Zero-error convergence of iterative learning control based on uniform quantisation with encoding and decoding mechanism

Chao Zhang¹, Dong Shen¹ ²

¹College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, People’s Republic of China
²E-mail: shendong@mail.buct.edu.cn

Abstract: In this study, the zero-error convergence of the iterative learning control for a tracking problem is realised by incorporating a uniform quantiser with an encoding and decoding mechanism. Under this scheme, the system output is first transformed and encoded. Then, the encoded information is transmitted back for updating the input. The results are extended to a finite quantisation level situation under the same framework and a simulation using a permanent magnet linear motor is performed to demonstrate the effectiveness of the proposed scheme.

1 Introduction

Iterative learning control (ILC) is an intelligent control method, which is suitable for systems that iterate the same task in a finite time interval. It incorporates the control experience of past iterations into the current control signal and can be used as a data-driven method that does not require accurate information about the system model. ILC has achieved great improvements, both in theory and applications [1, 2], ever since the original concept was first proposed by Arimoto et al. in 1984 [3].

Most of the studies have assumed that signals can be exchanged with infinite precision. However, this is not practical in real applications owing to the internal principle of digital computers, where the precision is limited owing to the fact that data is stored in binary form. Moreover, in recent decades, network techniques have been significantly developed and several applications of network communications have emerged. Depending on the network circumstances, there is a need to exchange more information with less bandwidth using cheaper devices. One effective method is to apply quantisation. This observation leads us to consider ILC incorporating with quantisation mechanisms.

Bu et al. have attempted to contribute to this topic in [4], where the output of a plant was quantised using a logarithmic quantiser and then transmitted to the controller to obtain the error between the quantised output and reference for updating the ILC scheme. The bounded error convergence is realised, where the upper bound depends on the quantisation density of the employed logarithmic quantiser and the value of the reference trajectory. In particular, a lower quantisation density or a larger value of the reference trajectory leads to a larger convergent zone of the upper bound. To achieve the zero-error convergence, Xu et al. proposed an error quantisation scheme to replace this output quantisation scheme [5]. In this scheme, the reference trajectory is first transmitted to the plant and compared with the real output to generate the tracking error at the local site. The error information is then quantised and transmitted back to update the input signal. It is shown that the zero-error convergence can be achieved asymptotically under this scheme. Moreover, the convergence is independent of the quantisation density of the logarithmic quantiser. This scheme was then extended to stochastic systems in [6], where the influence of the quantisation error was asymptotically eliminated even under stochastic noises. A recent paper [7] introduced the lifting representation for linear systems to resolve the above output-quantisation and error-quantisation problems.

We note that the error-quantisation scheme can help to derive a zero-error convergence due to the inherent characteristic of logarithmic quantiser in the above studies. The technique was also applied to multi-agent systems for asymptotical consensus in [8, 9]. Particularly, in [8] the quantisation was imposed on the differential of the tracking error for continuous-time agents, which implied that the consensus error should be known before quantising. In [9], the quantised consensus error was used to update the input signal in essence for discrete-time agents. In other words, they all benefited from the sufficient precision property within a given finite scope of the logarithmic quantiser. Moreover, in [10], another quantisation method called $\Delta$-quantisation, of which the parameters selection ensured a quantisation bound similarly to the logarithmic sector bounded property, was introduced. However, as the state of the logarithmic quantiser within a given finite scope is infinite, the memory requirements are too large. This disadvantage motivated us to wonder if we can achieve zero-error convergence with a quantiser whose state is finite within a given scope (for example, a uniform quantiser), and which does not need to first transfer the reference precisely through the network circumstance?

Herein, we select a simple uniform quantiser, in which an encoding and decoding method is employed to achieve zero-error convergence. The intrinsic concept of the encoding and decoding mechanism can be found in [11, 12], in which the entire scheme was successfully applied to the coordination problem of multi-agent systems with quantised information. For the application of the encoding and decoding mechanism, we redesign an iteration-axis-based scheme in this study instead of the time-axis-based scheme in [11, 12]. With the incorporation of the uniform quantiser and the encoding and decoding scheme, the zero-error tracking performance is guaranteed to be independent of the quantisation density. Moreover, in this study, we consider using a finite quantisation level quantiser (FQLQ) as a substitute for the infinite quantisation level quantiser (IQLQ). The main difference is that the FQLQ has finite quantisation levels (i.e. the output of the quantiser does not exist as there are infinite quantisation levels. If the input of the finite-level quantiser is larger than the output bound of the quantiser, then the output will be the bound. Consequently, the technical analysis for IQLQ would be easier than that for the FQLQ case, while the implementation of FQLQ would be more suitable than IQLQ.

The rest of the paper is arranged as follows: Section 2 provides the problem formulation and details of the encoding and decoding mechanism. Section 3 elaborates upon the P-type learning algorithm and the convergence property of IQLQ. Section 4 extends the encoding and decoding mechanism to a finite quantisation level situation (i.e. the FQLQ case). Section 5 presents illustrative examples to demonstrate the effectiveness of the proposed methods, followed by conclusions in Section 6.
are suitable matrices with appropriate dimensions. Without loss of
generality, it is assumed
where
system
∥ ⋅ ∥
\[ \mathbb{R} \]
denotes the space\( \mathbb{R} \times p \), the notation \( \| x \| \) denotes \( (x^T x)^{1/2} \), while for a matrix \( X \in \mathbb{R}^{n \times q} \), \( \| X \| = \sqrt{\text{trace}(X^T X)} \). In addition, \( \| \cdot \|_a \) denotes the \( \alpha \)-norm of a given real function, and it is
defined as follows: \( \| \cdot \|_a = \sup_{0 \leq t \leq \mathbb{N}} \| \cdot \|^a \) with \( a > 1 \).

2 Problem formulation
Consider the following linear discrete-time system:
\[
\begin{align*}
    \dot{x}_t &= Ax_t(t) + Bu_t(t), \\
    y_t &= Cx_t(t),
\end{align*}
\]  
(1)
where \( k = 1, 2, \ldots \) denotes the iteration number and \( t \in \mathcal{J}_N \) denotes different time instants during one iteration with \( N \) being the iteration length. The variables \( x_t \in \mathbb{R}^n \), \( u_t \in \mathbb{R}^p \), and \( y_t \in \mathbb{R}^q \) are the state, input, and output, respectively. \( A, B, \) and \( C \) are suitable matrices with appropriate dimensions. Without loss of any generality, it is assumed \( CB \neq 0 \). We should emphasise that although a linear time-invariant system \((A, B, C)\) is considered here, the following results can be extended to a linear time-varying system \((A_t, B_t, C_t)\) without any further efforts.

The desired trajectory is denoted by \( y_d(t) \), \( t \in \mathcal{J}_N \), and the actual tracking error is defined as \( e_t(t) = y_d(t) - y_t(t) \). For further analysis, the following assumptions are needed.

A 1: The desired reference \( y_d(t) \) is realisable in the sense that there exist \( x_d(0) \) and \( u_d(t) \) such that
\[
\begin{align*}
    \dot{x}_d(t + 1) &= Ax_d(t) + Bu_d(t), \\
    y_d(t) &= Cx_d(t),
\end{align*}
\]  
A 2: The identical initialisation condition is satisfied for all iterations, that is
\[
    x_d(0) = x_d(0),
\]  
(3)
where \( x_d(0) \) is the initial value of the desired state defined in A1.

The assumption A1 is usually called the realisable condition.

For the linear system (1), such an assumption is easy to be realised when the related signals lie in the spanning space of the input/output coupling matrix \( CB \); otherwise, a least-square solution can be obtained. The desired input \( u_d(t) \) can be recursively determined if the input/output coupling matrix \( CB \) is of full-column-rank for the relative degree being one case. The realisation issue is out of the scope of this study, thus we simply adopt the classic assumption A1. The assumption A2 is a well-known identical initialisation condition and has been applied in many ILC papers. Some papers have been published to relax this condition; however, most of them required additional information regarding the system or introduced additional learning mechanisms. As the re-initialisation topic is beyond the scope of this study, we simply consider the assumption A2 to concentrate on the subject at hand.

In this study, it is assumed that only quantised output can be transmitted through the network, as shown in Fig. 1, where the capital \( Q \) denotes the applied quantiser. Here, we apply a uniform quantiser \( Q(\cdot) \) to the system output. The uniform quantiser is defined as
\[
Q(m) = \begin{cases} 
0, & \text{if } -\frac{1}{2} \leq m \leq \frac{1}{2}, \\
i, & \text{if } \frac{2i-1}{2} < m \leq \frac{2i+1}{2}, \\
v, & \text{if } m > \frac{2v-1}{2}, \\
-Q(-m), & \text{if } m < \frac{1}{2}.
\end{cases}
\]  
(4)
where \( i = 1, 2, \ldots, v-1 \), \( m \) is an arbitrary value, and \( v \) denotes the largest quantisation level. If \( v \) is a finite integer, then the quantiser is a finite-level quantiser; otherwise \( v = \infty \), then the quantiser is an infinite-level quantiser. Both cases will be elaborated in turn in the sequel. For a vector \( m = [m_1, \ldots, m_m]^T \), the quantiser is formulated as \( Q(m) = [Q(m_1), \ldots, Q(m_m)]^T \). It is clear that \( Q(\cdot) \) is a map from \( \mathbb{R} \) to the set of quantisation levels \( \Omega = \{0, \pm 1, \ldots, \pm v\} \).

The control objective of this study is to design a suitable learning mechanism such that the generated input sequence \( u_d(t) \) can drive the tracking error to converge to zero as the iteration number goes to infinity, \( \forall t \). However, even if we only consider the case where \( v = \infty \), as shown in Fig. 1, the uniform quantiser always leads to a quantisation error in the system output unless the output is equal to the centre of the quantisation interval. The latter case is not suitable for an arbitrary tracking task. In other words, precise tracking performance cannot be achieved by the uniform quantiser. Instead, we can only guarantee a bounded tracking performance with the error bound determined by the resolution of the uniform quantiser. This does not consider the situation where \( v \) is finite, which would lead further to other design issues such as avoiding the saturation of the quantiser.

To solve this problem, we introduce an encoding-decoding mechanism to the system to achieve better tracking performance. The quantisation framework of the output incorporated with the encoding-decoding mechanism is illustrated in Fig. 2, where \( E \) and \( D \) denote the encoder and decoder, respectively. In particular, the output of the plant is first encoded and quantised at the plant site and the quantised data is then sent back to the learning controller. Before being employed by the learning controller, the received data is first decoded by a decoder to obtain an estimate of the sender’s information.

Similar to [11], the associated encoder \( \phi_d(t) \) is designed as
\[
\begin{align*}
\phi_d(t) &= 0, \\
\phi_d(t) &= Q\left(\frac{y_d(t) - \zeta_d(t)}{b_k}\right),
\end{align*}
\]  
(5)
where \( \phi_d(t) \) denotes a zero vector with the same dimension as the system output, \( t \in \mathcal{J}_N \), \( k = 0, 1, \ldots, \) and \( y_d(t) \) and \( y_t(t) \) are the input and output of the encoder \( \phi_d(t) \), respectively. \( \zeta_d(t) \) is the internal state of the encoder \( \phi_d(t) \), \( Q(\cdot) \) is the standard uniform quantiser defined in (4), \( b_k \) is a scaling sequence to improve the tracking performance and we assume \( \frac{b_k}{k} < C_b \). Note that the role of \( b_k \) is
to adjust the magnitude of the difference between the system output and the encoder state. On the contrary, the associated decoder \( \psi_k(t) \) is designed as

\[
\begin{align*}
\hat{y}_k(t) &= \mathbf{0}, \\
\hat{y}_k(t) &= b_k s_k(t) + \hat{y}_k(t),
\end{align*}
\]

where \( t \in \mathcal{J}_N, k = 0, 1, \ldots \), and \( \hat{y}_k(t) \) is the output of the decoder, which is actually an estimate of \( s_k(t) \).

**Remark 1:** To improve the comprehensibility of the proposed encoding and decoding mechanism, we provide a visual description in Fig. 3 of the encoding and decoding process according to the encoding–decoding mechanism, we provide a visual description in Fig. 3 of the encoding and decoding process. From this figure, one can easily comprehend the mechanism is assessed. By substituting the formulation of \( s_k(t) \) from the encoder to the estimate of the system output \( \hat{y}_k(t) \), we get

\[
\begin{align*}
\hat{y}_{k+1}(t) &= b_k s_k(t) + \hat{y}_k(t) \\
\hat{y}_{k+1}(t) &= b_k Q \left( \frac{y_k(t) - \hat{y}_k(t)}{b_k} \right) + \hat{y}_k(t) \\
\hat{y}_{k+1}(t) &= b_k [y_k(t) - \hat{y}_k(t) + \eta_k(t)] + \hat{y}_k(t) \\
\hat{y}_{k+1}(t) &= y_k(t) + b_k \eta_k(t) + \hat{y}_k(t) - \hat{y}_k(t),
\end{align*}
\]

where \( \eta_k(t) = Q \left( \frac{y_k(t) - \hat{y}_k(t)}{b_k} \right) - \frac{y_k(t) - \hat{y}_k(t)}{b_k} \) is the quantisation error. It is apparent from the property of a standard uniform quantiser with infinite level that \( \eta_k(t) \) is bounded. In particular, the absolute value of each dimension of \( \eta_k(t) \) is bounded by 1/2 and such boundedness is independent of the specific value of the output. Moreover, according to Fig. 3 and Remark 1, we have \( \hat{y}_k(t) = \hat{y}_k(t) \). Then we can get

\[
\hat{y}_{k+1}(t) = y_k(t) + b_k \eta_k(t). \tag{9}
\]

From this equation, we can find that the difference between the original output \( y_k(t) \) and the estimated value \( \hat{y}_{k+1}(t) \) is the product of a scaling sequence \( b_k \) and a bounded quantisation error \( \eta_k(t) \). Thus, if we can select a suitable scaling sequence \( b_k \), an acceptable tracking performance can then be achieved. This is summarised in the following theorem.

**Theorem 1:** Consider the system (1) and assume that \( A_1 \) and \( A_2 \) hold. The update law (7) is employed with the encoding–decoding mechanism (5) and (6). If the quantisation level \( v \) is infinity and the learning gain matrix \( L \) is designed such that

\[
\| I - CBL \| < 1, \tag{10}
\]

then the actual tracking error will satisfy

\[
\| e_{k+1}(t) \| \leq \alpha \rho^k C_M + M \sum_{i = 0}^{N} \rho^i |e_i(t)|, \tag{11}
\]

where \( C_M = \max_{0 \leq t \leq T} |e_i(t)| \),

\[
M = \| CBL \| + \sum_{i = 0}^{N} \alpha^i \times \| CA^{i+1}BL \| \| \eta_i(t) \|_a
\]

with \( \| \cdot \|_a \) denoting the \( a \)-norm of a real function,

\[
\rho = \| I - CBL \| + \sum_{i = 0}^{N} \alpha^i \| CA^{i+1}BL \|
\]

and \( a \) is a large enough constant which should be designed to make \( \rho < 1 \). From (11) we can observe that \( \| e_i(t) \| \) is bounded by a linear function of the scaling sequence \( b_k \). Moreover, if we further select the scaling sequence such that \( \sum_{i = 0}^{N} \rho^i b_k \rightarrow 0 \), then we can guarantee that the system output achieves an asymptotical zero-error tracking performance along the iteration axis. That is, \( e_i(t) \rightarrow 0 \) as \( k \rightarrow \infty \).

**Proof:** Denote \( \Delta x_k(t) \equiv x_k(t) - x_0(t) \). Then, from this definition and (1), we have

\[
\Delta x_k(t) = x_k(t) - x_0(t) = A x_{k-1}(t-1) + B u_k(t-1) - A x_0(t-1) - B u_0(t-1) \tag{12}
\]

where \( \Delta x_k(t) = A \Delta x_{k-1}(t-1) + B \Delta u(t-1) \),

\[
\begin{align*}
\Delta x_k(t) &= x_k(t) - x_0(t) \\
\Delta x_k(t) &= x_k(t) - x_0(t) = A x_{k-1}(t-1) + B u_k(t-1) - A x_0(t-1) - B u_0(t-1) \\
\Delta x_k(t) &= A \Delta x_{k-1}(t-1) + B \Delta u(t-1),
\end{align*}
\]
where $\Delta t_i(t) \triangleq u_{t,i}(t) - u_{t}(t)$. Recursively, by calculating the above equation, we get

$$\Delta t_i(t) = \sum_{i=0}^{t-1} A^i B \Delta u(t - 1 - i), \quad (13)$$

where the initialisation condition $A_2$ is applied. From the update law (7) and (9), it follows that

$$\Delta t_i(t) = L[y_i(t') - \hat{y}_i(t')]
\quad = L[y_i(t') - y_i(t') + \gamma_i(t') - \hat{y}_i(t')]
\quad = L[e_i(t') - b_k - \eta_i(t')]
\quad = L[e_i(t') - \eta_i(t')]. \quad (14)$$

Then, according to the definition of $e_i(t)$ and system (1)

$$\begin{align*}
  e_{t,i}(t') - e_{t}(t') \\
  &= y_{t,i}(t') - y_{t,i}(t') + y_i(t') \\
  &= - C_2 e_{t,i}(t') + C_3 e_{t}(t') \\
  &= - C_2 e_{t,i}(t') + C_3 e_{t}(t') \\
  &= - [C_3 A x_{t,i}(t) + B w_{t,i}(t)]e_{t}(t') + C_3 e_{t}(t') \\
  &= - C \Delta t_i(t) - C \Delta t_i(t).
\end{align*}\quad (15)$$

Moving $e_i(t')$ to the right-hand side of the last equation and combining (13) and (14), we have

$$\begin{align*}
  e_{t,i}(t') \\
  &= e_{t}(t') - C \sum_{i=0}^{t-1} A^i B \Delta u(t - 1 - i) - C \Delta t_i(t) \\
  &= e_{t}(t') - C \sum_{i=0}^{t-1} A^i B [L e_{t}(t - i) - B \Delta e_{t}(t - i)] \\\n  &= e_{t}(t') - C \sum_{i=0}^{t-1} A^i B [L e_{t}(t - i) - B \Delta e_{t}(t - i)] \\\n  &= e_{t}(t') - C \sum_{i=0}^{t-1} A^i B L e_{t}(t - i) \\\n  &= e_{t}(t') - C \sum_{i=0}^{t-1} A^i B L e_{t}(t - i) \\\n  &\quad + C \sum_{i=0}^{t-1} A^i B \Delta e_{t}(t - i) + C B L \Delta e_{t}(t - i).
\end{align*}\quad (16)$$

Inserting Euclidean-norm on both sides of the last equation, we obtain (see equation below). Now, apply the $\alpha$-norm to the last inequality, that is, multiply both sides of the last inequality with $\alpha^{-i}$ and take the supremum over the time interval. We then have

$$\begin{align*}
  \sup_{0 \leq t \leq N} \alpha^{-i} \| e_{t,i}(t') \| \\
  \leq \sup_{0 \leq t \leq N} \alpha^{-i} \| I - C B L \| \| e_{t}(t') \| \\\n  + \sup_{0 \leq t \leq N} \alpha^{-i} \sum_{i=0}^{t-1} \| C A^{i+1} B L \| \| e_{t-i}(t') \| \\\n  + \sup_{0 \leq t \leq N} \alpha^{-i} \sum_{i=0}^{t-1} b_{t-i} \| C A^{i+1} BL \| \| \eta_{t-i}(t') \| \\\n  + \sup_{0 \leq t \leq N} \alpha^{-i} b_{t-i} \| C B L \| \| \eta_{t-i}(t') \| .
\end{align*}\quad (17)$$

Recalling the definition of $\alpha$-norm, i.e.

$$\| \cdot \|_\alpha = \sup_{0 \leq t \leq N} \alpha^{-i} \| \cdot \|$$

with $\alpha > 1$

$$\begin{align*}
  \| e_{t}(t') \|_\alpha \leq \| I - C B L \| \| e_{t}(t') \|_\alpha \\
  + \sup_{0 \leq t \leq N} \alpha^{-i} \sum_{i=0}^{t-1} \| C A^{i+1} B L \| \| e_{t-i}(t') \| \\
  + \sup_{0 \leq t \leq N} \alpha^{-i} \sum_{i=0}^{t-1} b_{t-i} \| C A^{i+1} B L \| \| \eta_{t-i}(t') \| \\
  + \| b_{t-i} \| \| C B L \| \| \eta_{t-i}(t') \| .
\end{align*}\quad (18)$$

where $\rho$ and $M$ are defined as follows:

$$\begin{align*}
  \rho &\triangleq \| I - C B L \| + \sum_{i=0}^{N-1} \alpha^{-i} \| C A^{i+1} B L \| , \\
  M &\triangleq \| C B L \| + \sum_{i=0}^{N-1} \alpha^{-i} \| C A^{i+1} B L \| \| \eta_{t-i}(t') \|.
\end{align*}\quad (19)$$

Choose a sufficiently large $\alpha$ so that

$$\rho = \| I - C B L \| + \sum_{i=0}^{N-1} \alpha^{-i} \| C A^{i+1} B L \| < 1. \quad (20)$$

From (17), we can infer that $\sum_{i=0}^{t-i} \alpha^{-i} \| C A^{i+1} B L \|$ is a bounded term. Meanwhile, the quantisation error $\eta_{t-i}(t)$ is always bounded when applying the infinite level uniform quantiser, leading to $M$ also being bounded. By recursively substituting the relation shown in (16) along the iteration axis, the following expression can be obtained:

$$\begin{align*}
  \| e_{t,i}(t') \|_\alpha \leq \rho^i \| e_{t}(t') \|_\alpha + M \sum_{i=0}^{k-1} \rho b_{t-1-i} \cdot .
\end{align*}\quad (21)$$

From the definition of $\alpha$-norm, we can always find an upper bound of

$$\begin{align*}
  \| e_{t}(t') \|_\alpha = \sup_{0 \leq t \leq N} \alpha^{-i} \| e_{t}(t') \| \leq C_{M}\|
\end{align*}\quad (22)$$

where $C_{M} = \max_{0 \leq t \leq N} | e_{t}(t') |$. Then, we obtain

$$\begin{align*}
  \| e_{t,i}(t') \|_\alpha \leq \rho^i C_{M} + M \sum_{i=0}^{k-1} \rho b_{t-1-i} \cdot .
\end{align*}\quad (23)$$

Using the property of $\alpha$-norm again

$$\begin{align*}
  \| e_{t,i}(t') \|_\alpha \leq \| I - C B L \| \| e_{t}(t') \|_\alpha \quad + \sum_{i=0}^{t-i} \| C A^{i+1} B L \| \| e_{t-i}(t') \| \\
  + \sum_{i=0}^{t-i} b_{t-i} \| C A^{i+1} B L \| \| \eta_{t-i}(t') \| + b_{t-i} \| C B L \| \| \eta_{t-i}(t') \| .
\end{align*}\quad (24)$$
\[
\| e_{k+1}(t) \| \leq \alpha^k \| e_{1}(t) \|
\]

Moreover, if we further select the scaling sequence such that \( \sum_{i=0}^{k-1} \rho_i b_{k-i} \to 0 \), we can then guarantee that the system output achieves an asymptotical zero-error tracking performance along the iteration axis. This completes the proof. \( \square \)

Theorem 1 presents the tracking performance of the proposed ILC scheme using quantised output information incorporated with an encoding-decoding mechanism. Moreover, Theorem 1 also characterises the inner relation between the tracking error bound and the scaling sequence in (11), which provides us with a guideline for regulating the tracking performance by choosing a corresponding decreasing sequence. Specifically, it follows from (20) that the tracking error tends to zero as long as \( \sum_{i=0}^{k-1} \rho_i b_{k-i} \to 0 \). In addition, the scaling sequence \( b_k \) should be carefully designed because it is not only related to the quantisation level design but also correlated with the convergence speed of the proposed algorithm.

**Remark 2:** To achieve zero-error tracking performance of the desired trajectory, \( \sum_{i=0}^{k-1} \rho_i b_{k-i} \to 0 \) and \( \rho_i b_i < C_b \) should be satisfied. It is clear that \( b_k = \rho^k \) meets these requirements, thus it can be an alternative selection. Moreover, arbitrary decreasing sequence of \( b_k \) satisfying \( b_k \to 0 \) would infer the condition \( \sum_{i=0}^{k-1} \rho_i b_{k-i} \to 0 \). We have mentioned that the role of \( b_k \) is to adjust the magnitude of the difference between the system output and the encoder state. As the iteration number increases, the difference between system output \( y_k(t) \) and the encoder state \( \zeta_k(t) \) will tend to zero, so \( b_k \) is introduced to enlarge this difference. This means \( b_k \) should decrease to retain the enlarging ability. However, when \( b_k \) decreases too fast, it may increase the transmission burden. Consequently, there exists a trade-off in selection of \( b_k \).

## Finite quantisation level situation

In the last section, we discussed the P-type learning algorithm and its associated convergence property using quantised information when the quantisation level \( v \) is chosen as infinity. However, this is difficult to realise in practical applications, because we have to implement an infinite range of quantisers since we have little prior knowledge of possible output. This motivates us to design a quantiser that only has a finite quantisation level but can also achieve the same goal as the IQQL. Compared with the infinite quantisation level case, how to select suitable bounded quantisation levels, which still guarantee the convergence property of the proposed scheme, is the key issue that we need to cope with for this case. A detailed description of the quantisation level design and the corresponding proof will be provided in this section.

Similar to the derivations in Section 3 (see (9)), we can derive the inherent relationship between the estimated output and the actual output as follows:

\[
\hat{y}_{k+1}(t) = \zeta_{k+1}(t) = y_{k+1}(t) + b_k \eta_{k+1}(t).
\]

It is worth pointing out that the natural boundedness property of the quantisation error \( \eta_{k+1}(t) \) in Section 3 is no longer valid for the finite quantisation level situation. In particular, the fact that each dimension of \( \eta(t) \) is less than 1/2 is always valid under the infinite quantisation level situation, but we cannot guarantee this property naturally when the quantisation level is finite. This observation motivates us to consider the suitable selection of the scaling functions in this section. In other words, the problem of designing a quantisation level to guarantee the boundedness of \( \eta(t) \) is the key issue that we need to consider for the finite quantisation level case.

The main theorem for the finite quantisation level situation is given as follows, where the selection of the scaling functions is specified.

**Theorem 2:** Consider the system (1) and assume that A1 and A2 hold. The update law (7) is employed with an encoding-decoding mechanism (5) and (6). If the learning gain matrix \( L \) is designed such that

\[
\| I - CBL \| < 1, \tag{22}
\]

and the quantisation level \( v \) satisfies

\[
v \geq \begin{cases} 
\frac{\max_{0 \leq i \leq T} |y_i(t)| + \frac{C_M}{b_i}}{b_i} \quad & k = 0, \\
\frac{\|C\| \|BL\| \left( C_M + \frac{1}{2} \rho^k \sum_{i=0}^{k-1} \rho_i b_{k-i} \right) + b_k \rho^k + \frac{1}{2b_i}}{b_i (1 - b_1) \rho^k} \quad & k = 1, \\
\frac{\|C\| \|BL\| \alpha^k \left( \rho^k C_M + \sum_{i=0}^{k-1} \rho_i b_{k-i} \right) \rho^k + b_k \rho^k + \frac{1}{2b_i}}{b_i (1 - b_1) \rho^k} \quad & k \geq 2,
\end{cases}
\]

(23)

where \( C_M \triangleq \max_{0 \leq i \leq T} \| e_i(t) \|, \| A \| \leq b_k \).

\[
\rho \triangleq \| I - CBL \| + \sum_{i=0}^{N-1} \alpha^i \| CA^{i+1}BL \|
\]

with \( \alpha \) being a large enough constant, which should be designed to make \( \rho < 1 \), and

\[
M \triangleq a \| CBL \| + \sum_{i=0}^{N-1} \alpha^i \| CA^{i+1}BL \| \| \eta(t) \|_a
\]

Then the actual tracking error satisfies

\[
\| e_{k+1}(t) \| \leq \alpha^k \left( \rho^k C_M + \sum_{i=0}^{k-1} \rho_i b_{k-i} \right). \tag{24}
\]

In other words, the tracking error \( \| e(t) \| \) is bounded by a linear function of the scaling sequence \( b_k \). Moreover, if we further select the scaling sequence such that \( \sum_{i=0}^{k-1} \rho_i b_{k-i} \to 0 \), we can then guarantee that the system output achieves an asymptotical zero-error tracking performance along the iteration axis.

**Proof:** When \( k = 0 \), from (5) and (23), we can get

\[
\frac{|y(t) - \zeta(t)|}{b_0} = \frac{|y(t)|}{b_0} \leq \max_{0 \leq i \leq T} |y_i(t)| + \frac{C_M}{b_0} \tag{25}
\]

\[
\leq v + \frac{1}{2} \alpha^k
\]

where \( C_M \triangleq \max_{0 \leq i \leq T} \| e_i(t) \| \) as defined before denotes the maximum tracking error during the first iteration. Combining (4) with (25) leads to

\[
\max \| \eta(t) \| \leq \frac{1}{2}. \tag{26}
\]
Then we have

\[
\frac{|y(t) - \zeta(t)|}{b_k} \leq \frac{|C \sum_{i=0}^{r-1} A^i B \Delta u_i(t - 1 - i)|}{b_i} + \frac{b_h}{2b_i}.
\]

Similarly,

\[
\frac{|y(t) - \zeta(t)|}{b_k} \leq \frac{|C \sum_{i=0}^{r-1} C_M \sum_{j=0}^{N-1} b_j A^j \Delta u_j(t - 1 - i)|}{b_i} + \frac{b_h}{2b_i}.
\]

When \( k > 0 \), with (21), one can obtain \( \zeta(t) = b_k \cdot \eta_k(t) + y_k(t) \).

Then we have

\[
\frac{|y_k(t) - \zeta(t)|}{b_k} = \frac{|y_k(t) - \zeta(t) - b_k \cdot \eta_k(t)|}{b_k} \leq \frac{|y_k(t) - \zeta(t)|}{b_k} + \frac{|b_k \cdot \eta_k(t)|}{b_k}.
\]

Consider \( y_k(t) - \zeta(t) \) in (27)

\[
y_k(t) - \zeta(t) = C_{y_k(t)} - C_{\Delta \eta_k(t)} = C \Delta \eta_k(t).
\]

We have illustrated the property of \( \Delta \eta_k(t) \) in (12) and (13), whereas the property of \( \Delta u_k(t) \) is checked as follows:

\[
\Delta u_k(t) = I[y(t) - \zeta(t)] - \Delta \eta_k(t)
\]

(29)

where \( \Delta u_k(t) \triangleq u_k(t) - u(t) \). Then, (27) can be further rewritten in the following form:

\[
\frac{|y_k(t) - \zeta(t)|}{b_k} \leq \frac{|C \sum_{i=0}^{r-1} A^i B \Delta u_i(t - 1 - i)|}{b_i} + \frac{b_h}{2b_i}.
\]

When \( k = 1 \), combining (23), (26) and (29) yields (see (31)) , where \( \| A \| \leq b_k \). Combining (4) and (30) yields

\[
\max \{ \| \eta_i(t) \| \} \leq \frac{\rho}{1 - \alpha}.
\]

We will use the mathematical induction method to continue our proof. Assume that \( \max \{ \| \eta_i(t) \| \} \leq \frac{\rho}{1 - \alpha} \), for \( i = 1, 2, \ldots, k \). Then, combining (13), (15), and (29) leads to

\[
\frac{|e_{\alpha_i}(t') - e_{\alpha_i}(t)|}{b_k} = \frac{|e_{\alpha_i}(t') - e_{\alpha_i}(t) - b_k \cdot \eta_i(t)|}{b_k} \leq \frac{|e_{\alpha_i}(t') - e_{\alpha_i}(t)|}{b_k} + \frac{|b_k \cdot \eta_i(t)|}{b_k}.
\]

Choose a sufficiently large \( \alpha \) such that

\[
\rho = \| I - CBL \| + \sum_{i=0}^{N-1} \alpha^i \| CA^{i+1}BL \| < 1.
\]

Again, by applying the same steps as with the infinite-level quantisation situation, we get

\[
\| e_{\alpha_i}(t) \| \leq \alpha^i \rho^i C_M + M \sum_{i=0}^{N-1} \alpha^i \rho^i \| \eta_i(t) \| \leq \frac{\rho}{1 - \alpha}.
\]

Now it comes to the induction step of the mathematical induction method. In other words, we will prove that if \( \max \{ \| \eta_i(t) \| \} \leq \frac{\rho}{1 - \alpha} \), for \( i = 1, 2, \ldots, k \), then \( \max \{ \| \eta_i(t) \| \} \leq \frac{\rho}{1 - \alpha} \). In fact...
Therefore, we have proven that the finite-level quantiser is unsaturated in the $(k + 1)$th iteration by using the mathematical induction method. This means that max $|q_i(t)| \leq (1/2)$ will always be valid for all $l \in \mathbb{N}$. Therefore, (35) can be extended to all $k \in \mathbb{N}$. If we further select the scaling sequence such that $\sum_{i=0}^{k-1} \rho_i b_{k-i-1} \rightarrow 0$, for example, $b_i = \rho_i$, then we can guarantee that the system output achieves an asymptotical zero-error tracking performance along the iteration axis.

At last, we should prove the boundedness of quantisation levels given in formula (23) so that the proposed scheme can be called finite-level quantiser-based scheme. It is obvious that quantisation level is finite for $k = 0$ and $k = 1$. When $k \rightarrow \infty$, from condition $\sum_{i=0}^{k-1} \rho_i b_{k-i-1} \rightarrow 0$ and $(b_{k-i})/b_k < C_0$, we can get

$$\lim_{k \rightarrow \infty} \left| \frac{\alpha^N \rho_i^i C_b + M \sum_{i=0}^{k-1} \rho_i b_{k-i-1}}{b_k} + \frac{b_k - 1}{2} \right| \leq C_0,$$

and

$$\sum_{i=0}^{k-1} \rho_i b_{k-i-1} \rightarrow 0.$$ 

Hence, the proof is completed.

Theorem 2 shows the tracking performance of the proposed finite quantisation level method for the ILC tracking problem by employing the encoding and decoding mechanism. As we can see, the difference between the FQLQ case and IQLQ case only lies in the type of quantiser as both schemes employ the same system, update law, and encoding and decoding mechanism. In other words, avoiding saturation of the quantiser is the main difference and the most challenging requirement when addressing the FQLQ situation.

Remark 3: Whether the finite quantisation level situation or the infinite quantisation level situation is in effect, the transmission burden has been efficiently reduced. As we can learn from encoder equation (5) and decoder equation (6), the data that we need to transmit is just the magnified error information between the system output and the encoder estimation. If we do not introduce the encoding and decoding mechanism, then the data that we need to transmit is the real output of the system.

Remark 4: We should especially note that some bound values given in the theorem may be some large due to the conservative estimation in the proof. However, the actual quantisation level required by the algorithm would be much smaller. In other words, the actual upper bound of the finite level is independent of the analysis of the convergence, while depends on the intrinsic character of the algorithm and system. This property will be demonstrated in the illustrative simulation, from which we can observe that the actual quantised value is quite small inferring a smaller bound $\nu$ of the quantiser.

5 Simulation example

To verify the effectiveness and convergence property of the proposed encoding and decoding ILC quantisation method, a permanent magnet linear motor (PMLM) model is utilised. The discretised model of PMLM is given as follows [14]:

$$x_{t+1} = x_t + v(t)\Delta,$$

$$v(t+1) = v(t) - \frac{k_i q_f^2}{Rm} v(t) + \frac{k_i q_f}{Rm} u(t),$$

where the sampling period $\Delta$ is selected as 0.01 s, $x$ and $v$ denote the motor position and rotor velocity, $R = 8.6$, $m = 1.635$ kg, $q_f = 0.35$ Wb are the resistance of the stator, rotor mass and flux linkage, respectively. $k_i = \pi/\tau$ and $k_b = 1.5\pi/\tau$ with $\tau = 0.031$ m being the pole pitch. Furthermore, the desired trajectory is given as $\gamma_d = 1/3[\sin(5t) + 1 - \cos(15t)]$, $0 \leq t \leq 1$. As assumed in assumption A2, the initial state is chosen as $x_0(0) = 0$ for all $k$, and the initial input is simply set as $u_0(0) = 0$, $\forall \tau \in [0, 1]$.

Next, we will illustrate the encoding and decoding mechanism under an infinite quantisation level and a finite quantisation level, respectively. We first consider the infinite level case and the learning gain $L$ is chosen as 15. Then we can easily obtain $\|I - CBL\| = 0.4324 < 1$. Moreover, $\alpha$ is selected as 2 and $\rho = \|I - CBL\| + \sum_{i=0}^{N} \alpha^i \|CA^iBL\| = 0.6667 < 1$ is satisfied. As mentioned at the end of the proof, $b_k$ can simply be chosen as $\rho^i$. The algorithm is run for 20 iterations.

The tracking performance of the actual and estimated output at the final iteration and the desired trajectory are shown in Fig. 4. We call the output of decoder $\hat{y}(t)$ as the estimated output, and the output of the system $y(t)$ as the actual output. The solid, dashed-dotted, and dashed lines denote the desired trajectory, the estimated output, and the actual output, respectively. We can see that the three lines almost coincide with each other, meaning that the tracking performance is good at the 20th iteration and that we achieve the goal of zero-error convergence using a simple uniform quantiser.

Figs. 5 and 6 show the tracking performance of the actual output and estimated output at the 1st, 3rd, 5th, and 20th iteration, respectively. From both figures, we observe that the tracking performance is gradually improved as the iteration number increases. However, during the first several iterations, the quantisation effect of the estimated output is clearly shown.

The maximal tracking errors max $|e_{t+1}(t)|$ and max $|e_{t+1}(t)|$ for the actual error (defined as $e(t) = y(t) - \hat{y}_k(t)$) and auxiliary error (defined as $e(t) = \gamma_d(t) - \gamma_k(t)$) along the iteration axis are shown.
in Fig. 7, denoted by a solid line and a dashed line, respectively. It is seen that both profiles display a continuously decreasing trend along the iteration axis. This observation shows the effectiveness of the proposed scheme and verifies the asymptotical zero-error tracking property.

Next, we move to verify the finite quantisation level case. According to (23), the quantisation level can be executed and these values can be applied to the finite-level quantiser. When we employ the algorithm on the same motor and accordingly use the same simulation parameters, the simulation results demonstrate that an identical tracking performance can be achieved. In other words, we can obtain the same simulation results as shown in Figs. 4–7. Thus we omitted these repetitions.

From (23), it is observed that there exists an exponential term $\alpha^N$ inside the quantisation level calculation. However, because $\alpha$ and $N$ are both constants, the multiplication is bounded. This bounded exponential term is the main factor that makes the calculation results of the quantisation level fairly large. We need to mention that the calculation formula of the quantisation level mainly relies on the proof. If we can employ a tighter estimation method, the quantisation level will be compressed to a smaller range. Data extracted from the simulations is provided to verify this conjecture. In particular, we define $\max_t|\kappa_k(t)|$ as the maximum input value of the quantiser at the $k$th iteration, $\kappa_k(t) \triangleq (y_k + 1)(t + 1) - \zeta_k(t)/b_k$ (denoting the enlarged difference between the actual output and internal state of the encoder). The profile of such a value along the iteration axis is plotted in Fig. 8. In the figure, the actual quantisation level that we need is quite small. Actually, it is far less than the calculated result from (23).

**Table 1** Comparison between real bounds and calculation results

<table>
<thead>
<tr>
<th>Iteration</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>real bound</td>
<td>0.9135</td>
<td>1.7169</td>
<td>1.9576</td>
<td>2.3510</td>
<td>2.1079</td>
</tr>
<tr>
<td>calculated result</td>
<td>2.1388</td>
<td>$7.2122 \times 10^{30}$</td>
<td>$9.2664 \times 10^{30}$</td>
<td>$1.321 \times 10^{31}$</td>
<td>$1.3375 \times 10^{31}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iteration</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>real bound</td>
<td>1.6891</td>
<td>1.7214</td>
<td>2.0535</td>
<td>1.9968</td>
<td>2.1817</td>
</tr>
<tr>
<td>calculated result</td>
<td>$1.5429 \times 10^{31}$</td>
<td>$1.7483 \times 10^{31}$</td>
<td>$1.9538 \times 10^{31}$</td>
<td>$2.1592 \times 10^{31}$</td>
<td>$2.3646 \times 10^{31}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iteration</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>real bound</td>
<td>1.7567</td>
<td>1.6893</td>
<td>1.9529</td>
<td>1.4969</td>
<td>1.5131</td>
</tr>
<tr>
<td>calculated result</td>
<td>$2.5700 \times 10^{31}$</td>
<td>$2.7755 \times 10^{31}$</td>
<td>$2.9809 \times 10^{31}$</td>
<td>$3.1863 \times 10^{31}$</td>
<td>$3.3917 \times 10^{31}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iteration</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>real bound</td>
<td>1.4022</td>
<td>1.5542</td>
<td>1.5319</td>
<td>1.5045</td>
<td>1.8973</td>
</tr>
<tr>
<td>calculated result</td>
<td>$3.5972 \times 10^{31}$</td>
<td>$3.8026 \times 10^{31}$</td>
<td>$4.0080 \times 10^{31}$</td>
<td>$4.2134 \times 10^{31}$</td>
<td>$4.4189 \times 10^{31}$</td>
</tr>
</tbody>
</table>
The specific values of the actual maximum input and the calculated results of (23) are given in Table 1. From this table, it is clear that $v = 3$ (i.e. the possible quantised values are $0, \pm 1, \pm 2, \pm 3$) is sufficient for the simulation.

6 Conclusions

In this study, the tracking problem of quantised ILC is considered. By employing an infinite-levelled uniform quantiser incorporated with an encoding and decoding mechanism, the zero-error tracking performance is strictly proven and simulation results verify the theoretical consequences. Then, the extension to a finite-levelled uniform quantiser is also strictly analysed. It extends the practical application range of the proposed schemes. For further research, it is of great interest to consider the input quantisation case (for example, [15]) as most current studies focus on the output quantisation. Meanwhile, how to design the quantisation level for a finite-levelled uniform quantiser is also important for many applications.

7 Acknowledgments

This work was supported by the National Natural Science Foundation of China (61673045) and the Beijing Natural Science Foundation (4152040).

8 References