manufacturing Technology*, vol. 40, pp. 371–374, 1991.

[12] Y. Koren, "Cross-coupled biaxial computer control for manufacturing systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 102, pp. 265–272, 1980.

[13] P. K. Kulkarni and K. Srinivasan, "Optimal contouring control of multi-axial feed drive servomechanisms," *Journal of Engineering for Industry*, vol. 111, pp. 140–148, 1989.

[14] F.-J. Lin, P.-H. Chou, C.-S. Chen, and Y.-S. Lin, "DSP-based cross-coupled synchronous control for dual linear motors via intelligent compelementary sliding mode control," *IEEE Transactions on Industrial Electronics*, vol. 59, pp. 1061–1073, 2012.

[15] C.-C. Lo, "An improved algorithm for cross-coupling control of multi-axis machine tools," *Journal of Manufacturing Science and Engineering*, vol. 121, pp. 537–540, 1999.

[16] C.-C. Lo and C.-Y. Chung, "Tangential-contouring controller for biaxial motion control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 121, pp. 126–129, 1999.

[17] W. H. Moase, C. Manzie, and M. J. Brear, "Newton-like extremum-seeking for the control of thermoacoustic instability," *IEEE Transactions on Automatic Control*, vol. 55, pp. 2094–2105, 2010.

[18] Y.-T. Shih, C.-S. Chen, and A.-C. Lee, "A novel cross-coupling control design for bi-axis motion," *International Journal of Machine Tools and Manufacture*, vol. 42, pp. 1539–1548, 2002.

[19] K. Srinivasan and P. K. Kulkarni, "Cross-coupled control of biaxial feed drive servomechanisms," *Journal of Dynamic Systems, Measurement, and Control*, vol. 112, pp. 225–232, 1990.

[20] D. Sun, "Position synchronization of multiple motion axes with adaptive coupling control," *Automatica*, vol. 39, pp. 997–1005, 2003.

[21] X.-C. Xi, G.-S. Hong, and A.-N. Poo, "Improving CNC contouring accuracy by integral sliding mode control," *Mechatronics*, vol. 20, pp. 442–452, 2010.

[22] Y. Xiao, K. Zhu, and H. Liaw, "Generalized synchronization control of multi-axis motion systems," *Control Engineering Practice*, vol. 13, pp. 809–819, 2005.

[23] 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, pp. 193–211, 2005.

[24] 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, pp. 524–530, 1999.

[25] ——, "Estimation of the contouring error vector for the cross-coupled control design," *IEEE/ASME Transactions on Mechatronics*, vol. 7, pp. 44–51, 2002.

[26] Z.-M. Yeh, Y. Trang, and Y. Lin, "Cross-coupled fuzzy logic control for multiaxis machine tools," *Mechatronics*, vol. 7, pp. 663–681, 1997.