

A study on the optimal tracking problems with predefined data by using iterative learning control

Dang-Khanh Le¹ · Dang-Phuong Le² · Taek-Kun Nam[†]

(Received October 27, 2014 ; Revised November 25, 2014 ; Accepted December 12, 2014)

Abstract: In this paper, we present an iterative learning control (ILC) framework for tracking problems with predefined data points that are desired points at certain time instants. To design ILC systems for such problems, a new ILC scheme is proposed to produce output curves that pass close to the desired points. Unlike traditional ILC approaches, an algorithm will be developed in which the control signals are generated by solving an optimal ILC problem with respect to the desired sampling points. In another word, it is a direct approach for the multiple points tracking ILC control problem where we do not need to divide the tracking problem into two steps separately as trajectory planning and ILC controller. The strength of the proposed formulation is the methodology to obtain a control signal through learning law only considering the given data points and dynamic system, instead of following the direction of tracking a prior identified trajectory. The key advantage of the proposed approach is to significantly reduce the computational cost. Finally, simulation results will be introduced to confirm the effectiveness of proposed scheme.

Keyword: Iterative learning control, Optimal tracking, predefined data points, Without a reference trajectory, Direct tracking.

1. Introduction

Control schemes for tracking problems can be divided into two steps: trajectory planning and tracking control. In these schemes, the trajectory planner attempts to generate an optimal reference trajectory from a given set of points in the motion profile (i.e., specified data points that are desired points at some specified time instants). Then the controller, which is designed to track the reference trajectory, focuses on the system dynamics to generate a sequence of inputs. To improve the performance of trajectory tracking, various control schemes such as feedback control, robust control, and iterative learning control have been developed.

Iterative learning control (ILC) is a control methodology for tracking a reference trajectory in repetitive systems, found in applications such as robotics, semiconductors, and chemical processes. The prime strategy of ILC algorithms is to refine the input sequence from one trial in order to improve the performance of the system on the next trial. A number of

surveys[1]-[3] have effectively covered the novel ideas and development of ILC methodology and can be referred for more information on the basic ideas of ILC.

As discussed in related works [2][4][5] dividing the tracking problem into trajectory planning and ILC trajectory tracking shows drawbacks under certain circumstances. For that reason, this paper proposes and analyzes a direct approach for the multiple points tracking ILC control problem where it does not need to divide the tracking problem into two steps separately.

The strength of the proposed formulation is the methodology to obtain learning laws only considering the given data points and a control signal through dynamic system, instead of following the direction of tracking a prior identified trajectory. In effect, it may not be necessary to plan a reference trajectory as conventional approaches and to reduce the computational cost.

As mentioned, most ILC algorithms focus on tracking a fixed reference trajectory. There have been fewer studies on the connection between the trajectory planning and ILC control

† Corresponding Author (ORCID: <http://orcid.org/0000-0003-0432-3504>) : Division of Control Engineering, Mokpo National Maritime University, 91 H aeyangdaehakro, Mokpo-si, Cheollanam-do, Korea, 530-729, E-mail: tknam@mmu.ac.kr, Tel:

1 Department of Marine Engineering, Mokpo National Maritime University, 91 Haeyangdaehak-ro, Mokpo-si, Cheollanam-do, Korea, 530-729, E-mail: ledangkhanhmtb@gmail.com

2 Department of Marine Engineering, Mokpo National Maritime University, 91 Haeyangdaehak-ro, Mokpo-si, Cheollanam-do, Korea, 530-729, E-mail: ledangphuong85@gmail.com

algorithms, although there is increasing interest in this topic. In terminal ILC, which considers only one end point as a desired output point, Xu et al [6] and Gauthier et al. [5] have addressed the relationship between a point in the motion profile and the control update. Recently, there has been a series of works that consider the tracking problem with multiple specified data points [4][7]-[9] with a particular application to stroke rehabilitation to assist point-to-point movements of patients [10]. These works have shown that, the performance of multiple points tracking problems could be improved based on ILC techniques. Specifically, in [4], the authors developed an ILC framework in which the reference trajectory is updated. Similar to Freeman et al. [4] tracking problems with multiple pass points at specified time are also considered but in this paper, a novel approach for tracking problems with multiple pass points at specified data points is proposed. That is a direct tracking control without a reference trajectory.

The remainder of this paper is organized as follows. The formulation of tracking problem and the conventional approach to solving the problem is provided in Section 2. Section 3 proposes an ILC algorithm where the control signals are generated by solving an optimal ILC problem with respect to the desired sampling points. Simulation results are given in Section 4, and Section 5 concludes this work.

2. Tracking Problem with Specified Data Points

Similar to work in Le et al. [11], consider a linear, continuous-time system that operates on an interval $t \in [0, T]$ as

$$\begin{aligned} \dot{x}_k(t) &= Ax_k(t) + Bu_k(t) \\ y_k(t) &= Cx_k(t) \end{aligned} \quad (1)$$

where, k is the iteration index. The system is a MIMO system that has state $x_k(t) \in \mathfrak{R}^{N_{n_x}}$, control signal $u_k(t) \in \mathfrak{R}^{N_{n_u}}$, and output $y_k(t) \in \mathfrak{R}^{N_{n_y}}$. The matrices A , B and C have appropriate dimensions. In the tracking problem, the system trajectory must pass through, or close to, a limited number of specified points, at prescribed times. The specified time instants in the system operation are defined from the set $T^s = \{t_1^s, t_2^s, \dots, t_M^s\}$,

where $0 \leq t_1^s < t_2^s < \dots < t_M^s \leq T$, and the desired outputs at these points are given by $y_d(t_1^s), y_d(t_2^s), \dots, y_d(t_M^s)$. As a result, the control task is to construct a control law that drives the system output to go through the desired values as closely as possible:

$$y_k(t_i^s) = y_d(t_i^s), \quad i = 1, 2, \dots, M. \quad (2)$$

In typical tracking schemes, a trajectory planner generates a reference trajectory $r(t)$ such that it passes the desired points at $t_1^s, t_2^s, \dots, t_M^s$. In the trajectory planning stage, one of the common techniques to find a reference trajectory from the given points is ‘‘interpolating splines’’, which is a tool in numerical analysis. The generated function is a spline if it exactly interpolates the given points [12]. However, the reference trajectory may not be feasible for the system since it may vary too fast to allow the dynamics to follow the trajectory or it may vary too slowly, which gives a conservative control performance. Thus, the control theoretic interpolating splines were introduced in [13]. The suggested approach addresses these problems by considering auxiliary system dynamics, rather than the spline functions.

In a typical problem, after a trajectory has been generated, a tracking control is applied. To illustrate, consider the case where a linear discrete-time system operates repetitively, from the same initial condition at the start of each trial and we suppose an ILC controller is applied to drive the system output to go as close as possible to the reference trajectory. In ILC, the learning algorithm utilizes output errors and control inputs from the previous iterations to compute an updated control signal in [3][14]. To describe this, let the discrete-time system be described in the lifted system framework as

$$y_k = Pu_k, \quad (3)$$

where the input signal, output signal, and reference trajectory are written in the super-vector forms as

$$\begin{aligned} u_k &= [u_k^T(0) \quad u_k^T(1) \quad \dots \quad u_k^T(N-1)]^T \\ y_k &= [y_k^T(1) \quad y_k^T(2) \quad \dots \quad y_k^T(N)]^T \\ y_d &= [y_d^T(1) \quad y_d^T(2) \quad \dots \quad y_d^T(N)]^T, \end{aligned} \quad (4)$$

and the system matrix $P = [p_{ij}]$ is a Markov matrix which is a lower triangular Toeplitz matrix. For this system we then adopt the following ILC algorithm:

$$u_{k+1} = T_u u_k + T_e (y_d - y_k). \quad (5)$$

It is well known that, this ILC algorithm is convergent if $\rho(T_u - T_e P) < 1$. where $\rho(A) < 1$. is the spectral radius of the matrix A . The algorithm guarantees a monotonic convergence if $\bar{\rho}(T_u - T_e P) < 1$ the condition where $\bar{\rho}(A) < 1$ is the largest singular value of the matrix A , is satisfied. The tracking problem we are interested in, then, is to force the continuous-time system (1) or the discrete-time system (3) to pass through (or close) to the desired points (2) using an algorithm such as **Equation (5)**. In the next section an approach is proposed to solve this problem.

3. Direct Tracking Control without a Reference Trajectory

3.1 Optimal ILC

Let us consider the equivalent linear discrete-time invariant system (3), where $t = 0, 1, 2, \dots, N-1$.

Define

$$g^i(t) = \begin{cases} CA^{t_i^s - t - 1} & \text{if } t < t_i^s \\ 0 & \text{if } t \geq t_i^s \end{cases} \quad (6)$$

the output at the i -th specified time instant is expressed by

$$y_k(t_i^s) = C \sum_{j=0}^{t_i^s - 1} A^{t_i^s - j - 1} B u_k(j) = \sum_{t=0}^{N-1} g^i(t) u_k(t). \quad (7)$$

where, $t_1^s, t_2^s, \dots, t_M^s$ is pre-specified time of M desired points.

The cost function for the problem of tracking multiple desired points in the discrete time model is given as

$$Jc = (y_d - G u_{k+1})^T Q (y_d - G u_{k+1}) + u_{k+1}^T S u_{k+1} + (u_{k+1} - u_k)^T R (u_{k+1} - u_k) \quad (8)$$

where, $R, S = (rI, sI), Q$ are positive definite matrices, $G = [(g^1)^T \ (g^2)^T \ \dots \ (g^M)^T]^T$. The controller in the $(k+1)$ -th trial is achieved from the condition $\partial Jc / \partial u_{k+1} = 0$, or

$$-G^T Q (y_d - G u_{k+1}) + R (u_{k+1} - u_k) + S u_{k+1} = 0. \quad (9)$$

Let $u_k = G^T z_k$ then (9) is equal to

$$-Q (y_d - G G^T z_{k+1}) + R (z_{k+1} - z_k) + S z_{k+1} = 0. \quad (10)$$

Hence, (10) is an iterative learning algorithm, i.e.,

$$((r+s)I + Q W_d) z_{k+1} = (rI + Q W_d) z_k + Q e_k, \quad (11)$$

where, $W_d = G G^T$ is a symmetric positive definite matrix.

Matrix $((r+s)I + Q W_d)$ is non-singular, by defining

$$\begin{aligned} L_z &= ((r+s)I + Q W_d)^{-1} (rI + Q W_d), \\ L_e &= ((r+s)I + Q W_d)^{-1} Q, \end{aligned} \quad (12)$$

the ILC control algorithm for given system is derived as follow

$$\begin{cases} u_k = G^T z_k \\ z_{k+1} = L_z z_k + L_e e_k \end{cases} \quad (13)$$

In definition, the L_2 -norm of the control signal is

$$\|u_k\|^2 = \int_0^T u_k^T u_k dt. \quad (14)$$

Then,

$$\|u_k\|^2 = \int_0^T z_k^T G G^T u_k dt = z_k^T W_d z_k. \quad (15)$$

Since W_d is a positive definite matrix, $W_d = V^T V$, in which V has independent columns, leads to $z_k^T W_d z_k = \|V z_k\|^2$.

Therefore,

$$\|u_k\| \leq \|V\| \|z_k\|. \quad (16)$$

Consequently, the convergence of the control signal is guaranteed from the convergence of the z_k learning algorithm because of **Equation (13)**. Then, the control input generated by **Equation (13)** will drive the system output through or close to the desired motion profile.

Moreover, from (11):

$$((r + s)I + QW_d)z_\infty = rz_\infty + Qy_d, \quad (17)$$

or,

$$z_\infty = (sI + QW_d)^{-1}Qy_d, \quad (18)$$

Then the control input converges to a fixed point u_∞ as:

$$u_\infty = G^T(sI + QW_d)^{-1}Qy_d. \quad (19)$$

As a result, the error converges to a fixed point e_∞ as:

$$e_\infty = y_d - W_d(sI + QW_d)^{-1}Qy_d. \quad (20)$$

Furthermore, if $Q = qI$, where q is real positive, the algorithm achieves monotonic convergence. The reason is that now the matrix $((r + s)I + QW_d)^{-1}r$ is a symmetric positive definite matrix which has the largest singular value equals its spectral radius, and $\rho(((r + s)I + QW_d)^{-1}r) < 1$ [15].

3.2 Theory Analysis

Here, it is clear to see a significant decrease of the computational analyses. In proposed learning algorithm, vector $z_k \in R^M$ and L_z, L_e are $mM \times mM$ matrices where M is the number of terminal points. In comparison, the typical ILC algorithm updates the input with the system matrix $mN \times mN$. As the length of iteration increases ($N > 1000$), which is common in many applications such as robotics with a high sampling rate, the requirement of memory and time dramatically increases.

3.3 Input Constraints

Many control applications require constraints on the control signal u_{k+1} and its change with time δu_{k+1} due to physical limitations or performance requirements. Furthermore, the

constraint on the change of input in the iteration domain Δu_{k+1} should be incorporated.

Consider the constraints as

$$\begin{cases} u_l \leq u_{k+1} \leq u_h \\ \Delta u_l \leq \Delta u_{k+1} \leq \Delta u_h \\ \delta u_l \leq \delta u_{k+1} \leq \delta u_h \end{cases} \quad (21)$$

where,

$$\Delta u_{k+1} = u_{k+1} - u_k, \text{ and}$$

$$\begin{aligned} \delta u_{k+1}(t) &= u_{k+1}(t-1) - u_{k+1}(t-2), \\ t &= 1, 2, \dots, N. \end{aligned}$$

Note that the constraints are chosen to be feasible. Here, δu_{k+1} can be rewritten as

$$\delta u_{k+1} = E u_{k+1}. \quad (22)$$

where, E is a $N \times N$ matrix and,

$$E = \begin{bmatrix} I & 0 & \dots & 0 & 0 \\ -I & I & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & -I & I \end{bmatrix}. \quad (23)$$

Thus, the given constraints can be combined into the following constraints equation:

$$\Theta \Delta u_{k+1} \leq \Xi \quad (24)$$

where,

$$\Theta = \begin{bmatrix} -I \\ I \\ -E \\ E \end{bmatrix}, \Xi = \begin{bmatrix} -\max\{u_l - u_k, \Delta u_l\} \\ \min\{u_h - u_k, \Delta u_h\} \\ -(\delta u_l + E u_k) \\ \delta u_h - E u_k \end{bmatrix} \quad (25)$$

or equivalently,

$$\Theta G^T \Delta z_{k+1} \leq \Xi. \quad (26)$$

Since $e_{k+1} = e_k - W_d \Delta z_{k+1}$, $u_{k+1} = G^T (\Delta z_{k+1} + z_k)$, the problem of minimizing the cost function J_c in Equation (8) is derived as

$$\min_{\Delta z_{k+1}} \{ \Delta z_{k+1}^T (W_d^T Q W_d + (r + s) W_d) \Delta z_{k+1} + 2(s z_k^T - e_k^T Q) W_d \Delta z_{k+1} \} \quad (27)$$

As a result, the given minimization together with the constraints Equation (26) represents a standard quadratic programming problem. In addition, the solutions were proposed in [7][15].

4. Simulation Results

Performances of the proposed techniques are presented through an example of the tracking problem with a linear discrete-time system model to show advantages of the proposed method without reference trajectory. The proposed approach will be compared with the conventional approach.

The chosen system is the same in [11] which is shown in Equation (28) which operates on interval $t \in [0, 79]$. 5 points are selected as desired points in the motion profile, such as: (16, 10), (32, 28), (48, 8), (64, 12), (80, 16). Simulation is shown in Figures 1-5 as following.

$$\begin{aligned} x(t+1) &= \begin{bmatrix} 0 & 1.2 & 0 \\ 0 & 0 & 1.1 \\ -0.1 & -0.3 & -0.2 \end{bmatrix} x(t) + \begin{bmatrix} 0.2 \\ 0.5 \\ 0 \end{bmatrix} u(t) \\ y(t) &= [1 \ 0 \ 0] x(t) \end{aligned} \quad (28)$$

Figure 1 shows the norm of errors in linear scale versus iteration number of proposed approach. Figure 2 depicts a detail view for norm of error. As shown in Figure 1 and Figure 2, the tracking error converges close to a constant error and the convergent rate of error under proposed ILC algorithm without any reference trajectory is slightly better than the method of work in [11].

Figure 3 and Figure 4 illustrate achieved actual outputs of given system under proposed ILC controller. Accordingly, based on suitable chosen weighting matrices, the controller produces superior performance; specifically, output signals go through, or very close to, desired given data points after some iteration.

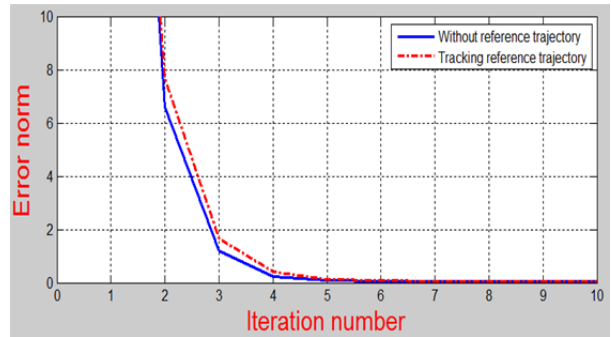


Figure 1: Performance of the ILC controller without reference trajectory with $q = 20$, $r = 2$, and $s = 0.01$.

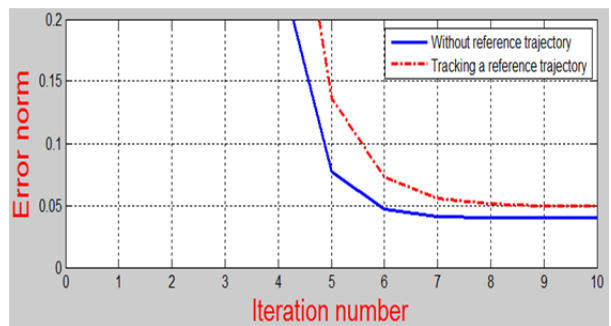


Figure 2: Detail norm of error of system under proposed ILC controller with $q = 20$, $r = 2$ and $s = 0.01$.

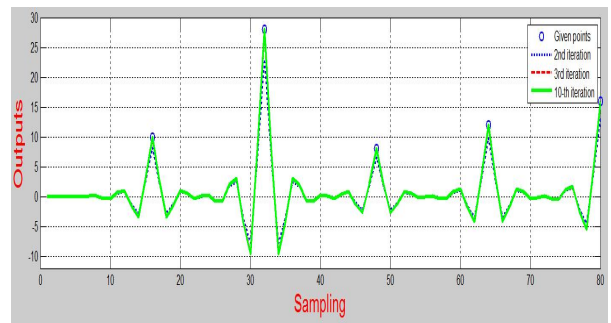


Figure 3: Output of system under proposed ILC controller with $q = 20$, $r = 2$ and $s = 0.01$.

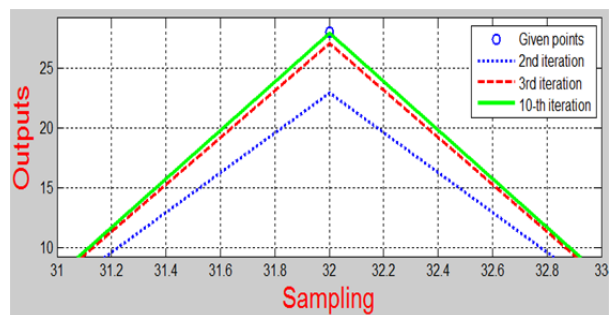


Figure 4: Detail output of system under proposed ILC controller with $q = 20$, $r = 2$ and $s = 0.01$.

By comparison, the conventional norm optimal solution of tracking the given reference trajectory is also simulated in **Figure 5**. It can be seen that the proposed ILC law achieves significantly decrease of control energy, while increasing rate of convergence.

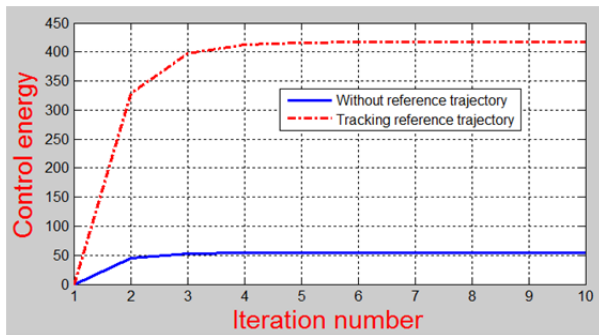


Figure 5: Comparison of control energy between the proposed controller and the conventional ILC ($q = 20$, $r = 2$, and $s = 0.01$).

5. Conclusion

This paper has presented optimal tracking strategies for problems where the system must pass through or close to desired data points at given specified time instants, for the case of systems that repeat their operation from the same initial conditions at each trial. Taking an iterative learning control approach, we have proposed an idea of using only the essential information of the specified data points without building the desired trajectory along the iteration. We showed that, this approach is better in both performance and control efforts, relative to the conventional ILC approaches. Here we have considered only linear systems from an optimization point of view. Future work will consider nonlinear systems as well as linear systems with model uncertainties.

Acknowledgements

The research was supported by the Green Marine Equipment RIS Center Program through the Ministry of Trade, Industry & Energy (MOTIE).

This paper is extended and updated from the short version that appeared in the Proceedings of the International symposium on Marine Engineering and Technology (ISMT 2014), held at Paradise Hotel, Busan, Korea on September 17-19, 2014.

References

- [1] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control: A learning-based method for high-performance tracking control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.
- [2] H. S. Ahn, Y. Q. Chen, and K. L. Moore, "Iterative learning control: Brief survey and categorization," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1099–1121, 2007.
- [3] K. L. Moore, "Iterative learning control: An expository overview," *Journal of Applied and Computational Control, Signals, and Circuits*, vol. 1, no. 1, pp. 151–214, 1999.
- [4] C. T. Freeman, Z. Cai, E. Rogers, and P. L. Lewin, "Iterative learning control for multiple point-to-point tracking application," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 3, pp. 590–600, 2011.
- [5] G. Gauthier and B. Boulet, "Terminal iterative learning control design with singular value decomposition decoupling for thermoforming ovens," *Proceedings of the American Control Conference*, pp. 1640–1645, 2009.
- [6] J. X. Xu and D. Huang, "Initial state iterative learning for final state control in motion systems," *Journal of Automatica*, vol. 44, no. 12, pp. 3162–3169, 2008.
- [7] C. T. Freeman and Y. Tan, "Point-to-point iterative learning control with mixed constraints," *Proceedings of American Control Conference*, pp. 3657–3662, 2011.
- [8] C. T. Freeman, C. Zhonglun, P. L. Lewin, and E. Rogers, "Iterative learning control for multiple point-to-point tracking," *Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 3288–3293, 2009a.
- [9] C. T. Freeman, C. Zhonglun, P. L. Lewin, and E. Rogers, "Objective-driven ILC for point-to-point movement tasks," *Proceedings of American Control Conference*, pp. 252–257, 2009b.
- [10] C. T. Freeman, E. Rogers, A. M. Hughes, J. H. Burrige, and K. L. Meadmore, "Iterative learning control in healthcare electrical stimulation and robotic-assisted upper limb stroke rehabilitation," *IEEE Control Systems Magazine*, vol. 32, no. 1, pp. 18–43, 2012.

- [11] D. K. Le and T. K. Nam, "Performance improvement of trajectory tracking control by using ILC," Proceedings of the 38th KOSME Spring Conference, pp. 194, 2014.
- [12] A. D. Luca, L. Lanari, and G. A. Oriolo, "Sensitivity approach to optimal spline robot trajectories," Journal of Automatica, vol. 27, no. 3, pp. 535–539, 1991.
- [13] S. Sun, M. Egerstedt, and C. F. Martin, "Control theoretic smoothing spline," IEEE Transactions on Automatic Control, vol. 45, no. 12, pp. 2271–2279, 2000.
- [14] N. Amann, D. H. Owens, and E. Rogers, "Iterative learning control using optimal feedback and feed forward actions," International Journal of Control, vol. 65, no. 2, pp. 277–293, 1996.
- [15] D. K. Le and T. K. Nam, "Optimal iterative learning control with model uncertainty," Journal of the Korean Society of Marine Engineering, vol. 37, no. 7, pp. 743–751, 2013.
- [16] J. X. Xu and J. Xu, "Iterative learning control for non-uniform trajectory tracking problems," Proceedings of the 15th IFAC World Congress, pp. 1048–1048, 2002.