Multidimensional Systems and Signal Processing manuscript No.
(will be inserted by the editor)

Multidimensional Control Systems: Case Studies in Design and Evaluation

E. Rogers · K. Galkowski · W. Paszke · K. L. Moore · P. H. Bauer · L. Hladowski · P. Dabkowski

the date of receipt and acceptance should be inserted later

Abstract Multidimensional control systems have been the subject of much productive research over more than three decades. In contrast to standard control systems, there has been much less reported on applications where the multidimensional setting is the only possible setting for design or produces implementations that perform to at least the same level. This paper addresses the latter area where case studies focusing on control law design and evaluation, including experimental results in one case, are reported. These demonstrate that movement towards the actual deployment of multidimensional control systems is increasing.

E.Rogers
Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ, UK
E-mail: etar@ecs.soton.ac.uk
K. Galkowski, W. Paszke and L. Hladowski
Institute of Control and Computation Engineering
University of Zielona Gora
65--246 Zielona Gora, Poland E-mail: k.galkowski@issi.uz.zgora.pl,
w.paszke@issi.uz.zgora.pl, l.hladowski@issi.uz.zgora.pl
K. L. Moore
Colorado School of Mines
College of Engineering and Computational Sciences
Colorado School of Mines
Golden, CO 80401 , USA E-mail: kmoore@mines.edu
P. H. Bauer
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556-5637, USA
E-mail: pbauer@nd.edu
P. Dabkowski
Institute of Physics
Faculty of Physics, Astronomy and Informatics
Nicolaus Copernicus University,
ul. Grudziadzka 5, 87--100, Torun, Poland
E-mail: p.dabkowski@fizyka.umk.pl

1 Introduction

Classical control theory studies systems governed by ordinary differential equations or difference equations in the discrete case, where the latter may result from sampling the former. Multidimensional, or nD, systems originally arose from systems described by partial differential or difference equations. For such systems the independent variables may represent different space coordinates, with examples in image processing applications or mixed time and space variables or in processing seismic data. Multidimensional models also arise in the analysis of systems described by particular types of functional differential equations in one independent variable, such as delay-differential systems. This is also the case for repetitive processes that have provided a way to address control systems design for both industrial examples and also a representation for the design and experimental verification of iterative learning control laws. Early literature includes [Bose (1982),Bose et al, (2003), Rogers and Owens (1992)].

Recently emerging areas for the application of nD systems theory include grid sensor networks, and evidence filtering. In particular, wireless sensor networks consist of large numbers of resource constrained, embedded sensor nodes and are a candidate for distributed applications. Some applications require regularly placed nodes in a spatial grid, often sampling the sensors periodically over time, with a potential application in structural integrity monitoring. Agriculture and environmental monitoring applications often favor a grid or mesh topology. Spatially distributed sensor lattices are also essential in surveillance, target location, and tracking applications.

Distributed information processing schemes are natural candidates for such networks with regularly placed nodes, yielding benefits in terms of scalability, reduced communication costs, energy savings and improved system lifetime. Furthermore, applications requiring local actuation in response to local detection are best supported by distributed algorithms, yielding minimum response delays compared to centralized schemes.

The solution of nD systems and control design problems require a mathematical setting to address problems whose formulation and solution, for linear dynamics, can involve the use of functions and polynomials in more than one complex or real variables, where fundamental differences with the standard, or 1D linear systems case, immediately arise. For example, transfer-function descriptions of the dynamics of linear time invariant systems release a wealth of results from the theory of polynomials in one indeterminate for use in analysis and design, e.g. coprimeness and Bezout identities. In the nD case, coprimeness is no longer a single concept and hence the polynomial approach in the nD case is much more complicated.

This paper first introduces the commonly used models for discrete nD linear systems that have been used in control and systems problems and then gives results from a series of case studies. These case studies begin with sensor networks, followed by iterative learning control and then proceed to applications in civil engineering and agriculture. The general aim is to report results

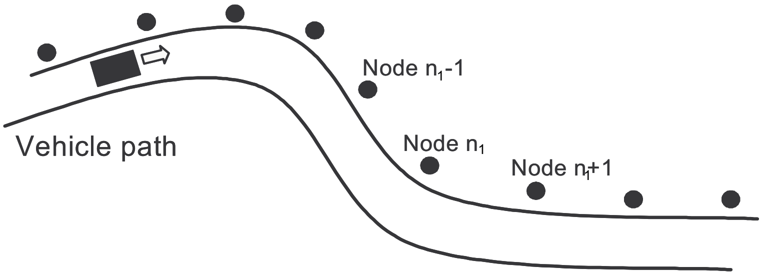


Fig. 1 A vehicle path under surveillance. The path is equipped with regularly spaced sensor nodes, represented by the black circles, placed in a 1D array.

that demonstrate that the nD systems approach does bring advantages in solving control problems. Finally, conclusions are drawn and possible future research briefly discussed.

Throughout this paper the null and identity matrices of compatible dimensions are denoted by 0 and I, respectively. Also $*$ is used to denote transposed entries in symmetric matrices. The notation $M \prec 0$ and $M \succ 0$ denotes that the symmetric matrix M is negative definite and positive definite respectively. Finally, $\rho(\cdot)$ denotes the spectral radius of its matrix argument, e.g., if λ_i, $1 \leq i \leq h$, are the eigenvalues of an $h \times h$ matrix H then $\rho(H) = \max_{1\leq i \leq h} |\lambda_i|$.

2 System Models

The Fornasini-Marchesini model [Fornasini and Marchesini (1978)] is an extensively studied state-space description of nD linear systems, and its dynamics can be illustrated by considering a regularly placed grid sensor network, such as the vehicle path under surveillance, equipped with regularly spaced sensor nodes placed in a 1D array as shown in Fig. 1.

In Fig. 1 the regularly spaced sensor nodes are denoted by black circles and the sensor number is denoted by n_1. The sensor node signals are sampled in time for discrete processing, n_2 denotes the sample number, and the result is a 2D discrete spatio-temporal signal.

The system of Fig. 1 can be modeled as the distributed system shown in Fig. 2. Two independent variables are required to define a signal, written as $x(n_1, n_2)$, where x is the variable or vector of interest, n_1 is the node number, and n_2 is the sampling instant. Consequently $x(n_1, n_2)$ is a spatio-temporal vector.

To describe the updating structure, consider sample instance n_2 at node $n_1 + 1$ and assume that the dynamics are linear. Then the sensor output, denoted by $y(n_1 + 1, n_2)$, is a linear combination of the state vector entries, and the node generates the state vector at the next time instant, that is, $x(n_1 + 1, n_2 + 1)$, by combining the current sample instance state vector at node $n_1 + 1$, that is $x(n_1 + 1, n_2)$, with the current sample instance state vector

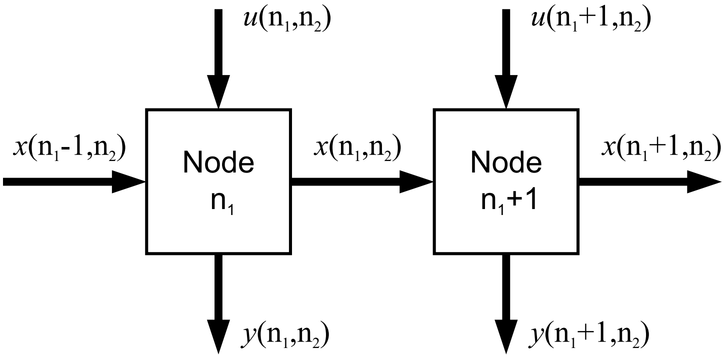


Fig. 2 A distributed 2D Fornasini-Marchesini state-space model for the 1D sensor array of Figure 1.

at the former node n_1, that is $x(n_1, n_2)$, and the input or control vector to the node at the current sample instance, that is $u(n_1 + 1, n_2)$. On completion of the computations, the node transmits its state information to the next node and so on. The state updating dynamics are described by the sensor network model

$$x(n_1 + 1, n_2 + 1) = A_1x(n_1 + 1, n_2) + A_2x(n_1, n_2) + B_1u(n_1 + 1, n_2). \quad (1)$$

The state-space model (1) is a special case of the 2D Fornasini-Marchesini state-space model

$$\begin{aligned} x(n_1 + 1, n_2 + 1) &= A_1x(n_1 + 1, n_2) + A_2x(n_1, n_2 + 1) + A_3x(n_1, n_2) \\ &\quad + B_1u(n_1 + 1, n_2) + B_2u(n_1, n_2 + 1) + B_3u(n_1, n_2), \\ y(n_1, n_2) &= Cx(n_1, n_2) + Du(n_1, n_2), \end{aligned} \quad (2)$$

where x is the $d_1 \times 1$ state vector, u is the $d_2 \times 1$ input vector, and y is the $d_3 \times 1$ output vector. Suppose also that n_1 and n_2 are restricted to nonnegative values. Then the dynamics described by (2) can be pictured as evolving over the positive quadrant of the 2D plane with axes n_1, and n_2, respectively, where each node, that is, a point in the 2D plane, is represented by a circle.

An alternative model that describes how a dynamic process evolves over the 2D plane is the Roesser state-space model [Roesser (1975)], where a state vector is defined for each axis. Denoting these vectors by $x^h(n_1, n_2)$, and $x^v(n_1, n_2)$, respectively, the state-space model is

$$\begin{aligned} \begin{bmatrix} x^h(n_1 + 1, n_2) \\ x^v(n_1, n_2 + 1) \end{bmatrix} &= \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} x^h(n_1, n_2) \\ x^v(n_1, n_2) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(n_1, n_2), \\ y(n_1, n_2) &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x^h(n_1, n_2) \\ x^v(n_1, n_2) \end{bmatrix}, \end{aligned} \quad (3)$$

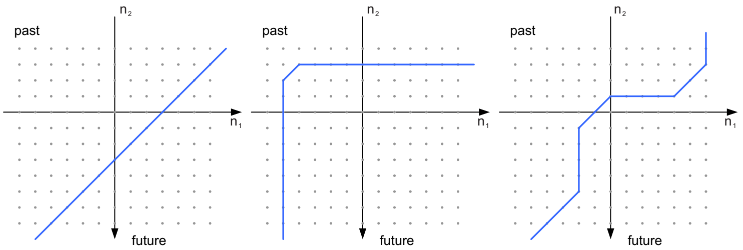


Fig. 3 The blue lines represent three of the possible separation sets for the 2D state-space models described by (3).

where also the augmented plant, input and matrices Φ, B, and C, respectively, for this model are defined as

$$\Phi = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \quad C = \begin{bmatrix} C_1 & C_2 \end{bmatrix}, \tag{4}$$

are used in some of the literature.

In 1D systems, the separation between past and future, that is, the present is given by a single time instant, that is, a point. Hence the recursive computation of a 1D trajectory consists of updating values on successive points of the domain. The situation is more complex in the nD case, where three examples of the possible separation-sets, denoted by the blue lines, for systems described by (3) are shown in Fig. 3. The 2D system dynamics evolves over a plane where n_1 is a spatial variable and n_2 a temporal variable and there can be no linear ordering on the plane and hence no time enforced separation into past, present and future. One way of interpreting the separation set is as a generalization of the idea where the past represents already computed or known values and the future those to be computed by a recursive algorithm, starting from the values that lie on this set.

To introduce the transfer-function description of the 2D dynamics considered in this paper consider a 2D sequence, say $x(n_1, n_2), n_1 \geq 0, n_2 \geq 0$. Then, using Z to denote the operation of taking the 2D z-transform of this sequence,

$$Z(x(n_1, n_2)) = \sum_{n_1 \geq 0, n_2 \geq 0} x(n_1, n_2) z_1^{-n_1} z_2^{-n_2}. \tag{5}$$

Applying the 2D z-transform to (3) with assumed zero boundary conditions gives, after routine algebraic manipulations,

$$y(z_1, z_2) = G(z_1, z_2) u(z_1, z_2), \tag{6}$$

where $G(z_1, z_2)$ is the 2D transfer-function matrix given by

$$G(z_1, z_2) = C \left[\begin{bmatrix} z_1 I & 0 \\ 0 & z_2 I \end{bmatrix} - \Phi \right]^{-1} B \tag{7}$$

and the constant entry matrices Φ, B and C are defined in (4).

To illustrate how nD, $n > 2$, state-space models can arise, consider again the 2D spatially distributed grid sensor network shown in Fig. 1 with the added feature that the data produced is now gathered over time. Then the 3D Fornasini-Marchesini state-space model describing the dynamics is

$$\begin{aligned} x(n_1, n_2, n_3 + 1) &= A_{001} x(n_1, n_2, n_3) + A_{101} x(n_1 - 1, n_2, n_3) \\ &\quad + A_{011} x(n_1, n_2 - 1, n_3) + Bu(n_1, n_2, n_3), \\ y(n_1, n_2, n_3) &= Cx(n_1, n_2, n_3), \end{aligned} \tag{8}$$

where x is the $d_1 \times 1$ state vector, u is the $d_2 \times 1$ input vector generated using sensor signals, and y is the $d_3 \times 1$ system output vector.

The quarter-plane causality of the Roesser and Fornasini state-space models considered in this paper impose a particular structure, or preferred direction of updating, on the computation of the state and output vectors over the quarter plane. Treating the concept of state in a behavioral setting for nD systems leads to first-order state-space models without a preferred direction which, in turn, requires a formalization of the concepts of past, future, and of the independence of the future of a trajectory given the past. This topic is extensively investigated in [Rocha and Wood (2001)] and in the 2D systems case leads to discrete state-space models defined by equations with the following structure,

$$E_1 x + F_1 z_1 x = 0, \tag{9}$$

$$E_2 x + G_2 z_2 x = 0, \tag{10}$$

$$E_3 x + F_3 z_1 x + G_3 z_2 x + H_3 z_1 z_2 x = 0, \tag{11}$$

$$Nx + Mw = 0, \tag{12}$$

where x is the state variable vector and the matrices E_1, E_2, E_3, F_1, F_3, G_2, G_3, H_3, N, and M have additional properties [Rocha and Wood (2001)]. The state equations (9), (10) and (11) are first-order in x and zeroth order in w, where these properties are a consequence of the formalization of past, future, and of the independence and not postulated a priori as is the case for both the Roesser and Fornasini-Marchesini state-space models considered in this paper. The behavioral setting for nD systems analysis has enabled the solution of many control systems theoretic questions but to this time little or no impact on control system design for applications. Hence this approach is not considered in this paper.

2.1 Models of Linear Repetitive Processes

The Roesser and Fornasini-Marchesini model based 2D linear systems are recursive over the positive quadrant of the 2D plane. It is also possible to write down models where information propagation in one direction is governed by a differential equation and in the other by a difference equation. It also possible that information propagation in one direction occurs only over a finite duration and is an intrinsic feature as opposed to an assumption made for modeling and analysis purposes. These two features are present in repetitive processes whose unique characteristic can be illustrated by considering machining operations where the material, or workpiece, is processed by a sequence of passes of the processing tool. Assuming the pass length $\alpha < \infty$ to be constant, the output vector, or pass profile, $y_k(t)$, $0 \leq t \leq \alpha$, where t denotes the independent spatial or temporal variable, generated on pass k acts as a forcing function on, and hence contributes to, the dynamics of the next pass profile $y_{k+1}(t)$, $0 \leq t \leq \alpha$, $k \geq 0$.

These processes have their origins in the coal mining and metal rolling industries and the details can be found in the original work [Edwards (1974), Edwards and Greenberg (1977)]. For more control-system related discussion of coal mining refer to [Einicke et al. (2008)]. Simulation studies [Edwards (1974), Edwards and Greenberg (1977),Rogers and Owens (1992)] immediately highlight the unique control problem for linear repetitive processes where the output sequence generated, that is, the sequence of pass profiles, can contain oscillations that increase in amplitude in the pass-to-pass direction. In longwall coal cutting the problem is caused by the weight of the machine as it rests on the previous pass profile during the cutting of the next pass profile, and the undulations caused can result in productive work having to stop. A stability theory for linear repetitive processes must prevent productive work stoppage in order to maximize production.

A differential linear repetitive process is described over $0 \leq t \leq \alpha$, $k \geq 0$, by

$$\begin{aligned} \dot{x}_{k+1}(t) &= Ax_{k+1}(t) + Bu_{k+1}(t) + B_0 y_k(t), \\ y_{k+1}(t) &= Cx_{k+1}(t) + Du_{k+1}(t) + D_0 y_k(t), \end{aligned} \tag{13}$$

where on pass k, $x_k(t) \in \mathbb{R}^n$ is the state vector, $y_k(t) \in \mathbb{R}^m$ is the output, or pass profile vector, and $u_k(t) \in \mathbb{R}^r$ is the input vector. For this model it is also necessary to specify boundary conditions, and the simplest possible is

$$y_0(t),\ 0 \leq t \leq \alpha,\ \ x_{k+1}(0) = d_{k+1},\ k \geq 0, \tag{14}$$

where $y_0(t)$ is a given initial pass profile vector, and d_{k+1} has known constant entries.

In a 2D systems setting, processes with state dynamics described by (13) can be referred to as mixed, that is, the along the pass dynamics are governed by a linear matrix differential equation, and the pass-to-pass dynamics by a discrete linear matrix equation. It is also possible to have discrete dynamics along the pass, and a discrete linear repetitive process state-space model over $0 \leq p \leq \alpha - 1$, $k \geq 0$, is

$$\begin{aligned} x_{k+1}(p+1) &= Ax_{k+1}(p) + Bu_{k+1}(p) + B_0 y_k(p), \\ y_{k+1}(p) &= Cx_{k+1}(p) + Du_{k+1}(p) + D_0 y_k(p), \end{aligned} \tag{15}$$

where on pass k, $x_k(p) \in \mathbb{R}^n$ is the state vector, $y_k(p) \in \mathbb{R}^m$ is the pass profile vector, and $u_k(p) \in \mathbb{R}^r$ is the input vector. The equivalent of the boundary conditions of (14) are

$$x_{k+1}(0) = d_{k+1},\ \ k \geq 0,\ \ y_0(p) = f(p),\ \ 0 \leq p \leq \alpha - 1, \tag{16}$$

where $y_0(p)$ is a given initial pass profile vector and d_{k+1} has known constant entries.

The state initial vector sequence $x_{k+1}(0)$, $k \geq 0$, for differential and discrete linear repetitive processes can be a function of points along the previous pass profile, and for discrete processes one choice is

$$x_{k+1}(0) = d_{k+1} + \sum_{j=0}^{\alpha-1} J_j y_k(j),\ k \geq 0, \tag{17}$$

where $J_j,\ 1 \leq j \leq \alpha - 1$, is an $n \times m$ matrix, and when combined with the initial pass profile $y_0(p)$ of (16) are termed dynamic boundary conditions.

An obvious route to analysis of repetitive process dynamics is to ignore (17) and join the pass profiles end-to-end to obtain the standard linear systems state-space model. In particular, write the variables in terms of $V = k\alpha + t$, in the exemplar case of differential along the pass dynamics, to convert the particular example under consideration into an equivalent infinite length single pass process where the relationships between variables are expressed in terms of V, termed the total distance traversed. Then a variable, say, $Y_{k+1}(t)$, $k \geq 0$, is identified as a function $Y(V)$ of V defined for $0 \leq V < \infty$. The problem that then arises is that the inherent structure of linear repetitive processes is not present in the resulting model and incorrect stability conclusions could be made.

The stability theory [Rogers and Owens (1992),Rogers et al. (2007)] for linear repetitive processes is based on an abstract model in a Banach space setting that includes a wide range of such processes as special cases, including those described above. Suppose that the pass profile $y_k \in E_\alpha$, where E_α is a suitably chosen Banach space with norm $||\cdot||$. Then the dynamics of a linear repetitive process of constant pass length $\alpha > 0$ are described by

$$y_{k+1} = L_\alpha y_k + b_{k+1}, \quad k \geq 0, \tag{18}$$

where $b_{k+1} \in W_\alpha$, W_α is a linear subspace of E_α, and L_α is a bounded linear operator mapping E_α into itself. In this model the term $L_\alpha y_k$ represents the contribution of pass k to pass $k+1$, and b_{k+1} represents other terms that enter on pass $k+1$, namely, control inputs, pass state initial conditions, and disturbances.

Repetitive process models can also be written where the pass is a rectangle in the plane, and the state-space model is then 3D. One model of this form is

$$x_{k+1}(l,m) = \sum_{i=-\epsilon}^{\epsilon} \sum_{j=-\varepsilon}^{\varepsilon} \left(A^{i,j} x_k(l+i, m+j) + B^{i,j} u_k(l+i, m+j) \right), \tag{19}$$

where on pass k $x_k(l,m) \in \mathbb{R}^n$ is the state vector, $u_k(l,m) \in \mathbb{R}^q$ is the input vector, and $\epsilon > 0$ and $\varepsilon > 0$ are positive integers. The boundary conditions are

$$
\begin{aligned}
x_k(l,m) &= 0,\ -\epsilon \le l < 0,\ 0 \le m \le \beta,\ k \ge 0, && (20)\\
x_k(l,m) &= 0,\ -\varepsilon \le m < 0,\ 0 \le l \le \alpha,\ k \ge 0, && (21)\\
x_0(l,m) &= d_0(l,m),\ 0 \le l \le \alpha,\ 0 \le m \le \beta, && (22)\\
x_k(\alpha - i,m) &= d_k(i,m),\ 0 \le m \le \beta,\ 0 \le i < \epsilon,\ k \ge 0, && (23)\\
x_k(l,\beta - j) &= d_k(l,j),\ 0 \le l \le \alpha,\ 0 \le j < \varepsilon,\ k \ge 0 && (24)
\end{aligned}
$$

and the process dynamics are defined over a finite fixed rectangle $0 \le l \le \alpha - \epsilon, 0 \le m \le \beta - \varepsilon$ but, at every point on pass $k+1$, only those points in the rectangle defined by $-\epsilon \le l \le \epsilon$, $-\varepsilon \le m \le \varepsilon$, on the previous pass contribute to the current pass profile. The updating structure for the case when $\epsilon = \varepsilon = 1$ is illustrated in Fig. 4. For processes described by (19) it is a rectangle of information that is propagated in the pass-to-pass direction.

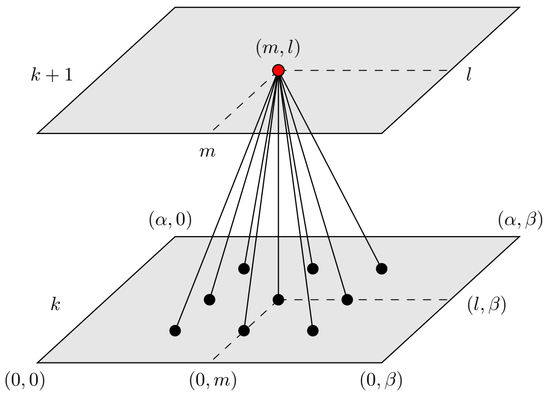


Fig. 4 Illustrating the updating structure of (19).

The repetitive process models considered previously in this paper require the assumption that the sole previous pass contribution to the current pass profile is at the same point along the pass. In some examples, such as long-wall coal cutting, the previous pass profile is modified before the start of the next pass, and this effect is termed inter-pass smoothing. In long-wall coal cutting the inter-pass smoothing is caused by the machines weight, up to 5 tons, as it comes to rest on the newly cut pass profile and cannot be realistically modeled by state-space models of the form (13) or (15). Instead, it is necessary to consider a model with all the points along the previous pass contributing to the pass profile at each point on the current pass.

One way of extending the model of (13) and (14) for differential linear repetitive processes to include inter-pass smoothing is

$$\begin{aligned} \dot{x}_{k+1}(t) &= Ax_{k+1}(t) + Bu_{k+1}(t) + B_0 \int_0^{\alpha} K(t,\tau)\, y_k(\tau)\, d\tau, \\ y_{k+1}(t) &= Cx_{k+1}(t),\ 0 \leq t \leq \alpha,\ k \geq 0, \end{aligned} \tag{25}$$

with, for simplicity, $x_{k+1}(0) = d_{k+1},\ k \geq 0$. In this representation, the inter-pass interaction term $\int_0^{\alpha} K(t,\tau) y_k(\tau)\, d\tau$ represents a smoothing out of the previous pass profile in a manner governed by the properties of the kernel $K(t,\tau)$. Also the particular choice of $K(t,\tau) = \delta(t-\tau)I_m$, where $\delta(\cdot)$ denotes the Dirac delta function, reduces (25) to the model of (13) and (14) with $D = 0$, $D_0 = 0$. One possible choice of the kernel is a double-sided exponential decay centered on the instance along pass under consideration. A detailed treatment of this and other practically motivated choices for the kernel can be found in [Edwards (1974),Edwards and Greenberg (1977)] and the extension of the abstract model based stability theory in [Rogers et al. (2007)].

The application of this stability theory in the design of iterative learning control laws is described in Section 4 together with experimental verification.

3 2D Systems Control of Sensor Networks

This section gives two applications of the Fornasini Marchesini 2D systems model in the general area of sensor networks, an area of strong research across a number of disciplines. The results include data from a testbed implementation in one case.

3.1 Spatially Distributed Grid Sensor Networks

Return to the sensor network model of (1), where one of the main challenges is to design the local state-space model matrices such that the desired filtering function is implemented in a distributed manner. In some cases, coefficient matching can be used to design the state-space matrices in order to achieve a desired response as represented by a 2D transfer-function, say $H(z_1, z_2)$.

Sensor networks are often used to implement functionality beyond the sensing tasks, that is, by coupling nodes with actuator devices so that if a certain event is detected, the node activates the actuator devices to achieve a specified task. A sensor network that detects a chemical spill may be required to release neutralizing agents to its surroundings. Typically, actions need to be coordinated using a leader node that gathers sensor data from all nodes that exchange data with it, and the information combined to make a decision on the actuation task. As a result, an overhead can be added to the system, draining limited system resources such as energy.

In contrast, the 2D systems approach requires each node to communicate its current state-information to only its immediate neighbors. After the in-node computation using the exchanged state values, and observing that its output has exceeded a specified threshold, a node can decide on its own regarding the actuation task. Such decision-making capability eliminates the need for receiving actuation commands from leader nodes, saving network bandwidth, and energy.

Consider the use of the 2D Fornasini-Marchesini state-space model (1) to describe a node in a sensor network. Then the window of events considered during computation of the local output y to decide on the local actuation task is fully governed by the 2D transfer-function $G(z_1, z_2)$ computed by applying the 2D z-transform defined by (5) to (1). As the example given below demonstrates, proper choice of parameters in $G(z_1, z_2)$ can ensure that a longer or a shorter window of events along the space and time-axes is considered during the computation of the state x, and hence the output y.

Current embedded wireless sensor platforms typically use 8-bit or 16-bit fixed point microprocessors. Hence, quantization effects during data processing inside each node can affect both system performance and the stability of the implemented filtering process. Moreover, the effects of quantization and overflow nonlinearities on nD system stability must also be considered.

In addition to quantization during processing, only a limited number of bits can be employed to propagate the state information over the low rate wireless channel to the neighboring nodes. In some cases the values are truncated down to 428 bit values and such coarse quantization can also affect the stability and performance of the distributed filter [Kar and Singh (2001), Dewasurendra et al. (2006)].

In the sensor networks described above, see (8), the spatial variables are bounded and this is a 3D distributed system application with a 2D spatial sensor grid, and 1D time, with sensors along the axes n_1 and n_2. Only the temporal variable t is unbounded because the temporal duration of the sensor signals collected by each node is several orders of magnitude longer than the spatial extent of the impulse response.

As an example implementation, suppose that the goal is to detect a vehicle moving at a constant velocity along a straight 1D path, which is a special case of Fig. 1, using a velocity filter. The effects of quantization nonlinearities discussed above are not considered, and a quantization word length of 16 bits is used for both inter-node communication, and in-node computation, to minimize its effects on system stability.

The testbed shown in Fig. 5, consisting of a linear 1D array of wireless sensor nodes, is used for the implementation. Each node is based on the TelosB [1] platform using IEEE 802.15.4 Zigbee wireless communication standard, and is connected to a multi-modality sensor board. The sensor board can sense multiple sensor modalities including visual light (L), sound (S), infrared (I), and magnetic (M), which correspond to the various properties of a rover vehicle to be detected. A combination of sensors is used mainly to improve the robustness of the detection process. To detect the vehicle characterized by all four sensor signals L, S, I, and M, the average of the sampled values of all four sensor-signals is used as the input term $u(n_1+1, n_2)$. As a particular example,

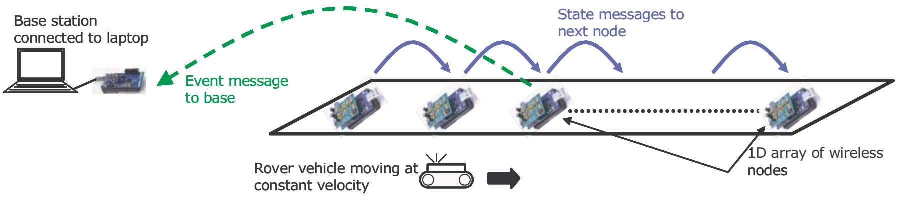

Base station connected to laptop
Event message to base
State messages to next node
Rover vehicle moving at constant velocity
1D array of wireless nodes

Fig. 5 The testbed used in the grid sensor network experiment.

consider the case when eleven nodes are placed in a periodic 1D array on an elevated platform over the observed path. The rover vehicle is set to move with a constant velocity underneath.

For ease of implementation, a simplified scalar Fornasini-Marchesini state-space model is used with a scalar local state x in (1), and real scalar constants $A_1 = 0$, $A_2 = a_{00}$, and $B_1 = b$. The corresponding 2D transfer-function description can be written as

$$G(z_1, z_2) = \frac{bz_2^{-1}}{1 - a_{00}z_1^{-1}z_2^{-1}} \tag{26}$$

and a velocity detection filter is given by the 2D transfer-function

$$H(z_1, z_2) = \frac{P(z_1, z_2)}{1 - az_1^{-1}z_2^{-1}}, \tag{27}$$

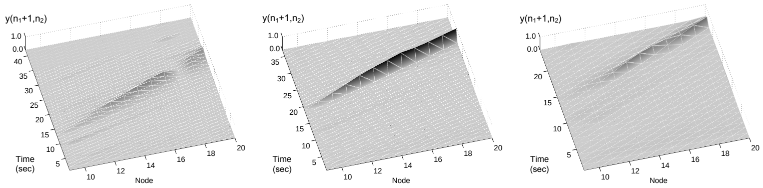
where the constant a is determined based on the velocity to be detected. For the case considered, coefficient matching gives the required 2D transfer-function

$$H(z_1, z_2) = \frac{z_2^{-1}}{1 - az_1^{-1}z_2^{-1}}. \tag{28}$$

As a numerical example, consider the application of (28) with $a = 0.75$, which determines the length of window of past events considered by the system during the detection process. Also the rover vehicle is moving at a constant velocity of 5 cm/sec, which corresponds to the distance between two consecutive sensor nodes or the sampling time. The filter response $y(n_1 + 1, n_2)$ is observed for this velocity and the two other rover velocities, that is, 3.8 cm/sec and 8.1 cm/sec.

In Fig. 6 the response of the distributed filter for three different rover speeds is given. Comparing Figs. 6a, 6b, and 6c, it follows that the distributed filter implemented using the Fornasini-Marchesini local state-space model detects the rover vehicle with a constant velocity of 5.0 cm/sec.

The 2D Fornasini-Marchesini state-space model based approach to distributed information processing in grid sensor networks is capable of implementing linear systems. Additional advantages include high scalability, ease of re-configurability, minimized communication costs, and the ability to execute local actuation tasks in response to local phenomena. The implementation of



(a) Low velocity (3.82 cm/sec) (b) Exact velocity (5.0 cm/sec) (c) High velocity (8.125 cm/sec)

Fig. 6 Velocity filter implementation results.

a velocity filter illustrates the approach, although it is of limited scope considering the wide application potential of the method. Another interesting application is the detection of wavefronts consisting of hazardous plumes in air, using a 2D or 3D grid sensor network placed in a suburban or urban area.

By using a higher value for a in (28), it is possible to consider a longer window of events in space-time to produce the distributed filter output, and vice versa, where such a 2D transfer-function coefficient is a critical parameter, particularly in situational awareness applications, where a shorter or longer time window is needed from time to time, depending on the desired event resolution, and current situational requirements.

3.2 Temporal Evidence Filtering

Surveillance, monitoring, and situational awareness applications often use sensors that are spatially distributed over the area of observation, generating data over time. These sensors are often coupled with microprocessors, and low-rate, short range wireless radios to create a distributed network of embedded sensor nodes [Szewczyk et al. (2004)]. Limitations in energy reserves, radio range, and data throughput in nodes require distributed processing of information in such networks.

Temporal evidence filtering (TEF) [Dewasurendra et al. (2006)] combines multimodal sensor information for local processing within the nodes in a sensor network, and is based on Dempster-Shafer evidence theory [Shafer 1976], which models sensor data as evidence supporting various observation events. Filtering of this form is capable of processing temporally ordered evidence inside a sensor node to directly infer the occurrence of periodic events characterized by multiple sensing modalities. The temporal evidence filtering approach can be extended to the case of spatio-temporal evidence filtering (STEF), to process evidence gathered from multi-modal sensors over multiple dimensions of both space and time.

A centralized implementation of STEF would be impractical, considering the resource limitations of an embedded sensor network. The Fornasini-Marchesini state-space model (2) and Roesser state-space model (3), can be used in a distributed implementation of spatio-temporal evidence filters. In such an implementation, each node in the network performs a portion of the computation locally, in order to generate the output of the STEF collectively, and the approach can offer advantages for resource-limited embedded sensor networks.

The evidence filter discussed in [Dewasurendra et al. (2006)] processes the evidence only over time and TEF can be further extended to process evidence gathered over multiple dimensions of both space and time. The basics of spatio-temporal evidence filtering versus alternative forms of filtering are illustrated in Fig. 7. An evidence filter adds one more dimension to spatio-temporal filtering and explores the correlations among sensor signals of various modalities to improve and enhance the sensitivity.

A mD spatio-temporal evidence filter takes the form

$$\begin{aligned}\mathrm{Bel}(B)(n_1,\cdots,n_m) &= \sum_{i_1}..\sum_{i_m}\alpha_{i_1,..,i_m}\mathrm{Bel}(B)(n_1-i_1,\cdots,n_m-i_m),\\ &+\sum_{j_1}..\sum_{j_m}\beta_{j_1,..,j_m}\mathrm{Bel}(B|A)(n_1-j_1,..,n_m-j_m),\end{aligned}\tag{29}$$

with indices $n_1, n_2, \ldots, n_m$ representing each dimension, and Bel(B) is the Dempster-Shafer theoretic belief function of proposition B [Shafer 1976]. The conditional belief of B given event A, denoted by Bel$(B|A)$, is defined in [Fagan and Halpem (1990)] and the constant coefficients $\alpha_{i_1,..,i_m}, \beta_{j_1,..,j_m} \in \mathbb{R}$ satisfy the sum and positivity constraints given in [Dewasurendra et al. (2006)].

Consider the 3D evidence filter consisting of 2D space, with indices n_1, n_2, and 1D time, indexed by t, characterized by the difference equation

$$\begin{aligned}\mathrm{Bel}(B)(n_1,n_2,t) &= \sum_{i_1}\sum_{i_2}\sum_{\tau}\alpha_{i_1,i_2,\tau}\mathrm{Bel}(B)(n_1-i_1,n_2-i_2,t-\tau),\\ &+\sum_{j_1}\sum_{j_2}\sum_{\tau}\beta_{j_1,j_2,\tau}\mathrm{Bel}(B|A)(n_1-j_1,n_2-j_2,t-\tau),\end{aligned}\tag{30}$$

where $(n_1, n_2) \in \mathbf{N}_0^2, t \in \mathbf{N}_0$. To compute the belief of B at time t and location (n_1, n_2), the beliefs of B computed at time $(t-\tau)$ at locations (n_1-i_1, n_2-i_2) are required, and can be extended to represent the occurrence of multiple events $A_1, \ldots, A_n$ supporting proposition B. Such higher dimensional evidence filters are potentially useful in real-world applications including the detection of various low-signature space-time events.

Consider a vehicle path under surveillance, equipped with equi-spaced multi-modal sensor nodes placed in the 1D array shown in Fig. 1. The goal is to detect a particular type of vehicle traveling along the path with a particular speed. Based on the sensor signals sampled at regular intervals with a sampling time t, the nodes update their local viewpoints or beliefs, and the system can be modeled as the distributed system shown in Fig. 5.

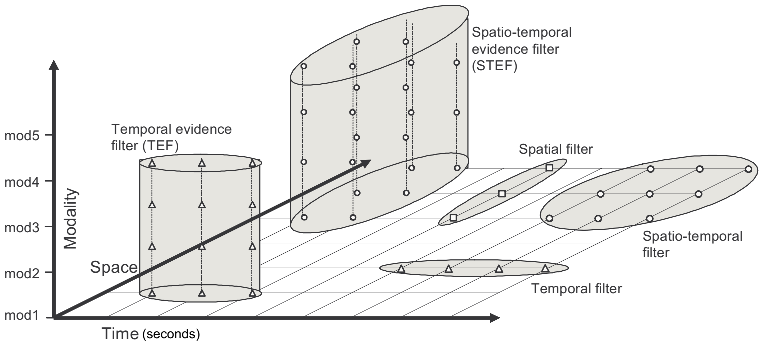


Fig. 7 Temporal evidence-filtering (TEF) for processing evidence gathered over multiple dimensions of both space and time. An evidence filter adds one more dimension to spatio-temporal filtering, and explores the correlations among sensor signals of various modalities to improve and enhance sensitivity.

A distributed sensor network based on the 2D Fornasini-Marchesini state-space model of (1) with a static output equation added is

$$\begin{aligned} x(n,t+1) &= A_2x(n,t) + A_1x(n-1,t) + Bu(n,t), \\ y(n,t) &= Cx(n,t) + Du(n,t), \end{aligned} \tag{31}$$

where $x \in \mathbb{R}^d$ is the state vector, $u \in \mathbb{R}^h$ is the input vector generated using beliefs of interest, and $y \in \mathbb{R}^q$ is the output vector, $n \in \{0, 1, \ldots, M\}$ denotes the node number, and $t \in \mathbf{N}$ denotes discrete time. The design problem is to determine the local state-space matrices A_1, A_2, B, C, and D such that the desired evidence filtering function is implemented in a distributed manner, where for some examples coefficient matching is again an option for design.

Further development of the area depends on progress with several open research problems. First, a formal method needs to be developed to obtain the state-space matrices in (31) based on the evidence filter specifications, and its transfer-function. The term corresponding to $A_2x(n_1, n_2 + 1)$ in the 2D Fornasini-Marchesini state-space model (2) does not appear in (31), and suggests that some quarter-plane causal evidence filters are not realizable. Determining the realizable frequency responses using a constrained form is an open research problem.

4 Repetitive Process based Iterative Learning Control Design

In recent years, there has been a substantial volume of research on using the repetitive process setting to design iterative learning control laws. This research has included experimental verification, which represents a major step forward for multidimensional control system design. This section overviews the results to date, gives some new results on disturbance rejection design.

4.1 Experimentally Verified Nominal Model Designs

Iterative Learning Control (ILC) emerged from industrial applications where the system involved executes the same operation many times over a fixed time interval. When each operation is complete, resetting to the starting location takes place and the next operation can commence immediately, or after a stoppage time. A common example is a gantry robot undertaking a pick and place operation in synchronization with a moving conveyor or assembly line. The sequence of operations is: a) the robot collects a payload from a fixed location, b) transfers it over a finite duration, c) places it on the moving conveyor, d) returns to the original location for the next payload and then e) repeats the previous four steps for as many payloads as required or can be transferred before it is required to stop.

To operate it is necessary to supply the robot with a trajectory to follow and the task for a control law is to ensure that the robot follows the prescribed trajectory exactly or, more realistically, to within a specified tolerance. In addition to controlling its own movement and that of the payload, the control law must prevent other effects, such as disturbances and signal noise, from degrading tracking and thereby forcing it outside of the tolerance bound. If the robot begins to operate outside permissible limits, the control task is to bring it back as quickly as required or is physically possible. This must be achieved without causing damage to, for example, the sensing and actuating technologies used.

In ILC, each completion or execution of the task is commonly termed a trial, but in this paper pass is used instead of trial to conform with the repetitive process terminology. Also the finite time each pass takes to complete will be referred to as the pass length. Once a pass is complete, all data used and generated during its completion is available for use in computing the control action to be applied on the next pass. The use of such data is a form of learning and is the essence of ILC, embedding the mechanism through which performance may be improved by past experience. The ILC mode of operation outlined above is the most common, that is, complete a pass, reset and then repeat. This is different from repetitive control where the system continuously executes over the period of the reference signal, that is, no stoppage time between passes (or executions).

The widely recognized starting point for ILC is [Arimoto et al. (1984)], which considered a simple structure linear servomechanism system for speed control of a voltage-controlled dc-servomotor. Suppose also that the system to be controlled has discrete linear time-invariant dynamics. Then in the ILC setting the dynamics are described by the state-space model over the pass length $\alpha < \infty$.

$$\begin{aligned} x_k(p+1) &= Ax_k(p) + Bu_k(p), \\ y_k(p) &= Cx_k(p), \;\; x_k(0) = x_0, \end{aligned} \tag{32}$$

where on pass k, $x_k(p) \in \mathbb{R}^n$ is the state vector, $y_k(p) \in \mathbb{R}^m$ is the output vector and $u_k(p) \in \mathbb{R}^l$ is the control input vector.

In this model it is assumed that the initial state vector does not change from pass-to-pass. The case when this assumption is removed has also been considered in the literature. Also the dynamics are assumed to be disturbance-free but again this assumption can be relaxed. It also possible to write the dynamics in input-output form involving the convolution operator or take the one-sided z transform and hence enable analysis and design in the frequency domain. To apply the z transform it is necessary to assume $\alpha = \infty$ but in most cases the consequences of this requirement have no detrimental effects. For a more detailed analysis of cases where there are unwanted effects arising from this assumption, see the relevant references in [Ahn et al. (2007), Bristow et al. (2006)] and more recent work in [Wallen et al. (2013)].

Let $r(p) \in \mathbb{R}^m$ denote the supplied reference vector. Then the error on pass k is $e_k(p) = r(p) - y_k(p)$ and the core requirement in ILC is to construct a sequence of input functions $u_{k+1}(p)$, $k \geq 0$, such that the performance achieved is improved with each successive pass and after a 'sufficient' number of these the current pass error is zero or within an acceptable tolerance. Mathematically this can be stated as a convergence condition on the input and error of the form

$$\lim_{k\to\infty} ||e_k|| = 0, \lim_{k\to\infty} ||u_k - u_\infty|| = 0, \tag{33}$$

where u_∞ is termed the learned control and $||\cdot||$ denotes an appropriate norm on the underlying function space. For example, if $||\cdot||_2$ denotes the Euclidean norm of its argument one possibility is $||e|| = \max_{p\in[0,\alpha]} ||e(p)||_2$. The reason for including the requirement on the control vector is to ensure that undue emphasis on reducing the pass-to-pass error does not come at the expense of unacceptable control signal demands. In application, only a finite number of passes will ever be completed but mathematically letting $k \to \infty$ is required in analysis of, for example, pass-to-pass error convergence.

The standard form of ILC algorithm or law constructs the current pass input as the sum of the input used on the previous pass and a corrective term, that is,

$$u_{k+1}(p) = u_k(p) + \varDelta(u_k(p), e_k(p)), \tag{34}$$

where $\varDelta(u_k(p), e_k(p))$ is the correction term and is a function of the error and input recorded over the previous pass. A large number of variations exist for computing the correction term, including algorithms that make use of information generated on a finite number (greater than unity) of previous passes.

Analysis of ILC, involves signals that propagate in two directions, from pass-to-pass (in k) and along the pass (in p) over a subset of the upper-right quadrant in the 2D plane. Hence it is to be expected that 2D systems theory can be applied and the first work in this area used the Roesser model [Kurek and Zaremba (1993)]. In this latter paper, it is shown how pass-to-pass error convergence of linear ILC laws in the discrete domain can be examined as a stability problem in terms of a Roesser state-space model interpretation of the dynamics. Given that the pass length is finite by definition, it follows that ILC fits naturally into the class of repetitive processes.

For analysis purposes, introduce the following vector defined from the dynamics of (33)

$$\eta_{k+1}(p+1) = x_{k+1}(p) - x_k(p) \tag{35}$$

and select the term $\varDelta(u_k(p), e_k(p))$ in (34) as

$$\varDelta(u_k(p), e_k(p)) = K_1\eta_{k+1}(p+1) + K_2 e_k(p+1). \tag{36}$$

Also introduce the notation

$$\begin{aligned} \tilde{A} &= A + BK_1, & \tilde{B}_0 &= BK_2, \\ \tilde{C} &= -C(A + BK_1), & \tilde{D}_0 &= I - CBK_2 \end{aligned} \tag{37}$$

and it follows on combining (32), (35) and (36) that the ILC dynamics are then described by

$$\begin{aligned} \eta_{k+1}(p+1) &= \tilde{A}\eta_{k+1}(p) + \tilde{B}_0 e_k(p), \\ e_{k+1}(p) &= \tilde{C}\eta_{k+1}(p) + \tilde{D}_0 e_k(p), \end{aligned} \tag{38}$$

which is a particular case of the discrete linear repetitive process state-space model (15) with pass profile vector $e_{k+1}(p)$ and current state vector $\eta_{k+1}(p)$ and no current pass input terms.

The stability theory for linear repetitive processes developed in terms of (18) demands that a bounded initial pass profile produces a bounded sequence of pass profiles either over the finite and fixed pass length of the process or, in stronger form, for all possible values of the pass length. Applying the first of the two stability properties, termed asymptotic stability [Rogers et al. (2007)], to the ILC dynamics generated by (38) requires that $\rho(\tilde{D}_0) < 1$. This condition is precisely that obtained by applying 2D discrete linear systems stability theory [Kurek and Zaremba (1993)] to (38) and hence ensure pass-to-pass error convergence only using the control law (setting $K_1 = 0$ in (36)) $u_{k+1}(p) = u_k(p) + K_2 e_k(p+1)$.

The repetitive process setting provides the alternative of imposing the stronger form of stability, known [Rogers et al. (2007)] as stability along the pass, where this property holds for dynamics described by (38) if and only if i) $\rho(\tilde{D}_0) < 1$, ii) $\rho(\tilde{A}) < 1$, and iii) all eigenvalues of

$$G(z) = \tilde{C}(zI - \tilde{A})^{-1}\tilde{B}_0 + \tilde{D}_0, \tag{39}$$

have modulus strictly less than unity for all $|z| = 1$, where it is assumed that the pair $\{\tilde{A}, \tilde{B}_0\}$ is controllable and the pair $\{\tilde{C}, \tilde{A}\}$ observable. If stability along the pass holds, the pass profile sequence $\{e_k\}_k$ generated by (38) converges in k to the limit profile e_∞ described by a stable 1D linear systems state-space model, where this property will be used in Section 4.2.

If ILC is to be applied to a discrete time linear system that is unstable one approach is to first design a stabilizing control law and then apply ILC to the resulting closed-loop system, where the application of the latter is based on first building a 1D discrete linear systems model of the pass-to-pass updating, that is, in a similar manner to the 1D equivalent model for discrete linear repetitive processes [Rogers et al. (2007),Rogers et al. (2002)]. The lifting approach to ILC design is therefore a two-stage process whereas the repetitive process setting allows for the design of a control law in one step. A comparison of the repetitive process approach against alternative design settings is given later in this section.

The area of ILC design based on repetitive process stability theory has seen a substantial body of results developed. Next, one method that has led to experimental verification is considered and then alternatives are discussed and some new results developed. This requires a Lyapunov function characterization of stability along the pass, where for the ILC dynamics (38) the Lyapunov function used has the form

$$V(k,p) = V_1(k,p) + V_2(k,p), \tag{40}$$

with

$$\begin{aligned} V_1(k,p) &= \eta_{k+1}^T(p) W_1 \eta_{k+1}(p), \\ V_2(k,p) &= e_k^T(p) W_2 e_k(p), \end{aligned} \tag{41}$$

where $W_1 \succ 0$ and $W_2 \succ 0$. Also the increment of this Lyapunov function is

$$\begin{aligned} \Delta V(k,p) = {} & V_1(k,p+1) - V_1(k,p) \\ & + V_2(k+1,p) - V_2(k,p), \end{aligned} \tag{42}$$

where, in physical terms, the first term $\Delta V_1(k,p)$ measures the difference in state (or along the pass) energy at two successive sample instants and $V_2(k,p)$ the difference in the error energy between two successive passes.

Applying a known result [Rogers et al. (2007)], stability along the pass of (38) holds if

$$\Delta V(k,p) \prec 0, \tag{43}$$

which can be rewritten in LMI form as

$$\hat{A}^T W \hat{A} - W \prec 0, \tag{44}$$

where

$$\hat{A} = \begin{bmatrix} \tilde{A} & \tilde{B}_0 \\ \tilde{C} & \tilde{D}_0 \end{bmatrix}, \quad W = \begin{bmatrix} W_1 & 0 \\ 0 & W_2 \end{bmatrix} \succ 0. \tag{45}$$

Theorem 1 *[Haldowski et al. (2010)] Stability along the pass holds for the ILC dynamics (38) if there exist matrices $X_1 \succ 0$, $X_2 \succ 0$, R_1 and R_2 such that the LMI*

$$\begin{bmatrix} -X_1 & 0 & X_1A^T + R_1^T B^T & -X_1A^TC^T - R_1^T B^T C^T \\ 0 & -X_2 & R_2^T B^T & X_2 - R_2^T B^T C^T \\ AX_1 + BR_1 & BR_2 & -X_1 & 0 \\ -CAX_1 - CBR_1 & X_2 - CBR_2 & 0 & -X_2 \end{bmatrix} \prec 0, \tag{46}$$

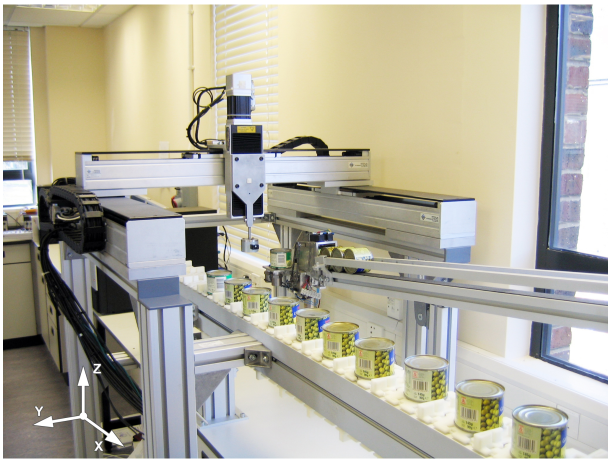


Fig. 8 The gantry robot with the three axes marked.

is feasible. If (46) holds, the control law matrices K_1 and K_2 are given by

$$K_1 = R_1 X_1^{-1},\ K_2 = R_2 X_2^{-1}. \tag{47}$$

Routine algebraic manipulations give the form of the control law for implementation as

$$u_{k+1}(p) = u_k(p) + K_1(x_{k+1}(p) - x_k(p)) + K_2(r(p+1) - y_k(p+1)). \tag{48}$$

In this control law the second term is state feedback, based on the difference between the state vectors on the current and previous passes and the third is phase-lead, where the argument $p+1$ is temporal information that would be non-causal in 1D linear systems designs. The use of such information is the critical novel feature of ILC.

Experimental verification of the ILC control law design of Theorem 1 has been reported using the multi-axis gantry robot shown in Fig. 8, which has also been used to test and compare the performance of a range of ILC laws. For details of the design and construction of this robot refer to [Ratcliffe et al. (2006)]. The task is to collect an object from the feeder system on the right-hand side of the photograph, place it on the moving conveyor, and then return for the next one. The experimental facility replicates requirements in the food processing and other industries, where the same task is to be completed over and over again with a specified accuracy over a finite duration. Here the requirement is for the robot head to track a given reference signal from right to left each time it picks up an object.

Each axis of the gantry robot is modeled based on frequency response tests, where, since the axes are orthogonal, it is assumed that there is minimal interaction between them. Consider the X-axis, the one parallel to the conveyor in Fig. 8 where frequency response tests, using the Bode gain and phase plots in

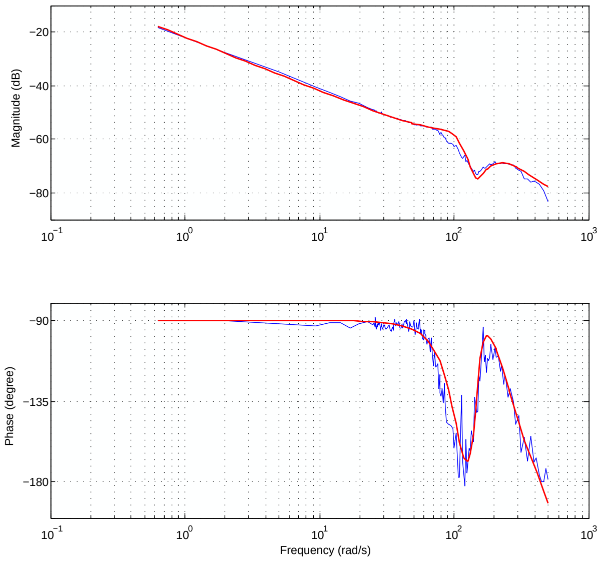


Fig. 9 Bode gain and phase plots for the X-axis of the gantry robot of Fig. 8, where the red line is the actual data and the blue that produced by the model.

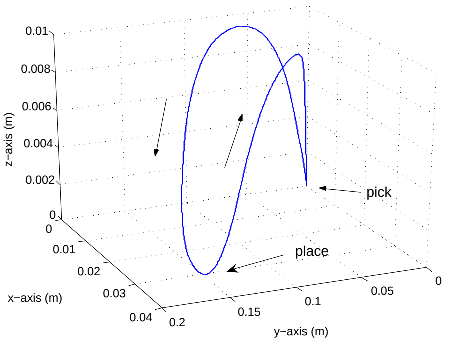


Fig. 10 The reference trajectory $r(p)$ for the gantry robot of Fig. 8.

Fig. 9, result in a 7th order continuous-time transfer-function as an adequate model of the dynamics to use for control law design. The transfer-function is discretized with a sampling time of $T_s = 0.01$ sec to develop a discrete linear state-space model. The required reference trajectory is designed to simulate a pick and place process of duration 2 sec, and the signal is used in all ILC law tests in order to make all results comparable. The 3D reference trajectory is given in Fig. 10 and that for the X-axis in Fig. 11. For a complete treatment of the modeling of this robot, including the transfer-functions for the Y and Z axes, again refer to [Ratcliffe et al. (2006)].

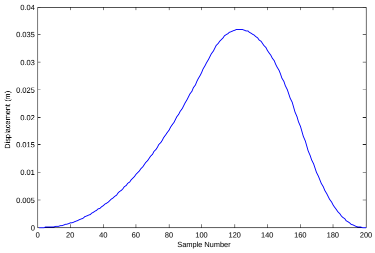
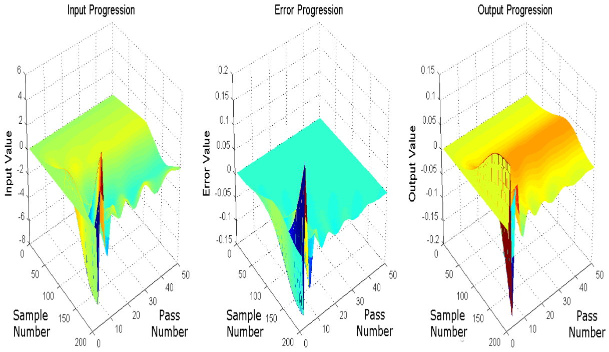


Fig. 11 The reference trajectory for the X-axis of the gantry robot.


Input Progression

Error Progression

Output Progression

Fig. 12 The input, error and output progression for the design of (50) applied to the X axis.

After discretization, the process state-space models for the X-axis are given by

$$A = \begin{bmatrix} 2.41 & -0.86 & 0.85 & -0.59 & 0.30 & -0.19 & 0.32 \\ 4.00 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.00 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.00 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.00 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.50 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.0313 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T,$$

$$C = \begin{bmatrix} 0.0095 & -0.0023 & 0.0048 & -0.0027 & 0.0029 & -0.0011 & 0.0029 \end{bmatrix}, \qquad (49)$$

where the state matrix A has all eigenvalues inside the unit circle except for one of value unity on the real axis of the complex plane.

It is of interest to first consider the design of an ILC control law for pass-to-pass error convergence only, that is, following [Kurek and Zaremba (1993)] and choose K_2 to satisfy $\rho(\hat{D}_0) = \rho(I - CBK_2) < 1$. One choice is

$$K_1 = 0, \; K_2 = 50. \qquad (50)$$

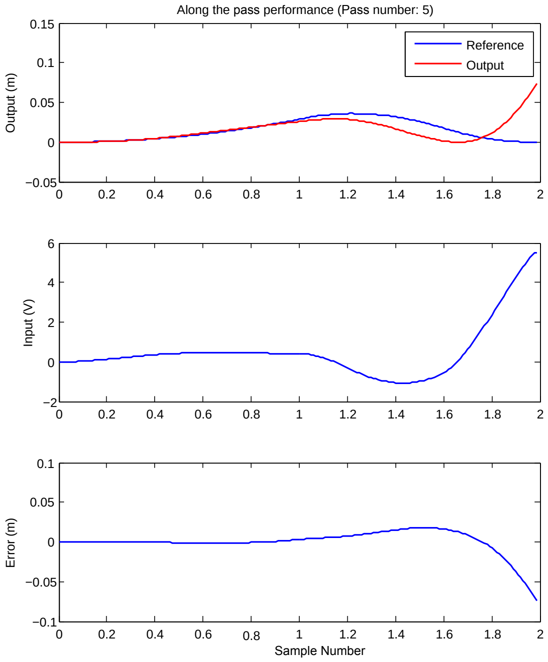


Fig. 13 The output (pass profile) on pass 5 (red line) compared to the reference (blue line) together with the input (middle plot) and the error (bottom plot).

The input, error and output progression for this design are shown in Fig.12 and Fig. 13 shows the response on pass 5. These are totally unacceptable and were not implemented experimentally as damage to the equipment would result.

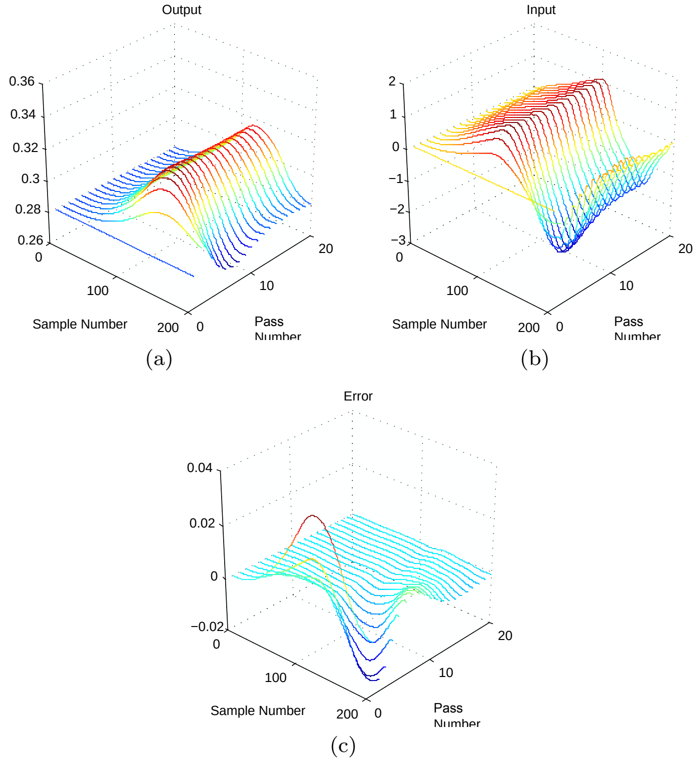
The LMI based design of Theorem 1 produces a family of solutions, and as one example consider the control law matrices

$$\begin{aligned} K_1 &= \left[7.3451\ -2.7245\ 0.1499\ 7.6707\ 2.7540\ -3.6088\ -20.4519\right], \\ K_2 &= 82.4119. \end{aligned} \tag{51}$$

As representative of the performance possible, and making use of the tuning opportunities offered by the LMI solutions, Fig. 14 shows the experimentally measured input, error, and output progression over 20 passes, where the tracking error is reduced to a low value.

The input, error and output signals on pass 200 are shown in Fig. 15 and these are highly acceptable. The design has been repeated for the Y (perpendicular to the X-axis in the same plane) and Z (perpendicular to the X-Y plane)-axes. As a comparison between simulated and experimental results, Fig. 16 shows the mean square error, denoted by MSE, for all axes generated by the ILC law with the matrices of (51). These show good agreement between simulation and experimentation.

In, for example, [Longman (2000)], it is reported that ILC laws can exhibit higher frequency noise build up as the number of passes increases and tracking



(a) (b)

(c)

Fig. 14 The input, error and output progression for the design of (51).

of the reference signal then begins to diverge due to numerical problems in both computation and measurement. In this design, the higher frequency component buildup was observed in some cases, resulting in vibrations that increase the pass error. One relatively simple option in such cases is to employ a zero-phase Chebyshev low-pass filter. Here the filter used has transfer-function

$$\begin{aligned} H(z) = & \frac{0.0002 + 0.0007z^{-1} + 0.0011z^{-2}}{1 - 3.5328z^{-1} + 4.7819z^{-2}} \cdots \\ & \frac{+0.0007z^{-3} + 0.0002z^{-4}}{-2.9328z^{-3} + 0.6868z^{-4}}, \end{aligned} \tag{52}$$

with cut-off frequency 10 Hz and is applied to the pass error after each pass is complete and before computation of the control law for the next pass. A case where the pass error without filtering starts to diverge after (approximately) 100 passes but the addition of the filter (52) of this type is able to maintain (this aspect of overall) performance is shown in Fig. 17. Such a filter is not required in all cases and an open research question is whether or not its selection can be included in the overall design algorithm.

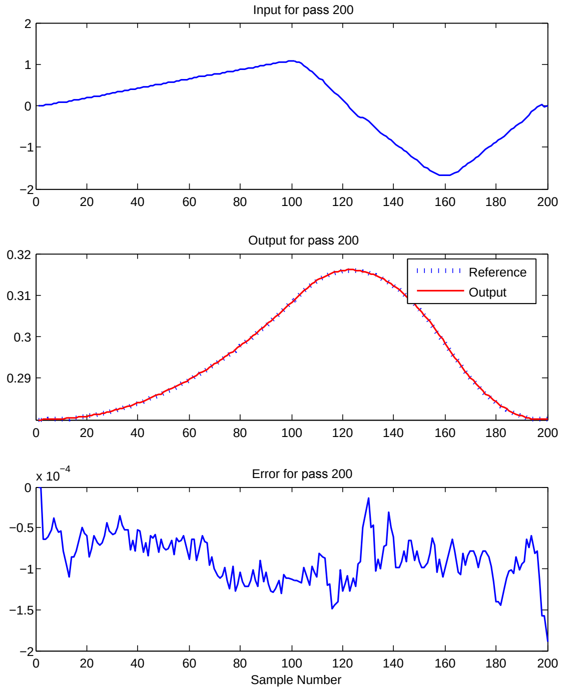


Fig. 15 The input (top plot), the reference and output (middle plot) and the error (bottom plot) on pass 200.

The ILC law (48) requires access to the state vector on both the current and previous passes. If an observer is not used, one option is to use the pass profile vectors on the current and previous passes. To develop this design, the ILC law is again of the form (34) and also (35) is used. Also select $\Delta(u_k(p), e_k(p))$ as

$$\Delta(u_k(p), e_k(p)) = K_1\mu_{k+1}(p+1) + K_2\mu_{k+1}(p) + K_3e_k(p+1), \tag{53}$$

with

$$\mu_k(p) = y_k(p-1) - y_{k-1}(p-1) = C\eta_k(p). \tag{54}$$

After routine analysis (53) can be written as

$$\Delta u_{k+1}(p-1) = K_1C\eta_{k+1}(p) + K_2C\eta_{k+1}(p-1) + K_3e_k(p) \tag{55}$$

and hence on introducing

$$\tilde{\eta}_{k+1}(p+1) = \begin{bmatrix} \eta_{k+1}(p+1) \\ \eta_{k+1}(p) \end{bmatrix}, \tag{56}$$

the controlled dynamics can be written as

$$\begin{aligned} \tilde{\eta}_{k+1}(p+1) &= \hat{A}\tilde{\eta}_{k+1}(p) + \hat{B}_0e_k(p), \\ e_{k+1}(p) &= \hat{C}\tilde{\eta}_{k+1}(p) + \hat{D}_0e_k(p), \end{aligned} \tag{57}$$

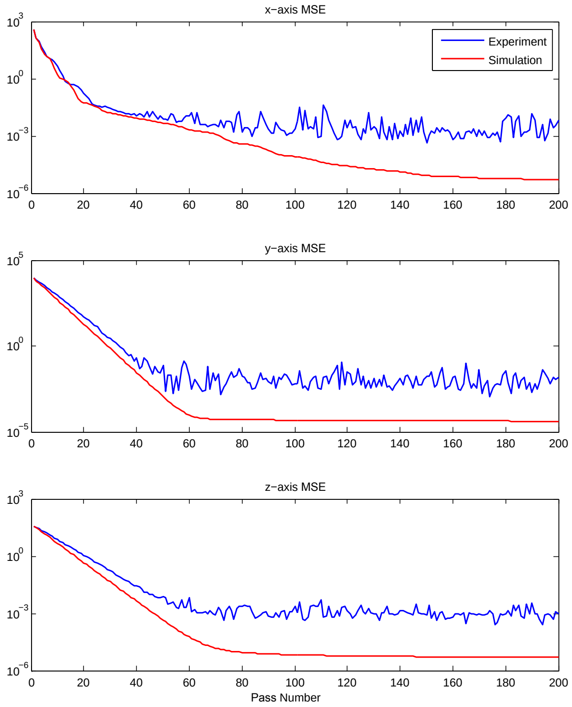


Fig. 16 MSE for all axes.

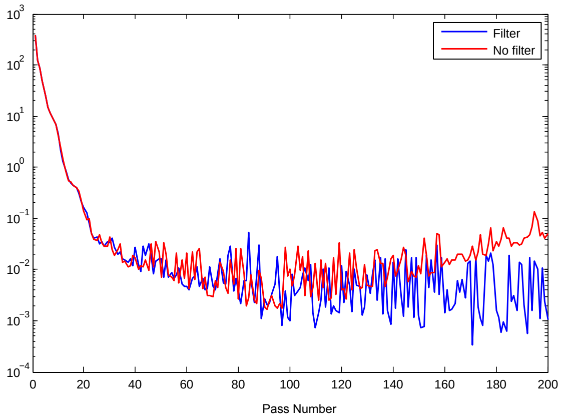


Fig. 17 The effect of filtering — red line without, blue line with filter added.

where

$$\begin{aligned}\hat{A} &= \begin{bmatrix} A + BK_1C & BK_2C \\ I & 0 \end{bmatrix}, \ \hat{B}_0 = \begin{bmatrix} BK_3 \\ 0 \end{bmatrix}, \\ \hat{C} &= \begin{bmatrix} -CA - CBK_1C & -CBK_2C \end{bmatrix}, \ \hat{D}_0 = (I - CBK_3).\end{aligned} \tag{58}$$

Theorem 2 *[Hladowski et al. (2012)] Stability along the pass holds for the ILC dynamics (57) if there exist matrices* $Y \succ 0$, $Z \succ 0$, N_1, N_2 *and* N_3 *such that the following LMI with linear constraints is feasible*

$$\begin{bmatrix} Z-Y & * & * \\ 0 & -Z & * \\ \Omega_1 & \Omega_2 & -Y \end{bmatrix} \prec 0, \\ CY_1 = PC, \;\; CY_2 = QC, \tag{59}$$

where

$$Y = \begin{bmatrix} Y_1 & 0 & 0 \\ 0 & Y_2 & 0 \\ 0 & 0 & Y_3 \end{bmatrix} \tag{60}$$

and

$$\Omega_1 = \begin{bmatrix} AY_1 + BN_1C & BN_2C & BN_3 \\ Y_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \\ \Omega_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -CAY_1 - CBN_1C & -CBN_2C & Y_3 - CBN_3 \end{bmatrix}. \tag{61}$$

The matrices P *and* Q *are additional decision variables. If the LMI with equality constraints of (59) is feasible, the control law matrices* K_1, K_2 *and* K_3 *are given by*

$$K_1 = N_1P^{-1}, \;\; K_2 = N_2Q^{-1}, \;\; K_3 = N_3Y_3^{-1}. \tag{62}$$

To apply the control law (55), simple algebraic manipulations give

$$\begin{aligned} u_k(p) = {} & u_{k-1}(p) + K_1(y_k(p) - y_{k-1}(p)) \\ & + K_2(y_k(p-1) - y_{k-1}(p-1)) \\ & + K_3(y_{ref}(p+1) - y_{k-1}(p+1)). \end{aligned} \tag{63}$$

The last term in this law is ILC phase-lead activated by the previous pass error. A variable advance is also possible and has been found to lead to accurate tracking in practice on a range of applications [Freeman et al. (2007), Wallen et al. (2008)]. The second and third terms are proportional in nature acting on the error between the current and previous passes at p and $p-1$ respectively. The use of current pass data has appeared in many approaches to manipulate the plant dynamics along the pass, see, for example, [Norloff and Gunnarsson (1999)] and has been found to increase initial tracking and disturbance rejection [Ratcliffe (2005)], (63) uses both current and previous pass data, where the latter is composed of ILC phase lag (the second term) and lead (the third term). It is therefore a higher order ILC algorithm with a relatively simple structure. This ILC design has also been experimentally verified on the gantry robot [Hladowski et al. (2012)].

4.2 Alternative Designs and Comparisons

For discrete dynamics an alternative setting for ILC design uses the lifted model, see, for example, [Ahn et al. (2007)] and [Bristow et al. (2006)] and the many subsequent publications on this approach where the first step in design is to define the super-vector

$$E_k = \left[\, e_k^\top(0)\ e_k^\top(1)\ \cdots\ e_k^\top(\alpha-1) \,\right]^\top$$

and with, as one example, the phase-lead ILC law

$$u_{k+1}(p) = u_k(p) + Ke_k(p+1), \tag{64}$$

applied the controlled dynamics can be written in the form $E_{k+1} = \mathcal{Q}E_k$, where $\mathcal{Q}$ is a block lower triangular matrix whose non-zero entries are formed from the Markov parameters of the system state-space model. This approach subsumes the along the pass dynamics and assumes that any requirements beyond pass-to-pass error convergence arising in a particular application are, if required, met by first designing a feedback control loop for the system and then applying lifting to the resulting state-space model. If the lifting setting is to be applied to the discretized dynamics of the gantry robot of the previous section then a stabilizing feedback control law must be applied before the ILC design as the state matrix is unstable in the case considered (the matrix A of (49)).

In comparison, the use of the repetitive process setting for ILC design enables simultaneous design for pass-to-pass error convergence and performance along the passes. Also the repetitive process setting is applicable to differential dynamics, for which no lifted model analysis is possible. Hence the repetitive process setting is applicable to applications where design by emulation is required or is the only feasible approach.

The ILC designs in [Haldowski et al. (2010)] and [Hladowski et al. (2012)] extend without difficulty to robust design where the uncertainty is described by either the norm-bounded or polytopic models. Many alternative approaches to robust ILC control design for discrete linear systems have been reported and one option is again to use the lifted model of the dynamics but this will encounter serious difficulties due to the presence of matrix products in the resulting model. For example, the ILC law (64) results in pass-to-pass error dynamics described by the lifted model

$$E_{k+1} = \mathcal{Q}E_k, \tag{65}$$

where

$$\mathcal{Q} = \begin{bmatrix} I-KCB & 0 & \cdots & 0 \\ -KCAB & I-KCB & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -KCA^{\alpha-1}B & -KCA^{\alpha-2}B & \cdots & I-KCB \end{bmatrix}. \tag{66}$$

Hence when the state-space model matrices A, B, C have uncertainty associated with them that belong to a convex set, $\mathcal{Q}$ does not belong to a convex set and only a bound is possible that increases the conservativeness of the results.

The repetitive process based ILC designs given in the previous section are based on sufficient but not necessary stability conditions and the question of how conservative these designs are arises. In particular, how conservative are these designs against those based on the necessary and sufficient conditions for stability along the pass given in the previous section. No general answer is available for this question but the necessary and sufficient conditions require frequency attenuation over the complete frequency spectrum and by analogy with the standard linear systems case is likely to be very stringent. One option is to use the GKYP lemma [Paszke et al. (2013)] to permit design over finite frequency ranges of interest and again these designs have been experimentally benchmarked on the gantry robot. As another problem demonstrating the scope of repetitive process models in ILC design, [Cichy et al. (2014)] has used the wave repetitive process setting to examine how best to make use of previous pass data.

4.3 Strong Practical Stability and Disturbance Rejection

Stability along the pass requires uniform boundedness of the pass profiles generated for all k and α. As a result, the condition expressed in terms of the transfer-function matrix (39), that is, the transfer-function matrix that describes the contribution of the previous pass dynamics to the current pass dynamics is imposed. Consider the single-input single-output case for simplicity. Then this condition, in terms of control law design, requires frequency attenuation of the error dynamics from pass-to-pass. As in the 1D case, this condition is very stringent and unnecessary in many practical cases where the system involved is only required to operate over finite frequency ranges.

Physical repetitive processes will only ever complete a finite number of passes and motivated by this fact strong practical stability removes the uniform boundedness requirement as both $k \to \infty$ and $\alpha \to \infty$ but still demands this property when a) both k and p are finite, b) the pass index $k \to \infty$ and the pass length α is finite and c) the pass index k is finite and the pass length $\alpha \to \infty$: Cases a) and b) have obvious practical motivation and Case c) is the mathematical formulation of an application where the process completes a finite number of passes but the pass length is very long, and there is a requirement to control the along the pass dynamics. In particular, in control law design to force the pass profile to track a reference attenuation at each frequency component is relaxed to that of requiring this property at a subset of frequencies.

Strong practical stability of processes described by (15) holds by the analysis in [Dabkowski et al. (2009)] if and only if **[a]** $\rho(D_0) < 1$, **[b]** $\rho(A) < 1$, **[c]** $\rho(A + B_0(I - D_0)^{-1}C) < 1$, and **[d]** $\rho(C(I - A)^{-1}B_0 + D_0) < 1$, where the matrix inverses in **[c]** and **[d]** hold if **[a]** and **[b]** hold, respectively. To formulate

$\mathcal{H}_\infty$ disturbance attenuation for this model one way would be to use stability along the pass and the norm on 2D signals defined over $(k, p) \in [0, \infty] \times [0, \infty]$, considered in, for example, [Du and Xie (2002)] for 2D discrete linear systems.

This approach for control law design would have the same implications in terms of frequency attenuation detailed in the previous section for stability along the pass. Instead, the analysis that follows uses strong practical stability and measures of disturbance attenuation defined in terms of the evolution over one independent variable with the other fixed, that is, with p fixed and k ranging over $[0, \infty]$ and also k fixed and p ranging over $[0, \infty]$. The state-space model of the process dynamics now is

$$\begin{aligned} x_{k+1}(p+1) &= Ax_{k+1}(p) + Bu_{k+1}(p) + B_0 y_k(p) + B_1 \omega_{k+1}(p), \\ y_{k+1}(p) &= Cx_{k+1}(p) + Du_{k+1}(p) + D_0 y_k(p) + D_1 \omega_{k+1}(p), \end{aligned} \tag{67}$$

where $x_k(p)$, $y_k(p)$ and $u_k(p)$ are as in (15) and $\omega_k(t) \in \mathbb{R}^{n_w}$ is a disturbance vector acting on both the state and pass profile vectors. In some cases, it may be required to consider different disturbances acting on the state and pass profile dynamics. Let $B_1^a \omega_k^a(p)$ and $D_1^b \omega_k^b(p)$ denote the additive disturbances on the state and pass profile dynamics, respectively, and this case is included in the model of (67) on setting

$$\omega_k(p) = \begin{bmatrix} \omega_k^a(p) \\ \omega_k^b(p) \end{bmatrix}, \; B_1 = \begin{bmatrix} B_1^a \; 0 \end{bmatrix}, \; D_1 = \begin{bmatrix} 0 \; D_1^b \end{bmatrix}.$$

Consider an asymptotically stable process described by (67) with zero input vector $(u_{k+1}(p) = 0)$ and a sequence $\{\omega_k\}_{k \geq 1}$ that converges strongly to ω_∞ as $k \to \infty$. Then (following the analysis for the input only case in [Rogers et al. (2007)]) the repetitive process dynamics converge as $k \to \infty$ to the limit profile

$$\begin{aligned} x_\infty(p+1) &= (A + B_0(I - D_0)^{-1}C)x_\infty(p) + (B_1 + B_0(I - D_0)^{-1}D_1)\omega_\infty(p), \\ y_\infty(p) &= (I - D_0)^{-1}Cx_\infty(p) + (I - D_0)^{-1}D_1\omega_\infty(p), \end{aligned} \tag{68}$$

where $(I - D_0)^{-1}$ exists since $\rho(D_0) < 1$. This is a discrete linear systems state-space model with indeterminate p and condition **[c]** is the requirement for this model to be stable. Furthermore, under condition **[b]** for strong practical stability the repetitive dynamics converge as $p \to \infty$ for any finite k are (again following the analysis in for the input only case in [Rogers et al. (2007)]) to the limit profile

$$\begin{aligned} y_{k+1}(\infty) &= \left(C(I - A)^{-1}B_0 + D_0\right) y_k(\infty) + \left(C(I - A)^{-1}B_1 + D_1\right) \omega_{k+1}(\infty), \\ x_{k+1}(\infty) &= (I - A)^{-1}B_0 y_k(\infty), + (I - A)^{-1}B_1 \omega_{k+1}(\infty), \end{aligned} \tag{69}$$

where $(I - A)^{-1}$ exists since $\rho(A) < 1$. This a discrete linear systems state-space model with indeterminate k and condition **[d]** is the requirement for this model to be stable. In what follows, the initial pass profile at $p = 0$ with zero state initial vector sequence and control input vector, respectively, is used, that is,

$$y_{k+1}(0) = D_0 y_k(0) + D_1 \omega_{k+1}(0). \tag{70}$$

Consider $\mathcal{H}_\infty$ disturbance attenuation for these processes, where the 2D systems formulation would consider this property over $(k,p) \in [0,\infty] \times [0,\infty]$. Under strong practical stability, this requirement is relaxed to the 'boundaries' of this domain, that is, is imposed on (a) $y_k(0)$ for all values of the variable k, (b) $y_k(\infty)$ for all values of k and (c) $y_\infty(p)$ for all values of p. These disturbance attenuation measures are practically relevant since control design for applications should aim to reach a limit profile, that is, produce the same output of each pass, with acceptable along the pass dynamics and ensure that each pass completed also has acceptable along the pass dynamics. The required conditions can be formulated in terms of the standard linear systems H_∞ norm as detailed next.

Definition 1 (Performance): Suppose that the discrete linear repetitive process (67) with zero boundary conditions $x_k(0) = 0$ for $k \geq 1$ and zero input $u_k(p) = 0$ for $k, p \geq 0$, has the strong practical stability property. Given any 2D disturbance sequence $\omega = \{\omega_k\}_{k\geq 0}$, of p-indexed sequences $\{\omega_k(p)\}_{p\geq 0}$, that has the 1D strong limit $\bar{\omega} := \{\omega_\infty(p)\}_{p\geq 0}$ in k and/or 1D strong limit $\hat{\omega} := \{\omega_k(\infty)\}_{k\geq 0}$ in p, define the corresponding 1D sequences $\bar{y} := \{y_\infty(p)\}_{p\geq 0}$ and/or $\hat{y} := \{y_k(\infty)\}_{k\geq 0}$ according to (68) and/or (69). Moreover, let $\tilde{\omega} := \{\omega_k(0)\}_{k\geq 0}$ and define the corresponding 1D sequence $\tilde{y} := \{y_k(0)\}_{k\geq 0}$ according to (70). Then the process (13) is said to achieve $\mathcal{H}_\infty$ strong practical performance at the level of $\gamma_1 > 0$ in k (or trial) if

$$\sup_{0\neq\tilde{\omega}\in\ell_2} \frac{\|\tilde{y}\|_2}{\|\tilde{\omega}\|_2} < \gamma_1 \quad \text{and} \quad \sup_{0\neq\hat{\omega}\in\ell_2} \frac{\|\hat{y}\|_2}{\|\hat{\omega}\|_2} < \gamma_1$$

and/or at the level of $\gamma_2 > 0$ in p (or pass) if

$$\sup_{0\neq\bar{\omega}\in\ell_2} \frac{\|\bar{y}\|_2}{\|\bar{\omega}\|_2} < \gamma_2,$$

where ℓ_2 denotes the Hilbert space of square summable 1D sequences with norm $\|\{u_k\}_{k\geq 0}\|_2 = \sqrt{\sum_{k\geq 0} u_k^T u_k}$ (or $\|\{u(p)\}_{p\geq 0}\|_2 = \sqrt{\sum_{p\geq 0} u^T(p)u(p)}$).

Theorem 3 *[Dabkowski et al. (2012)] A discrete linear repetitive process described by (67) is strongly practically stable and has $\mathcal{H}_\infty$ disturbance attenuation $\gamma_1 > 0$ in the direction k, and $\gamma_2 > 0$ in the direction p if and only if there exist matrices $W_1 \succ 0$, $W_2 \succ 0$, $Q_1 \succ 0$, $Q_2 \succ 0$ and nonsingular matrices G_1, G_2, and S_2, such that the following set of LMIs is feasible*

$$\begin{bmatrix} -W_2 & 0 & S_2^T D_0^T & S_2^T D_0^T \\ 0 & -\gamma_1^2 I & D_1^T & D_1^T \\ D_0 S_2 & D_1 & -I & 0 \\ D_0 S_2 & D_1 & 0 & W_2 - S_2 - S_2^T \end{bmatrix} \prec 0, \tag{71}$$

$$\begin{bmatrix} -W_1 & W_1 A^T \\ AW_1 & -W_1 \end{bmatrix} \prec 0, \tag{72}$$

$$\begin{bmatrix} -Q_1 & 0 & G_1^T \bar{A}_1^T & G_1^T \bar{C}^T \\ 0 & -\gamma_2^2 & \bar{B}^T & 0 \\ \bar{A}_1 G_1 & \bar{B} & Q_1 - \bar{E}_1 G_1 - G_1^T \bar{E}_1^T & 0 \\ \bar{C} G_1 & 0 & 0 & -I \end{bmatrix} \prec 0, \tag{73}$$

$$\begin{bmatrix} -Q_2 & 0 & G_2^T \bar{A}_2^T & G_2^T \bar{C}^T \\ 0 & -\gamma_1^2 & \bar{B}^T & 0 \\ \bar{A}_2 G_2 & \bar{B} & Q_2 - \bar{E}_2 G_2 - G_2^T \bar{E}_2^T & 0 \\ \bar{C} G_2 & 0 & 0 & -I \end{bmatrix} \prec 0, \tag{74}$$

where

$$\bar{A}_1 = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} 0 & I \end{bmatrix}, \bar{E}_1 = \begin{bmatrix} I & -B_0 \\ 0 & I - D_0 \end{bmatrix}, \bar{B} = \begin{bmatrix} B_1 \\ D_1 \end{bmatrix},$$
$$\bar{A}_2 = \begin{bmatrix} 0 & B_0 \\ 0 & D_0 \end{bmatrix}, \bar{E}_2 = \begin{bmatrix} I - A & 0 \\ -C & I \end{bmatrix}.$$

Returning to the ILC setting, the dynamics to be controlled are assumed to be represented by discrete linear time-invariant state-space model, written in the ILC setting as

$$\begin{aligned} x_{k+1}(p+1) &= Ax_{k+1}(p) + Bu_{k+1}(p) + B_1 \omega_{k+1}^x(p), \\ y_{k+1}(p) &= Cx_{k+1}(p) + D_1 \omega_{k+1}^y(p), \end{aligned} \tag{75}$$

where $x_k(p) \in \mathbb{R}^n$, $y_k(p) \in \mathbb{R}^m$ and $u_k(p) \in \mathbb{R}^r$ are as in (32) and $\omega^x \in \mathbb{R}^{s_1}$ and $\omega^y \in \mathbb{R}^{s_2}$ are disturbances acting on the current pass state and output vectors.

Introduce, for analysis purposes only, the following

$$\begin{aligned} \eta_{k+1}(p+1) &= x_{k+1}(p) - x_k(p), \\ \Delta u_{k+1}(p) &= u_{k+1}(p) - u_k(p), \\ \zeta_{k+1}^x(p) &= \omega_{k+1}^x(p) - \omega_k^x(p), \\ \zeta_{k+1}^y(p) &= \omega_{k+1}^y(p) - \omega_k^y(p), \end{aligned} \tag{76}$$

where it is assumed that $\omega_{k+1}^x(p) \neq \omega_k^x(p)$ and $\omega_{k+1}^y(p) \neq \omega_k^y(p)$. Consider also an ILC law in the form defined by (34), (35) and (36) and again $e_k(p) = r(p) - y_k(p)$. Then the controlled ILC dynamics can be written in the form

$$\begin{aligned} \eta_{k+1}(p+1) &= \widehat{A}\eta_{k+1}(p) + \widehat{B}_0 e_k(p) + \widehat{B}_1 \widehat{\zeta}_{k+1}(p), \\ e_{k+1}(p) &= \widehat{C}\eta_{k+1}(p) + \widehat{D}_0 e_k(p) + \widehat{D}_1 \widehat{\zeta}_{k+1}(p), \end{aligned} \tag{77}$$

where

$$\begin{aligned} \widehat{A} &= A + BK_1, \ \widehat{B}_0 = BK_2, \\ \widehat{B}_1 &= \begin{bmatrix} B_1 & 0 \end{bmatrix}, \ \widehat{C} = -C(A + BK_1), \\ \widehat{D}_0 &= (I - CBK_2), \ \widehat{D}_1 = \begin{bmatrix} -CB_1 - D_1 \end{bmatrix}, \\ \widehat{\zeta}_{k+1}(p) &= \begin{bmatrix} \zeta_{k+1}^x(p-1) \\ \zeta_{k+1}^y(p) \end{bmatrix}. \end{aligned} \tag{78}$$

This is a linear repetitive process state-space model of the form (67) with current pass state and pass profile vectors $\eta_{k+1}(p)$ and $e_{k+1}(p)$ respectively. Hence the following results can be established based on Theorem 3.

Lemma 1 *The ILC dynamics described by* (77) *cannot be strongly practically stable if the matrix* CBK_2 *is singular.*

Theorem 4 *Strong practical stability holds for the ILC dynamics described by (77) with* $\mathcal{H}_\infty$ *disturbance attenuation* $\gamma_1 > 0$ *in the direction* k*, and* $\gamma_2 > 0$ *in the direction* p *if there exist matrices* $W_1 \succ 0$*,* $W_2 \succ 0$*,* $Q_1 \succ 0$*,* $Q_2 \succ 0$*, a nonsingular matrix* $G = diag(\bar{G}_1, \bar{G}_2)$ *and rectangular matrices* $\widetilde{N}_1 = \begin{bmatrix} N_1 \; 0 \end{bmatrix}$ *and* $\widetilde{N}_2 = \begin{bmatrix} 0 \; N_2 \end{bmatrix}$ *such that the following set of LMIs is feasible*

$$\begin{bmatrix} -W_2 & * & * & * \\ 0 & -\gamma_1^2 I & * & * \\ \bar{G}_2 - CBN_2 & D_1 & -I & 0 \\ \bar{G}_2 - CBN_2 & D_1 & 0 & W_2 - \bar{G}_2 - \bar{G}_2^T \end{bmatrix} \prec 0, \tag{79}$$

$$\begin{bmatrix} -W_1 & * \\ A\bar{G}_1 + BN_1 & W_1 - \bar{G}_1 - \bar{G}_1^T \end{bmatrix} \prec 0, \tag{80}$$

$$\begin{bmatrix} -Q_1 & * & * & * \\ 0 & -\gamma_2^2 & * & * \\ \widetilde{A}_1 G + \mathit{\Pi} \widetilde{N}_1 & \bar{B} & \mathit{\Gamma}_1 & * \\ \bar{C}G & 0 & 0 & -I \end{bmatrix} \prec 0, \tag{81}$$

$$\begin{bmatrix} -Q_2 & * & * & * \\ 0 & -\gamma_1^2 & * & * \\ \widetilde{A}_2 G + \mathit{\Pi} \widetilde{N}_2 & \bar{B} & \mathit{\Gamma}_2 & * \\ \bar{C}G & 0 & 0 & -I \end{bmatrix} \prec 0, \tag{82}$$

where

$$\mathit{\Gamma}_1 = Q_1 - \bar{E}_1 G - (\bar{E}_1 G)^T = Q_1 - \widetilde{E}_1 G + \mathit{\Pi} \widetilde{N}_2 - G^T \widetilde{E}_1^T + \widetilde{N}_2^T \mathit{\Pi}^T,$$

$$\mathit{\Gamma}_2 = Q_2 - \bar{E}_2 G - (\bar{E}_2 G)^T = Q_2 - \widetilde{E}_2 G + \mathit{\Pi} \widetilde{N}_1 - G^T \widetilde{E}_2^T + \widetilde{N}_1^T \mathit{\Pi}^T,$$
$$\bar{C} = \begin{bmatrix} 0 \; I \end{bmatrix}$$

and

$$\mathit{\Pi} = \begin{bmatrix} B \\ -CB \end{bmatrix}, \quad \widetilde{A}_1 = \begin{bmatrix} A & 0 \\ -CA & 0 \end{bmatrix}, \widetilde{E}_1 = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix},$$
$$\widetilde{A}_2 = \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix},$$
$$\widetilde{E}_2 = \begin{bmatrix} I - A & 0 \\ CA & I \end{bmatrix}.$$

If this set of LMIs is feasible, stabilizing control law matrices are given by

$$K_1 = N_1 \bar{G}_1^{-1},\ K_2 = N_2 \bar{G}_2^{-1}. \tag{83}$$

Proof This result follows from applying the results of Theorem 3 to (77) and applying routine operations. Hence the details are omitted.

Remark 1 To find stabilizing K_1 and K_2 it is necessary to use $G = diag(\bar{G}_1, \bar{G}_2)$ in Theorem 4 to avoid introducing additional strong links between these matrices and a potential cost of additional conservativeness.

Remark 2 This last theorem gives sufficient conditions for solvability of $\mathcal{H}_\infty$ control law design problem as specified by γ_1 from pass-to-pass and γ_2 along the pass. Hence the interest in minimizing the values of γ_1 and γ_2 but this may not be possible as the two criteria involved do not need to have a joint minimum. An alternative is to create a convex combination of γ_1 and γ_2 and design the control law to minimize the resulting function. One choice is to solve the convex optimization problem:

$$\begin{gathered}\min\ \sigma \quad \text{subject to (79) – (82)} \\ (\text{where} \quad \sigma = \bar{\alpha}_1\gamma_1^2 + \bar{\alpha}_2\gamma_2^2)\end{gathered} \tag{84}$$

for some $0 \le \bar{\alpha}_1 \le 1$, $0 \le \bar{\alpha}_2 \le 1$, such that $\bar{\alpha}_1 + \bar{\alpha}_2 = 1$. Also varying the selected values of $\bar{\alpha}_1$ and $\bar{\alpha}_2$ increases the options in this respect.

5 *n*D Systems Applications in Civil Engineering and Agriculture

With a view to further increasing the range of applications for nD systems and/or repetitive processes, two areas are considered in this section. The first of these argues the case for extending the linear model stability theory to nonlinear dynamics and the second concerns systems described by partial differential equations. These add to the ongoing work on an nD systems approach to spatially interconnected and related systems, which have received some attention in the literature, see, for example, [D'Andrea and Dullerud (2003), Recht and D'Andrea (2004),Liu et al. (2014)].

5.1 Soil Compaction

Compacting soil is a key task in construction and farming applications where the machine, or compactor, makes repeated passes over an area of soil that is periodically built up with additional layers. The performance criterion is to achieve a specified height above some datum line with a desired compaction to meet the requirements of a particular application.

Consider the schematic diagram [Mooney and Adam (2007)] of Fig. 18 where a compactor makes repeated passes over an area of soil that is periodically built up with additional layers, termed lifts. The ultimate goal is to achieve a desired compaction to meet the requirements of the application. It is possible to determine the stiffness resulting from a pass due to the transmission of known energy from the compactor to the soil, and accelerometers can be used to measure the energy that is reflected back to the roller. The availability of this measurement motivates the idea of controlling the applied energy as a function of spatial location from pass-to-pass, in an effort to reduce the overall number of passes needed and also reduce the risk of over-compaction without unnecessary manual measurements.

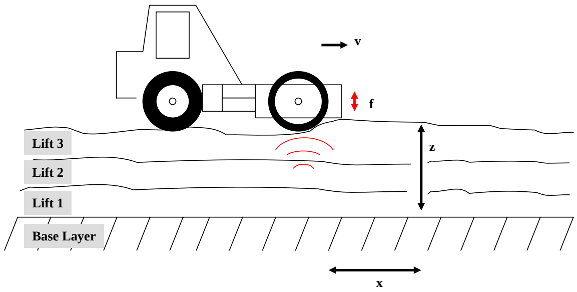


Fig. 18 Schematic diagram of soil compaction.

Fig. 19 shows a vibratory roller of mass M acting on the soil with a force $F(x,t)$, and moving at a velocity $V(x,t)$. The objective is to deliver some amount of energy to the soil within a given unit of distance Δx, termed the impact interval, and the energy, which is a function of spatial location x, is the control variable. Moreover, the force is produced by a combination of the static weight of the system, and the oscillation frequency ω and amplitude A of an eccentric mass rotated inside the roller. The net effect is that an amount of energy is delivered to the soil at each point in space x, denoted by $E(x)$. The energy is treated as a function of spatial location x, as the control variable for the problem, though in practice the control variables are the velocity $V(x,t)$, and the values of ω and A.

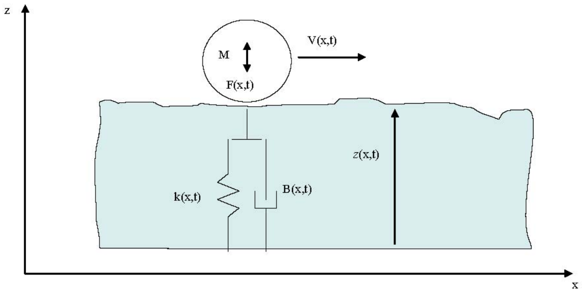


Fig. 19 A simplified model of the soil compaction problem used to obtain a model for control-related studies.

Typically the soil can be characterized by its mass, stiffness, and damping, and modeled by a mass-spring-damper system as shown in Fig. 19. By constructing a more relevant model of the roller, and its associated eccentric rotation mass, it is possible to extract the value of the soil stiffness, as measured by the spring constant k(x), using accelerometer measurements associated with the motion of the roller [Mooney and Adam (2007)].

All the variables in the system except the mass of the roller are spatio-temporal, but the scale of the problem makes it reasonable to assume initially that the time-variation can be ignored, and hence it is necessary only to consider the steady-state behavior of each variable or parameter. Therefore it is necessary only to consider spatial variation in the variables.

As discussed above, the time variation of the roller's velocity, and its applied force can be aggregated to an effective energy value as a function of x, and, since the interest is in pass-to-pass behavior, it is reasonable to assume that time-domain transients have died out between passes. Hence the need to solve a partial differential equation, to determine the values of height, stiffness, and damping in response to the application of force as a function of time from the roller, can be replaced by the problem of solving an ordinary differential equation in space.

Suppose that, before a given pass of the roller, the soil has a current state of compaction given by $k_j(x)$, where the subscript j denotes the pass number. Suppose also that during the pass the compactor adds energy $E_j(x)$ to the soil, where the energy results from the combined effect of the velocity of the roller along the pass, and the frequency and amplitude of the rotating eccentric mass inside the roller. The assumption needed to proceed is that there is a function that computes $k_{j+1}(x)$ from $k_j(x)$, and the applied energy $E_j(x)$, that is, a mapping that describes the pass-to-pass updating structure. Hence, as illustrated in Fig. 20, in the simplest case a relationship of the form

$$k_{j+1}(x) = f(k_j(x), E_j(x)), \tag{85}$$

can be expected.

Having started the process with the soil in a current state of compaction given by $k_j(x)$, and introducing energy $E_j(x)$ to the soil during the pass using the compactor, resulting in the next state of compaction $k_{j+1}(x)$, the task now is to design a suitable controller update algorithm that can compute the energy to be applied on the next pass. In algorithm terms hence

$$E_{j+1}(x) = h(E_j(x), k_d(x) - k_j(x)), \tag{86}$$

where $k_d(x)$ is the desired compaction or stiffness.

The term $k_d(x) - k_j(x)$ is the difference between the soil stiffness that resulted from the applied energy on pass j and the desired stiffness $k_d(x)$. On pass $j + 1$, (86) applies energy that is a function of the energy applied on the previous pass and the error between the desired stiffness and that at the completion of pass j. Hence the unique feature of a repetitive process is present, that is, pass-to-pass updating over a finite duration or pass length.

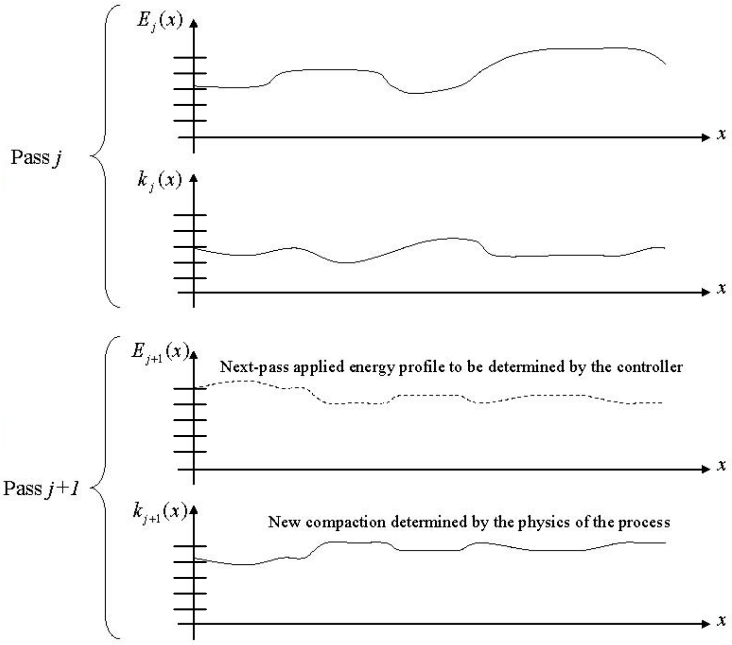

Pass j
$E_j(x)$
x
$k_j(x)$
x
Pass j+1
$E_{j+1}(x)$
Next-pass applied energy profile to be determined by the controller
x
$k_{j+1}(x)$
New compaction determined by the physics of the process
x

Fig. 20 A schematic representation of model building for the soil compaction.

Here it is assumed that during a pass it is possible to measure the compaction behind the compactor, that is, before the pass starts the soil has an existing compaction level. As the compactor travels over the soil there is a resulting change in compaction and it is assumed that the compaction be measured during the pass but no correcting action can be applied until the next pass.

Combining (85) and (86) gives the state-space model

$$\begin{aligned} k_{j+1}(x) &= f(k_j(x), E_j(x)), \\ E_{j+1}(x) &= h(E_j(x), k_d(x) - k_j(x)), \end{aligned} \tag{87}$$

for the controlled system, with $E_0(x)$ and $k_0(x)$ given. Assuming that X is quantized over the pass length, (87) describes a discrete nonlinear repetitive process but there are cases where the model may need extending to provide a more realistic description of the dynamics. The reason is the assumption in (87) that $k_{j+1}(x)$ is a nonlinear function of k and E at the same point on the previous pass j, that is, the point contact only occurs between the vibratory roller and the soil. In practice, the compactor has an impact on the soil in front of it as it moves along the pass and therefore a wave discrete nonlinear repetitive process is one way of representing the dynamics.

Examples such as this one motivate the need for further research to develop a stability theory for nonlinear repetitive processes and nD nonlinear systems. Early work in the nD systems case can be found in [Kurek (1995)], with more recent results in [Yeganefar et al. (2013),Emelianova et al. (2014), Emelianova et al. (2015)]. Another emerging application area for repetitive processes that will require a nonlinear stability theory on which to base control law design is laser metal deposition process modeling and control, see, for example, [Sammons et al. (2013)].

5.2 Dry Land Farming

All of the repetitive process models previously considered executed multiple passes that start from the same location. Dry-land farming results in a repetitive process model where the along the pass dynamics are defined spatially rather than with respect to a time axis and the control degrees of freedom are coupled multiplicatively.

In dry land farming, where irrigation is required to support crop growth, it is common [Moore and Chen (2006)] to find center pivot irrigators as shown in Fig. 21. The sprayer rotates about a central pumping station, with the result that each part of the field is irrigated once per cycle. Flow rates and fertilizer additions can be variably controlled along the length of the sprayer. The sprayer may be programmed to provide less water over areas of the field with clay-like soil, with a slower diffusion rate, and more over areas with sandy soil and a higher diffusion rate, in an attempt to provide more uniform moisture levels. Such usage is open-loop, but using low-power, commercial off-the-shelf radio transmitters, with associated transducers and embedded computers, such as Motes, or aerial-obtained hyperspectral images, moisture measurements over the entire area of a field can be obtained in real-time and then used to control flow rates and fertilizer addition.

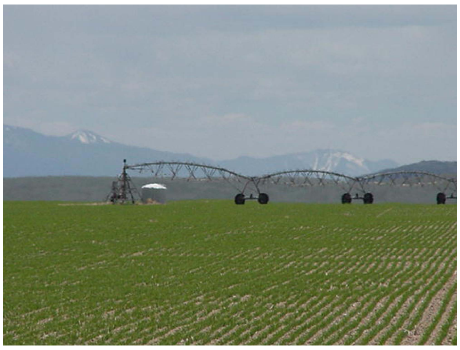


Fig. 21 Dry-land farming.

A schematic representation of model building for the irrigator is shown in Fig. 22. Starting with the specification of a variable of interest, which could be the concentration of a nutrient or the moisture content of a unit volume of soil, which varies in three spatial dimensions and with time, and the units of variables and constants are taken to be normalized with no loss of meaning. The irrigator is the control input. Suppose there is some variable of interest,

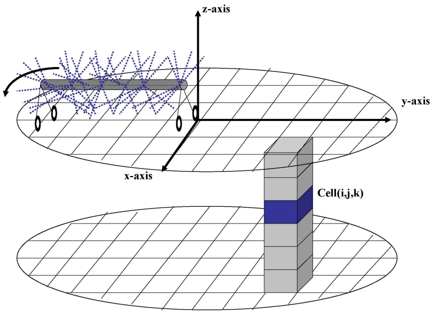


Fig. 22 A schematic representation of model building for the irrigator.

such as the concentration of a nutrient or the moisture content of a unit volume of soil, which varies in three spatial dimensions and with time, and the units of variables and constants are taken to be normalized with no loss of meaning from this point onwards.

Let $c(x, y, z, t)$ denote this variable and assume that the surface source of $c(x, y, z, t)$ is the center pivot irrigator, which travels around the field at a constant rate with period T_p. The irrigator is the control input in the process. It is also assumed that the irrigator can supply the variable of interest using a liquid spray that is applied with a flow rate $F(x, y, t) = v(x, y, 0, t)$, where $v(x, y, z, t)$ is the velocity through the soil of the variable, and the input is applied on the surface only. Further, it is assumed that the variable of interest is carried by the applied liquid in some proportion, given by $c(x, y, 0, t) = k_v v(x, y, 0, t)$. One further assumption is that the flow process evolves spatially only in the vertical direction.

The spatial decay rate of the velocity variable is denoted by γ and $D = D(x, y)$ denotes the diffusion-rate as a function of space in two dimensions only. Writing $c = c(x, y, z, t)$ and $v = v(x, y, z, t)$ when the subscripts are obvious from the context, the model of the system is given by the diffusion process with coupled flow [Cantrell and Cosner (2003)] described by

$$\frac{\partial v}{\partial t} = -\gamma \frac{\partial v}{\partial z}, \tag{88}$$

$$\frac{\partial c}{\partial t} = \frac{\partial}{\partial x}(D\frac{\partial c}{\partial x}) + \frac{\partial}{\partial y}(D\frac{\partial c}{\partial y}) + D\frac{\partial^2 c}{\partial z^2} - \frac{\partial (vc)}{\partial z}, \tag{89}$$

with all initial and boundary conditions equal to zero except

$$c(x, y, 0, t) = k_v v(x, y, 0, t), \ \ v(x, y, 0, t) = F(x, y, t). \tag{90}$$

The first three terms in (88) and (89) account for unforced diffusion, and the last term represents the effect of the flow containing the variable of interest.

The equations (88), (89), and (90) define the dynamics of the system, and it remains to model the input to the system, starting from treating the action of center pivot as defining N sectors in a single cycle. Also when the irrigator's angle places it in a given sector, only that sector receives water. Hence a masking function for each sector that only allows that sector to be on is required.

Let R denote the radius of a circle in the $x-y$ plane with center at (x_c, y_c). Suppose also that N, and T_p are given and set $\theta(x,y) = \arctan(y/x)$. Then the masking function $\delta_i(x,y)$, $i = 1, \ldots, N$, is defined by

$$\delta_i(x,y) = \begin{cases} 1, & (i-1)\frac{2\pi}{N} \le \theta(x,y) < i\frac{2\pi}{N} \text{ and} (x-x_c)^2 + (y-y_c)^2 \le R^2, \\ 0, & \text{otherwise,} \end{cases} \tag{91}$$

Also define the periodic function $\delta_i(t) = \delta_i(t - T_p)$ as

$$\delta_i(t) = \begin{cases} 1, & \frac{T_p}{N} \le t < i\frac{T_p}{N}, \\ 0, & \text{otherwise,} \end{cases} \tag{92}$$

and let

$$\delta_i(x,y,t) = \delta_i(x,y)\delta_i(t). \tag{93}$$

Figs. 23 and 24 illustrate $\delta_i(x,y)$, and $\delta_i(t)$, respectively, when $R = 12, N = 10, T_p = 20, x_c = 15$, and $y_c = 15$. Fig. 23 plots the function $\delta_2(x,y)$, and illustrates that $\delta_i(x,y)$ can be considered as a masking function in the $x-y$ plane. Finally, Fig. 24 plots $\delta_1(t), \delta_2(t)$, and $\delta_{10}(t)$ and illustrates that $\delta_i(t)$ can be considered as a masking function in time with pulse width $T_p/N = 2$.

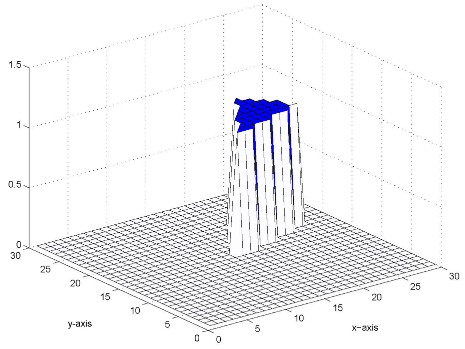


Fig. 23 System input modeling for the dry land farming application with dynamics described by (88), (89), and (90). For this purpose, it is necessary to use several specialized forms of the Dirac delta function given by (91), (92), and (93). The function $\delta_2(x,y)$ obtained by setting $i = 2$ in (93) is plotted for $R = 12, N = 10, x_c = 15$, and $y_c = 15$.

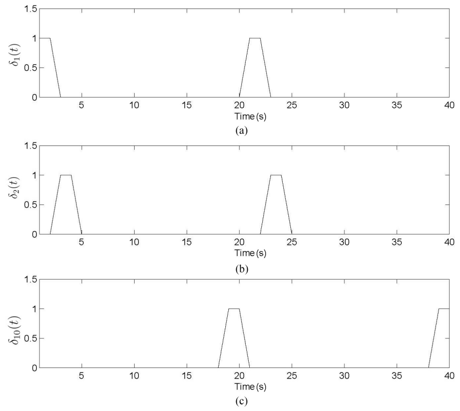


Fig. 24 System input modeling for the dry land farming application. The dynamics are described by (88), (89), and (90) and it is necessary to use several specialized forms of the Dirac delta function defined by (91), (92), and (93). In (a), (b) and (c), respectively, of this plot are the functions $\delta_1(t)$, $\delta_2(t)$, and $\delta_{10}(t)$ obtained by setting $i = 1, 2$, and 3, respectively, in (93) plotted for $N = 10$, and $T_p = 20$.

The center-pivot irrigator travels with a velocity that is defined by the period T_p and the model assumes a type of discrete in-space and in-time motion of the center pivot. In particular, it is assumed that it begins in sector 1, defined by the masking function $\delta_1(x, y)$, and stays there for T_p/N time units. The function $\delta_i(t)$ enables the sprayer when it is in sector 1, with zero input in all remaining sectors. After the time has elapsed in sector 1, the center pivot then moves to sector 2, stays there for T_p/N time units when the sprayer is enabled in this sector only, then moves on to sector 3 and so on. It is also assumed that an instantaneous change between sectors is possible.

Suppose that a constant flow rate F is the output of the irrigator. Then the input to the system in (90) can be written as

$$F(x, y, t) = F \sum_{i=1}^{N} \delta_i(x, y, t). \tag{94}$$

It is known that a multidimensional partial differential equation can be described as a discrete-time nD system by making finite difference approximations in time, and space [Rabenstein and Trautmann (2000)]. Let the region of interest be divided into a grid as shown in Fig. 22, with spacings Δx, Δy, and Δz and let the values of x, y, and z at each node be given as $x_i = i\Delta x$, $y_j = j\Delta y$, and $z_k = k\Delta z$, respectively. Also let time be quantized as $t_n = n\Delta T$, where ΔT is the sampling time. For simplicity, variables are subscripted with their indexes, where, for example, $c(x_i, y_j, z_k, t_n)$ is denoted by $c_{i,j,k,n}$. Then using the finite difference approximations [Crank (1975)], (88) and (89) can be rewritten as

$$\begin{aligned} v_{i,j,k,n+1} &= v_{i,j,k,n} + k_0(v_{i,j,k-1,n} - v_{i,j,k,n}),\\ c_{i,j,k,n+1} &= k_1 c_{i,j,k,n} + k_2 c_{i-1,j,k,n} + k_3 c_{i-1,j,k,n}\\ &\quad + k_4 c_{i,j+1,k,n} + k_5 c_{i,j-1,k,n}\\ &\quad + k_6(c_{i,j,k+1,n} - c_{i,j,k-1,n})\\ &\quad + k_7(v_{i,j,k,n}c_{i,j,k,n} - v_{i,j,k,n-1}c_{i,j,k,n-1}), \end{aligned} \tag{95}$$

with suitably defined constants k_1, which has an nD discrete systems structure.

The design objective now is to develop an ILC law for application to the center pivot system by treating each cycle of the irrigator as a pass. After the first cycle, whenever the irrigator is at a given point (x, y), the flow-rate applied to that point is adjusted according to the flow rate the last time the irrigator was at that point, combined with some measure of what has happened since that time. The measure chosen is a function of the measurement of the variable of interest at some fixed depth, corresponding to the idea of burying one sensor in each sector at that depth for every sample interval along the length of the center-pivot irrigator. It is also assumed that control of the flow rate for each sample interval along the irrigator's length is possible.

Assume that the input is given by (94) for the first cycle. Thereafter, however, the update equation employed is

$$F(x,y,t) = \sum_{i=1}^{N} \left[F(x,y,t-T_p) + \tau f(y_d - c(x,y,z_s,t'))\right] \delta_i(x,y,t), \tag{96}$$

where z_s is the sensor depth, τ is the learning gain, y_d is the desired setpoint, and $t' \in [t - T_p, t]$. The function $f(\cdot)$ is to be defined, and could be related to t'. For example, if the time lag to reach the depth z_s is known to be Δ, one possible choice is

$$f(y_d - c(x,y,z_s,t')) = y_d - c(x,y,z_s,t-T_p+\Delta)$$

and alternatives include

$$f(y_d - c(x,y,z_s,t')) = \max_{t' \in [t-T_p,t]} (y_d - c(x,y,z_s,t')). \tag{97}$$

The finite difference representation of the dynamics obtained by combining (95) and (97) gives the ILC law

$$F_{i,j,n} = \begin{cases} F \sum\limits_{l=1}^{N} \delta_l(x,y,n) & \text{for } t < T_p \\ \sum\limits_{l=1}^{N} \left[F_{i,z,n-T_p} + \tau \max\limits_{n' \in [n-T_p,n]} (y_d - c_{i,j,k_s,n'})\right] \delta_l(i,j,n) & \text{for } t \geq T_p, \end{cases} \tag{98}$$

Some illustrative simulation results for the case when $\Delta x = 1, \Delta y = 1, \Delta z = 0.5$, and $\Delta t = .05$ are now given, for a grid size of $30 \times 30 \times 7$, and the sensor at a depth of $z_s = 4$ units in the z-direction and $N = 36$ sectors with period $T_p = N = 36$. Hence the irrigator is assumed to stay in each sector for one time unit. The initial flow rate is $F = 10$, with $k_v = \frac{\Delta T}{\Delta x \Delta y \Delta z} = 0.1$, and the velocity-decay constant with respect to space taken as $\gamma = 2$. The learning gain is also chosen as $\tau = 2$ and the center pivot is placed on the grid at $(x_c, y_c) = (15, 15)$, with radius $R = 12$, and a diffusion rate of either 0.5 or 0.1. Finally, the control objective is defined using $y_d = 1$ in (97).

The simulation was run for 24 passes, or irrigation cycles, with the ILC turned off during the first 4 passes in order to allow the system to come to its natural open-loop steady-state, given a constant flow rate. Fig. 25 shows a snapshot of the process at the surface, $z = 7$ in the plot, and two different depths 3 and 6 corresponding to $z = 4$ and $z = 1$ respectively. These plots show the variable $c_{i,j,k,n}$, where the left-hand plot is for time $n = 24$, which is two-thirds the way through the first cycle, and the right-hand plot is for time $n = 48$, or one-third the way through the second pass. From these plots it follows that variable c slowly sinks down and also diffuses radially at each level.

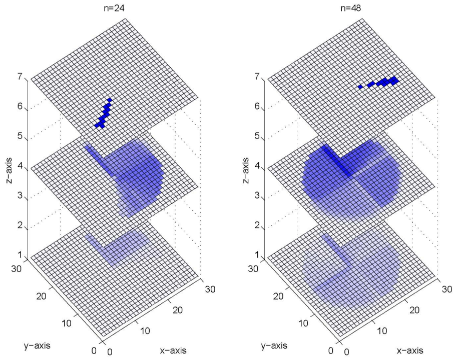


Fig. 25 ILC simulation results for the dry land farming example.

To further illustrate the process characteristics, the top plot in Fig. 26 shows the variables $c_{18,18,k,n}$ for all seven depths k, through the first four passes, and the bottom shows $c_{12,18,k,n}$, again for all seven depths k and through the first four passes. The distinction is that the top plot in Fig. 26 shows the evolution of the concentration at a point in a region of the field where the diffusion rate is higher with value 0.5, and the bottom shows the evolution of the concentration at a point in a region of the field where the diffusion rate is lower with value 0.1. It is seen that at the same flow rate the region with the lower diffusion rate has a higher level of c at a given depth than the one at the higher rate.

Given the results in Fig. 26, it follows that the ILC control law needs to enable a higher flow rate for the higher diffusion rate region in order to achieve uniform levels of c over all the regions. Fig. 27 confirms that the control law

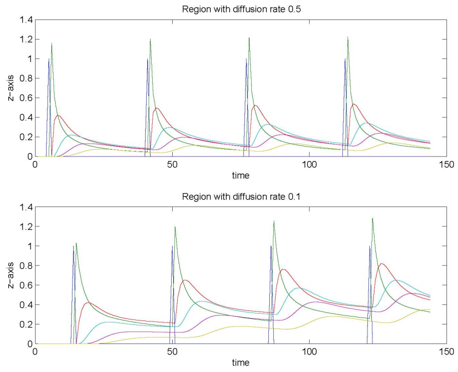


Fig. 26 The concentration at two different rates in the dry land farming example.

meets the objective. The upper two plots in the figure show the evolution of the flow rate over the course of 24 passes, four without ILC, and 20 with, for the variables $v_{18,18,0,n}$ on the left-hand side, and $v_{12,18,0,n}$ on the right-hand side. The input in the higher diffusion-rate region is larger, reflecting the fact that a higher input is required in this region because of the faster decay. One case where this situation can occur is for sandy soil as opposed to clay, where the lower plots in this same figure show the variables $c_{18,18,4,n}$ on the left-hand side, and $c_{12,18,4,n}$ on the right-hand side. Although the integral of the variable over a cycle is still larger for the lower diffusion rate region, the maximum is kept at one, as desired.

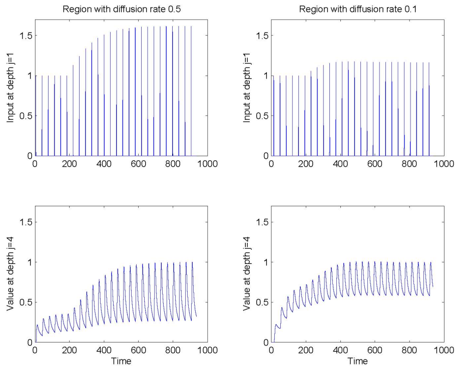


Fig. 27 Various flow rates computed by the ILC controller for the dry land farming example.

For comparison to the results in Fig. 25, Fig. 28 again shows a snapshot of the variable $c_{i,j,k,n}$ at the surface, $z = 7$ for the same depths considered in Fig. 25, for $n = 828$ on the left-hand side and $n = 864$ at the right after the

ILC law has converged. The plots are at the beginning of passes 23 and 24, and on comparing Figs. 25 and 28, the ILC law has affected the dynamics.

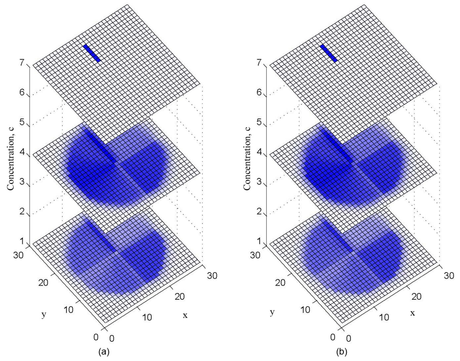


Fig. 28 Comparison of the dry land farming simulation results.

Finally, it is useful to consider a global cost function, where Fig. 29 plots the function

$$J = \sum_{x} \sum_{y} \left[\max_{t' \in [t - T_p, t]} \left(y_d - c\left(x, y, z_s, t'\right) \right) \right]^2 , \tag{99}$$

for passes after cycle number 7, and demonstrates that the ILC law reduces the error.

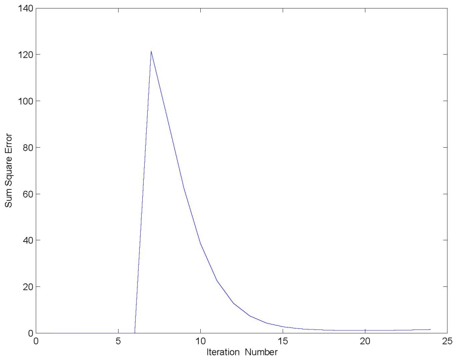


Fig. 29 Convergence of global cost function (99).

6 Conclusions and Further Research

The case studies considered in this paper demonstrate that nD control systems theory is making considerable impact in terms of translating the designs from analysis to in some cases experimental verification. In ILC, there has also been a transfer of laws from the industrial domain to next generation healthcare and, in particular, robotic-assisted upper-limb stroke rehabilitation with supporting clinical trials [Freeman et al. (2012),Meadmore et al. (2012)]. For this application it is the repeated performance of a finite duration task (with the input on the current pass computed by adding a corrective term that is directly influenced by the previous pass error to the previous pass input) that makes ILC particulary suitable, as the patients attempt to relearn lost functionality due the stroke, such as reaching out to a cup, by repeated attempts with assistive stimulation applied to the relevant muscles that are electrically stimulated. If the patient is improving with repeated attempts then his/her voluntary effort should be increasing and the level of assistive stimulation decreasing, and this feature was found in the clinical trials with an ILC law adjusting the stimulation before the start of each new attempt.

This healthcare application has not used the repetitive process setting for design but as the area develops and there is a continual move away from supplying the patient with a fixed trajectory and strapping the affected arm to the robot there will be a need to do control law design for pass-to-pass error convergence and along the pass performance and hence the repetitive process setting is directly applicable. These case studies have also highlighted the need for further research in control theory, especially for nonlinear dynamics. Moreover, onward development of the control law design algorithms, especially to reduce the possible conservativeness from the use of sufficient but not necessary conditions for stability along the pass have immediate applications. Another application area for, in particular, strong practical stability of linear repetitive processes is large-scale irrigation channels, see [Soltanian (2014)].

References

[Bose (1982)] Bose N. K. 1982. Applied Multidimensional Systems Theory, Van Nostrand Reinhold.

[Bose et al, (2003)] Bose, N. K., Buchberger, B., and Guiver, J. P. 2003. Multidimensional Systems Theory and Applications, Springer.

[Rogers and Owens (1992)] Rogers, E., and Owens, D. H. 1992. Stability Analysis for Linear Repetitive Processes, Springer LNCIS, vol. 175.

[Fornasini and Marchesini (1978)] Fornasini, E., and Marchesini, G. 1978. Doubly-indexed dynamical systems: State-space models and structural properties, Mathematical Systems Theory, 12(1), 59–72.

[Roesser (1975)] Roesser, R. P. 1975. A discrete state-space model for linear image processing, IEEE Transactions on Automatic Control, AC-20, 1–10.

[Rocha and Wood (2001)] Rocha, P. and Wood, J. 2001. Trajectory control and interconnection of 1D and nD systems, SIAM Journal on Control and Optimization, 40(1), 107–134.

[Edwards (1974)] Edwards, J. B. 1974. Stability problems in the control of multipass processes, IEE Proceedings on Control Theory and Applications, 124(11), 1425–143.

[Edwards and Greenberg (1977)] Edwards, J. B., and Greenberg, J. M. 1974. Longitudinal interaction in multipass processes, IEE Proceedings on Control Theory and Applications, 124(4), 385–392.

[Einicke et al. (2008)] Einicke, G., Ralston, Hargrave, J. J., Reid, D., and Hainsworth, D. 2008. Longwall mining automation: an application of minimum-variance smoothing, IEEE Control Systems Magazine, 28(6), 28–37.

[Rogers et al. (2007)] Rogers, E., Galkowski, K., and Owens, D. H. 2007. Control Systems Theory and Applications for Linear Repetitive Processes, Springer LNCIS, vol. 349.

[Kar and Singh (2001)] Kar, H., and Singh V. 2001. Stability analysis of 1D and 2D fixed point state-space filters using any combination of overflow and quantization nonlinearities, IEEE Transactions on Signal Procesing, 49(5), 1095–1105.

[Dewasurendra et al. (2006)] Dewasurendra, D. A., Liang, Y., and Bauer, P. H. 2006. Distributed multidimensional evidence filtering, Proceedings of the International Conference for Young Researchers (ICYR'06), Zielona Gora, Poland, CD Rom Proceedings.

1. Crossbow Inc., TelosB Mote Research Platform Datasheet [Online], Available: http://www.xbow.com/.

[Szewczyk et al. (2004)] Szewczyk, R., Osterweil, E., Polastre, J., Hamilton, A., Mainwaring, A., and Estrin, D. 2004. Habitat monitoring with sensor networks, Communications of the ACM, 47(6), 34–40.

[Shafer 1976] G. Shafer, G. 1976. A Mathematical Theory of Evidence, Princeton University Press, Princeton.

[Fagan and Halpem (1990)] Fagin, R., and Halpem, J. 1990. A new approach to updating beliefs, Proceedinsg of the Sixth Annual Conference on Uncertainty in Artificial Intelligence, 347–374.

[Dewasurendra et al. (2006)] Dewasurendra, D. A., Bauer, P. H., and Premaratne, K. 2006. Distributed evidence filtering in networked embedded systems, In: Network Embedded Sensing and Control, Eds. P. J. Antsaklis and P. Tabuada, Springer, 183–198.

[Arimoto et al. (1984)] Arimoto, S., Kawamura, S., and F. Miyazaki, F. 1984. Bettering operation of robots by learning. Journal of Robotic Systems, 2(1) 123–140.

[Bristow et al. (2006)] Bristow, D. A., Tharayil, M., Alleyne, A. G. 2006. A survey of iterative learning control: a learning based method for high performance tracking control. *IEEE Control Systems Magazine* 26(3), (2006), pp. 96–114.

[Ahn et al. (2007)] Ahn, H.-A., Chen, YQ., and Moore. K. L. 2007. Iterative learning control: brief survey and characterization, IEEE Transactions on Systems Man and Cybernetics Part C, 37(6), 1099–1121.

[Wallen et al. (2013)] Wallen, J., Gunnarsson, S., and Norrloff, M. 2013. Analysis of boundary effects in iterative learning control, International Journal of Control, 86(3), 410–415.

[Kurek and Zaremba (1993)] Kurek, J. E. and Zaremba, M. B. 1993. Iterative learning control synthesis based on 2D system theory, IEEE Transactions on Automatic Control, 38(1), 121–125.

[Rogers et al. (2002)] Rogers, E., Galkowski, K., Gramacki, A., Gramacki, J. and Owens, D. H. 2002. Stability and controllability of a class of 2D linear systems with dynamic boundary conditions, IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 49(2), 181-195.

[Haldowski et al. (2010)] Hladowski, L., Galkowski, K., Cai, Z., Rogers, E., Freeman, C. T., and Lewin, P. L. 2010. Experimentally supported 2D systems based iterative learning control law design for error convergence and performance, Control Engineering Practice, 18(4), 339–348.

[Ratcliffe et al. (2006)] Ratcliffe, J. D, Lewin, P. L., Rogers, E., Hatonen, J. J., and Owens, D. H. 2006. Norm-optimal iterative learning control applied to gantry robots for automation applications, IEEE Transactions on Robotics, 22(6), 1303-1307.

[Longman (2000)] Longman, R. W. 2000. Iterative learning control and repetitive control for engineering practice, International Journal of Control, 73(10), 930-954.

[Freeman et al. (2007)] Freeman, C. T., Lewin, P. L., and Rogers, E. 2007. Further results on the experimental evaluation of iterative learning control algorithms for non-minimum phase plants, International Journal of Control, 80(4), 569-582.

[Wallen et al. (2008)] Wallen, J., Norrlof, M., and Gunnarsson, S. 2008. Arm-side evaluation of ILC applied to a six-degrees-of-freedom industrial robot, Proceedings of 17th IFAC World Congress, 13450-13455.

[Chen et al. (1996)] Chen, Y., Xu, J-X, and Lee, T. H. 1996. Current iteration tracking error assisted iterative learning control of uncertain nonlinear discrete-time systems, Proceedings of the 35th IEEE Conference on Decision and Control, 3038-3043.

[Norloff and Gunnarsson (1999)] Norrlof, M. and Gunnarsson, S. 1999. A frequency domain analysis of a second order iterative learning control algorithm, Proceedings of the 38th Conference on Decision and Control, 1587–1592.

[Ratcliffe (2005)] Ratcliffe, J. D. 2005. Iterative learning control implemented on a multi-axis system, PhD Thesis, University of Southampton, UK

[Hladowski et al. (2012)] Hladowski, L., Galkowski, K., Cai, Z., Rogers, E., Freeman, C. T., and Lewin, P. L. 2012. Output information based iterative learning control law design with experimental verification, Journal of Dynamic Systems, Measurement and Control, 134(2), 021012/1-021012/10.

[Paszke et al. (2013)] Paszke, W., Rogers, E., Galkowski, K., and Cai, Z. 2013. Robust finite frequency range iterative learning control design and experimental verification, Control Engineering Practice, 21(10), 1310-1320.

[Cichy et al. (2014)] Cichy, B., Galkowski, K., and Rogers, E. 2014. 2D systems based robust iterative learning control using noncausal finite-time interval data, Systems and Control Letters, 64, 36–42.

[Dabkowski et al. (2009)] Dabkowski, P., Galkowski, K., Rogers, E., and Kummert, A. 2009. Strong practical stability and stabilization of discrete linear repetitive processes, Multidimensional Systems and Signal Processing, 20, 311–331.

[Du and Xie (2002)] Du, C., and Xie, L. 2002. H_∞ Control and Filtering of Two-Dimenional Systems, Springer LNCIS, vol. 278.

[Dabkowski et al. (2012)] Dabkowski, P., Galkowski, K., Bachelier, O. and Rogers, E. 2012. Control of discrete linear repetitive processes using strong practical stability and H_∞ disturbance attenuation, Systems and Control Letters, 61(12) 1138-1144.

[Mooney and Adam (2007)] Mooney, M. A., and Adam, D. 2007. Vibratory roller integrated measurement of earthwork compaction: an overview, Proceedings of the 7th International ASCE Symposium. Field Measurements in Geomechanics, CD Rom Proceedings.

[Kurek (1995)] Kurek, J. E. 1995. Stability of nonlinear parameter-varying digital 2-D systems, IEEE Transactions on Automatic Control, 40(8), 1428–1432.

[Yeganefar et al. (2013)] Yeganefar, Nima, Yeganefar, Nader, Ghamgui, M., and Moulay, M. 2013. Lyapunov theory for 2-D Nonlinear Roesser models: Application to asymptotic and exponential stability, IEEE Transactions on Automatic Control, 58(5), 1299-1304.

[Emelianova et al. (2014)] Emelianova, J., Pakshin, P., Galkowski, K., and Rogers, E. 2014. Stability of nonlinear 2D systems described by the continuous-time Roesser model, Automation and Remote Control, 75(5), 845-858.

[Emelianova et al. (2015)] Emelianova, J., Pakshin, P., Galkowski, K., and Rogers, E. 2015. Stability of nonlinear discrete repetitive processes with Markovian switching, Systems and Control Letters, 75, 108–116.

[Sammons et al. (2013)] Sammons, P.M., Bristow, D. A., and Landers, R. G. 2013. Height dependent laser metal deposition process modeling, ASME Journal for Manufacturing Science and Engineering, 135(5), pp. 054501-(1-7).

[Moore and Chen (2006)] Moore, K. L., and Chen, Y-Q. 2006. Iterative learning control approach to a diffusion control problem in an irrigation application, Proceedings of the International Conference on Mechatronics and Automation, Luoyang, P. R. China.

[Cantrell and Cosner (2003)] Cantrell, R. S., and Cosner, C. 2003. Spatial Ecology via Reaction-Diffusion Equations, Wiley.

[Rabenstein and Trautmann (2000)] Rabenstein, R., and Trautmann, L. 2000. Models for continuous and discrete multidimensional systems, Proceedings of the 2nd International Workshop on Multidimensional (nD) Systems, 47-54.

[Crank (1975)] Crank, J. 1975. The Mathematics of Diffusion, Oxford Science Publications, Oxford University Press.

[D'Andrea and Dullerud (2003)] D'Andrea, R., and Dullerud, G. 2003. Distributed control design for spatially interconnected systems, IEEE Tranactions on Automatic Control, 48, 1478-1495.

[Recht and D'Andrea (2004)] Recht, B., and DAndrea, R. 2004. Distributed control of systems over discrete groups, IEEE Transactions on Automatic Contol, 49(9), 14461452.

[Liu et al. (2014)] Qin Liu, Q., Gonzalez M. A., and Werner, H. 2014. Distributed control of spatially-interconnected parameter-invariant and LPV models for actuated beams, Proceedingss of 2014 American Control Conference, 3381-3386.

[Soltanian (2014)] Soltanian, L. 2014. Distributed distant-downstream controller design for large-scale irrigation channels, PhD thesis, University of Melbourne, Australia.

[Freeman et al. (2012)] Freeman, C. T., Rogers, E., Hughes, A. M., Burridge, J. H. and Meadmore, K. L. 2012. Iterative learning control in healthcare electrical stimulation and robotic-assisted upper limb stroke rehabilitation, IEEE Control Systems Magazine, 32(1), 18-43.

[Meadmore et al. (2012)] Meadmore, K. L., Hughes, A. M., Freeman, C. T., Cai, Z., Tong, D., Burridge, J. H., and Rogers, E. 2012. Functional electrical stimulation mediated by iterative learning control and 3D robotics reduces motor impairment in chronic stroke, Journal of Neuroengineering and Rehabilitation, 9(32), 1-11.