Norm Optimal Iterative Learning Control with Application to Problems in Accelerator based Free Electron Lasers and Rehabilitation Robotics

E. Rogers D. H. Owens, H. Werner C. T. Freeman P. L. Lewin
S. Kichhoff G. Lichtenberg C. Schmidt

This paper gives an overview of the theoretical basis of the norm optimal approach to iterative learning control followed by results that describe more recent work which has experimentally benchmarking the performance that can be achieved. The remainder of this paper then describes its actual application to a physical process and a very novel application in stroke rehabilitation.

I. INTRODUCTION

Iterative Learning Control (ILC) has been especially developed to improve the performance of systems that operate in a repetitive manner where the task is to follow some specified trajectory in a specified finite time interval, also known as a pass or a trial in the literature, with high precision. The novel principle behind ILC is to suitably use information from previous trials, often in combination with appropriate current trial information, to select the current trial input to sequentially improve performance from trial-to-trial. In particular, the aim is to improve performance from trial-to-trial in the sense that the tracking error, the difference between the output on a trial and the specified reference trajectory, is sequentially reduced to either zero, ideal case, or some suitably small value.

The original work in this area is credited to and since then there have been substantial developments in both systems theoretic and applications terms. For an overview of the algorithm development side see, for example, where this reference has the added feature of a categorization of algorithms developed up to 2004. Applications areas where ILC has been successfully applied include robotics, automated manufacturing plants and food processing. For more details, including some those where there is clear potential for significant added benefit from fully developed ILC, one possible source is the survey article.

In general, current research and development in ILC can, as in other areas, be broadly partitioned into starting form either a linear or nonlinear model of the plant dynamics but here we restrict attention to the former where there are still many open problems. This is especially true at the interface between theory and applications. Given the diverse range of algorithms which have been developed over the years, there is a clear need to develop tools and case studies which allow for valid comparison of competing designs. This paper begins by giving the basic ideas and results that led to the development of so-called norm optimal class of algorithms. The development then continues to describe experimentally benchmarked using laboratory based facilities especially designed and constructed for this purpose. This is followed by more recent results where such algorithms have been applied, using a computationally more efficient implementation resulting from the laboratory based experimental benchmarking, to free electron lasers. Following this, a novel application of norm optimal ILC to stroke rehabilitation is described and the paper concludes by a brief overview of some areas of ongoing work and possible future research directions.

II. NORM OPTIMAL CONTROL THEORY

The mathematical definition of ILC used in this paper has the following general form, where the results in this section are mainly from.

Definition 1 Consider a dynamic system with input $u$ and output $y$. Let $\mathcal{Y}$ and $\mathcal{U}$ be the output and input function spaces respectively and let $r \in \mathcal{Y}$ be a desired reference trajectory for the system. An ILC algorithm is successful if, and only if, it constructs a sequence of control inputs $\{u_k\}_{k \geq 0}$ which, when applied to the system (under identical experimental conditions), produces an output sequence $\{y_k\}_{k \geq 0}$ with the following properties of convergent learning

$$\lim_{k \to \infty} y_k = r, \quad \lim_{k \to \infty} u_k = u_\infty$$

Here convergence is interpreted in terms of the topologies assumed in $\mathcal{Y}$ and $\mathcal{U}$ respectively.

A major advantage of this general statement of the problem is that it allows a simultaneous description of linear and nonlinear dynamics, continuous or discrete plant with either time-invariant or time varying dynamics.

Let the space of output signals $\mathcal{Y}$ be a real Hilbert space and $\mathcal{U}$ also be a real (and possibly distinct) Hilbert space of input signals. The respective inner products (denoted by $\langle \cdot, \cdot \rangle$) and norms $\| \cdot \|$ are indexed in a way that reflects the space if it is appropriate to the discussion e.g. $\|x\|_\mathcal{Y}$ denotes the norm of $x \in \mathcal{Y}$. The Hilbert space
structure induced by the inner product is essential in what follows but is not restrictive, as specific choices of this structure enables the analysis of, for example, continuous or discrete-time systems.

The dynamics of the systems considered here are assumed to be linear and represented in operator form as

$$y = Gu + z_0$$

where $G : \mathcal{U} \rightarrow \mathcal{Y}$ is the system input/output operator (assumed to be bounded and typically a convolution operator) and $z_0$ represents the effects of system initial conditions. If $r \in \mathcal{Y}$ is the reference trajectory or desired output then the tracking error is defined as

$$e = r - y = r - Gu - z_0 = (r - z_0) - Gu$$

Hence without loss of generality, it is possible to replace $r$ by $r - z_0$ and consequent assume that $z_0 = 0$.

It is clear that the ILC procedure, if convergent, solves the problem $r = Gu_\infty$ for $u_\infty$. If $G$ is invertible, then the formal solution is just $u_\infty = G^{-1}r$. A basic assumption in ILC is that direct inversion of $G$ is not acceptable since, for example, this would require exact knowledge of the plant and involve derivatives of the reference trajectory. This high-frequency gain characteristic would make the approach sensitive to noise and other disturbances. Also inversion of the whole plant $G$ is unnecessary as the solution only requires finding the pre-image of the reference trajectory $r$ under $G$.

The problem considered here can easily be seen to be equivalent to finding the minimizing input $u_\infty$ for the optimization problem

$$\min_u \{\|e\|^2 : \ e = r - y, \ y = Gu\}$$

This optimization problem needs an iterative solution that is traditionally seen as a numerical problem in numerical analysis but in the ILC setting it is seen as an experimental procedure. The difference between the two viewpoints is the fact that an experimental procedure has an implicit causality structure that is not naturally there in numerical computation.

There are many iterative procedures to solve the optimization problem (1) but there is a clear advantage in the use of descent algorithms of a suitable type as considered for learning systems by, for example,\(^7\). The gradient based ILC algorithm class generates the control input to be used on trial $k+1$ as

$$u_{k+1} = u_k + \epsilon_{k+1} G^* e_k$$

where $G^* : \mathcal{Y}_* \rightarrow \mathcal{U}_*$ is the adjoint operator of $G$ and $\epsilon_{k+1}$ is a step length to be chosen at each trial. This approach suffers from the need to choose a step length and the feedforward structure of the trial which takes no account of current trial effects including disturbances and plant modeling errors.

The norm optimal algorithm class has the following two important properties.

- Automatic choice of step-size.
- Potential for improved robustness through the use of causal feedback of current trial data and feedforward of data from previous trials.

This is achieved by, on completion of trial $k$, computing the control input on trial $k+1$ as the solution of the minimum norm optimization problem

$$u_{k+1} = \arg\min_{u_{k+1}} \{J_{k+1}(u_{k+1}) : \ e_{k+1} = r - y_{k+1}, \ y_{k+1} = Gu_{k+1}\}$$

where the performance index, or optimality criterion, used is defined to be

$$J_{k+1}(u_{k+1}) := \|e_{k+1}\|_2^2 + \|u_{k+1} - u_k\|_U^2$$

(2)

The initial control $u_0 \in \mathcal{U}$ can be arbitrary in theory but, in practice, will be a good first guess at the solution of the problem.

This problem can be interpreted as the determination of the control input on trial $k+1$ with the properties that: (i) the tracking error is reduced in an optimal way; and (ii) this new control input does not deviate too much from the control input used on trial $k$. The relative weighting of these two objectives can be absorbed into the definitions of the norms in $\mathcal{Y}$ and $\mathcal{U}$.

The benefits of this approach are immediate from the simple interlacing result

$$\|e_{k+1}\|^2 \leq J_{k+1}(u_{k+1}) \leq \|e_k\|^2, \ \forall k \geq 0$$

(3)
which follows from optimality and the fact that the (non-optimal) choice of \( u_{k+1} = u_k \) would lead to the relation
\[
J_{k+1}(u_k) = ||e_k||^2.
\]
Hence we considering a descent algorithm as the norm of the error is monotonically non-increasing in \( k \). Also equality holds if, and only if, \( u_{k+1} = u_k \), i.e. when the algorithm has converged and no more input-updating takes place.

The control law on trial \( k + 1 \) is obtained from the stationarity condition, necessary for a minimum, by Fréchet differentiation of (2) with respect to \( u_{k+1} \) (and substitution using the plant equation and the error) as
\[
u_{k+1} = u_k + G^*e_{k+1}, \ \forall k \geq 0
\tag{4}
\]
This equation is the formal update relation for the class of norm optimal ILC algorithms.

Using \( e = r - Gu \) now gives the tracking error update relation
\[
e_{k+1} = (I + GG^*)^{-1}e_k, \ \forall k \geq 0
\]
and the recursive relation for the input evolution
\[
u_{k+1} = (I + G^*G)^{-1}(u_k + G^*r), \ \forall k \geq 0
\]

Norm optimal has a number of other useful properties. For example, monotonicity immediately shows that the following limits exist
\[
\lim_{k \to \infty} ||e_k||^2 = \lim_{k \to \infty} J_k(u_k) =: J_\infty \geq 0
\]
Also an inductive argument and the inequality \( ||y|| \leq ||G|| ||u|| \) yields the relations
\[
\sum_{k \geq 0} ||u_{k+1} - u_k||^2 < ||c_0||^2 - J_\infty < \infty
\]
\[
\sum_{k \geq 0} ||e_{k+1} - e_k||^2 < ||G||^2(||c_0||^2 - J_\infty) < \infty
\tag{5}
\]
and hence
\[
\lim_{k \to \infty} ||u_{k+1} - u_k||^2 = 0, \ \lim_{k \to \infty} ||e_{k+1} - e_k||^2 = 0
\tag{6}
\]

The properties given in (6) show that the algorithm has an implicit choice of step size (the first distinguishing property of norm optimal ILC noted above) as the incremental input converges to zero. This asymptotic slow variation is a prerequisite for convergence. Also the summation of the energy costs from the first to the last trial is bounded, as shown by (5). This implicitly contains information on convergence rates.

The result following result gives proof of convergent learning, where the notation \( \mathcal{R}(\mathcal{A}) \) is used to denote the range of an operator \( \mathcal{A} \).

**Theorem 1** If either \( r \in \mathcal{R}(G) \) or \( \mathcal{R}(G) \) is dense in \( \mathcal{Y} \), then the ILC tracking error sequence \( \{e_k\}_k \) converges in norm to zero in \( \mathcal{Y} \), i.e. the ILC algorithm has guaranteed convergence of learning.

The guaranteed convergence together with the monotonicity of the tracking error sequence represent powerful properties of the algorithm. To be applicable to an example, obviously requires a causal implementation where the presence of this property is not immediately obvious here as the relation \( u_{k+1} = u_k + G^*e_{k+1} \), although apparently of a feedback form, suggests that the relationship is not causal. For example, if \( G \) is the convolution operator in \( L^2_{\mathbb{R}}[0,T] \), with the inner product \( \langle w,v \rangle_{L^2_{\mathbb{R}}[0,T]} = \int_0^T w^T(t)v(t)dt \), described by the relation \( (Gu)(t) := \int_0^T K(t-\tau)u(\tau)d\tau \), then
\[
(G^*e)(t) = \int_0^T K^T(\tau-t)e(\tau)d\tau
\]
This means that evaluation of \( G^*e_{k+1} \) requires knowledge of future values of the tracking errors. Such data is not, of course, available in practice. The special causality structure of ILC allows, however, the transformation of the algorithm into a causal procedure, as detailed for one case of particular interest below.

In mathematical terms, the trial-to-trial error always goes to zero but this does not imply convergence of the input sequence in \( \mathcal{U} \) unless this space is chosen appropriately. As an example, suppose that both \( \mathcal{Y} \) and \( \mathcal{U} \) are \( L^2 \) type spaces and \( G \) arises from a linear time-invariant system described by a state-space model. Then if the state initial vector \( x(0) \) does not generate an output that matches the value of the reference vector at \( t = 0 \), the required \( u_\infty \) will contain distributions such as the Dirac delta function. Hence \( u_\infty \not\in \mathcal{U} \) and a proof that \( u_k \to u_\infty \) in \( \mathcal{U} \) is impossible. Consequently the following results are conditional on additional assumptions on either the input sequence applied or the plant itself. Here we use the latter and the following result deals with general convergence of the input (the proof is that for Theorem 3 in \( \text{i} \)).
Theorem 2 The sequence \( \{u_k\}_{k \geq 0} \) satisfies
\[
\lim_{k \to \infty} \|G^*(r - G u_k)\|_U = 0
\]
(7)
Also if \( G^*G \) has a bounded inverse in \( U \) the input sequence converges in norm to
\[
u_\infty = (G^*G)^{-1}G^*r \in U
\]
(8)
Also if \( \sigma := \frac{1}{\|G^{-1}\|} > 0 \), the convergence is bounded by a geometric relation of the form
\[
\|u_{k+1} - u_\infty\| \leq \frac{1}{1 + \sigma^2}\|u_k - u_\infty\|
\]
(9)
This last result only holds with the boundedness assumption imposed on the plant inverse by the assumption that \( \sigma^2 > 0 \). The following result, whose proof is that for Theorem 4 in \(^4\) relaxes this assumption.

Theorem 3 If the sequence \( \{u_k\}_{k \geq 0} \) is bounded in \( U \), the desired, or learned control, input \( u_\infty \in U \) and \( G^*G \) has range dense in \( U \), then this sequence converges to \( u_\infty \) in the weak topology in \( U \).

For finite-dimensional spaces, weak convergence is equivalent to convergence in norm. In such cases, we have also proved convergence in norm. This fact includes discrete-time systems.

The modified ILC law
\[
u_{k+1} = \alpha u_k + G^*e_k
\]
(10)
provides a link to almost singular optimal control\(^\text{10}\). Here \( \alpha \) is a relaxation parameter, as used commonly in numerical analysis to improve algorithm robustness or a forgetting factor in adaptive control. Setting \( \alpha = 1 \) recovers the case considered above and substituting from \( e = r - Gu \) in the expression for \( e_{k+1} - e_k \) gives
\[
e_{k+1} = (I + GG^*)^{-1}[\alpha e_k + (1 - \alpha)r]
\]
(11)
Theorem 1 has proved convergence when \( \alpha = 1 \). When \( \alpha \neq 1 \), it is possible to use results from the stability theory of linear repetitive processes\(^9\) to show that convergence in this case holds if, and only if, \( |\alpha| < 1 \) with non-zero limit error \( \hat{e}_\infty \in \mathcal{Y} \) given by
\[
\hat{e}_\infty = \left( I + \frac{GG^*}{1 - \alpha} \right)^{-1} r
\]
(12)
Using the plant equation, it follows that the input sequence also satisfies
\[
u_{k+1} = (I + GG^*)^{-1}(\alpha u_k + G^*r)
\]
(13)
The norm of the recursion operator \( (I + GG^*)^{-1}\alpha \) in this case is \( |\alpha| \), and it follows immediately that the input sequence converges geometrically in norm in \( U \) if, again, \( |\alpha| < 1 \) to the limit
\[
u_\infty = ((1 - \alpha)I + G^*G)^{-1}G^*r
\]
(14)
with geometric constant \( |\alpha| \). Note that the control input vector in this case converges in norm if relaxation is used. Also for convergence to a solution close to \( u_\infty \), \( \alpha \) must be chosen to be close to, but slightly less than, unity. This follows from the fact that \( u_\infty \) and \( \hat{e}_\infty \) are the solutions of the optimization problem
\[
\min_u [J(u) = ||e||^2 + (1 - \alpha)||u||^2 : y = Gu, \ e = r - y]
\]
(15)
It now follows that it is possible to make \( ||e||^2 \) arbitrarily small with controls \( u \in U \) and hence the minimum value of \( J \) tends to zero as \( \alpha \) approaches one from below. The following result can therefore be established (Theorem 5 in \(^4\)).

Theorem 4 Under the assumptions of Theorem 1, the ILC algorithm with modified update law \( (10) \) with \( |\alpha| < 1 \) converges in norm in \( U \) to a control input that produces a non-zero limit error with norm that can be made arbitrarily small by choosing \( \alpha \) arbitrarily close to unity.

If \( \alpha < 1 \) is close to unity then the control input weighing in \( (15) \) is very close to zero. This is the essential link to the almost singular or cheap control problem\(^8,\text{10}\).

For applications, the results given above have to be converted into computational procedures. Central to this is that the resulting algorithms are causal in the ILC sense. The formal definition of this property is as follows.

European Journal of Control, page 4
Definition 2 An ILC algorithm is causal if, and only if, the value of the input \( u_{k+1}(t) \) at time \( t \) on trial \( k + 1 \) is computed only from data that is available from this trial in the time interval \([0, t]\) and from previous trials over the complete trial duration \([0, T]\).

Note that this definition differs from the classical one since as data from times \( \hat{t} > t \) can be used, but only from previous trials. In the next section, we switch to the case of discrete linear time-invariant plant models as a prelude to the application studies which follow in the remainder of the paper. Consideration is also given to maximizing computational efficiency of the resulting algorithm.

A natural extension of norm-optimal ILC, termed predictive norm-optimal ILC, can also be derived. The intuition that motivated this work\(^5\) is that predictive control contains the key to improved performance. The proposed form of ‘norm optimal predictive ILC’ extends the performance criterion to take future predicted error signals into account. The extended criterion for computing the input \( u_{k+1} \) on trial \( k + 1 \) is

\[
J_{k+1}(u_{k+1,N}) := \sum_{i=1}^{N} \lambda^{i-1} (\|e_{k+i}\|^2 + \|u_{k+i} - u_{k+i-1}\|^2)
\]

This criterion includes the error not only of the next trial, but of the next \( N \) trials, together with the corresponding changes in input. The weight \( \lambda > 0 \) determines the importance of more distant (future) errors and incremental inputs compared to the present ones.

The actual ILC algorithm follows uniquely from minimization of the proposed cost criterion. It only remains to compute the minimizing input. This is done by dynamic programming. Once the input is found, a recursive formulation for the evolution of the error (and input) is computed. All interesting properties and characteristics of the norm-optimal predictive ILC algorithm can, as in the norm optimal case, be obtained by examining the properties of the operators appearing in the recursive formulation.

### III. NORM OPTIMAL ILC COMPUTATION

The following discrete linear time-invariant systems state-space model is considered

\[
\begin{align*}
x(t + 1) &= Ax(t) + Bu(t), \quad x(0) = x_0, \quad 0 \leq t \leq N_s \\
y(t) &= Cx(t)
\end{align*}
\]

where \( N_s \) is the number of samples, \( x(t) \) is the \( n \times 1 \) state vector, \( y(t) \) is the \( m \times 1 \) output vector and \( u(t) \) is the \( l \times 1 \) control input vector. Also since \( N_s \) is finite (\( N_s < \infty \) samples), introduce the supervectors

\[
y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N_s) \end{bmatrix}, \quad u = \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N_s-1) \end{bmatrix}
\]

Then, using the transition matrix solution for \( y(t) \) of (16), we can describe the process dynamics by the matrix equation

\[
y = y_0 + Gu
\]

where the \( mN_s \times lN_s \) matrix \( G \) is given by

\[
G = \begin{bmatrix} CB & 0 & \ldots & 0 \\ CAB & CB & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N_s-1}B & CA^{N_s-2}B & \ldots & CB \end{bmatrix}
\]

and

\[
y_0 = \left[ (CA)^T \ (CA^2)^T \ \ldots \ (CA^{N_s})^T \right]^T x_0
\]

In the single-input single-output (SISO) case the Toeplitz matrix \( G \) is invertible if, and only if, \( CB \neq 0 \). This matrix could also be of very large dimensions but this is not a problem as it does not appear in the final calculations. If the system has a delay, and hence \( CB = 0 \), then it can be regularized as detailed in\(^6\).

European Journal of Control, page 5
In the abstract setting, the spaces $\mathcal{Y}$ and $\mathcal{U}$ are taken as $\ell_2$ spaces of $m \times 1$ and $l \times 1$ vectors on $[1, n]$ and $[0, N-1]$ respectively. Writing the norms out as sums give the performance index as

$$J_{k+1}(u_{k+1}) = \frac{1}{2} \left( \sum_{t=1}^{N} (r(t) - y_{k+1})^T(t) Q(r(t) - y_{k+1}(t)) + \sum_{t=0}^{N-1} (u_{k+1}(t) - u_k(t))^T R(u_{k+1}(t) - u_k(t)) \right)$$

where the weighting matrices $Q$ and $R$ are of compatible dimensions, and symmetric positive definite. This is the familiar linear quadratic performance criterion from linear quadratic optimal control theory which, in effect, a combination of the optimal tracking (tracking of $r(t)$) and the disturbance accommodation problem (regarding $u_k(t)$ as a known disturbance on trial $k+1$).

Forming the block diagonal matrices $\hat{Q}$ and $\hat{R}$ with $Q$ and $R$ as the diagonal entries respectively, the definitions of the inner products in $\mathcal{Y}$ and $\mathcal{U}$ are

$$\langle y_1, y_2 \rangle_\mathcal{Y} = y_1^T \hat{Q} y_2 = \sum_{t=1}^{N} y_1(t)^T Q y_2(t)$$

$$\langle u_1, u_2 \rangle_\mathcal{U} = u_1^T \hat{R} u_2 = \sum_{t=0}^{N-1} u_1(t)^T R u_2(t)$$

The initial control $u_0 \in \mathcal{U}$ can be arbitrary in theory but, in practice, it will be chosen to be a good first guess at the solution of the problem.

The control input on trial $k+1$ that minimizes the cost function here is obtained from the stationary condition

$$\frac{\partial J_{k+1}}{\partial u_{k+1}} = -G^T Q e_{k+1} + R(u_{k+1} - u_k) = 0$$

or, since $R$ is invertible,

$$u_{k+1} = u_k + R^{-1} G^T Q e_{k+1}, \quad \forall \ k \geq 0$$

Here $R^{-1} G^T Q$ is equivalent to the adjoint operator $G^*$ with respect to the weighted inner products (21) and (22) (recall (4) from the analysis of the previous section) and we can treat the former term here as an abbreviation for the latter. Moreover, the presence of the transpose of $G$ in (24) implies that $u_{k+1}(t) - u_k(t)$ depends on values of $e_{k+l}(t)$ for $t < t \leq N$. Consequently this control law cannot be implemented but, as shown below, can be converted to one that is implementable.

The first part of Theorem 2 specialized to this case shows that the input sequence minimizes the error in a least squares sense, even if $G$ is singular or non-square. Also if $G$ is square and nonsingular, it is guaranteed that there exists a scalar $\sigma > 0$ such that $\|G u\|_\mathcal{U} \geq \sigma^2 \|u\|_\mathcal{U}$ holds, where $\sigma$ is the smallest singular value of $G$.

If $G$ has inverse with norm $\frac{1}{\sigma}$ then

$$\|e_{k+1}\| \leq \frac{1}{1 + \sigma^2} \|e_k\|$$

This is a corollary of the second part of Theorem 2 for this case and establishes that an exponential convergence rate is possible for ‘regular’ plants. Note also that $\sigma$, and hence the rate of convergence of $\|u_{k+1} - u_\infty\|$ and $\|e_k\|$, can be changed arbitrarily by varying the weighting matrices $Q$ and $R$ in the cost function. This follows from

$$u^T G^T Q u \geq \sigma^2 u^T R u, \quad \forall \ u \in \mathcal{U}$$

and then if $R = p R_0$, where $R_0$ is fixed and the scalar $p$ is a variable parameter, it follows that $\sigma = \frac{\sigma_0}{p}$ where $\sigma_0$ is the smallest singular value of $R_0$. The parameter $p$ thus provides complete control over the convergence rate: the smaller $p$ is, the faster the convergence rate of the input. For example, to obtain a guaranteed reduction of the error of about $\frac{1}{2}$ on each trial $p$ should be chosen to be of the order of magnitude of $\sigma_0$.

Simulation results suggest, however, that this may be overcautious as the smallest singular value stems from a worst-case consideration and the convergence for typical reference signals is (at least initially) much faster than is guaranteed by the bound here. This control over the convergence rate is one of the biggest advantages of this algorithm.
over alternatives where there is typically neither an exponential rate of convergence nor any possibility of increasing the speed of convergence. 

To obtain an implementable form of this algorithm, first note that the adjoint (or transpose) for the class of plants considered here involves the operations of time reversal plus an appropriate change of the state-space parameters. Hence in

\[ u_{k+1} - u_k = G^* e_{k+1} = R^{-1} G^T Q e_{k+1} \]  

(25)

the adjoint operator \( G^* \) becomes the well known costate system

\[
\xi_{k+1}(t) = A^T\xi_{k+1}(t + 1) + C^T Q e_{k+1}(t + 1), \quad \psi_{k+1}(N_s) = 0 \\
u_{k+1}(t) = u_k(t) + R^{-1} B^T\xi_{k+1}(t), \quad N_s > t \geq 0
\]  

(26)

This system has a terminal condition at \( t = N_s \) instead of an initial condition, marking it (as expected) as an anti-causal representation of the solution. It cannot therefore be implemented in this form, but a causal implementation can be found when assuming full state knowledge. The optimal control is transformed by writing for the costate

\[
\xi_{k+1}(t) = [-K(I + BR^{-1}B^T K)]^{-1} A(x_{k+1}(t) - x_k(t))] + \zeta_{k+1}(t)
\]  

(27)

and then using these last two equations and standard techniques in optimal control theory to show that the matrix gain \( K(t) \) is the solution of the familiar discrete matrix Riccati equation on the interval \([0, N_s - 1]\)

\[
K(t) = A^T K(t+1) A + C^T Q C - [A^T K(t+1) B] \\
\times (B^T K(t+1) B + R)^{-1} B^T K(t+1) A
\]  

(28)

with terminal condition \( K(N_s) = 0 \). Also the predictive or feedforward term is generated by

\[
\xi_{k+1}(t) = (I + K(t)BR^{-1}B^T)^{-1}(A^T \xi_{k+1}(t + 1) + C^T Q e_k(t + 1)) \\
\xi_{k+1}(N_s) = 0
\]  

(29)

and the input update equation now is

\[
u_{k+1}(t) = u_k(t) - \left[ (B^T K(t) B + R)^{-1} B^T K(t) \\
\times A(x_{k+1}(t) - x_k(t)) \right] + R^{-1} B^T \xi_{k+1}(t)
\]  

(30)

This is hence a causal ILC algorithm consisting of current trial full state feedback combined with feedforward from the previous trial output tracking error data. This representation of the solution is causal because (29) and (30) can be solved offline, between trials, by reverse time simulation using available previous trial data. For a time-invariant system, as here, the matrix \( K(t) \) for \( 0 \leq t < N_s \), needs to be computed only once before the sequence of trials begins.

The main disadvantage limiting the practical application of the norm optimal ILC algorithm is the large amount of computation which must be performed between each sample interval. To remedy this problem, a faster version of the algorithm can be used, which allows the majority of calculations to be performed during the design and commissioning of the control law. The remaining calculations are significantly reduced in number and consist solely of multiplications, additions and subtractions.

Implementation of the algorithm is as follows. The matrix gain \( K(t) \) defined by (28) can, as noted above, be calculated before the system operates, and hence, does not contribute to the real-time processing load. The predictive term (29) must be calculated between each trial. Note again that this equation has a terminal, as opposed to an initial, condition and must therefore be computed in descending sample order. The input update (30) must be calculated at each sample instant. It is therefore the input update equation which particularly contributes to the real-time processing load and has a significant influence on the minimum sample time that can be used in an application.

This improved implementation is derived by identifying simplifications which can be made to the computation of the original. Consider the predictive component (29); the only variables in this equation are the tracking error \( e_k \) and the predictive term itself \( \xi_{k+1} \), and all of the other terms can be combined together to produce constant matrices

\[
\alpha(t) = (I + K(t)BR^{-1}B^T)^{-1}, \quad \beta(t) = \alpha(t)A^T \\
\gamma(t) = \alpha(t)C^T Q
\]  

(31)

(32)

European Journal of Control, page 7
First level (before operation):
\[
K(t) = A^T K(t+1)A + C^T QC
- [A^T K(t+1)B (B^T K(t+1)B + R)^{-1}
B^T K(t+1)A]
\]
\[
\alpha(t) = (I + K(t)BR^{-1}B^T)^{-1}
\beta(t) = \alpha(t)A^T
\gamma(t) = \alpha(t)C^T Q
\omega(t) = R^{-1}B^T
\lambda(t) = (B^T K(t)B + R)^{-1}B^T K(t)A
\]

Second level (between trials):
\[
\xi_{k+1}(t) = \beta(t)\xi_{k+1}(t+1) + \gamma(t)\epsilon_{k}(t+1)
\]

Third level (between sampling instants):
\[
u_{k+1}(t) = u_k(t) - \lambda(t)\{x_{k+1}(t) - x_k(t)\} + \omega(t)\xi_{k+1}(t)
\]

TABLE I: Norm optimal Computation

leading to the computationally simpler predictive component equation
\[
\xi_{k+1}(t) = \beta(t)\xi_{k+1}(t+1) + \gamma(t)\epsilon_{k}(t+1) \quad (33)
\]

Exactly the same concept can be applied to the input update equation (30), resulting in the simplified input update equation:
\[
u_{k+1}(t) = u_k(t) - \lambda(t)\{x_{k+1}(t) - x_k(t)\} + \omega(t)\xi_{k+1}(t)
\]
\[
\omega(t) = R^{-1}B^T
\lambda(t) = (B^T K(t)B + R)^{-1}B^T K(t)A
\]

The resulting implementation therefore requires seven matrices \(A, B, C, \beta, \gamma, \lambda\) and \(\omega\) to be supplied to the real-time controller.

This reformulated algorithm uses significantly more memory than the original, because the memory allocation is static rather than dynamic, but norm optimal ILC can recycle memory once calculations are complete. However it is worth observing that the process of recycling the memory takes time and decreases the amount of time available for computation of the algorithm. The approach here is preferable because it is relatively easier and cheaper to upgrade memory than to upgrade the processor.

In terms of improvement in computation speed, due to the reduced number of calculations, it is possible to calculate exactly the time required to perform each algebraic operation for both the basic algorithm and its reformulation, then find the total time for each variant. However, the results of this process still ultimately depend on the characteristics of the controller, the operating system and the efficiency of the program functions. In simulation comparisons there was a factor of three increase in available speed, when using an identical setup for both algorithms and running them at maximum simulation rate, without specifying the need for strict sample intervals. However, care must be exercised when transferring this result to experimental implementation. Although some increase in speed will be achieved by the reformulated algorithm, a quantifiable guarantee is beyond the scope of this paper.

The computations required here are summarized in the following table.

IV. MEASURING ILC PERFORMANCE

In norm optimal ILC the weighting matrices \(Q\) and \(R\) are used to adjust the balance between trial-to-trial error convergence speed and robustness respectively. A critical task therefore is to investigate just how much the values

European Journal of Control, page 8
chosen affect algorithm performance. However it is first necessary to discuss exactly what ILC performance is and how to measure it.

**Note:** For the applications considered in Sections V and VII, the cost function weighting matrices $Q$ and $R$ are selected as scalars. The development below is for this case but extends naturally for plants with more than one input and/or output.

It is generally recognized that there are three variables which are of particular importance when describing the performance of an ILC algorithm\textsuperscript{14}, these being

- convergence speed,
- minimum tracking error, and
- long-term performance.

Although the instantaneous data recorded during each trial (such as input voltage and output error) is useful for analyzing the learning process and its performance, it is clearly necessary to calculate some general measure of the tracking accuracy for each trial, and observe how this changes as the number of trials increases. This can specifically indicate minimum error, time to reach minimum error (convergence speed) and any sign of performance degradation. In implementation, it is possible that the error decreases from trial-to-trial and then, after a large number of trials have elapsed, start to grow again. This is sometimes termed long term instability\textsuperscript{15,16} but is in fact a performance issue that is still not completely understood, and for the remainder of this paper is referred to as long-term performance.

As for standard linear systems, popular measures of tracking accuracy are in terms of a suitable error norm. Figure 1 shows the typical such plot for an ILC system with long-term performance degradation, where key parameters used to describe performance are also indicated. In particular, $e_1$ is the initial error norm value, $i_{me}$ is the number of trials required to reach minimum error, $e_{min}$ is the minimum error norm value and $i_u$ is the number of trials until the error norm begins to increase. The typical plot for a system without long-term performance degradation is similar, except that the $i_u$ point is never reached and the error norm does not increase as the number of trials increases. Of these parameters, $i_{me}$ and $e_{min}$ are most commonly used to describe ILC performance.

![FIG. 1: Typical error norm curve for an ILC system with long-term performance degradation.](image)

The performance index that will be adopted involves simply integrating the area under the error norm curve for the first $N$ trials, where $N$ is selected appropriately for the system being considered. This results in the performance index for $N$ trials, $PI_N$, given by

$$PI_N = \sum_{k=1}^{N} p(e)_k$$

where $p(e)_k$ denotes the error norm on trial $k$.

To allow a fair comparison of algorithm performance, test parameters such as the plant model parameters and reference trajectory must be held constant. In the case that the ILC algorithm does not involve parameter-dependent current trial feedback, the value of $e_1$ will be theoretically parameter-invariant, and the $PI_N$ can be normalized by setting $e_1 = 1$. Since this does not hold for norm optimal ILC, $PI_N$ will instead be normalized using the error norm produced in the absence of ILC current trial feedback. Approximate upper and lower bounds on the value of $PI_N$ are then respectively given by $PI_N(max) = Ne_1$ (assuming that no improvement occurs for $k > 1$), and $PI_N(min) = e_1$ (assuming that the error is zero for $k > 1$).
V. APPLICATION TO A GANTRY ROBOT TEST FACILITY

The gantry robot, shown in Figure 2, is a commercially available system found in a variety of industrial applications whose task is to place a sequence of objects onto a moving conveyor under synchronization. The sequence of operations is that the robot collects the object from a specified location, moves until it is synchronized (in terms of both position and speed) with the conveyor, places the object on the conveyor, and then returns to the same starting location to collect the next object and so on. This is sometimes referred to as ‘pick and place’ and is clearly suitable for the application of ILC. In particular, the time taken to re-initialize to the starting location after placing an object on the conveyor can be used to update the control input for the next ‘pick and place’ trial.

The gantry robot can be treated as three SISO systems (one for each axis) which can operate simultaneously to locate the end effector anywhere within a cuboid work envelope. The lowest axis, X, moves in the horizontal plane, parallel to the conveyor beneath. The Y-axis is mounted on the X-axis and moves in the horizontal plane, but perpendicular to the conveyor. The Z-axis is the shorter vertical axis mounted on the Y-axis. The X and Y-axes consist of linear brushless dc motors, while the Z-axis is a linear ball-screw stage powered by a rotary brushless dc motor. All motors are energized by performance matched dc amplifiers. Axis position is measured by means of linear or rotary optical incremental encoders as appropriate.

To implement norm optimal ILC, it is necessary to obtain a model for the plant which is to be controlled. Each axis of the gantry was modeled independently by means of sinusoidal frequency response tests. From this data it was possible to construct Bode plots for each axis and hence determine approximate transfer-functions. These were then refined, by means of a least mean squares optimization technique, to minimize the difference between the frequency response of the real plant and that of the model. The resulting X-axis Bode plot comparing the plant and the model is given in Figure 3 (the remaining plots appear in13). From each Bode plot an approximate transfer-function was constructed and from that, a minimal order state-space model. Here we only give detailed ILC design for the X-axis where a 7-order transfer-function was used in design, and the efficient implementation procedure17 was used (with all axes) to obtain the results given below.

A. Test Parameters

With all axes operating simultaneously, the reference trajectories for the axes produce a three dimensional synchronizing ‘pick and place’ action, shown in Figure 4. The trajectories generate a work rate of 30 units per minute which is equivalent to a trial time period of 2 seconds. Using a sampling frequency of 1kHz, this generates 2000 samples per trial.

A 2 second stoppage time exists between each trial, during which the next input to the plant is calculated. The stoppage time also allows vibrations induced in the previous trial to die away and prevents their propagation between trials. Before each trial, the axes are homed to within ±30 microns of a known starting location to minimize the effects of initial state error.

The plant input voltage for the first trial is zero. Therefore the algorithm must learn to track the reference in its entirety. There is no assistance from any other form of controller. In the practical implementation, the system states are estimated by means of a full-state Kalman estimator.

FIG. 2: The gantry robot

European Journal of Control, page 10
FIG. 3: Bode gain (top) and phase (bottom) plots for the X-axis.

FIG. 4: Three dimensional reference trajectory

B. Experimental Results

As representative results from the experiments performed, Figure 5 shows the error norm calculated for each axis during a 5000 trial test designed to investigate the long-term performance, where for the X and Y-axes $Q = 100$ and $R = 0.01$ and for the Z-axis $Q = 1000$ and $R = 0.1$. The mean square error (mse) has been used to enable comparison with other control methodologies that have been implemented on the gantry robot\textsuperscript{13,17}, and the most important feature here is that there is no sign of an increasing error norm, indicating that the algorithm can achieve long-term performance compared to other algorithms implemented on the same plant which resulted in error divergence occurring after 100 or, in severe cases, just 3 trials.

To investigate the effect of varying $Q$ and $R$, a batch of tests was performed using different combinations of these parameters. Table II displays the values of $Q$ and $R$ which were used to produce a total of 56 combinations. Each combination was implemented for 100 trials and the $PI_{100}$ performance index described in Section IV was calculated. Given that there are 2 tuning parameters, it is particularly suitable to plot the algorithm performance on a three dimensional surface chart, as shown in Figure 6 for the X-axis. The performance plots corresponding to the other two axes are very similar, particularly the Y-axis where the low frequency gain of the linear motor is practically identical to that of the X-axis.

Noting that $Q$ affects the rate of error reduction and $R$ limits the input change, interpreting the plots becomes a simple task. To the right of the chart is a region of poor tracking performance where the $PI_{100}$ value is approximately 100 indicating that virtually nothing is learnt during the test. As could be expected, this corresponds to a small value for $Q$ and a large value for $R$. With these settings, the algorithm is far too conservative. As the ratio of $Q$ to $R$ increases, gradually $PI_{100}$ reduces, indicating that the performance is improving. This is represented by the slope to the right side of the chart. As the $Q/R$ ratio continues to increase, $PI_{100}$ is reduced to values very close to 1, indicating that the perfect trajectory is learnt in almost one trial. The balance of error reduction to input change
FIG. 5: Plot of the error norm over 5000 trials.

\[
\begin{array}{|c|c|}
\hline
Q & R \\
\hline
0.1 & 0.0001 \\
1 & 0.001 \\
10 & 0.01 \\
100 & 0.1 \\
1000 & 1 \\
10000 & 10 \\
100000 & 100 \\
\hline
\end{array}
\]

TABLE II: \(Q\) and \(R\) values used in experiments

is now approaching optimality. Temporarily increasing \(Q/R\) has little effect on the performance, until the system becomes unstable and \(PI_{100}\) jumps back to 100. This is represented by the channel and then the steep slope to the left of the chart. It is important to note that the ratio of \(Q\) to \(R\) is what determines the algorithm performance rather than the absolute values of each parameter.

FIG. 6: X-axis \(PI_{100}\) for various \(Q\) and \(R\)

The results that have been presented form part of an extensive programme of experiments\(^{17}\), which have confirmed the excellent performance of norm optimal ILC in terms of convergence speed, minimum error and long term performance. The approach has proven well suited to the gantry robot due to the accuracy of the linear model of each axis, and the lack of interaction between axes. In the next section the algorithm is applied to an actual physical system associated with free electron lasers. Then in the following section will be applied to control the movement of human
Subjects, this being a challenging application in which there exists strong nonlinear behavior, and significant difficulty in obtaining an accurate system model.

VI. APPLICATION TO FREE ELECTRON LASERS

The results in this section are in the main from\textsuperscript{18}.

Free Electron Lasers use linear particle accelerators that increase the energy of the electrons by interaction with electromagnetic radio frequency (RF) fields\textsuperscript{19}. They are operated in pulsed mode, e.g. every second there is a pulse for approximately one millisecond. This pulsed system has the following properties:

- the characteristic disturbances and uncertainties only show small changes from pulse to pulse,
- between pulses, several hundred milliseconds could be used for computing optimal parameters and driving signals for the next pulse,
- the Field Programme Gate Array (FPGA) structure of the digital intra pulse controller allows arbitrary input signals at a frequency of 1 MHz,
- appropriate models could be identified by standard methods from measurement data.

Clearly the second of these properties, in particular, makes this an application for which ILC appears to be well suited. In this section, we describe the application of norm optimal ILC using the same implementation as for the gantry robot and begin with some background leading on to the construction of a model for design.

A. Basic System Structure

In numerous research areas, a light source that is able to resolve objects on an atomic level is required, e.g. in molecular biology. Also laser light is used for a variety of experiments because it can be better focused compared to other light sources, is monochromatic, and very short pulses can be produced.

A Free Electron Laser (FEL) produces laser radiation with tunable wavelength. Here we focus on a research program that is aiming to build an FEL that can operate in the X-ray wavelength by the year 2012\textsuperscript{19}. The process uses a linear particle accelerator, which increases the energy of electrons by interaction with electromagnetic radio frequency (RF) fields to a desired value. These fields are required to be very precise in amplitude and phase stability. Figure 8 shows the structure of the Vacuum Ultraviolet FEL (FLASH) which is already working in the same establishment where this research program is based. The linear accelerator consists of resonators for the RF-fields housed in shape cryomodules. The RF-fields inside these superconducting resonator cavities are supplied by an actuator system for a finite time interval and then turned off again periodically.

In Figure 7 the amplitude of the desired envelope of the RF-field is displayed for one RF-pulse as a function of time. The field inside the accelerator cavities has to be kept constant once the required amplitude for the appropriate energy gain of the electrons has been reached at the end of the so called filling phase. During the flat top phase the...
electron beam is injected into the accelerator. When the electron beam has passed, the RF-field is turned off and the field amplitude decays. The envelope of the RF-field oscillation must be kept constant in amplitude and phase during the flat top time interval to transfer a precise amount of energy to the electrons.

Once the system is set up to a desired operating point, the pulse trajectory remains unchanged for a large number of pulses. Therefore repetitive disturbances can be suppressed by finding an optimal feedforward control signal to minimize the deflection from the reference. The adaption of the driving signal is calculated using norm optimal ILC, for which a system model must be developed.

### B. RF System of the Linear Accelerator

The acceleration takes part in the resonators where standing RF-waves (modes) provide the energy. The resonance frequency is determined by the geometry, and for the desired acceleration mode is $1.3\,\text{GHz}$. If the length of the cavity changes, the resonance frequency changes as well. Due to the relatively thin walls, the resonators become susceptible to mechanical vibrations called microphonics which detune the resonance frequency. The high power RF-fields in the cavities lead to deformation of the cavity walls and therefore detuning as well. Induced currents cause Lorentz forces acting on the metal surroundings during the pulse sequence. Measurements have shown that detuning can take up to $\Delta f \approx 500\,\text{Hz}$. Since the Lorentz force is induced every time the electric field is generated, the Lorentz force detuning is considered deterministic and repetitive.

Another source of disturbance is the electron beam itself. While it is passing the accelerating structure, the charged particles gain energy from the present RF-field which leads to fluctuations in the present amount of energy stored in the system. The following bunches of charged particles will be influenced by these fluctuations, which have to be minimized by the control system. It can be assumed that the bunch arrival time will be constant from pulse to pulse, thus having the properties of a repetitive disturbance. Next we describe the general architecture and related aspects of the digital control scheme used in this application.

### C. Digital Control System

The actuator system receives a precise RF signal of $1.3\,\text{GHz}$ from the Master Oscillator (MO). This low power sinusoidal signal can be changed by the vector modulator in amplitude and phase. The output signal of the vector modulator is amplified by a klystron, which is a radio frequency amplifier. The amplified RF waves are transferred from the klystron to the cavities inside the cryomodules via a waveguide transmission and distribution system. For economical reasons, one high power klystron supplies all $8 - 32$ cavities of an RF station, thus RF fields can not be influenced in each cavity individually and the system is therefore underactuated.

The superconduction cavity simulator and controller (SIMCON) is based on FPGA structures that enables the implementation of fast algorithms. A block diagram Low Level Radio Frequency (LLRF) control system is shown in Figure 9, where the bottom part shows the digital FPGA controller. The LLRF control system has the task of keeping the pulsed RF fields in the superconducting cavities of the RF station at the reference value during the flat top phase of one RF pulse shown in Figure 7.

FIG. 8: An FEL with front end, accelerating structures and undulators.
After measuring the actual RF-field by pickup antennas, the signals are downconverted to an intermediate frequency of 250 kHz. The real (I) and imaginary (Q) field components are digitalized in analog-digital-converters (ADC) with a sampling frequency of 1 MHz. An overview of the signals shown in Figure 9 are represented in terms of I and Q is given next.

- The input signals \( u_I, u_Q \) are produced by actuator system and act directly on the vector modulator.
- The output signals \( y_I, y_Q \) are the real and imaginary parts respectively of the sum of the RF-field voltage vectors of eight cavities.
- The reference signals \( r_I, r_Q \) are the real and imaginary parts respectively of the vector sum of the RF-field’s voltage vectors given by look-up tables for the specified field gradient.
- The feedforward signals \( f_I, f_Q \) are among the control signals determined by open-loop control.
- The control signals \( u_{c,I}, u_{c,Q} \) are the ILC controller output signals, updating the previous, trial, or iteration, input signals.
- The control error signals \( e_I, e_Q \) are the deviations in real and imaginary parts respectively of the output signals from the reference signals.

Calibration of the measurement signals is done for compensation of effects resulting e.g. from different cable lengths. The control algorithm usually uses the vector sums of all calibrated measurement signals of the individual cavities as signals to be controlled, because of the lack of individual action for each cavity. To reach the desired setpoint values even in open loop requires that an adaptation of the feedforward signals is sufficient to track the reference trajectory. As far as the disturbance sources are concerned, mainly the strong effect of the Lorentz forces which are deterministic from pulse-to-pulse. It is possible to compensate the drift away from the operating point by a smooth increase of the driving signal over the flattop. Transients induced by the beam are predictable, and the arrival time is known. An increase in the driving power will keep this fluctuation low. Both compensation measures have a positive effect on a feedback control performance while keeping the control error signals small. To predict the system behavior, it is essential to have an adequate model of the underlying system dynamics. Next we give an outline of the identification procedure used for model building.

### D. SYSTEM MODELING

Although additional external disturbances and a number of nonlinearities in the actuator system are known to be relevant for a broad range of operation setpoints, standard identification procedures for linear time invariant models can be used to estimate models that can be validated at specific setpoints in the manner outlined in, for example,\(^\text{20}\). The subspace identification method \( \textit{N4SID} \) from Matlab’s System Identification Toolbox\(^\text{21}\) was used to estimate the
matrices $A, B, C, D$ of the state-space model

$$
\begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) \\
y(t) &= Cx(t) + Du(t)
\end{align*}
$$

where $u = (u_I, u_Q)^T$ and $y = (y_I, y_Q)^T$ denote the system input and output vectors respectively and $x$ the state vector. The flat top phase of the pulse is the main interest for control and is at the same time marking the systems operation point. Only measurements from this period are used for system identification. Persistent excitation signals can be injected into the accelerator system by superimposing random signals on standard feedforward tables with defined setpoints. A typical input sequence for the feedforward table is shown in Figure 10.

In the first $500\mu$s (filling phase), the actuator system is operated at maximum power. In the flat top phase starting from $500\mu$s, the inputs are first reduced by a factor of 0.5 to reach the setpoint and soon after, the excitation signal is added to both inputs, see Figure 10. A high amplitude leads to a good signal to noise ratio. In Figure 11 measured vs. simulated signals are shown for an identified 3rd order model.

E. Application of Norm Optimal ILC

The system to be controlled is driven in pulsed mode. Moreover, the accelerating process is considered to be repetitive and the disturbances also show this behavior. Hence ILC is suitable for this application where $e_k = \begin{bmatrix} e_I & e_Q \end{bmatrix}^T$ and here the norm optimal approach with cost function of the form (20) is employed. Fast and efficient computation is important when computing the updated input signal before the next trial starts. The time interval between two consecutive RF-pulses is approximately $0.1\,\text{s}$. Furthermore, a state feedback control input has to be
TABLE III: Available and required computation time for different levels

<table>
<thead>
<tr>
<th></th>
<th>1st level</th>
<th>2nd level</th>
<th>3rd level</th>
</tr>
</thead>
<tbody>
<tr>
<td>before operation</td>
<td>24.71</td>
<td>0.0147</td>
<td>0.0047</td>
</tr>
<tr>
<td>between trials</td>
<td>∞</td>
<td>0.1</td>
<td>0.000001</td>
</tr>
<tr>
<td>between samples</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

computed in each sampling period of 1 µs. Using a workstation computer with an Intel® Pentium® 4 processor with a clock speed of 3 GHz, the computation time needed for the three different time levels is given in Table VIE.

Note here that the time needed between samples exceeds the time that is available. As a consequence, the third level computations are executed in the second level for the results below. The input signals are then computed for the whole trial at once, instead of for every sample separately. To prevent damage to system components when the algorithm is implemented on the real plant, the input signals are limited as shown in Fig. 12. The limits are set to the maximum and minimum values of the input signals during the filling and the decay phase.

The following weighting matrices gave results which could be implemented on the real plant

\[ Q = 100 \times I_{2 \times 2} \quad \text{and} \quad R = I_{2 \times 2} \]  \hspace{1cm} (37)

The state variables required for the state feedback component in the algorithm are obtained by simulation using the plant model. Including input and output disturbances in the simulation, the results for the flat top phase are shown in Fig. 13. Without losing information, only the signals of the first input and output are given. The shape of the simulated disturbances can be computed as the deviation from the smooth trajectory of the first trial. As the number of trials increases, the output signals approach the desired setpoint (SP) trajectory. Rejection of the included input and output disturbances can also be observed. Since the input signal reaches the given limits in the beginning of the phase, the output signal only approaches the setpoint slowly in the first 100 µs.

F. Experimental Results

The norm optimal ILC scheme designed as explained in this paper has been successfully implemented on the real plant at the DESY test facility. The results are shown in Figures 14 to 16. The state variables required for the state feedback component are computed using the identified model of the plant. The weighting matrices are set to the values in (37). Because of the large computation time the input signals are computed for a whole trial at each trial. The number of trials is set to 10.

Figures 14 and 15 show an increasing and decreasing trend of the first and the second output signal during the flat top phase, respectively. This is the general behavior caused by the detuning effects described before. However, increasing the number of trials, both output signals approach the desired setpoint. After 10 trials, the output signals show only small deviations from the reference trajectory. Since only the signals during the flat top phase are controlled, the input signals of the filling and the decay phase are kept constant as illustrated in Figure 16.

Considering the output signals \( y_I \) and \( y_Q \) as the real and imaginary part of an output vector, respectively, the amplitude and phase of the vector can be computed. In order to evaluate the performance of the norm optimal ILC,
the peak–to–peak error of the amplitude ($A$) is computed as follows as
\[
\epsilon_{A,p2p} = \frac{\|\max(e_A(t)) - \min(e_A(t))\|}{\|\text{setpoint}_{A}\|},
\]
with $e_A(t) = y_A(t) - r_A(t)$, where $y(t)$ and $r(t)$ are the output and reference signals respectively. The RMS-error is computed as
\[ e_{A, \text{rms}} = \sqrt{\frac{1}{T_{ft}} \int_{t_{0,ft}}^{t_{0,ft}+T_{ft}} e_A^2 \, dt} \]  

(39)

with \( t_{0,ft} \) defining the beginning of the flat top phase and \( T_{ft} \) the time interval of the flat top phase. For the phase \( (\phi) \) the errors are computed similarly. The results are given in Table VIF.

<table>
<thead>
<tr>
<th></th>
<th>Amplitude ( A )</th>
<th>Phase ( \phi )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Norm optimal ILC</td>
<td>1.06 %</td>
<td>31.76</td>
</tr>
<tr>
<td></td>
<td>2.42 %</td>
<td>0.2879°</td>
</tr>
<tr>
<td>P-controller</td>
<td>0.5 %</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>1 %</td>
<td>0.1</td>
</tr>
<tr>
<td>Objectives</td>
<td>( \leq 0.1 % )</td>
<td>( \leq 0.01 % )</td>
</tr>
</tbody>
</table>

TABLE IV: Performance objectives and performance achieved with norm optimal ILC

For comparison, the defined objectives as well as the performance achieved using the already implemented proportional feedback controller are also given in the table. It can be seen that the performance of the norm optimal does not reach the given objectives in open loop without feedback. However, the tracking errors of the norm optimal after only 10 trials are close to the values that were achieved with the proportional controller and close to the desired values defined by the objectives.

In summary, the experimental results show performance improvements are possible with norm optimal ILC of the system to be controlled, even though only tests with a small number of trials could be executed. It has not yet been possible yet to investigate the performance of the controller when an electron beam is injected into the system, but experiments including the beam loading are planned. For further improvement of the performance, tests with an increased number of trials will be performed to confirm long term performance. Moreover, implementing the control algorithm in FPGA will decrease the computational time needed so that input signals can be computed in every sampling period. A combination of the ILC with the existing proportional controller or a multiple-input multiple output linear time-invariant controller which is currently being developed\(^22\), is expected to further increase the control performance.

VII. APPLICATION TO STROKE REHABILITATION

There are 300,000 people in the UK living with moderate to severe disabilities as a result of a stroke\(^23\), of which 85% had an initial deficiency in the upper limb\(^7\) and less than 50% have recovered useful upper limb function\(^24,25\). These demographics reflect those across the European Union and, due to an aging population and better acute care, prevalence of stroke is likely to increase. Electrical Stimulation (ES) has been shown to improve motor control in a growing number of clinical applications\(^26,27\), and the approach is supported by neurophysiological\(^28\) and motor learning research\(^29\). There is also clear evidence that recovery is enhanced when stimulation is applied coincidentally with a patient’s own voluntary intention whilst performing a task\(^30\).
The need to accurately apply ES to achieve a movement has motivated significant interest in the development and application of techniques that can provide a high level of precision. A range of model-based control schemes have been proposed, including optimal, $H_\infty$, and fuzzy control of standing, sliding mode control of shank movement, and data-driven control of the knee joint. Artificial neural networks are a popular approach, and have been applied to both the upper and lower limbs of paretic subjects.

However, these control laws have not transferred to clinical practice, a setting in which there is very limited time available for tuning parameters, and in which a high level of performance is required over a wide range of participants and tasks. Only open-loop methods, or those triggered through voluntary activity (such as the myoelectric activity of the same muscle) have been used in patient trials, but cannot provide the precise control over the applied stimulation necessary to fully exploit the association between the intended movement of the subject and the application of stimulation that allows them to achieve it. In order to maximize the potential of the treatment, it is necessary to apply simulation that results in accurate tracking in a small number of trials, but also exerts close control over the level of stimulation and error attained in order to promote the necessary sustained voluntary effort by the subject.

In an ongoing research programme, a workstation has been designed and constructed in order to provide the controlled environment needed to assist patients in performing reaching tasks. ILC has enabled a high level of tracking performance to be achieved in this demanding setting, and recent clinical trials of 8 week duration have been conducted with five stroke participants. Following these, results have indicated statistically significant improvement in several areas, including patients’ level of unassisted tracking and their ability to exert isometric force.

**A. System Description**

The task presented to the seated patient is to track reaching trajectories using their impaired arm. So that the objective is presented with maximum clarity, only trajectories in a fixed horizontal plane are used and the patient’s hand is constrained to move in this plane by a custom-built robot. A data projector mounted above the subject shines an image of the trajectory path, as well as a moving spot to indicate the reference point, onto a target mounted above the subject’s hand. In response to clinical need, ES is applied to the triceps muscle to provide forearm extension. A patient using the workstation is shown in Figure 17. The stimulation comprises a sequence of bi-phasic pulses at 40Hz with pulsewidth, $u$, in the range 0 to 350 $\mu$ seconds (resolution 1 $\mu$ second). The subject’s arm is strapped to a five-bar robotic arm which is actuated using two dc brushless servomotors, and a force/torque sensor, situated between the penultimate and final links, measures the force applied by the subject.

A model of the combined human arm and robotic manipulator system is shown in Figure 18, in which the fifth robotic link is strapped to the human arm. The robotic arm has a base co-ordinate system with components $x_0$, $y_0$ and $z_0$. Likewise the human arm has a base co-ordinate system with components $x_0'$, $y_0'$ and $z_0'$. The stimulated
The upper arm angle. This was achieved through selection of variables in (42) to result in the relationship
\[ B(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(q, \dot{q}) = \tau + J^T(q)h \] (40)

where
\[ q = \begin{bmatrix} \dot{\theta}_u \\ \dot{\theta}_f \end{bmatrix}, \ B(\cdot) = \begin{bmatrix} b_1 & b_2 \\ b_2 & b_3 \end{bmatrix}, \ C(\cdot) = \begin{bmatrix} -2c_1\dot{\theta}_f - c_1\dot{\theta}_f \\ c_1\dot{\theta}_u \end{bmatrix}, \ F(\cdot) = \begin{bmatrix} F_u(\theta_u, \dot{\theta}_u) \\ F_f(\theta_f, \dot{\theta}_f) \end{bmatrix}, \ \tau = \begin{bmatrix} 0 \\ -T\sigma(\dot{\theta}_f) \end{bmatrix} \] (41)

and \( J(\cdot) \) is the Jacobian\(^39\). Here \( \tau \) is the moment produced by the stimulated muscle, with \( \sigma(\dot{\theta}_f) = s_f c_2/\sqrt{1 - c_2^2 c_3^2} \) representing the \( z_0 \) component of a unit vector parallel to the elbow axis. The variables \( \dot{\theta}_u \) and \( \dot{\theta}_f \) denote the upper arm angle and forearm angles respectively. The robotic arm has link lengths \( l_1 = 0.45m, l_2 = 0.2m \) and \( l_3 = 0.66m \), and the subject interacts with the robot by applying a force with components \( F_x \) and \( F_y \), in the directions of \( x_0 \) and \( y_0 \) respectively, at the point \( P \) which has a \( z_0 \) component of \( l_z \) in this system. A form of impedance control is used to govern the torque demand supplied to the motors and ensure safe interaction with the patient\(^42\). This produces the relationship
\[ -h = \begin{bmatrix} F_x \\ F_y \end{bmatrix} = K_{Kx} \ddot{x} + K_{Bx} \dot{x} + K_{Mx} x \] (42)

at \( P \), where \( \ddot{x} = \ddot{\theta} - \ddot{x}, \dot{x} \) is the robot reference, \( x = k(q) \) is the direct kinematics equation for the human arm, and \( K_{Kx}, K_{Bx} \) and \( K_{Mx} \) are gain matrices.

Control laws using ES for actuation of the forearm have been tested using unimpaired subjects who apply no voluntary effort, hence providing no means for control over upper arm movement. Rather than fixing the upper arm and restricting the reference trajectories used, the robotic arm was instead used to provide the necessary control over the upper arm angle. This was achieved through selection of variables in (42) to result in the relationship
\[ h = J^{-T}(q) \begin{bmatrix} K_{Kx} & 0 & 0 \\ 0 & K_{Bx} & 0 \\ 0 & 0 & K_{Mx} \end{bmatrix} \dot{\theta} + \begin{bmatrix} K_{B1} & 0 \\ 0 & K_{B2} \end{bmatrix} \dot{q} + \begin{bmatrix} K_{M1} \\ 0 \\ 0 \end{bmatrix} \dot{q} \] (43)

Here \( \dot{x}, K_{Kx}, K_{Bx} \) and \( K_{Mx} \) are used to impose a moment about the upper arm to keep the target on the trajectory path, while dynamics about the stimulated forearm are specified by \( K_{B1} \) and \( K_{M2} \). If the reference path is then

European Journal of Control, page 21
chosen to provide a smooth upper arm component variation with respect to that of the forearm, the joint coupling terms in (40) may be neglected and the forearm dynamics assumed independent of the upper arm. The resulting forearm dynamics are given by

\[ b_3 \ddot{f} + F_f(\dot{f}, \ddot{f}) = -T\sigma(\ddot{f}) + K_{B_f}\dot{f} + K_{M_f}\ddot{f} \]  

(44)

and a complete derivation appears in\(^\text{43}\). The moment, \(T\), developed by the stimulated triceps muscle may be accurately modeled using

\[ T(\beta, \dot{\beta}, u, t) = g_m(u, t) \times F_m(\beta, \dot{\beta}) \]  

(45)

where \(g_m(u, t)\) is a Hammerstein structure incorporating a static nonlinearity, \(f_m(u)\), representing the isometric recruitment curve, cascaded with a critically damped second order system\(^\text{44}\). This is one of the few reported models that may be identified using excitation inputs suitable for application to stroke patients, although recent work has begun to develop alternative models for this application\(^\text{45}\). Since the elbow angle is given by \(\beta(\theta, t) = \arccos(-c_f c_{\gamma})\), the function \(F_m(\beta, \dot{\beta})\) may be written as \(\hat{F}_m(\dot{\beta}, \ddot{f})\). Combining (44) and (45) results in the electrically stimulated arm system shown in Figure 19. To identify the necessary parameters, the subject’s arm is first moved about the workspace to yield the geometric relationships appearing in Figure 18 via nonlinear optimization. The arm is then held stationary while ES is applied in order to identify parameters in the Hammerstein model, \(g_m(u, t)\), using deconvolution and a nonlinear optimization procedure. Stimulation sequences and kinematic trajectories, imposed on the arm by the robot, are then applied, and an LMS optimization is used to provide all those parameters appearing in (40). Full details of the identification for both stroke and unimpaired subjects can be found in\(^\text{19}\).

**B. ILC Implementation**

Two approaches involving norm optimal ILC have been implemented to control the human arm. In the first of these the system is linearized through the introduction of a controller to compensate for the effect of the nonlinear terms prior to the application of norm optimal ILC. The output of this controller, \(u\), is the stimulation signal applied to the triceps and it comprises of i) the inverse of the recruitment curve, \(f_m^{-1}(\cdot)\), ii) the inverse of the muscle multiplier, \(1/F_m(\cdot)\), and iii) the inverse of the elbow joint angle effect, \(1/\sigma(\cdot)\). Since these functions vary only slowly when the trajectory is tracked\(^\text{20}\), the control action is to cancel their effect whilst neglecting the intervening muscle dynamics. This approach has been supported both theoretically and experimentally\(^\text{43}\), and results in the linear system approximation

\[ G(s) = \frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2} \cdot \frac{1}{s((b_3 + K_{M_f})s + K_{B_f})} \]  

(46)

A feedback controller is then implemented to act as a pre-stabilizer and provide satisfactory tracking over the initial trial, as shown in Figure 20. Implementation of norm optimal ILC is then achieved through discretization of the closed-loop system, and the states required in practice are provided by a Kalman filter based state estimator. Experimental results confirming the efficacy of this approach are given in Section VII.C.

*European Journal of Control, page 22*
The second approach does not attempt to compensate for the plant nonlinearity. Instead, Newton method based norm optimal ILC is applied to the time-varying linear model formed by linearizing the system shown in Figure 19 along each trial. Here the starting point is a discrete-time system representation, and accordingly the linear activation dynamics are first represented by the state-space model \([\Phi_m, \Gamma_m, H_m]\). The relationship between \(w_1\) and \(w_2\) is then given by

\[
x_m(t + 1) = \Phi_m x_m(t) + \Gamma_m w_1(t), \quad x_m(0) = x_{m0}
\]

and similarly the arm dynamics are represented by the state-space model \([\Phi_p, \Gamma_p, H_p]\). Hence the relationship between \(w_3\), \(\vartheta_f\) and \(\dot{\vartheta}_f\) is given by

\[
\begin{bmatrix}
\vartheta_f(t) \\
\dot{\vartheta}_f(t)
\end{bmatrix} = \begin{bmatrix}
H_{p1} \\
H_{p2}
\end{bmatrix} x_p(t) = H_p x_p(t)
\]

The stimulated arm system on trial \(k\) is now described by

\[
x_k(t + 1) = \Phi x_k(t) + \Gamma \left[ d(x_k(t)) f_m(u_k(t)) \right] = f(x_k(t), u_k(t))
\]

\[
\vartheta_{f,k}(t) = H_{p1} x_k(t) = h(x_k(t))
\]

\[
x_k(0) = x_0, \quad t \in [0, N_s]
\]

where \(x(t) = [x_p(t) x_m(t)]^T\), \(x_0 = [x_{p0} x_{m0}]^T\), \(\Phi = \text{diag} \{ \Phi_p, \Phi_m \}\), \(\Gamma = \text{diag} \{ \Gamma_p, \Gamma_m \}\), \(\Phi_m = \begin{bmatrix} 0 & H_m \end{bmatrix}\), \(\Phi_{p1} = \begin{bmatrix} H_{p1} & 0 \end{bmatrix}\) and \(\Phi_{p2} = \begin{bmatrix} H_{p2} & 0 \end{bmatrix}\). The integer \(N_s\) is equal to \(\frac{T_s}{T_p} + 1\), where \(T_s\) is the sample time, and

\[
d(x_k) = -\hat{H}_m x_k \hat{F}_m (\hat{H}_{p1}, x_k, \hat{H}_{p2} x_k) \sigma (\hat{H}_{p1} x_k)
\]

\[
- F_f (\hat{H}_{p1}, x_k, \hat{H}_{p2} x_k)
\]

in which the explicit time dependence of \(x_k\) has been omitted.

Following the procedure given in\(^6\), (49) is replaced with a set of algebraic equations in \(\mathbb{R}^{N_s}\) by defining the shifted input and output vectors as

\[
u_k = [u_k(0), u_k(1), \ldots, u_k(N_s - 1)]^T
\]

\[
\vartheta_{f,k} = [\vartheta_{f,k}(1), \vartheta_{f,k}(2), \ldots, \vartheta_{f,k}(N_s)]^T
\]

and the relationship between the input and output time-series can be expressed by the algebraic functions

\[
\vartheta_{f,k}(1) = h(x_k(1)) = h(f(x_k(0), u_k(0)))
\]

\[
g_1(x_k(0), u_k(0))
\]

\[
\vartheta_{f,k}(2) = h(x_k(2)) = h(f(x_k(1), u_k(1)))
\]

\[
g_2(x_k(0), u_k(0), u_k(1))
\]

\[\vdots\]

\[
\vartheta_{f,k}(N_s) = h(x_k(N_s))
\]

\[
h(f(x_k(N_s - 1), u_k(N_s - 1))
\]

\[
g_{N_s}(x_k(0), u_k(0), u_k(1), \ldots, u_k(N_s - 1))
\]

and hence system (49) can be represented as

\[
\vartheta_{f,k} = g(u_k), \quad g(\cdot) = [g_1(\cdot), g_2(\cdot), \ldots, g_{N_s}(\cdot)]^T
\]

European Journal of Control, page 23
The ILC task of finding the input which drives the dynamic system (49) to track the desired output, equates to finding the solution that satisfies the nonlinear function (54) with \( \vartheta_{f,k} \) substituted by \( \vartheta_f^* = [\vartheta_f^*(1), \vartheta_f^*(2), \ldots, \vartheta_f^*(N_k)]^T \). The Newton method is selected to solve this nonlinear equation, and is given in ILC notation as

\[
u_{k+1} = u_k + \alpha_{k+1} g'(u_k)^{-1} e_k
\]

(55)

where the scalar \( \alpha_{k+1} \geq 0 \) is a relaxation parameter, and \( e_k = \vartheta_f^* - \vartheta_{f,k} \). The derivative \( g'(u_k) \) is equivalent to the linearization of (49), on trial \( k \) at \( (u_k, x_k) \) which can be represented by the time-varying system

\[
\vec{\vartheta}_f = g'(u_k) \vec{u}
\]

(56)

which is approximated by

\[
\vec{x}(t+1) = A(t)\vec{x}(t) + B(t)\vec{u}(t) \\
\vec{\vartheta}_f(t) = C(t)\vec{x}(t)
\]

(57)

with

\[
A(t) = \left( \frac{\partial f}{\partial x} \right)_{u_k(t),x_k(t)} = \Phi + \Gamma \begin{bmatrix} p(t) \\ 0 \end{bmatrix} \\
B(t) = \left( \frac{\partial f}{\partial u} \right)_{u_k(t),x_k(t)} = \Gamma \begin{bmatrix} 0 \\ f_m'(u_k(t)) \end{bmatrix} \\
C(t) = \left( \frac{\partial h}{\partial x} \right)_{u_k(t),x_k(t)} = H_{p1}
\]

(58)

where \( \vec{x} = x_{k+1} - x_k, \vec{u} = u_{k+1} - u_k, \vec{\vartheta}_f = \vartheta_{f,k+1} - \vartheta_{f,k}, \vec{x}(0) = x_{k+1}(0) - x_k(0) = 0 \) and

\[
p(t) = -H_m\vec{F}_m\sigma - H_mxH_{p1}\vec{F}_m'\sigma - H_mx\vec{F}_mH_{p1}\sigma' - H_{p1}\vec{F}_f' + H_mxH_{p2}\vec{F}_m\sigma - H_{p2}\vec{F}_f
\]

(59)

Here \( \ast \) denotes differentiation with respect to the second variable, and the functional dependence has been omitted. If the system (56) can be forced to track \( e_k \), then the corresponding input is \( \vec{u} = g'(u_k)^{-1} e_k \) which can then be used in the update (55). Since this is itself an ILC problem, the norm optimal approach is an obvious candidate for application to the time-varying system (57). Proceeding accordingly, the input to (56) on trial \( m+1 \) of norm optimal ILC is chosen to minimize

\[
J_{m+1} = \frac{1}{2} \left[ \sum_{t=1}^{N_s} [e_m(t) - \vartheta_{f,m+1}(t)]^T Q [e_m(t) - \vartheta_{f,m+1}(t)] \right]
\]

\[
+ \sum_{t=0}^{N_s-1} [\vec{u}_{m+1}(t) - \vec{u}_m(t)]^T R [\vec{u}_{m+1}(t) - \vec{u}_m(t)]
\]

(60)

and \( M \) trials are performed in order to obtain the input \( \vec{u} \). Its subsequent application in (55) again means the ability of norm optimal ILC to transparently influence the error and stimulation levels, is able to directly translate into control over the stimulation input. Hence the potential for rehabilitation can be maximized.

C. Human Subject Results

The control laws were tested on a 60 year old subject (and so age-matched with stroke participants). The reference path used comprised the extension component of elliptical reaching trajectories, as shown in Figure 21, which were similar to those used in the clinical trials. The reference was followed at two constant speeds to produce 2.5 and 5 second duration movements, which were then preceded by a 5 second stationary period to produce ‘slow’, and ‘fast’ trajectories. The goal of the two ILC approaches is for the forearm angle to track the forearm component, \( \vartheta_f^* \), of these trajectories. In both cases \( N = 20 \) trials were performed since this is the maximum number used in clinical practice. Due to the time required to identify the stimulated arm model, as well as the effect of muscle fatigue, it
was not possible to perform sufficient tests on the subject to generate a $PI_{20}$ plot similar to Figure 5, and gains were instead selected heuristically. A sample frequency of 40Hz was used in all tests undertaken.

Focusing initially on the linearized system implementation, a proportional plus derivative (PD) controller was first selected with values of $K_p = 6$, $K_d = 0.5$ chosen to affect a compromise between disturbance rejection and tracking performance. To provide the controlled system state vector elements, a Kalman estimator was implemented, using output and state covariance weights of 1 and 10 respectively.

Figures 22 and 23 show error norm results for the slow and fast trajectories respectively, each using $Q$ values of 2 and 10, with $R$ set at unity. The root mean square (rms) error is used to allow comparison with simpler ILC laws. The larger $Q$ value leads to faster convergence in each case, although increased error fluctuation is apparent in the latter. Although larger error values occur for the fast trajectory, accurate tracking is achieved in both cases. Figure 24 shows the tracking, stimulation, and ILC input for the slow trajectory with $Q = 10$. It can be seen from b) that norm optimal ILC causes the maximum level of stimulation applied to decrease as the trials progress, and leads to its application during the initial 5 second waiting period. Figures 24 a) and c) illustrate the convergence of the forearm angle, $\vartheta_f$, and norm optimal ILC update, $u_k$, respectively.

Turning to the Newton method, the norm optimal ILC parameters were selected as $M = 30$, $Q = 50000$ and $R =$
1 in order to produce an update which tracks the error well but does not produce an excessively large input for the subject. Figures 25 and 26 shows error results for slow and fast trajectories respectively, using various values of $\alpha$. In both cases $\alpha = 0.3$ yields the most rapid convergence.

Figure 27 a) shows tracking results using the fast trajectory with $\alpha = 0.3$. The reference is seen to be closely followed by the 5th trial, and the corresponding ILC update, $u_k$, (which corresponds to the stimulation pulsewidth in this open-loop system), is shown in Figure 27 b). Since it directly addresses system nonlinearity instead of using approximate cancelation, the Newton based implementation generally leads to superior tracking when applied to more rapid trajectories, and results in a smoother stimulation signal which is more comfortable for the patient.
VIII. CONCLUSIONS AND FURTHER WORK

This paper has given an overview of norm optimal ILC, starting with the theoretical foundations together with results from experimental benchmarking on a gantry robot system specially designed for this task where the pick and place operation used closely resembles task found in many industrial applications in, for example, the food processing industry. This has been followed by results from an application area involving a real process where even the first experimental test results show sufficient performance to merit further development work concerned with robotics. The final section of the paper describes a recent application to stroke rehabilitation, involving electrical stimulation applied to the human arm. The high level of performance achieved in these contrasting applications demonstrate that the potential revealed by the theory of this form of ILC can be translated successfully to application domains. In doing so, norm optimal ILC is able to provide a high level of performance that can be tuned in a transparent manner.

These results demonstrate that the ILC area in general has much potential for rapid transition from the basic theory through to applications take up. This is still a very active area, where are two examples, the recent conference papers


