Multidimensional Systems and Signal Processing
https://doi.org/10.1007/s11045-021-00780-1

Modeling and iterative learning control of spatially distributed parameter systems with sensing and actuation over a selected area of the domain

Blazej Cichy[1] · Petr Augusta[2] · Krzysztof Galkowski[3] · Eric Rogers[4]

Received: 11 December 2020 / Revised: 24 March 2021 / Accepted: 16 April 2021

Abstract
This paper gives new contributions to the development of iterative learning control for distributed parameter systems, based on using finite difference schemes to construct a finite-dimensional approximate model of the dynamics for control law design. To form a basis for the new results, systems whose dynamics are described by a fourth-order partial differential equation are considered together with the associated accuracy and numerical stability checks. Some previous control law designs use only a spatial variable as the control input, which can be a serious obstacle to practical implementation since many actuators and sensors must be deployed. This paper's new design is based on spatially homogeneous sensing and excitation over a selected sub-area of the domain considered. Supporting numerical case studies are given to support the analysis.

Keyword Spatially distributed parameter systems and Iterative learning control

✉ Blazej Cichy
blcichy@gmail.com

Petr Augusta
augusta@utia.cas.cz

Krzysztof Galkowski
k.galkowski@issi.uz.zgora.pl

Eric Rogers
etar@ecs.soton.ac.uk

[1] Municipal Department of Transport, 65-713 Zielona Gora, Poland

[2] Institute of Information Theory and Automation, The Czech Academy of Sciences, Prague, Czech Republic

[3] Institute of Automation, Electronic and Electrical Engineering, University of Zielona Gora, Zielona Gora, Poland

[4] School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

Published online: 31 May 2021

1 Introduction

Iterative Learning Control (ILC) is a well-established area for systems that repeat the same finite duration task repeatedly. Each execution is known as a pass (or trial or iteration) in the literature, and the associated duration is termed the pass (or trial or iteration) length. Suppose, therefore, that a pass has been completed. Then all information generated over the complete pass length is available for use in updating the control input to the next pass. Hence ILC aims to improve performance from pass-to-pass, but it is also necessary to regulate the dynamics produced along the passes (if required).

The origins of ILC lie in robotics (Arimoto et al. 1984; Bristow et al. 2006; Ahn et al. 2007), but since this first work, many other application areas have arisen. Moreover, there is available a wide range of design methods for linear and (more recently) nonlinear dynamics. Recent application areas in engineering include various forms of additive manufacturing processes, e.g., Rafajlowicz et al. (2019), Lim et al. (2017)), nanopositioning, e.g., de Rozario et al. (2019), path following for center-articulated industrial vehicles (Dekker et al. 2019) and within healthcare, robotic-assisted upper limb stroke rehabilitation with clinical trials, e.g. Freeman et al. (2012) and ventricular assist devices, e.g., Ketelhut et al. (2019).

A substantial percentage of the available ILC literature is on the case when the dynamics of the application area can be modeled as a finite-dimensional system. There has also been some work on ILC design for distributed parameter systems, where two basic approaches are possible. The first of these is to use a semi-group or related setting, see, e.g., Huang et al. (2014). An alternative is to approximate the distributed parameter dynamics by a finite-dimensional model, which is the starting point for this paper.

Discretization of partial differential equations (PDE) describing systems with spatial and temporal dynamics is required to obtain discrete models that can form a basis for the design and digital implementation of control laws. A critical factor in this general approach is numerical stability, i.e., the discrete approximation must produce trajectories close to those produced by the PDE with identical stability properties. One group of methods which can be applied to the discretization of PDEs are based on a finite difference approximation, see, e.g. Strikwerda (1989).

Commonly used explicit discretization methods are conditionally numerically stable, i.e., the time discretization period is related to its spatial counterpart. This feature requires the use of dense time and spatial discretization grids. One way of avoiding the difficulties in analysis and design that would arise is to use the so-called singular methods, see Rabenstein and Steffen (2011), Rabenstein and Steffen (2009) and, in particular, the Crank Nicolson method (Crank and Nicolson 1947), which frequently produces an unconditionally stable discrete approximation to the dynamics of the original PDE. Discretization of PDEs describing systems or processes with one temporal and one spatial variable, such as the one- dimensional heat transfer equation results in models that are very similar to repetitive processes (Rogers et al. 2007), a distinct class of 2D systems.

Previous research on using the repetitive process setting to the control design for spatially distributed systems, described by PDEs, includes the results given in Cichy et al. (2008). As a particular example, a thin, flexible membrane is considered as model for vibrations resulting from a transverse external force's application. This paper builds on the preliminary results for dynamics modeling in Augusta et al. (2015). For this case, an ILC scheme, which is an essential novelty compared to previous research, is developed, analyzed, and supported by numerical examples. Also, the numerical analysis of the considered case is significantly extended.

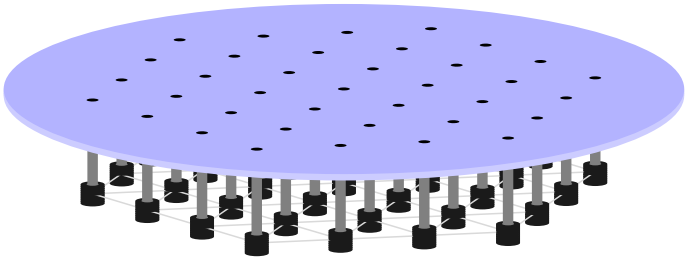


Fig. 1 A schematic of the form of actuation and sensing considered

2 Construction of the finite-dimensional approximate model

As a basis for the results in this paper, the following Lagrangian PDE is considered, where this model is standard in the study and application of membranes (or plates) and shells in mechanics and in this setting has the form, see, e.g. Timoshenko and Woinowski-Krieger (1959)

$$\frac{\partial^4 w(t,x,y)}{\partial x^4} + 2\frac{\partial^4 w(t,x,y)}{\partial x^2 \partial y^2} + \frac{\partial^4 w(t,x,y)}{\partial y^4} + \frac{\rho}{D}\frac{\partial^2 w(t,x,y)}{\partial t^2} = \frac{f(t,x,y)}{D}, \tag{1}$$

where

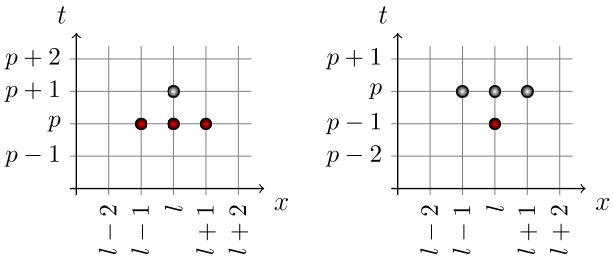
x, y are spatial variables [m],
t is time [s],
w is the lateral deflection in the z direction [m] perpendicular to x, and y,
ρ is the mass per unit area [$\mathrm{kg\,m^{-2}}$],
f is the transverse external force, with dimension of force per unit area [$\mathrm{N\,m^{-2}}$],
$\frac{\partial^2 w}{\partial t^2}$ is the acceleration in the z direction [$\mathrm{m\,s^{-2}}$],
$D = E h^3/(12(1-\nu^2))$,
ν is Poisson ratio,
h is the thickness of the membrane [m],
E is Young modulus [$\mathrm{N\,m^{-2}}$].

This PDE has two spatial indeterminates and one that is temporal. In this paper, both spatial indeterminates are finite. The analysis that follows applies to all linear systems with dynamics described by this form of PDE.

This paper considers the practically relevant case where the control action is discrete and based on an array of actuators and sensors. A schematic illustration of this form of sensing and actuating is given in Fig. 1. The particular case considered is where the actuators and sensors used are distributed over a circle of diameter a, but the actuators are only used in an area of diameter $d < a$. Hence the forcing function f can be modeled using a Heaviside function H as

$$f(t,x,y) = \left(1 - H(x^2 + y^2 - d^2)\right) q(t,x,y).$$

Since the function $1 - H(x^2 + y^2 - d^2) = 1$ within the region where the load is applied, the distributed system input is set to $f(t,x,y) = q(t,x,y)$ in the area of the membrane defined by the diameter d and to $f(t,x,y) = 0$ outside this area.



(a) Explicit scheme **(b)** Implicit scheme

Fig. 2 An example of explicit and implicit difference schemes showing the points needed for computation. The input mask is marked in red and output values currently being computed are marked in white

The discretization of (1) is based on finite difference methods, where, in general terms, the following steps must be implemented:

1. cover the region where a regular grid seeks a solution,
2. replace the derivative terms in the PDE by differences using only values at nodal points, i.e., approximate the derivatives.

These steps result in either an explicit or an implicit scheme. In general terms, an explicit difference scheme enables each value of the solution at sample time t_{p+1} to be independently calculated from the values known at time t_p, see Fig. 2a. However, such approximation's numerical stability critically depends on the quotient of the sampling period, δ_t secs, and the distance between nodes δ_x and δ_y. Conversely, discretization using an implicit difference scheme often results in an unconditionally numerically stable approximation of system dynamics. Still, more than one value of the solution has to be computed at the new time t_p, see Fig. 2b. Moreover, all values at t_p must be obtained simultaneously, which, in turn, requires the solution of a system of equations at each time step. In this paper, a difference scheme of this type is used and for a detailed treatment of these schemes and the differences between them, see, e.g., Smith (1985) In the spatial variables, we use the grid of Figs. 3, 4.

Derivatives arising in (1) are replaced by differences, and using the notation specified, this step results in

$$\begin{aligned}
\frac{\partial^4 w}{\partial x^4} \approx \frac{1}{4\delta_x^4}\big(&w_{p+2,l+2,m} - 2w_{p+2,l+1,m+1} \\
&- 2w_{p+2,l+1,m-1} + 6w_{p+2,l,m} - 2w_{p+2,l-1,m+1} \\
&- 2w_{p+2,l-1,m+1} + w_{p+2,l-2,m} + 2w_{p+1,l+2,m} \\
&- 4w_{p+1,l+1,m+1} - 4w_{p+1,l+1,m-1} + 12w_{p+1,l,m} \\
&- 4w_{p+1,l-1,m+1} - 4w_{p+1,l-1,m+1} + 2w_{p+1,l-2,m} \\
&+ w_{p,l+2,m} - 2w_{p,l+1,m+1} - 2w_{p,l+1,m-1} + 6w_{p,l,m} \\
&- 2w_{p,l-1,m+1} - 2w_{p,l-1,m+1} + w_{p,l-2,m}\big), \qquad (2) \\
\frac{\partial^4 w}{\partial y^4} \approx \frac{1}{4\delta_y^4}\big(&w_{p+2,l,m+2} - 2w_{p+2,l+1,m+1} \\
&- 2w_{p+2,l-1,m+1} + 6w_{p+2,l,m} - 2w_{p+2,l+1,m-1} \\
&- 2w_{p+2,l+1,m-1} + w_{p+2,l,m-2} + 2w_{p+1,l,m+2}
\end{aligned}$$

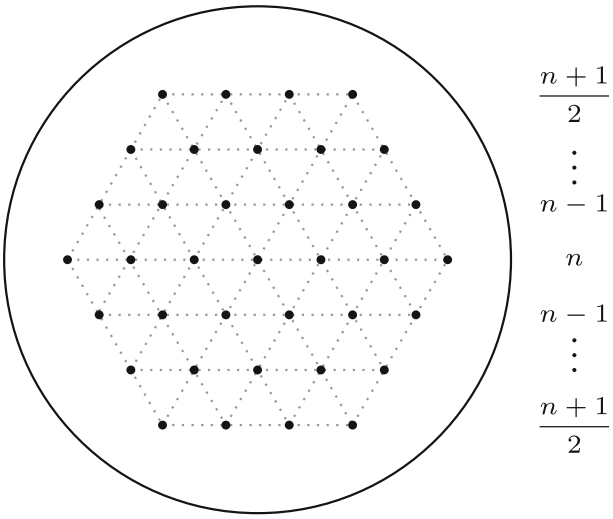


Fig. 3 An example of the triangular grid for $n = 7$, the number of nodal points in each row is written on the right-hand side

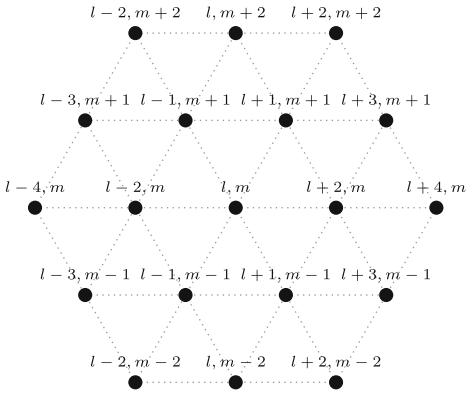


Fig. 4 A more detailed view of the triangular grid for $n = 7$

$$
\begin{aligned}
&- 4w_{p+1,l+1,m+1} - 4w_{p+1,l-1,m+1} + 12w_{p+1,l,m} \\
&- 4w_{p+1,l+1,m-1} - 4w_{p+1,l+1,m-1} + 2w_{p+1,l,m-2} \\
&+ w_{p,l,m+2} - 2w_{p,l+1,m+1} - 2w_{p,l-1,m+1} + 6w_{p,l,m} \\
&- 2w_{p,l+1,m-1} - 2w_{p,l+1,m-1} + w_{p,l,m-2}\big), \qquad (3)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial^4 w}{\partial x^2 \partial y^2} \approx \frac{1}{4\delta_x^2 \delta_y^2}\big(&- w_{p+2,l+1,m+1} - w_{p+2,l+1,m-1} \\
&+ 4w_{p+2,l,m} - w_{p+2,l-1,m+1} - w_{p+2,l-1,m-1} \\
&- 2w_{p+1,l+1,m+1} - 2w_{p+1,l+1,m-1} + 8w_{p+2,l,m} \\
&- 2w_{p+2,l-1,m+1} - 2w_{p+2,l-1,m-1} - w_{p,l+1,m+1} \\
&- w_{p,l+1,m-1} + 4w_{p,l,m} - w_{p,l-1,m+1} - w_{p,l-1,m-1}\big), \qquad (4)
\end{aligned}
$$

$$\frac{\partial^2 w}{\partial t^2} \approx \frac{1}{\delta_t^2}\left(w_{p+2,l,m} - 2w_{p+1,l,m} + w_{p,l,m}\right). \tag{5}$$

Substituting (2)–(5) into (1) followed by routine manipulations gives the following partial recurrence equation as a discrete approximate model of the original PDE dynamics described by (1)

$$\begin{aligned}
&\left(\frac{1}{4\delta_x^4}\right)\left(w_{p+2,l+2,m} + w_{p+2,l-2,m}\right)\\
&+\left(-\frac{1}{2\delta_x^4} - \frac{1}{2\delta_y^4} - \frac{1}{2\delta_x^2\delta_y^2}\right)\left(w_{p+2,l+1,m+1}\right.\\
&\left.+w_{p+2,l+1,m-1} + w_{p+2,l-1,m+1} + w_{p+2,l-1,m-1}\right)\\
&+\left(\frac{1}{4\delta_y^4}\right)\left(w_{p+2,l,m+2} + w_{p+2,l,m-2}\right)\\
&+\left(\frac{3}{2\delta_x^4} + \frac{3}{2\delta_y^4} + \frac{2}{\delta_x^2\delta_y^2} + \frac{\rho}{D\delta_t^2}\right)w_{p+2,l,m}\\
&+\left(\frac{1}{2\delta_x^4}\right)\left(w_{p+1,l+2,m} + w_{p+1,l-2,m}\right)\\
&+\left(-\frac{1}{\delta_x^4} - \frac{1}{\delta_y^4} - \frac{1}{\delta_x^2\delta_y^2}\right)\left(w_{p+1,l+1,m+1}\right.\\
&\left.+w_{p+1,l+1,m-1} + w_{p+1,l-1,m+1} + w_{p+1,l-1,m-1}\right)\\
&+\left(\frac{1}{2\delta_y^4}\right)\left(w_{p+1,l,m+2} + w_{p+1,l,m-2}\right)\\
&+\left(\frac{3}{\delta_x^4} + \frac{3}{\delta_y^4} + \frac{4}{\delta_x^2\delta_y^2} - \frac{2\rho}{D\delta_t^2}\right)w_{p+1,l,m}\\
&+\left(\frac{1}{4\delta_x^4}\right)\left(w_{p,l+2,m} + w_{p,l-2,m}\right)\\
&+\left(-\frac{1}{2\delta_x^4} - \frac{1}{2\delta_y^4} - \frac{1}{2\delta_x^2\delta_y^2}\right)\left(w_{p,l+1,m+1}\right.\\
&\left.+w_{p,l+1,m-1} + w_{p,l-1,m+1} + w_{p,l-1,m-1}\right)\\
&+\left(\frac{1}{4\delta_y^4}\right)\left(w_{p,l,m+2} + w_{p,l,m-2}\right)\\
&+\left(\frac{3}{2\delta_x^4} + \frac{3}{2\delta_y^4} + \frac{2}{\delta_x^2\delta_y^2} + \frac{\rho}{D\delta_t^2}\right)w_{p,l,m} = \frac{1}{D}q_{p,l,m}.
\end{aligned} \tag{6}$$

The next stage is to investigate this approximation's numerical stability properties, for which von Neumann stability analysis is used. Von Neumann analysis is a standard mathematical method for checking the stability of finite difference scheme approximations to a PDE, see, e.g., Strikwerda (1989). This analysis gives a relation between the sampling period δ_t and the distance between nodal points δ_x, δ_y, respectively, required for stability. Since an implicit difference scheme is used, i.e., a particular case of unconditionally stable schemes, von Neumann analysis should result in always satisfied conditions.

Consider (6) with a zero right-hand side and substitute $g^p e^{jl\theta_1} e^{jm\theta_2}$ for $w_{p,l,m}$, where g is termed the amplification factor and θ_1 and θ_2 denote the spatial frequencies. Then by von Neumann analysis, (6) is stable if and only if $|g| \leq 1$ for all values of θ_1 and θ_2. Also these substitutions result in a polynomial in g of the form

$$A_2 g^2 + A_1 g + A_0, \tag{7}$$

with non-constant, uniquely defined coefficients $A_0(\theta_1, \theta_2)$, $A_1(\theta_1, \theta_2)$, $A_2(\theta_1, \theta_2)$ that follow immediately from (6), see Rabenstein and Steffen (2011) for details. Moreover, it follows from algebraic manipulations, see also Rabenstein and Steffen (2011), that $|g| \leq 1$ if and only if

$$A_2 + A_1 + A_0 \geq 0, \tag{8}$$
$$A_2 - A_1 + A_0 \geq 0, \tag{9}$$
$$A_2 - A_0 \geq 0, \tag{10}$$

for all values of θ_1 and θ_2, where, for the case considered, (8) can be expressed as

$$\begin{aligned} A_2 + A_1 + A_0 = {} & \frac{8}{\delta_x^2 \delta_y^2} [1 - \cos\theta_1 \cos\theta_2] \\ & + \frac{1}{\delta_x^4} \left[6 + 2\cos(\theta_1)^2 - 2\sin(\theta_1)^2 - 8\cos\theta_1 \cos\theta_2\right] \\ & + \frac{1}{\delta_y^4} \left[6 + 2\cos(\theta_2)^2 - 2\sin(\theta_2)^2 - 8\cos\theta_1 \cos\theta_2\right] \geq 0. \end{aligned}$$

Using Euler's formula and routine manipulations gives

$$\begin{aligned} & \frac{8}{\delta_x^2 \delta_y^2} [1 - \cos\theta_1 \cos\theta_2] + \frac{4}{\delta_x^4} \left[(\cos\theta_1 - \cos\theta_2)^2 + \sin(\theta_2)^2\right] \\ & + \frac{4}{\delta_y^4} \left[(\cos\theta_1 - \cos\theta_2)^2 + \sin(\theta_1)^2\right] \geq 0, \end{aligned} \tag{11}$$

which is always satisfied and hence (8) is also always satisfied. Inequalities (9) and (10) now take the form

$$\begin{aligned} A_2 - A_1 + A_0 = 4\frac{\rho}{D\delta_t^2} \quad & \geq 0 \\ A_2 - A_0 = 0 \quad & \geq 0, \end{aligned} \tag{12}$$

respectively, and also are always satisfied. Hence (8)–(10) always hold and therefore (6) is a stable approximation for arbitrary values of the discretization parameters δ_t, δ_x and δ_y.

At this stage, (6) can be written in the equivalent form

$$A W_{p+2} + B W_{p+1} + A W_p = C Q_p, \tag{13}$$

where the nodes (l, m) are ordered row by row as illustrated in Fig. 5 with $p \geq 0$. In (13) W_p and Q_p are, respectively, the vectors of the deflections and the external forces at the nodal points and given by

$$W_p = \begin{bmatrix} w_{p,0,0} \\ \vdots \\ w_{p,l,m} \end{bmatrix}_{l+m=N-1}, \quad Q_p = \begin{bmatrix} q_{p,0,0} \\ \vdots \\ q_{p,l,m} \end{bmatrix}_{l+m=N-1},$$

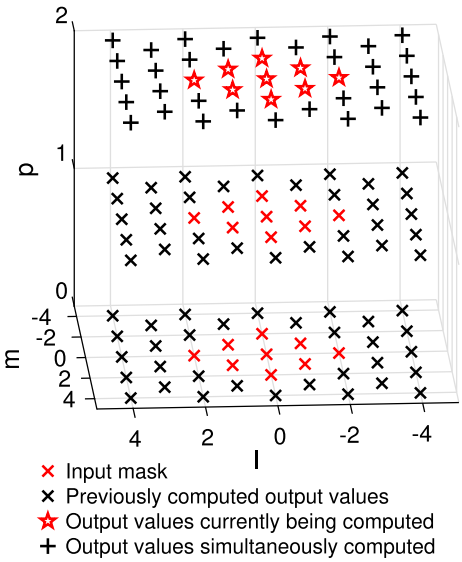


Fig. 5 Computation mask associated with the partial recurrence equation (6)

and A, B and C are Toeplitz matrices constructed from the coefficients of (6). Since the deflections at all boundary nodal points and those external to the membrane are always zero, they are not included in W_p; similarly, for Q_p. The entries in W_p are the variables to be controlled, and those in Q_p the control signals to be designed.

In (13), the matrices A, B and C have the dimension $N \times N$ and A can be written as (51) given in the appendix, where $A_1^{(X)}$ for a positive integer X is an $X \times X$ matrix and O is the zero matrix with compatible dimensions. Also the following matrices are used

$$
\begin{aligned}
S &= \frac{1}{\delta_x \delta_y^3} + \frac{1}{\delta_y \delta_x^3} + \frac{1}{2\delta_x^4} + \frac{1}{2\delta_y^4} + \frac{\rho}{D\delta_t^2}, \\
Q &= \frac{\delta_x \delta_y - \delta_y^2 - 2\delta_x^2}{4\delta_x^5 \delta_y + 4\delta_x^4 \delta_y^2}, \quad R = \frac{\delta_x \delta_y - 2\delta_y^2 - \delta_x^2}{4\delta_x^2 \delta_y^4 + 4\delta_x \delta_y^5}, \\
P &= \frac{1}{2\delta_x^2 \delta_y^2} - \frac{1}{2\delta_x \delta_y^3} - \frac{1}{2\delta_x^3 \delta_y}.
\end{aligned} \tag{14}
$$

The matrix B has the same structure as A with

$$
\begin{aligned}
S &= \frac{2}{\delta_x \delta_y^3} + \frac{2}{\delta_y \delta_x^3} + \frac{1}{\delta_x^4} + \frac{1}{\delta_y^4} - 2\frac{\rho}{D\delta_t^2}, \\
Q &= \frac{\delta_x \delta_y - \delta_y^2 - 2\delta_x^2}{2\delta_x^5 \delta_y + 2\delta_x^4 \delta_y^2}, \quad R = \frac{\delta_x \delta_y - 2\delta_y^2 - \delta_x^2}{2\delta_x^2 \delta_y^4 + 2\delta_x \delta_y^5}, \\
P &= \frac{1}{\delta_x^2 \delta_y^2} - \frac{1}{\delta_x \delta_y^3} - \frac{1}{\delta_x^3 \delta_y}.
\end{aligned} \tag{15}
$$

and

$$A_4 = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & R \end{bmatrix}, \; C = \begin{bmatrix} 1/D & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1/D \end{bmatrix}. \tag{16}$$

The node distances in the x and y directions are given by

$$\delta_x = \frac{a}{n+1} \text{ and } \delta_y = \frac{\sqrt{3}}{2}\delta_x.$$

An alternative method to that given above is to solve (13) directly by a multigrid algorithm, see Rabenstein and Steffen (2012). Then, (13) can be written as

$$W_{p+2} = -A^{-1}BW_{p+1} - W_p + A^{-1}CQ_p, \tag{17}$$

provided that the matrix A is invertible.

3 Problem formulation

3.1 Background on repetitive processes

The simplest form of repetitive processes is those where at any instant along the current pass the contributions from the previous pass are only from the same instant. Let the nonnegative integer k denote the pass number and $0 \leq p \leq N-1$ denote a sample instant along a trial formed of N samples. Then the state equation of such an example on pass k is of the form

$$x_k(p+1) = \tilde{A}x_k(p) + \tilde{B}u_{k+1}(p) + \tilde{B}_0 y_{k-1}(p)$$

where, with compatible dimensions, $x_k(p)$ is the state vector, $u_k(p)$ is the input vector and $y_{k-1}(p)$ is the previous pass output vector at sample p, termed the pass profile. The algebraic equation for computing the current pass profile vector is of the form

$$y_k(p) = \tilde{C}x_k(p) + \tilde{D}u_k(p) + \tilde{D}_0 y_{k-1}(p)$$

A further generalization of wave repetitive processes is possible to the case where a 'window' of previous instances along the previous pass contributes to a 'window' of instances along the current pass, see Fig. 6. It is this form of repetitive process that provides the setting for ILC design in the next section.

The stability theory for linear (and nonlinear) repetitive processes (Rogers et al. 2007) is of bounded-input bounded-output form. In particular, given the unique control problem, this theory requires that a bounded initial pass profile (y_0) produces a bounded sequence of pass profiles ($\{y_k\}_k$), where the meaning of 'bounded' is in the sense of the norm on the underlying function space. Moreover, this property can be required over the finite and fixed pass length or uniformly, i.e., for all possible values of the pass length where this property can be examined mathematically for discrete processes by considering the case when $N \to \infty$.

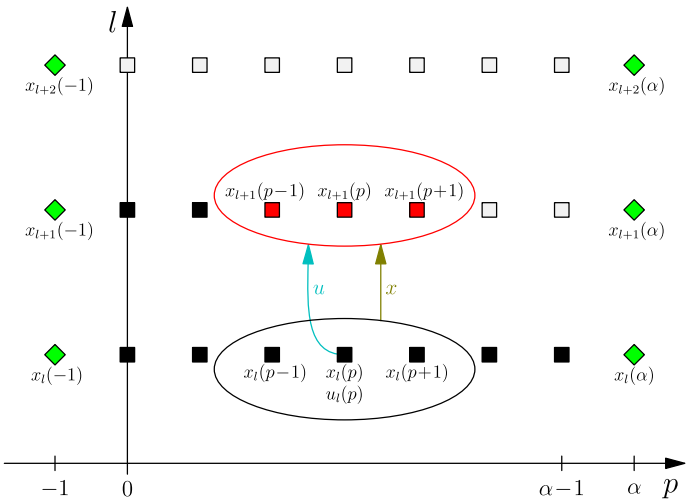


Fig. 6 An alternative updating structure of a wave repetitive process

3.2 ILC problem formulation

The control approach adopted in the current application is to apply control through Q_p of (13) over a finite duration. A stoppage occurs, the control action is updated and then applied over the same finite duration, i.e., pass length. The aim is to force the membrane to deflect to a specified profile by repeated application of control action updated between two successive passes using an ILC law. Next, the design problem is formulated.

Note 1 From this point onwards, I and O, respectively, denote the compatibly dimensioned identity and null matrices.

Rewrite (17) in the form

$$W_{p+2} = \hat{A}_1 W_{p+1} + \hat{A}_2 W_p + \hat{B} Q_p, \tag{18}$$

where

$$\hat{A}_1 = -A^{-1}B, \qquad \hat{A}_2 = -I, \qquad \hat{B} = A^{-1}C. \tag{19}$$

This is a second-order difference equation and to transform it to first-order, introduce

$$\mathbf{W}_p = \begin{bmatrix} W_{p+1} \\ W_p \end{bmatrix}, \quad \mathbf{Q}_p = Q_p, \tag{20}$$

to obtain the state equation

$$\mathbf{W}_{p+1} = \mathbf{A}\mathbf{W}_p + \mathbf{B}\mathbf{Q}_p, \tag{21}$$

with $p \geq 0$ and

$$\mathbf{A} = \begin{bmatrix} \hat{A}_1 & \hat{A}_2 \\ I & O \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \hat{B} \\ O \end{bmatrix}. \tag{22}$$

To formulate the ILC design problem, a positive integer variable k denoting the pass-to-pass update is introduced. Then (21) can be written as

$$\mathbf{W}_{p+1}(k) = \mathbf{A}\mathbf{W}_p(k) + \mathbf{B}\mathbf{Q}_p(k). \tag{23}$$

Introduce the output equation

$$\mathbf{Y}_p(k) = \mathbf{C}\mathbf{W}_p(k) = W_{p+1}(k), \tag{24}$$

with

$$\mathbf{C} = \begin{bmatrix} I & O \end{bmatrix}. \tag{25}$$

Also define the tracking error $\mathbf{E}_p(k)$ as

$$\mathbf{E}_p(k) = \mathbf{Y}_p^* - \mathbf{Y}_p(k) = W_{p+1}^* - W_{p+1}(k), \tag{26}$$

where $\mathbf{Y}_p^*$ denotes a spatial/temporal reference signal and hence W_{p+1}^* denotes deflections in spatial/temporal reference signal.

Introduce the state and control increments as

$$\Theta_{p+1}(k+1) = \mathbf{W}_p(k+1) - \mathbf{W}_p(k), \tag{27}$$

$$\Delta\mathbf{Q}_p(k+1) = \mathbf{Q}_p(k+1) - \mathbf{Q}_p(p) \tag{28}$$

and apply the ILC law

$$\Delta\mathbf{Q}_p(k+1) = \mathbf{K}_1\Theta_{p+1}(k+1) + \mathbf{K}_2\mathbf{E}_{p+1}(k) \tag{29}$$

to obtain the following ILC dynamics written in the form of a discrete linear repetitive process (Rogers et al. 2007)

$$\begin{aligned} \Theta_{p+1}(k+1) &= \bar{A}\Theta_p(k+1) + \bar{B}\mathbf{E}_p(k), \\ \mathbf{E}_p(k+1) &= \bar{C}\Theta_p(k+1) + \bar{D}\mathbf{E}_p(k), \end{aligned} \tag{30}$$

where

$$\begin{aligned} \bar{A} &= \mathbf{A} + \mathbf{B}\mathbf{K}_1, \quad & \bar{B} &= \mathbf{B}\mathbf{K}_2, \\ \bar{C} &= -\mathbf{C}\bar{A}, & \bar{D} &= I - \mathbf{C}\bar{B}. \end{aligned} \tag{31}$$

In the repetitive process setting, the pass error ($\mathbf{E}_p(k)$)is the output, and $\Theta_p(k+1)$ is the current pass state vector (there is no pass input term as this is the model of the controlled dynamics). Hence the stability theory for linear repetitive processes can be applied.

Previous research on ILC design using repetitive process stability theory has, uniquely for 2D systems based ILC analysis, been followed through to experimental validation, see, e.g., Hładowski et al. (2010); Paszke et al. (2016). This previous work was for systems that are finite-dimensional, instead of using a finite-dimensional model as an approximation to PDE dynamics.

4 ILC design

The aim is to design the ILC law using the strong form of the repetitive process stability theory (Rogers et al. 2007) applied to (30). This design will guarantee monotonic pass-to-pass error convergence and also regulate the dynamics produced along each trial. One means

of applying this theory is by use of the Lyapunov function

$$\begin{aligned} V_p(k) = \Theta_p(k+1)^T P_1 \Theta_p(k+1) \\ + \mathbf{E}_p(k)^T P_2 \mathbf{E}_p(k) \end{aligned} \tag{32}$$

where P_1 and P_2 are compatibly dimensioned symmetric positive definite matrices, denoted from this point onwards as $P_1 \succ 0$ and $P_2 \succ 0$. This function can be viewed in physical terms where the quadratic form in $\Theta_p(k+1)$ measures the energy along a pass and that in $\mathbf{E}_p(k)$ the 'energy' from pass-to-pass.

Introduce the associated increment of $V_p(k)$ as

$$\begin{aligned} \Delta V_p(k) = \Theta_{p+1}(k+1)^T P_1 \Theta_{p+1}(k+1) \\ -\Theta_p(k+1)^T P_1 \Theta_p(k+1) \\ +\mathbf{E}_p(k+1)^T P_2 \mathbf{E}_p(k+1) - \mathbf{E}_p(k)^T P_2 \mathbf{E}_p(k). \end{aligned} \tag{33}$$

then the stability theory for discrete linear repetitive processes (Rogers et al. 2007) gives that pass-to-pass error convergence occurs when

$$\Delta V_p(k) < 0, \quad \forall k, p > 0, \tag{34}$$

or

$$\begin{bmatrix} \bar{A}^T P_1 \bar{A} - P_1 + \bar{C}^T P_2 \bar{C} & \bar{A}^T P_1 \bar{B} + \bar{C}^T P_2 \bar{D} \\ \bar{B}^T P_1 \bar{A} + \bar{D}^T P_2 \bar{C} & \bar{B}^T P_1 \bar{B} + \bar{D}^T P_2 \bar{D} - P_2 \end{bmatrix} \prec 0, \tag{35}$$

where $\prec$ denotes the symmetric negative definite property. Also introduce

$$\bar{\mathcal{A}} = \begin{bmatrix} \bar{A} & \bar{B} \\ \bar{C} & \bar{D} \end{bmatrix}, \quad \bar{\mathcal{P}} = \text{diag}\{P_1, P_2\} \tag{36}$$

and then (35) can be rewritten as

$$\bar{\mathcal{A}}^T \bar{\mathcal{P}} \bar{\mathcal{A}} - \bar{\mathcal{P}} \prec 0. \tag{37}$$

Moreover, rewriting $\bar{\mathcal{A}}$ as

$$\bar{\mathcal{A}} = \begin{bmatrix} \mathbf{A} + \mathbf{B}\mathbf{K}_1 & \mathbf{B}\mathbf{K}_2 \\ -\mathbf{C}\mathbf{A} - \mathbf{C}\mathbf{B}\mathbf{K}_1 & I - \mathbf{C}\mathbf{B}\mathbf{K}_2 \end{bmatrix} \tag{38}$$

and introducing

$$\begin{aligned} \bar{\mathbb{A}} = \begin{bmatrix} \mathbf{A} & O \\ O & I \end{bmatrix}, \quad \bar{\mathbb{B}} = \begin{bmatrix} \mathbf{B} & \mathbf{B} \\ O & O \end{bmatrix}, \quad \bar{\mathbb{C}} = \begin{bmatrix} I & O \\ -\mathbf{C} & I \end{bmatrix}, \\ \mathbb{K} = \text{diag}\{\mathbf{K}_1, \mathbf{K}_2\}, \quad \mathbb{A} = \bar{\mathbb{C}}\bar{\mathbb{A}}, \quad \mathbb{B} = \bar{\mathbb{C}}\bar{\mathbb{B}}. \end{aligned} \tag{39}$$

Moreover, (38) is equivalent to

$$\bar{\mathcal{A}} = \mathbb{A} + \mathbb{B}\mathbb{K} \tag{40}$$

and then (37) can be written as

$$(\mathbb{A} + \mathbb{B}\mathbb{K})^T \bar{\mathcal{P}} (\mathbb{A} + \mathbb{B}\mathbb{K}) - \bar{\mathcal{P}} \prec 0. \tag{41}$$

Application of known results in Rogers et al. (2007) gives that the ILC dynamics written as the discrete linear repetitive process (30)is stable along the pass and hence the monotonic

Table 1 Plate parameters

Parameter	Value	Unit
h-thickness	0.01	m
ρ-area density	25	$\mathrm{kg\,m^{-2}}$
E-Young's modulus	70×10^9	$\mathrm{N\,m^{-2}}$
ν-Poisson's ratio	0.22	–

trial-to-trial error convergence occurs if there exists $\tilde{\mathcal{P}} \succ 0$, where $\tilde{\mathcal{P}} = \mathrm{diag}\left\{\tilde{\mathcal{P}}_1, \tilde{\mathcal{P}}_2\right\}$ and $\tilde{\mathcal{N}} = \mathrm{diag}\left\{\tilde{\mathcal{N}}_1, \tilde{\mathcal{N}}_2\right\}$ such that the Linear Matrix Inequality (LMI)

$$\begin{bmatrix} -\tilde{\mathcal{P}} & \tilde{\mathcal{P}}\mathbb{A}^T + \tilde{\mathcal{N}}^T\mathbb{B}^T \\ \mathbb{A}\tilde{\mathcal{P}} + \mathbb{B}\tilde{\mathcal{N}} & -\tilde{\mathcal{P}} \end{bmatrix} \prec 0 \tag{42}$$

is feasible. Moreover, if this LMI is feasible, the control law matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ in (29) are given by

$$\begin{aligned} \mathbb{K} &= \tilde{\mathcal{N}}\tilde{\mathcal{P}}^{-1} = \mathrm{diag}\left\{\mathbf{K}_1, \mathbf{K}_2\right\} \\ &= \mathrm{diag}\left\{\tilde{\mathcal{N}}_1\tilde{\mathcal{P}}_1^{-1}, \tilde{\mathcal{N}}_2\tilde{\mathcal{P}}_2^{-1}\right\}. \end{aligned} \tag{43}$$

5 Numerical case studies

The numerical value of the parameters used is given in Table 1 and discretization is by applying the regular hexagonal grid of Fig. 4.

The first example addresses the key issues in obtaining the approximate finite-dimensional model to be used for control law design.

Example 1

It is assumed that the edges of the membrane are clamped and hence both the deflection at the edge and its derivative are zero. This leads to the boundary conditions

$$\begin{aligned} &w(t, x, y)\Big|_{x,y\in\mathcal{B}} = 0, \\ &\frac{\partial w(t, x, y)}{\partial x}\Big|_{x,y\in\mathcal{B}} = 0, \quad \frac{\partial w(t, x, y)}{\partial y}\Big|_{x,y\in\mathcal{B}} = 0, \end{aligned} \tag{44}$$

where $\mathcal{B}$ denotes the boundary of the region where a solution is to be found.

Applying the discretization scheme, the conditions of (44) become

$$\begin{aligned} &w_{k,l,m} = 0, \\ &w_{p,l,m} - w_{p,l-2,m} = 0, \quad w_{p,l,m} - w_{p,l,m-2} = 0 \end{aligned} \tag{45}$$

at the boundary nodal points. It follows from (45) that $w_{p,l,m} = 0$, $w_{p,l-2,m} = 0$, $w_{p,l,m-2} = 0$ for all p at a boundary nodal point. The deflection is assumed to be zero at all boundary nodal points and nodal points outside the plate. However, in the simulations, the derivatives (44) are approximated using points on the boundary and outside the region where the solution

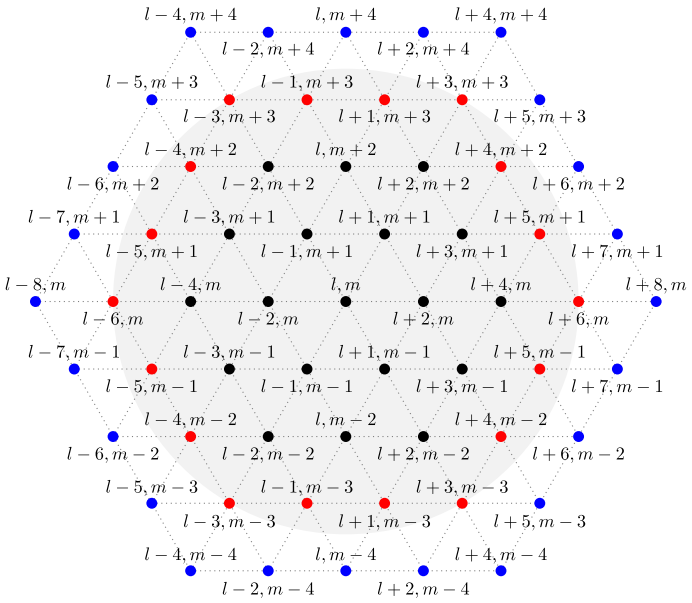


Fig. 7 The circular membrane covered by the hexagonal grid with $n = 5$. The deflection at the red points are zero because of the first line of (45) and those at the blue points are also zero by (46)

is sought. Derivatives in the x-direction are approximated by, see Fig. 7,

$$\begin{aligned} w_{p,l-6,m} - w_{p,l-8,m} &= 0 \text{ at } w_{p,l-6,m}, \\ w_{p,l-5,m+1} - w_{p,l-7,m+1} &= 0 \text{ at } w_{p,l-5,m+1}, \\ w_{p,l+4,m+2} - w_{p,l+6,m+2} &= 0 \text{ at } w_{p,l+4,m+2}, \\ w_{p,l-4,m-2} - w_{p,l-6,m-2} &= 0 \text{ at } w_{p,l-4,m-2}, \\ w_{p,l+3,m-3} - w_{p,l+5,m-3} &= 0 \text{ at } w_{p,l+3,m-3}, \end{aligned} \tag{46}$$

etc., with similar approximations for the derivatives in the y-direction. It follows from (46) that the deflection is still zero at all boundary nodal points and all external nodal points denoted, respectively, by the red and blue colors, in Fig. 7. Hence the deflections at these points do not need to be computed.

The initial conditions of (17) are W_0 and W_1 and are given in Fig. 8. Note also that alternative way to undertake the simulations is to solve (13) directly by a multigrid algorithm, see Rabenstein and Steffen (2012).

The matrix representation (17) is used to compute the response for various values of the sampling time with the node distances in the x, y directions, respectively, given by

$$\delta_x = \frac{a}{n+1}, \qquad \delta_y = \frac{\sqrt{3}}{2}\delta_x.$$

As a first case, consider the choice of $a = 1$ and $n = 19$, which results in $\delta_x = 0.05$ m. The deflection in the middle of the membrane for the sampling periods $\delta_t = 0.001$ secs and $\delta_t = 0.01$ secs, respectively, are shown in Figs. 9 and 10.

From these plots, it is seen that the deflections for both sampling periods are equal as required. Finally, the deflection of the membrane at time 1.5 sec for $\delta_t = 0.001$ sec is shown in

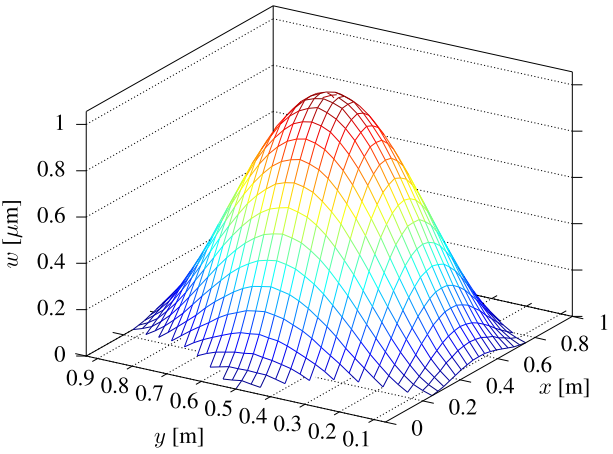


Fig. 8 Initial conditions—membrane deflection at time 0 secs

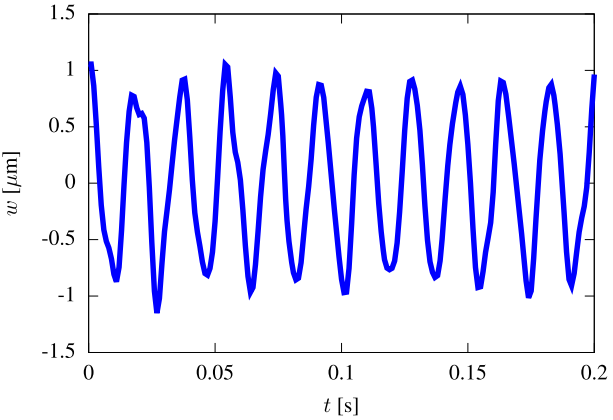


Fig. 9 Deflection at the membrane of the mirror, $\delta_t = 0.001$ secs

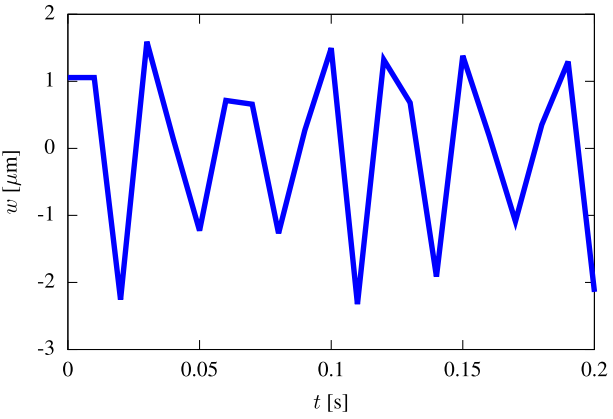


Fig. 10 Deflection at the middle of the membrane, $\delta_t = 0.01$ secs

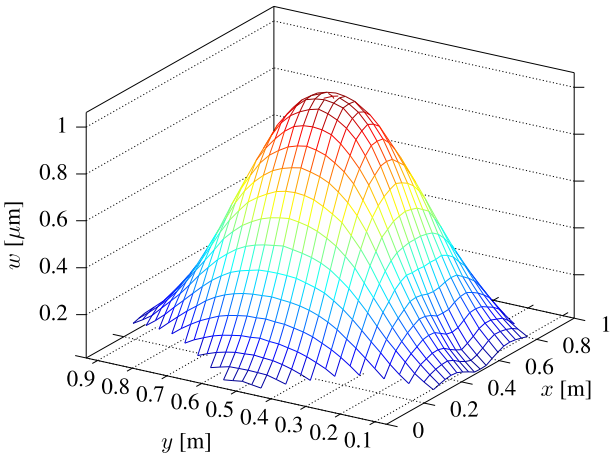


Fig. 11 Membrane deflection at time 0.202 s, $\delta_t = 0.001$ secs

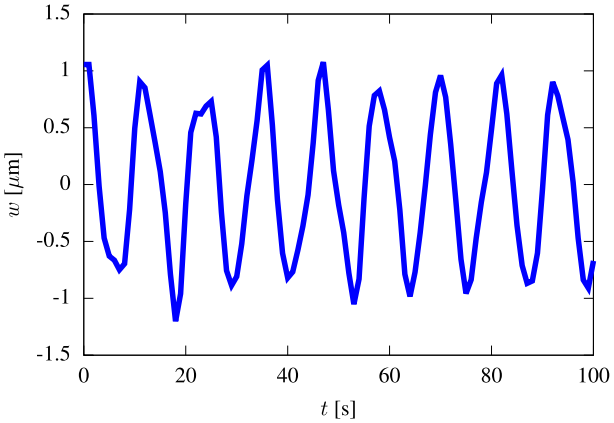


Fig. 12 Deflection at the middle of the membrane, $a = 25$ m, $\delta_t = 1$ s, $\delta_x = 1.25$ m, $\delta_y = 1.0825$ m

Fig. 11, where all values are finite, and this confirms that the difference scheme is numerically stable.

As another case, consider $\delta_t = 1$ sec and $n = 19$, $a = 25$ m and hence $\delta_x = 1.25$ m, $\delta_y = 1.0825$ m. Figure 12 shows the deflections at the middle of the membrane and their finite values confirm that the approximation (6) in this case is also numerically stable.

The following two examples illustrate ILC design using the total (Example 2) and then a limited (Example 3) actuator set.

Example 2

The dynamics considered are those generated by the parameters of Table 1 and the discretization is by the regular hexagonal grid of Fig. 7, with the same boundary conditions, i.e., (44) and (45).

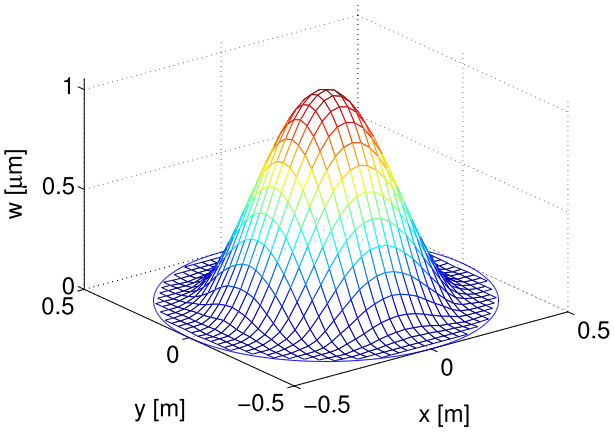


Fig. 13 Reference signal at time instants $t = 4, 5, 6, 7$

Consider the case when the membrane diameter is $a = 1$ m and the consider the discretization grid generated by $n = 11$. These values result in $\delta_x = 0.0833$ m and $\delta_y = 0.0722$ m. Also let the sampling period be $\delta_t = 1$ secs with zero initial conditions and a pass length of $t_f = 11$ secs (discrete time $p = 0, 1, \ldots, 11$). The reference signal increases linearly within discrete time period $p = 0, 1, 2, 3$ to achieve its maximum shown in Fig. 13 and remains constant for $p = 4, 5, 6, 7$ and then decreases linearly for $p = 8, 9, 10, 11$ to zero. The midpoint of the membrane corresponds to the maximum value of the reference signal within the spatial region.

The matrix A of (13) in this case is nonsingular, and the Toeplitz structure makes it easy to invert. Completing the ILC design results in the control law matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ of (43) that are not shown here due to their large dimensions ($N \times 2\,N = 91 \times 182$ and $N \times N = 91 \times 91$). Also the Root Mean Square (RMS) error along the passes is defined as

$$\text{RMS}\,(\mathbf{E}(k)) = \sqrt{\left(\mathbf{E}(k)^T\,\mathbf{E}(k)\right)/(\beta\,N)} \tag{47}$$

and is given in Fig. 14, which shows fast pass-to-pass error convergence. For this case, $\beta = 12$ holds in (47) and hence the control law is applied at 19 of the $N = 91$ points: $q_{p,l-2,m+2}$, $\ldots$, $q_{p,l,m}$, $\ldots$, $q_{p,l+2,m-2}$. The control signal is shown in Fig. 15 for pass $k = 20$ for $p = 5$, where it reaches its maximum value.

After 6 passes, the ILC has reduced the RMS to a very low value and maintains it for all subsequent passes, i.e., in agreement with the theory.

Example 3

Again the parameters are those given in Table 1 and the discretization is undertaken using the regular hexagonal grid of Fig. 7 with the same boundary conditions as in the previosu example, i.e., (44) and (45) and the reference signal is again that of Fig. 13.

Spatially distributed control action was used in Example 2, which is difficult to implement practically due to a large number of sensors and actuators distributed over the system must be used. Here, an alternative ILC law is designed to achieve a given spatial/temporal reference based on the application of a spatially homogeneous control signal at chosen excitation points.

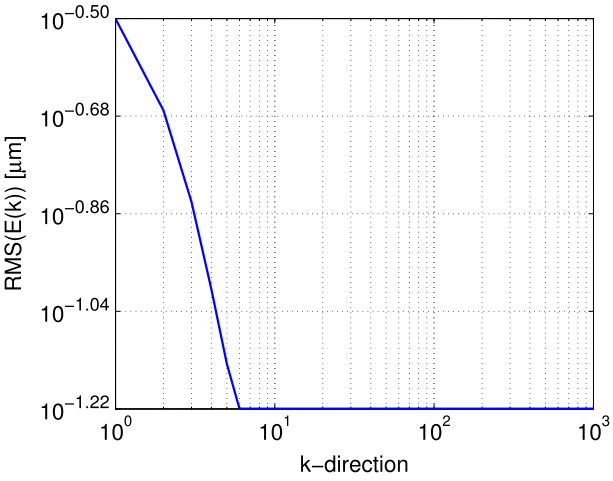


Fig. 14 RMS error (47) for Example 2

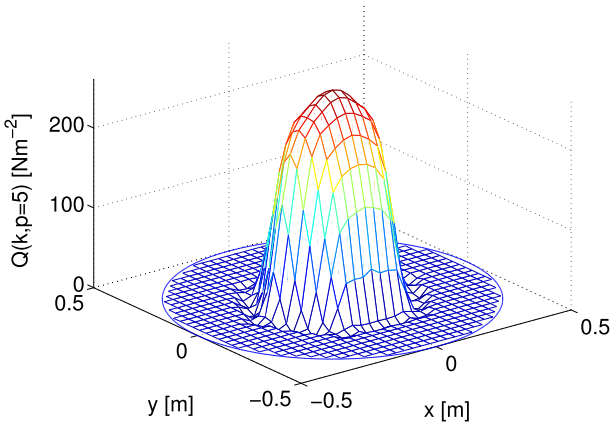


Fig. 15 Control signal at $k = 20$ and $p = 5$

The ILC law matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ of (43) are not shown here due to their large dimensions ($1 \times 2N = 1 \times 182$ and $1 \times N = 1 \times 91$). To limit the number of actuators required to realize the control action, a spatially homogeneous input is available at 7 central points of the plate, from a total of 91 points. This spatially homogeneous control signal is calculated using by suitable selection of the matrix C in the state-space model. In this case, the matrix applies actuation only at the selected seven points

$$\begin{aligned}&(l-1, m+1), (l+1, m+1),\\ &(l-2, m), (l, m), (l+2, m),\\ &(l-1, m-1), (l+1, m-1),\end{aligned} \tag{48}$$

which correspond to the following set of numbers

$$\{35, 36, 45, 46, 47, 56, 57\}, \tag{49}$$

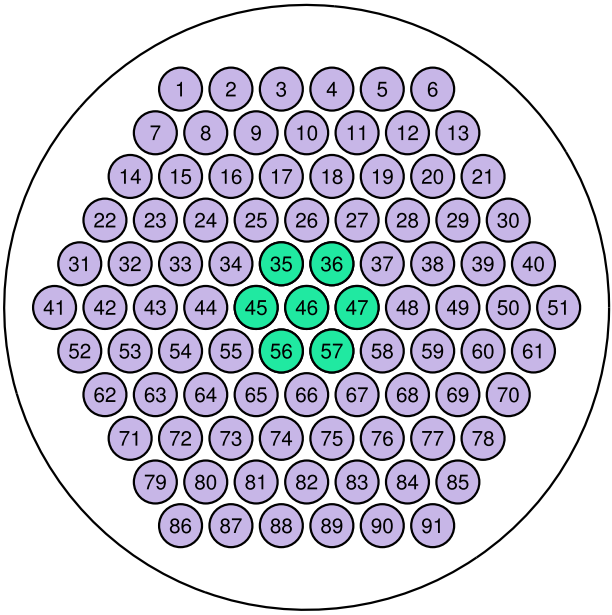


Fig. 16 The seven actuation points marked in green

shown in Fig. 16. Hence, the matrix C according to (16) is replaced by the one given in (50) below, which is a single-column matrix with unit entries in the selected positions and the remainder set to zero

$$\tilde{C}^T = \frac{1}{D}\left[\underbrace{0}_{1-34}\ 1\ 1\ \underbrace{0}_{37-44}\ 1\ 1\ 1\ \underbrace{0}_{48-55}\ 1\ 1\ \underbrace{0}_{58-91}\right]. \tag{50}$$

The numbers under the braces in this last equation denote the positions on the membrane where the input signal is not active.

The RMS is computed as in (47), where $\beta = 12$ is the number of temporal sampling instants along a pass. The result is shown in Fig. 17 and confirms rapid convergence of the ILC design.

Again, this RMS error plot is in agreement with the theory. To improve this result and to speed up the pass-to-pass convergence, an optimization method developed in Cichy et al. (2017) has been applied.

The input signal for the design is shown in Fig. 18 on pass $k = 10$ and $p = 5$ and over the passes in Fig. 19.

Suppose the system model coincides with that used for the evaluation of RMS error. In that case, the approach is identical to the design of an optimal dynamical feedforward control sequence, as the ILC control is trial-independent. There would be a mismatch between the simulated model in experiments, and the actual behavior and adaptation of $q_{k,p}$ would continue until the mismatch has been compensated.

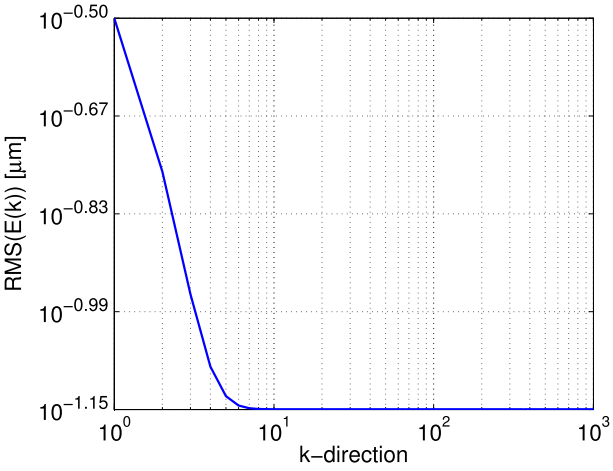


Fig. 17 RMS error (47) for Example 3

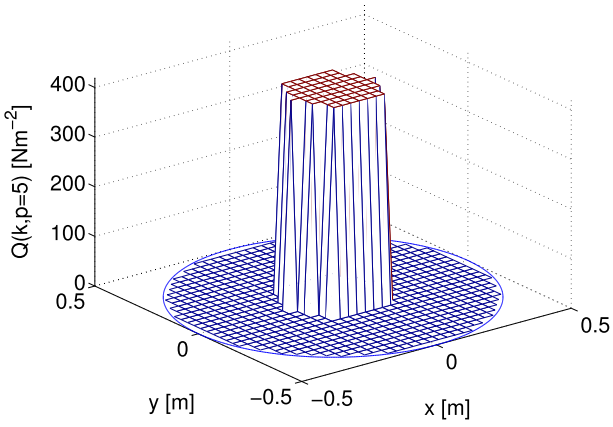


Fig. 18 Input signal at $k = 10$ and $p = 5$

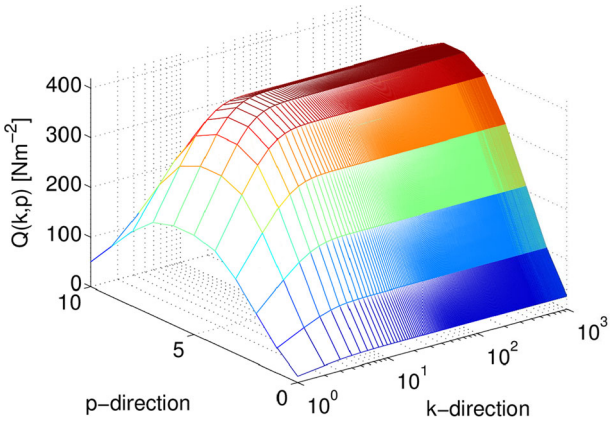


Fig. 19 Input signal over the trials

6 Conclusions and future research

This paper has produced new results on the 2D systems approach to ILC design for PDEs. To form a basis, a particular PDE is considered, which allows for control to be applied over only a subset of the domain. Supporting numerical examples have also been given. The use of the hexagonal girds for circular domains is even more accurate than the rectangular alternative.

The results in this paper also form the basis for possible future research. Potential areas include other forms of the ILC law and robust control design, which conceptually should be straightforward. One difficulty, particularly in terms of computations, is high dimensional matrices, resulting from the use of, in the main, the vector $\mathbf{W}_p$ of (20) (and the vectors from which it is formed) and the need to have a large number of nodal points. Further research could be directed to developing computationally more efficient algorithms for this task.

Funding The work was partially supported by the National Science Centre in Poland, grant no. 2020/37/B/ST7/03280.

Appendix A Notation

$$A = \begin{bmatrix}
A_1^{(\frac{n+1}{2})} & A_2 & A_3 & O & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & O \\
A_2^{\mathrm{T}} & A_1^{(\frac{n+3}{2})} & A_2 & A_3 & O & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
A_3^{\mathrm{T}} & A_2^{\mathrm{T}} & A_1^{(\frac{n+5}{2})} & A_2 & A_3 & O & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
O & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & A_3 & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & A_1^{(n-1)} & A_2 & A_4 & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & A_3^{\mathrm{T}} & A_2^{\mathrm{T}} & A_1^{(n)} & A_2 & A_3 & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & A_4^{\mathrm{T}} & A_2^{\mathrm{T}} & A_1^{(n-1)} & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & A_3^{\mathrm{T}} & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & O \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & O & A_3^{\mathrm{T}} & A_2^{\mathrm{T}} & A_1^{(\frac{n+5}{2})} & A_2 & A_3 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & O & A_3^{\mathrm{T}} & A_2^{\mathrm{T}} & A_1^{(\frac{n+3}{2})} & A_2 \\
O & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & O & A_3^{\mathrm{T}} & A_2^{\mathrm{T}} & A_1^{(\frac{n+1}{2})}
\end{bmatrix},$$

$$A_1^{(X)} = \begin{bmatrix} S & Q & 0 & \cdots & 0 \\ Q & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & Q \\ 0 & \cdots & 0 & Q & S \end{bmatrix}, \quad A_2 = \begin{bmatrix} P & P & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & P & P \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & R & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & R & 0 \end{bmatrix}. \tag{51}$$

References

Ahn, H. S., Chen, Y., & Moore, K. L. (2007). Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man and Cybernetics, Part C, 37*(6), 1109–1121.

Arimoto, S., Kawamura, S., & Miyazaki, F. (1984). Bettering operations of robots by learning. *Journal of Robotic Systems, 1*(1), 123–140.

Augusta, P., Cichy, B., Gałkowski, K., & Rogers, E. (2015). An unconditionally stable finite difference scheme systems described by second order partial differential equations. In *Proceedings of the 2015 IEEE 9th international workshop on multidimensional (nD) systems* (pp. 134–139).

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), 96–114.

Cichy, B., Augusta, P., Rogers, E., Gałkowski, K., & Hurák, Z. (2008). On the control of distributed parameter systems using a multidimensional systems setting. *Mechanical Systems and Signal Processing, 22,* 1566–1581.

Cichy, B., Gałkowski, K., Rauh, A., Aschemann, H., Rogers, E., & Rehak, B. (2017). Modeling and iterative learning control scheme of a circular deformable) mirror. In *20th IFAC world congress* (pp. 3172–3177).

Crank, J., & Nicolson, P. (1947). A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Proceedings of the Cambridge Philosophical Society, 43,* 50–67.

de Rozario, R., Fleming, A., & Oomen, T. (2019). Finite-time learning control using frequency response data with application to a nanopositioning stage. *IEEE/ASME Transactions on Mechatronics, 24*(5), 2085–2096.

Dekker, L. G., Marshall, J. A., & Larsson, J. (2019). Experiments in feedback linearized iterative learning-based path following for center-articulated industrial vehicles. *Journal of Field Robotics, 36*(5), 955–972.

Freeman, C. T., Rogers, E., Hughes, A.-M., Burridge, J. H., & 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.

Hładowski, L., Gałkowski, K., Cai, Z., Rogers, E., Freeman, C. T., & 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.

Huang, D., Li, X., Xu, J.-X., Xu, C., & He, W. (2014). Iterative learning control of inhomogeneous distributed parameter systems-frequency domain design and analysis. *Systems & Control Letters, 72,* 22–29.

Ketelhut, M., Stemmler, S., Gesenhues, J., Hein, M., & Abel, D. (2019). Iterative learning control of ventricular assist devices with variable cycle durations. *Control Engineering Practice, 83,* 33–44.

Lim, I., Hoelzle, D. J., & Barton, K. L. (2017). A multi-objective iterative learning control approach for additive manufacturing applications. *Control Engineering Practice, 64,* 74–87.

Paszke, W., Rogers, E., & Galkowski, K. (2016). Experimentally verified generalized KYP lemma based iterative learning control design. *Control Engineering Practice, 53,* 57–67.

Rabenstein, R., & Steffen, P. (2009). Implicit discretization of linear partial differential equations and repetitive processes. In *Proceedings of the 6th international workshop on multidimensional (nD) systems*

Rabenstein, R., & Steffen, P. (2011). Stability analysis for implicit second order finite difference schemes. In *Proceedings of 7th international workshop on multidimensional (nD) systems (nDs)*.

Rabenstein, R., & Steffen, P. (2012). Numerical iterative methods and repetitive processes. *Multidimensional Systems and Signal Processing, 23*(1–2), 163–183.

Rafajlowicz, W., Jurewicz, P., Reiner, J., & Rafajlowicz, E. (2019). Iterative learning of optimal control for nonlinear processes with applications to laser additive manufacturing. *IEEE Transactions on Control Systems Technology, 27*(6), 2647–2654.

Rogers, E., Gałkowski, K., & Owens, D. H. (2007). Control systems theory and applications for linear repetitive processes. In *Lecture notes in control and information sciences* (Vol. 349). Springer.

Smith, G. D. (1985). *Numerical solution of partial differential equations. Finite difference methods*. Oxford: Oxford University Press.

Strikwerda, J. C. (1989). *Finite difference schemes and partial differential equations*. Belmont: Wadsworth and Brooks.

Timoshenko, S., & Woinowski-Krieger, S. (1959). *Theory of plates and shells*. New York: McGraw Hill.