# Gaussian Process Repetitive Control With Application to an Industrial Substrate Carrier System With Spatial Disturbances

Mooren, Noud; Witvoet, Gert; Oomen, Tom

**DOI**
10.1109/TCST.2022.3177000

**Publication date**
2023

**Document Version**
Final published version

**Published in**
IEEE Transactions on Control Systems Technology

# Gaussian Process Repetitive Control With Application to an Industrial Substrate Carrier System With Spatial Disturbances

Noud Mooren, *Member, IEEE*, Gert Witvoet, *Member, IEEE*, and Tom Oomen, *Senior Member, IEEE*

***Abstract*— Repetitive control (RC) can perfectly attenuate disturbances that are periodic in the time domain. The aim of this article is to develop an RC approach that compensates for disturbances that are time-domain nonperiodic but are repeating in the position domain. The developed position-domain buffer consists of a Gaussian process (GP), which is learned using appropriate dynamic filters and nonequidistant data. This approach estimates position-domain disturbances resulting in perfect compensation. The method is successfully applied to a substrate carrier system, demonstrating performance robustness against time-domain nonperiodic disturbances that are amplified by traditional RC.**

***Index Terms*— Gaussian processes (GPs), repetitive control (RC), spatial disturbances.**

## I. Introduction

SPATIALLY periodic disturbances, including cogging, imbalances, eccentricity, and commutation errors, are encountered in many mechatronic applications [1], [2]. These often appear nonperiodic in the time domain, while these are reproducible in the spatial domain, e.g., with the (angular-)position [3], [4]. In particular, these spatial disturbances appear nonperiodic in the time domain for general tasks. In the case of repeating tasks, these appear periodic, which is a highly restrictive situation.

Repetitive control (RC) enables the perfect attenuation of periodic disturbances through the internal model principle (IMP). The IMP [5] states that a model of the disturbance generating system must be present in the stable feedback loop to enable asymptotic rejection. In traditional RC, a disturbance model is generated in a time-based fixed memory loop, allowing for attenuation of disturbances that have a fixed time-domain period, see, e.g., [6]–[8]. Only in the case of periodic tasks, i.e., if the position-domain disturbances appear periodic in time, RC is effective.

RC is not effective for disturbances that have a varying period or appear nonperiodic in the temporal domain due to inadequate temporal buffer [7], [9]. In the case of nonperiodic tasks, the resulting performance can be significantly degraded. Hence, a traditional time-domain memory loop is not effective for spatially periodic disturbances in the case of nonperiodic tasks.

High-order RC [9], [10] has been developed to increase flexibility for disturbances with uncertain period times by placing multiple delay lines in series. This allows optimizing a tradeoff between variations in the period time, on one the hand, and robustness against nonrepeating errors, on the other hand. In [11], multiperiod RC is designed, which allows for multiple disturbances with different periods, where, for time-domain nonperiodic disturbances, a sum of periodic signals is selected *a priori*. Furthermore, a variety of adaptive RC schemes, including [12]–[15], are developed, exploiting the adaptation of RC parameters, such as the sample time or buffer length, to cope with spatial disturbances. However, practical applications mainly operate at a fixed sample rate. While the above approaches also improve the performance for spatial disturbances with small task variations, they do not generalize to arbitrary task variations.

Spatial disturbances can be modeled efficiently in the position domain, where they appear periodic, in contrast to existing approaches where a time-based memory is employed in RC. Spatial disturbance models have been developed for RC [1], [16], where the disturbance is modeled as a nonlinear potentially time-varying parametric model. Alternatively, in [17]–[19], a discrete buffer is presented, which contains position information to model the spatial disturbance. These discrete buffers require additional interpolation to deal with the inherently nonequidistant data points in the spatial domain. Hence, both spatial RC approaches require additional modeling effort or interpolation, which complicates their practical implementation.

Although improvements have been made to traditional RC to cover wider ranges of disturbances, including spatial dis-

Manuscript received 24 September 2021; revised 22 February 2022; accepted 14 May 2022. Date of publication 2 August 2022; date of current version 28 December 2022. This work was supported in part by the European Union H2020 Program ECSEL-2016-1 under Grant 737453 (I-MECH) and in part by the ECSEL Joint Undertaking under Grant 101007311 (IMOCO4.E). Recommended by Associate Editor K. Barton. *(Corresponding author: Noud Mooren.)*

Noud Mooren is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands (e-mail: n.f.m.mooren@tue.nl).

Gert Witvoet is with the Optomechatronics Department, TNO Technical Sciences, 2628 CK Delft, The Netherlands, and also with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands (e-mail: g.witvoet@tue.nl).

Tom Oomen is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands, and also with the Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: t.a.e.oomen@tue.nl).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCST.2022.3177000.

Digital Object Identifier 10.1109/TCST.2022.3177000

turbances, a systematic approach to generate spatial buffers efficiently with the nonequidistant observations in the spatial domain is not yet available. The aim of this article is to present a spatial RC approach that can compensate for position-domain disturbances by using a new type of spatial memory.

Beyond typical buffers in RC, i.e., by a discrete- or continuous-time delay, recent developments have enabled modeling functions from data, e.g., as a Gaussian process (GP). The key idea in GP regression is to estimate a distribution over functions from data and prior knowledge, see [20], [21] for a complete overview. This enables efficient interpolation of continuous disturbance models from finite measurement points [22], [23]. The aim is to employ a GP-based disturbance model in a spatial memory loop for RC.

The main contribution of this article is a continuous GP-based spatial disturbance model in RC to enable attenuation of spatial disturbance for arbitrary task variations. This includes the following subcontributions:

- C1) developing a novel continuous spatial disturbance model and efficiently utilizing nonequidistant observations in combination with suitable prior knowledge based on GPs;
- C2) implementation of the GP-based disturbance model in a memory loop for spatial RC, including stability analysis, learning filter design, and prior selection;
- C3) a computationally efficient implementation of the GP-based spatial memory loop;
- C4) a simulation case study, with full GP, sparse GP, and traditional RC;
- C5) experimental validation on a substrate carrier system.

Preliminary results are presented in [24]. Extensions in this article are: 1) an improved spatial buffer design; 2) stability analysis; 3) an efficient implementation of the disturbance model using a sparse GP; and 4) experimental validation confirming the benefits of the method.

This article is outlined as follows. In Section II, the problem setting and spatial disturbance are introduced. In Section III, a continuous function of the spatial disturbance is identified as a GP with suitable prior (C1). In Section IV, the spatial RC framework is presented, including the stability analysis, the learning filter design, and the integration of the spatial model in a spatial memory loop (C2). In Section V, a computationally efficient implementation of GP regression is provided (C3). Finally, in Sections VI and VII, simulations and experimental validation are carried out (C4 and C5), and conclusions are presented in Section VIII.

## II. Problem Formulation

In this section, an industrial relevant motivational application is presented, and the considered type of disturbance is defined. Furthermore, the control setting and spatial disturbance rejection problem are defined.

### A. Application Motivation

Disturbances that are repeating in the position domain appear in many industrial applications, e.g., due to imbalances or imperfections in rotary systems or nonperfect commutation leading to position-dependent disturbances in positioning systems, such as a wafer stage. In this article, the industrial substrate carrier in Fig. 1 with schematic representation in Fig. 2 is considered and used for experimental validation.

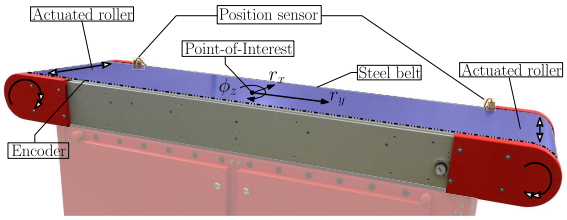


Fig. 1. Industrial substrate carrier setup, where to aim is to position the PoI, subject to spatially periodic disturbances. The belt is controller using actuated rollers.

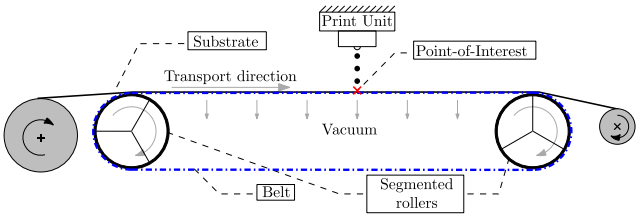


Fig. 2. Schematic representation of the industrial substrate carrier setup.

The aim of the substrate carrier setup is to accurately position the point-of-interest (PoI) on the substrate with respect to a print unit. This is enabled by fixating the substrate to a steel belt using a vacuum; consequently, the position of the steel belt is controlled in the three degrees of freedom (DoFs): $r_x$, $r_y$, and $\phi_z$; see [25] and [26] for details. Positioning of the belt is performed with two actuated rollers for the longitudinal direction $r_y$; these rollers also contain three segments that are actuated in the lateral direction to control $r_x$ and $\phi_z$ (see Figs. 1 and 2). The position of the belt is measured with an encoder in $r_y$, and two sensors on the side of the belt measure $r_x$ and $\phi_z$. The focus of the remainder of this article is on the positioning performance in the lateral direction $r_x$.

The rotational nature of this system and slight imperfections in the rollers and their segments induces a position-dependent disturbance that repeats every roller rotation. Depending on the belt velocity, which may vary over time, the disturbance can appear periodic or nonperiodic in the time domain. The aim of this article is to reject the spatial disturbance, independent of roller velocity variations. Furthermore, the roller position in the experiment setup is measured very accurately and fast, which is also considered in the remainder of this article.

### B. Control Problem

The control configuration is depicted in Fig. 3, where the substrate carrier is denoted by $P$ and assumed to be a single-input–single-output (SISO) stable linear time-invariant (LTI) system, and $C$ is a stabilizing feedback controller. The reference to be tracked is $r(t)$, which is zero in the industrial setup, and $y(t)$ is the position output corresponding to $\phi_z$ in the setup. A roller-induced input disturbance $d(t)$ is present, which is generated by an unknown static position-domain mapping $\bar{d}(p) \in \mathbb{R}$, driven by an exogenous and known position signal $p(t)$, i.e., the roller position in the considered experimental setup. Note that the position signal $p(t)$ and the system output

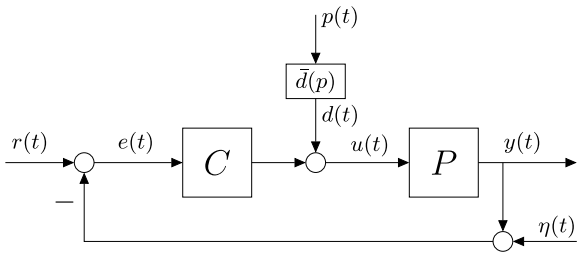


Fig. 3. Spatial RC problem with input disturbance $d(t)$.

$y(t)$ are not related in this setting. Furthermore, $\eta(t)$ represents measurement noise that is independent and identically distributed Gaussian zero mean, i.e., $\eta(t) \sim \mathcal{N}(0, \sigma_\eta^2)$. Throughout this article, $t \in \mathbb{R}$ is the continuous time, $k \in \mathbb{Z}$ is the discrete time, and the notation $\bar{\cdot}$ indicates a spatial signal. The aim is formulated in the following definition.

*Definition 1:* Define the measured tracking error

$$e_m(t) = r(t) - y(t) - \eta(t) \tag{1}$$

in the presence of a position-domain disturbance $d(t)$, reference $r(t)$, and measurement noise $\eta(t)$. Assume that the frequency contents of $r(t)$ and $d(t)$ are significantly different from the frequency content of $\eta(t)$, such that the aim is to minimize the disturbance- and reference-induced errors.

The disturbance- and reference-induced errors can be written as

$$e = S(r - Pd) = e_r + e_d \tag{2}$$

with $S = 1/(1 + PC)$ being the sensitivity function, and $e_d$ and $e_r$ the disturbance-induced error and the reference-induced error, respectively. This article focuses on the former by making the following assumption.

*Assumption 1:* The reference-induced error $e_r(t)$ is negligibly small after a certain (transient) time $t_0$, that is,

$$|e_r(t)| \leq \beta \quad \forall t \geq t_0 \tag{3}$$

with $\beta \in \mathbb{R}$ being a sufficiently small constant. This can be obtained by a suitable design choice of $C$ and possible feedforward control, i.e., $C_{\text{ff}} = P^{-1}$ and $u_{\text{ff}} = C_{\text{ff}} r$ that is added to the input $u(t)$, see [27].

Hence, when $t \geq t_0$, we assume that $e(t) = e_d(t)$ so that the control goal reduces to the attenuation of the spatially periodic disturbance, as outlined next.

### C. Spatial Disturbance

The disturbance $d(t)$ acting on the system may appear periodic or nonperiodic in time. However, it has an equivalent counterpart $\bar{d}(p)$ that is repeating in the spatial domain. Hence, the roller-induced disturbance in the substrate carrier system is modeled as a static but unknown function of position, defined as follows.

*Definition 2:* The disturbance $d(t)$ is composed from the exogenous and known position signal $p(t)$ and the unknown static spatial disturbance function $\bar{d}(p)$, that is,

$$d(t) = \bar{d}(p(t)) \tag{4}$$

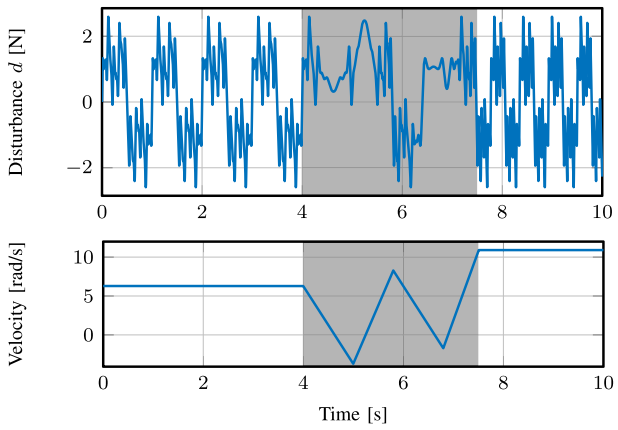


Fig. 4. Simulation example of a position-domain disturbance (top) with corresponding velocity of $p(t)$ (bottom) as a function of time. The gray area indicates where the velocity changes, which causes the disturbance to become nonperiodic in the time domain.

where $\bar{d}(p)$ is periodic in the spatial domain, that is,

$$\bar{d}(p) = \bar{d}\big(p + n \cdot p_{\text{per}}\big), \quad \text{for } n \in \mathbb{N} \tag{5}$$

with $p_{\text{per}} \in \mathbb{R}$ being the spatial period.

*Remark 1:* For this specific application in RC, the spatial disturbance function $\bar{d}$ is assumed to be periodic. However, this is not required in the general case.

*Remark 2:* In contrast to other existing approaches [4], [28], the exogenous position signal $p(t)$ is not equal to the plant output; also, the spatial disturbance is a static function of position, i.e., not a dynamical model, which removes the need to convert the system dynamics to the position domain, resulting in a time-varying system.

The spatial disturbance leads to two major challenges. First, if $p(t)$ is periodic, then $d(t)$ is periodic. However, if $p(t)$ is nonperiodic, then $d(t)$ is in general also nonperiodic. This implies that nonperiodicity of $p(t)$ leads to $d(t)$ being nonperiodic in the time domain. For instance, if $\dot{p}(t)$ is (piecewise) constant, then $d(t)$ is (piecewise) periodic in time, and if $\dot{p}(t)$ varies, then $d(t)$ is nonperiodic in time, see Fig. 4. Second, samples of the position-domain disturbance are in the general case nonequidistant in the spatial domain. The sampled position signal is denoted by $p(t_k)$, where $t_k = kT_s$ with $T_s$ being the sample time and $k \in \mathbb{N}$, which are equidistant in time. As a consequence, the corresponding spatial samples are nonequidistant, that is,

$$p(t_k) - p(t_{k+1}) \neq p(t_i) - p(t_{i+1}) \text{ for some } \quad i, k \in \mathbb{N} \tag{6}$$

which implies that observations of $\bar{d}(p)$ are nonequidistant in the spatial domain; see Fig. 5 where $d(t)$ and several samples that are equidistant in time are shown both in the time and position domains where they are nonequidistant. This leads to major challenges for writing to and reading from the buffer, which are both essential aspects in RC.

### D. Problem Definition

The aim of this article is to reject the spatially periodic disturbance-invariant under velocity variations in $p(t)$. This is established in the spatial RC framework in Fig. 6, where,

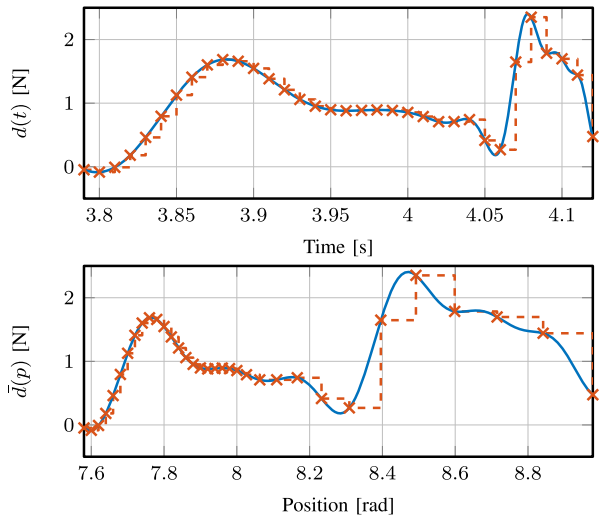


Fig. 5. Simulation example: the top plot shows a continuous disturbance as a function of time $d(t)$ (—) from which equidistant samples (×) are taken with a fixed sampling frequency. The bottom plot shows the same disturbance as a function of position yielding a different shape, where, due to velocity variations, the same samples that where equidistant in time are now nonequidistant in position (×).

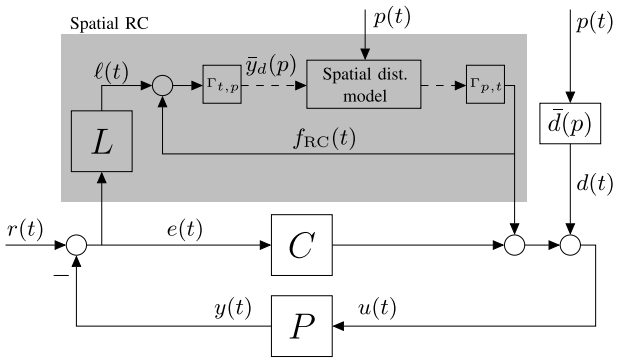


Fig. 6. Spatial RC framework, where (- - -) represents a position-domain signal and (—) is a time-domain signal.

analog to traditional RC, the IMP is applied by learning a continuous disturbance model through a time-domain filter. The crucial difference is that a disturbance model is learned in a spatial memory loop instead of a temporal one in traditional RC. The disturbance model is learned from nonequidistant observations $\bar{y}_d$ and predicted at other positions for compensation. To enable learning in the spatial domain, transformations between time-domain signals and position-domain signals are used, and a suitable time-domain learning filter $L$ is developed. Next, in Section III, the spatial disturbance model is derived, and in Section IV, the spatial RC framework is designed and analyzed.

## III. Gaussian Process Spatial Disturbance Model

In this section, a continuous model of the spatial disturbance is identified from nonequidistant observations by means of GP regression, constituting C1.

### A. Identifying a Spatial Disturbance Model

According to the IMP, the spatial buffer in Fig. 6 must contain a model of $\bar{d}(p)$, which is learned from $N$ training samples $\bar{d}(p_i)$ at nonequidistant positions $p_i \in \mathbb{R}$ for $i = 1, 2, \ldots, N$. At the same time, the RC provides a control action by evaluating the disturbance model at another position $p_j \in \mathbb{R}$, which is, in general, not equal to the training samples. This implies that interpolation or extrapolation is required to estimate $\bar{d}(p_j)$ from data $\bar{d}(p_i)$.

GP regression enables estimation of $\bar{d}(p)$ as a continuous function from nonequidistant training data and suitable prior knowledge, essentially automating the interpolation that is required in other spatial approaches [18]. The resulting GP-based disturbance model is a distribution over functions, i.e., a collection of random variables in the position domain determined by a mean and variance, see Fig. 7 for an illustrative example. Loosely speaking, GP regression is a projection of data on a set of basis functions that do not need to be defined explicitly, which is a major advantage over parametric estimation methods. Instead, a class of a potentially infinite set of basis functions is defined in the form of a kernel function; it will become clear in Section IV how to design the kernel function for spatial RC. Furthermore, GP regression can be split up into two key steps: a training step and a prediction step by performing inference; these are considered in the remainder of this section and used in the spatial RC framework in Section IV.

*Remark 3:* In contrast to existing spatial RC approaches, the nonequidistant data are used to estimate a distribution over (periodic) function, whereas traditional approaches either require interpolation in a discrete buffer [17]–[19] or require to reformulate the systems in the spatial domain resulting in time-varying dynamics [2], [4]. This article presents a systematic design approach for continuous buffers without interpolation while also removing the need for complex time-varying models.

Next, consider the disturbance function that is assumed to generate the data. Thereafter, a suitable disturbance estimation for spatial RC is formulated as a GP regression problem (see Section IV).

### B. Spatial Disturbance Function and Training Dataset

Observations of the spatial disturbance $\bar{y}_d$ are subject to independent and identically distributed zero-mean Gaussian noise with variance $\sigma_n^2$, that is,

$$\bar{y}_d(p_k) = \bar{f}(p_k) + \epsilon, \quad \text{with } \epsilon \sim \mathcal{N}\big(0, \sigma_n^2\big) \tag{7}$$

where $p_k \in \mathbb{R}$ is the position at which observation $\bar{y}_d(p_k)$ is taken. Note that $\epsilon$ is the result of measurement noise $\eta$, filtered by the closed loop and learning filter, which remains normally distributed. It will be shown later in Section IV that $\bar{f}(p_k)$ approximates the actual disturbance function, i.e., $\bar{f}(p_k) = \bar{d}(p_k)$. The function $\bar{f}$ is parameterized as a linear combination of a potential infinite number of basis functions and parameters

$$\bar{f}(p_k) = \phi(p_k)^\top w, \quad \text{with } w \sim \mathcal{N}\big(0, \Sigma_p\big) \tag{8}$$

where $\phi(p_k)$ is a vector of basis functions and $\Sigma_p = \mathbb{E}(ww^\top)$ is the covariance matrix of the weights $w$. Finally, the training dataset containing noisy observations and corresponding

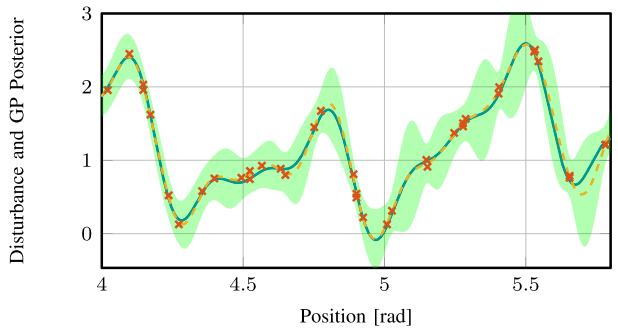


Fig. 7. GP regression example with the disturbance (—) from which noisy observations (×) are generated. With these observations, the GP posterior mean (- - ) and variance ( ) are computed.

positions is denoted as $\mathcal{D} = (X, \bar{\mathbf{y}}_d)$ with

$$X = \begin{bmatrix} p_1 & p_2 & \cdots & p_N \end{bmatrix}^\top \in \mathbb{R}^N$$
$$\bar{\mathbf{y}}_d = \begin{bmatrix} \bar{y}_{d,1} & \bar{y}_{d,2} & \cdots & \bar{y}_{d,N} \end{bmatrix}^\top \in \mathbb{R}^N$$

containing $N$ training positions and observations, respectively, and is used to train the GP.

Next, the training data $\mathcal{D}$ are used to identify a continuous function $\bar{f}$ that represents the spatial disturbance $\bar{d}$ and allows to make predictions outside the training set.

### C. Gaussian Process Regression

In this section, a GP regression problem is formulated to model the continuous disturbance function from data $\mathcal{D}$ and prior knowledge, i.e., training of the GP, and to make a prediction at new position $p_*$ by performing inference; for further details, see [20]. This separation is used later in Section IV to establish the spatial memory loop.

First, define a set of $N_*$ test positions $\mathbf{p}_* \in \mathbb{R}^{N_*}$ with corresponding function values $\bar{\mathbf{f}}_* = \bar{f}(\mathbf{p}_*)$, and the training points $\bar{\mathbf{y}}_d$. Next, assume that $\bar{\mathbf{f}}_*$ and $\bar{\mathbf{y}}_d$ have a joint Gaussian distribution $p(\bar{\mathbf{y}}_d, \bar{\mathbf{f}}_*)$ given by

$$\begin{bmatrix} \bar{\mathbf{y}}_d \\ \bar{\mathbf{f}}_* \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K_{XX} + \sigma_n^2 I_N & K_{X*} \\ K_{X*}^\top & K_{**} \end{bmatrix} \right) \tag{9}$$

where $K_{XX} \in \mathbb{R}^{N \times N}$ denotes the matrix of covariances evaluated at all pairs of training positions in $X$ and similar for $K_{X*} \in \mathbb{R}^{N \times N^*}$ and $K_{**} \in \mathbb{R}^{N^* \times N^*}$, which also includes test positions with $*$. The covariance matrix $K$ is selected by the user (see Section IV-E) and also known as the kernel matrix that expresses prior knowledge on the function $\bar{f}$ to be estimated.

Predictions of $\bar{f}$ at a test point $p_*$ from training data $\mathcal{D}$ are given by the conditional posterior distribution of $\bar{f}^*$

$$p(\bar{f}_* | p_*, \mathcal{D}) = \mathcal{N}(\bar{\mu}_{\text{post}}, P_{\text{post}}) \tag{10}$$

where

$$\bar{\mu}_{\text{post}} = K_{X*}^\top \left(K_{XX} + \sigma_n^2 I_N\right)^{-1} \bar{\mathbf{y}}_d \tag{11a}$$
$$P_{\text{post}} = K_{**} - K_{X*}^\top \left(K_{XX} + \sigma_n^2 I_N\right)^{-1} K_{X*} \tag{11b}$$

are the posterior predictive mean and covariance, respectively. The GP mean can be computed efficiently for a single test point $p_*$ as

$$\bar{\mu}_{\text{post}}(p_*) = \sum_{i=1}^{N} \alpha_i \kappa(p_i, p_*) \tag{12}$$

where $\kappa(p_i, p_*)$ is the kernel function evaluated at training points and test point, and

$$\alpha = \left(K_{XX} + \sigma_n^2 I_N\right)^{-1} \bar{\mathbf{y}}_d = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{bmatrix}^\top \tag{13}$$

is essentially the link between the trained GP and the prediction step, as will become clear later.

The posterior mean, i.e., (11a) or (12), involves inverting an $N \times N$ matrix that solely depends on the training data $\mathcal{D}$. Due to a lack of structure, the computational complexity of this inversion scales cubically with $N$. Note that this inversion, i.e., training of the GP, can be done independently of the prediction step such that new predictions are less time-consuming. In addition, in Section V, a sparse GP approximation is presented, which reduces the computational effort.

*Remark 4:* Inversion of the matrix $(K_{XX} + \sigma_n^2 I) \in \mathbb{R}^{N \times N}$ can be done efficiently using the Sherman–Morrison formula, essentially using rank 1 updates on the previous inverse, similar to [22]. This reduces the computational complexity to $\mathcal{O}(N^2)$ for each training update instead of $\mathcal{O}(N^3)$ if standard inversion algorithms are used.

*Remark 5:* If the position signal $p(k)$ in the training data $X$ is not known sufficiently accurate, then input noise can be taken into account for GP regression, see, e.g., [29], [30].

### D. Prior Selection

GP regression relies on data and suitable prior knowledge; the latter is essential to extrapolate the estimated model beyond the training points, which is a key in the spatial RC framework. The function $\bar{f}$ in (8) is defined by a set of basis function $\phi$ and parameters $w$. Instead of selecting these basis functions explicitly, the prior distribution (9) contains a covariance function that specifies a *class* of basis functions. Consequently, the prior spans a potentially infinite set of basis functions, without explicitly defining them, which is a result known as Mercer's Theorem, see [20, Sec. 4.3].

To show this, the covariance of two training observations of the disturbance is written as

$$\begin{aligned} \text{cov}\left(\bar{y}_d^i, \bar{y}_d^j\right) &= \mathbb{E}\left[ \left(\psi(p_i)^\top w + \epsilon_i\right) \left(\psi(p_j)^\top w + \epsilon_j\right)^\top \right] \\ &= \phi^\top(p_i) \Sigma_p \phi(p_j) + \delta_{ij} \sigma_n^2 \end{aligned} \tag{14}$$

such that

$$\text{cov}(\bar{\mathbf{y}}_d) = \Phi^\top \Sigma_p \Phi + \sigma_n^2 I_N \tag{15}$$

and the mean of a training observation

$$\mu(\bar{y}_d) = \mathbb{E}\left[\phi(p)^\top w + \epsilon\right] = \phi(p)^\top \mathbb{E}[w] \tag{16}$$

is assumed to be zero, i.e., $p(\bar{\mathbf{f}}) \sim \mathcal{N}(0, \Phi^\top \Sigma \Phi)$ with $\bar{\mathbf{f}} = \bar{f}(X)$.

*Remark 6:* GP regression can easily be extended for nonzero mean in the case that a prior estimate of the disturbance function is known, see [21].

By comparing the covariance of the observations in (15) with the joint prior in (9), it follows that the kernel expresses prior on the function $\bar{\mathbf{f}}$, i.e., $K_{XX} = \Phi^\top \Sigma \Phi$, such that the covariance matrix (15) becomes

$$\operatorname{cov}(\bar{\mathbf{y}}_d) = K_{XX} + \sigma_n^2 I_N. \tag{17}$$

Details on how to choose the kernel matrix for the application in RC follow in Section IV-E.

*Remark 7:* It follows from Mercer's theorem that any symmetric positive definite kernel matrix can be mapped into an inner product $K(p, p') = \langle \psi(p), \psi(p') \rangle$ with $\phi(p) = \Sigma_p^{(1/2)} \psi(p)$ being a potentially infinite sum of basis functions, see [20], [21]. This enables to perform inference with an infinite set of basis functions, while the kernel matrix is of size $N \times N$.

This completes the GP-based disturbance modeling; in Section IV, the disturbance model is incorporated into the spatial RC framework.

## IV. Spatial Gaussian Process Repetitive Control

In this section, the spatial GP-based RC as briefly introduced in Section II is further analyzed (contribution C2). This includes stability analysis, suitable learning filter design, integration of GP-disturbance model in a memory loop, and prior selection for the GP. Finally, a procedure is provided to implement spatial GP-based RC.

### A. Spatial Repetitive Controller

The spatial repetitive controller indicated in gray in Fig. 6 is depicted in detail in Fig. 8, where the solid lines represent time-domain signals, and dashed lines are position-domain signals. The GP-based disturbance model is included in two blocks, i.e., $GP_y$ that represents training of the GP by computing (13) from data $\mathcal{D}$, and $GP(p)$ is the prediction step at test point $p^* = p$ to generate the RC output $f_{\mathrm{RC}}$. The link between the training and prediction steps is given by (13) and indicated by the dotted line. Furthermore, transitions between time and position domains are indicated by $\Gamma_{t,p}$ and $\Gamma_{p,t}$, respectively, which are defined as follows.

*Definition 3:* The mappings $\Gamma_{t,p}$ and $\Gamma_{p,t}$ map a time domain signal $x(k)$ at sample $k$ to the spatial signal $\bar{x}(p)$ and vice versa, that is,

$$\Gamma_{t,p} : x(k) \mapsto \bar{x}(\bar{p}), \quad \bar{x}(\bar{p}) = x(k) \tag{18a}$$

$$\Gamma_{p,t} : \bar{x}(p) \mapsto x(k), \quad x(k) = \bar{x}(p) \tag{18b}$$

where $\bar{p}$ is the position at which the observation $x(k)$ is stored in $\mathcal{D}$, and $p$ is the current position at sample $k$.

A stable learning filter $L$ is present, which filters the error to obtain the learning signal $\ell$ that is used to update the GP. Since the disturbance will be suppressed over time, the error $e$ and, consequently, $\ell$ will converge to zero. Hence, the learning signal $\ell$ cannot be fed directly to the GP. A memory element in the form of a feedback loop, i.e., similar to traditional RC (see [7]), is present in the spatial RC such that the identified model remains present in the loop. This constitutes the update

$$y_d(k) = \ell(k) + f_{\mathrm{RC}}(k) \tag{19}$$

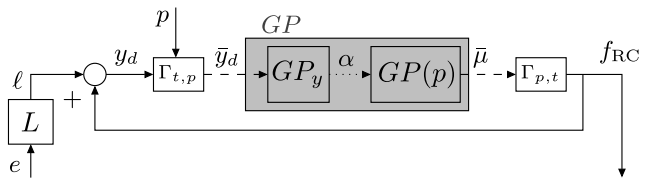


Fig. 8. Spatial RC with GP-based memory loop, where (- - -) is a position-domain signal and (——) a time-domain signal (·····).

where $\ell = Le$, and $f_{\mathrm{RC}}$ is the RC output at sample $k$. Using Definition 3, the spatial observation is generated by transforming $y_d$ to the spatial domain, that is,

$$\bar{y}_d(p) = y_d(k) \tag{20}$$

essentially including $y_d(k)$ at position $p(k)$ in the training dataset $\mathcal{D}$. The spatial RC output $f_{\mathrm{RC}}$ is generated by evaluating the GP at test position $p(k)$ and converting to the time domain as in Definition 3, that is,

$$f_{\mathrm{RC}}(k) = \bar{\mu}(p) \tag{21}$$

with $\bar{\mu}(p)$ being the mean of the GP at position $p$, i.e., (12) with $p_* = p$.

*Remark 8:* Note that, in classical RC, the learning filter can be placed either before or after the buffer because of the commutative property of SISO LTI filters. In spatial RC, the commutative property no longer holds, and the position of the learning filter $L$ before the buffer determines the interpretation of the GP, which is of crucial importance.

In the remainder of this section, it is shown how to design $L$ to ensure closed-loop stability such that $\bar{y}_d$ represents an estimate of the disturbance, i.e., $\bar{y}_d(p) = \hat{d}(p)$.

### B. Stability Analysis

To analyze closed-loop stability with spatial RC, the input–output gain of the GP buffer is upper-bounded. This is a reasonable assumption since the GP will mainly take care of the interpolation/extrapolation of the data given a certain prior. Hence, if a perfect model is obtained, then the gain of the GP buffer, i.e., from $y_d$ to $f_{\mathrm{RC}}$, is one. The mean of the GP is initialized as zero; hence, if no data are available, then the gain tends to zero.

*Assumption 2:* There exists $\gamma \in \mathbb{R}$ such that the $\ell_2$-induced gain of the spatial GP buffer is upper bounded as

$$\|f_{\mathrm{RC}}\|_{\ell_2} \leq \gamma \|y_d\|_{\ell_2}. \tag{22}$$

Assumption 2 implies that the output energy of the GP buffer is less than or equal to $\gamma$ times input energy, which is required to provide stability results for GP-based RC. In contrast to traditional RC, where the buffer is a delay $z^{-N}$ that has a gain of one, the GP buffer depends on many variables, including the hyperparameters and the position $p$, and often $\gamma \neq 1$. Assumption 2 provides a condition to be tested on given a set of hyperparameters, rather than providing direct design guidelines for the hyperparameters, as there is not a direct relation between $\gamma$ and the hyperparameters.

The following lemma is introduced before the main stability result is stated.

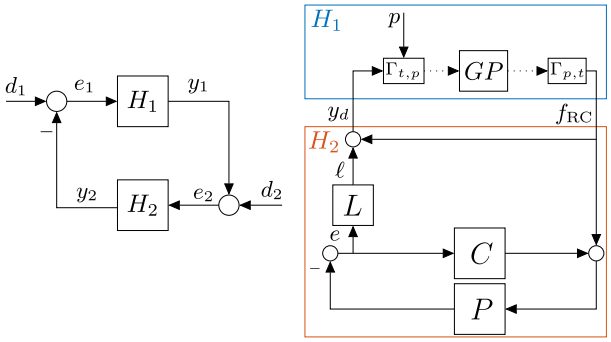


Fig. 9. Standard feedback interconnection (left) and GP-based RC casted in the standard feedback interconnection (right).

*Lemma 1:* For the system interconnection in Fig. 9, with

$$\|y_1\|_{\ell_2} \leq \gamma_1 \|e_1\|_{\ell_2} \tag{23}$$

$$\|y_2\|_{\ell_2} \leq \gamma_2 \|e_2\|_{\ell_2} \tag{24}$$

the inputs $\|d_1\|_{\ell_2}$, $\|d_2\|_{\ell_2} < \infty$ have finite $\ell_2$-induced norm. Then, for $\gamma_1\gamma_2 < 1$, the signals $e_1$, $e_1$, $y_1$, and $y_2$ have bounded $\ell_2$-induced norms, i.e., the interconnection is internally stable. For a proof, see [31, Ch. 2].

The stability of the closed loop is given by the following theorem.

*Theorem 1:* The GP-based spatial RC is closed-loop stable under Assumption 2 if

$$\left|1 - L\left(e^{j\omega}\right) S\left(e^{j\omega}\right) P\left(e^{j\omega}\right)\right| < \frac{1}{\gamma} \; \forall \, \omega \in [0, 2\pi]. \tag{25}$$

*Proof:* First, the spatial RC framework is casted in the standard feedback interconnection, as in Lemma 1, and then, it is shown that, if Theorem 1 is satisfied under Assumption 2, such that Lemma 1 is satisfied, the system is internally stable.

First, set $H_1$ as the nonlinear mapping from $y_d(k)$ to $f_{\mathrm{RC}}(k)$; then, $H_2$ is a linear system

$$y_d = H_2 f_{\mathrm{RC}} - \mathrm{LSP}d \tag{26}$$

with $H_2 = 1 - \mathrm{LSP}$, where LSP is stable since $L$ is stable by design and $C$ stabilizes the feedback loop, i.e., SP is also stable.

Second, it remains to show that (25) satisfies Lemma 1 for some $\gamma_1\gamma_2 < 1$. Note that $H_2$ corresponds to the mapping $y_d \mapsto f_{\mathrm{RC}}$, which is linear; thus, the signal norm bound (24) is satisfied if

$$\|1 - \mathrm{LSP}\|_\infty \leq \gamma_2 \tag{27}$$

for some $\gamma_2 < 1$, which is equivalent to (25) for SISO systems. From Assumption 2, it follows that $\gamma_1 \leq 1$, i.e., $\gamma_1\gamma_2 < 1$, and hence, Lemma 1 is satisfied, which completes the proof. ■

In condition (25) in Theorem 1 an additional robustness filter $Q$ can be introduced at the output of the GP-based buffer similar to traditional RC, see [32] for detailed design guidelines.

*Remark 9:* Note that the stability condition in Theorem 1 can be tested on the basis of an identified frequency response function of the system, see [11].

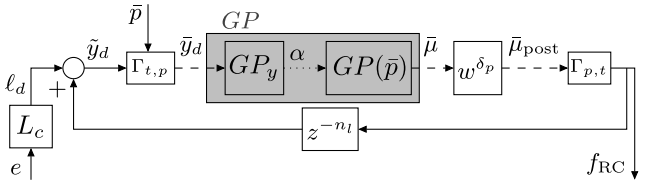


Fig. 10. Spatial RC with GP-based memory loop including preview.

### C. Learning Filter Design

From Theorem 1, it follows that, if $L$ is designed as the inverse of SP, then (25) is satisfied, and the closed-loop stability is obtained. In addition, recall from (2) that $d = -(\mathrm{SP})^{-1} e_d$, i.e., this learning filter design is also a sensible choice since the learning signal $\ell = Le = d$ generates an estimate of $d$ to train the GP. Next, a procedure is outlined to design the learning filter.

**Procedure 1: Learning Filter Design**

1) Identify a parametric model $\widehat{SP} = \hat{P}\,(I + C_{\mathrm{fb}}\hat{P})^{-1}$.
2) Inversion, i.e., $L = \widehat{SP}^{-1}$.

Direct inversion of $\widehat{\mathrm{SP}}$ may lead to a noncausal or unstable inverse, i.e., if the plant contains nonminimum phase zeros or delays. By employing finite preview, a bounded inverse can be obtained, see [33], [34], leading to

$$L = q^{n_l} L_c \tag{28}$$

where $q$ is the forward time-shift operator, $L_c$ is the causal part of $L$, and $n_l$ is the number of samples of preview in $L$.

In Section IV-D, the RC scheme in Fig. 8 is extended to implement the noncausal part of the learning filter.

### D. Incorporating Spatial Preview

In traditional RC, preview is incorporated in the time-domain buffer. This principle is extended toward spatial preview. A noncausal $L$ filter can be implemented in spatial RC by filtering the error with the causal part of $L$ and implementing the noncausal part as a preview in the spatial memory.

To show this, define $\ell_d$ as the error filtered with the causal part of $L$

$$\ell_d(k) = L_c(q) e_d(k) = \hat{d}(k - n_l) \tag{29}$$

which corresponds to the disturbance estimate with $n_l$ samples of delay. Substituting (29) in the update (19) gives

$$y_d(k) = q^{n_l} \ell_d(k) + f_{\mathrm{RC}}(k) \tag{30}$$

which requires future values of $\ell_d$. Multiplying both sides with $q^{-n_l}$ gives the following causal update:

$$\tilde{y}_d(k) = \ell_d(k) + q^{-n_l} f_{\mathrm{RC}}(k) \tag{31}$$

where $\tilde{y}_d(k) = q^{-n_l} y_d(k)$; this is schematically presented in the RC memory loop in Fig. 10. Next, from (29), and by using that $q^{-n_l} f_{\mathrm{RC}}(k) = \bar{\mu}(p(k - n_l))$, the observation

$\tilde{y}_d$ corresponds to the position $p(k-n_l)$. Hence, the spatial observation $\bar{y}_d$ is included in $\mathcal{D}$ at position $p(k-n_l)$, that is,

$$\bar{y}_d(\bar{p}) = \tilde{y}_d(k) \tag{32}$$

with $\bar{p} = p - \delta_p$ and $\delta_p(k) = p(k) - p(k-n_l)$ is a spatial shift.

*Definition 4:* The spatial forward shift operator is defined as

$$w^{\delta_p}\bar{x}(p) = \bar{x}\big(p+\delta_p\big). \tag{33}$$

To generate $f_{\mathrm{RC}}(k)$, the GP is evaluated at the current position $p$ by employing a spatial preview of $\delta_p$ to $\bar{\mu}(\bar{p})$ resulting in

$$\bar{\mu}_{\mathrm{post}}(p) = w^{\delta_p}\bar{\mu}(\bar{p}) \tag{34}$$

as shown in Fig. 10. Finally, the RC output becomes

$$f_{\mathrm{RC}}(k) = \bar{\mu}_{\mathrm{post}}(p). \tag{35}$$

The spatial forward shift allows to compensate for the non-causal part of $L$ essentially; observations at sample $k$ are stored in the GP at position $p(k-n_l)$ and used to perform inference at the test point $p(k)$. In contrast to traditional RC, the spatial preview $\delta_p$ is not limited by the buffer size $N$. The GP model can be evaluated at an arbitrary position in the future, potentially resulting in a posterior mean equal to the prior mean (zero by default) if there is little correlation between the test point and the training data.

Finally, the spatial RC with and without spatial preview is identical; therefore, the stability condition in Theorem 1 remains valid under Assumption 2 where, now, the preview is included.

### *E. Periodic Kernel Design*

What remains is to select a suitable kernel function that represents the spatial disturbance prior knowledge for GP regression. The kernel function imposes prior knowledge on the disturbance function $\bar{d}$, as shown in (15) and (17).

According to Definition 2, the underlying spatial disturbance is smooth and periodic with period $p_{\mathrm{per}}$. Hence, a periodic kernel that reflects the class of smooth and periodic functions is suitable. Note that traditional kernels as often used in system identification approaches (see [35]–[38]) do not impose these type of priors.

The periodic kernel function is given by

$$\kappa\big(p,\,p'\big) = \sigma_f^2 \exp\left(\frac{-2\sin^2\left(\frac{\pi\left(p-p'\right)}{\lambda}\right)}{l^2}\right) \tag{36}$$

with hyperparameters $\sigma_f$ being a scaling, $l$ the length scale, and $\lambda$ the period, see Fig. 11 for an example. The hyperparameter $\lambda$ is known in RC and equal to the period $p_{\mathrm{per}}$; the additional parameters $\sigma_f$ and $l$ have to be tuned. Tuning can be done based on measurement data, i.e., increasing the smoothness $l$ yields more correlation between data points resulting in a smoother estimate, and $\sigma_f$ is used to express prior on the amplitude of the disturbance. From an engineering point of view, choosing $l$ large may be desired to take high-frequency modeling errors into account, see Remark 10.

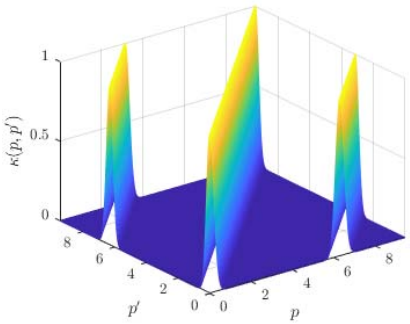


Fig. 11. Example of a periodic kernel with hyperparameters $\lambda = 2\pi$, $l = 0.2$, and $\sigma_f = 1$.

Including periodicity and smoothness as prior helps to extrapolate beyond the currently known training data for fast learning. Moreover, note that the period $\lambda$ may be a real number in contrast to traditional time-domain memory loops where the periodicity is always an integer multiple of the sample time. This is an advantage of the GP-based approach, i.e., it also allows to suppress disturbances with noninteger period times, which is not possible with traditional RC [7], or requires interpolation in other spatial approaches [18].

*Remark 10:* The high-frequency content in the GP-RC output $f_{\mathrm{RC}}$ is limited due to the smoothness, i.e., fulfilling a similar role as a robustness filter in traditional RC.

### *F. GP-RC Procedure*

Based on the above, a procedure to implement GP-RC is outlined in the following.

Procedure 2: Position-Domain RC Using Full GP

**(A) Initialization and prior**

1) Design $L$ through Procedure 1.
2) Select hyperparameters $\sigma_f$, $\lambda$, $l$ and noise variance $\sigma_n$.
3) Set counters $k = i = 1$.

**(B) At every sample** $k$

- Obtain $\ell_d(k)$ by filtering $e(k)$ with $L_c$.

1) **if** $k > n_l$
   - Add the $i^{\mathrm{th}}$ training observation to $\mathcal{D}(X, \bar{\mathbf{y}}_d)$

$$X(i) = \bar{p} \tag{37}$$

$$\bar{y}_d(i) = \ell_d(k) + q^{-n_l} f_{\mathrm{RC}}(k) \tag{38}$$

   increase counter $i = i + 1$.
   - Train the GP with data $\mathcal{D}$, i.e., compute $\alpha$ in (13).
   - Prediction at test position $p^*$ by computing (43b) with precomputed $\alpha$.
   - Set $f_{\mathrm{RC}}(k) = \bar{\mu}_{\mathrm{post}}(p)$.
2) **else**
   - No training data available yet, i.e., $f_{\mathrm{RC}}(k) = 0$.
3) **end** set $k = k + 1$.

An interesting observation is that the GP-RC output is nonzero from the $n_l^{\text{th}}$ sample onward, in contrast to classical RC, where a delay of one period is required before the disturbance can be compensated. This is caused by the extrapolation capabilities of the GP, which allows predicting the disturbance model $n_l$ samples in the future based on the current observations, leading to much faster learning in the first period, in contrast to existing spatial approaches.

*Remark 11:* In this procedure, all samples are included in the GP training dataset $\mathcal{D}$; furthermore, at every sample, the GP is trained with the available data. Due to high sample rates in combination with smoothness and periodicity in the kernel, there may be a significant amount of redundant data available. Hence, the above procedure can significantly speed up by only including a subset of the samples in the training set; in addition, training of the GP can be done at a lower rate than the predictions.

## V. Computationally Efficient GP-Based RC

The computational complexity of GP regression hampers its practical implementation in RC, i.e., the computation time of a full GP scales cubically with the number of data points $\mathcal{O}(N^3)$. In this section, a sparse GP approximation based on inducing points is presented, which, in combination with a periodic kernel, is highly suitable for the implementation in spatial GP-based RC. Also, optimization of hyperparameters and inducing points for sparse GP regression is presented [39].

### A. Efficient GP for RC

The cubic complexity is caused by the inversion of the $N \times N$ matrix $(K_{XX} + \sigma_n^2 I_N)$, which appears in the posterior distribution (10). Several methods have been investigated to address the complexity requirement, including discarding data [40], full GP approximations [41], and prior approximations, such as the fully independent training conditional (FITC) [39]. In view of RC, the FITC approximation is particularly suitable since it relies on a set of $M \ll N$ inducing points $X_m$. Due to the periodic kernel, the inducing point positions can be concentrated within one spatial period, i.e., a small number of inducing points are sufficient. This reduces the computation complexity to $\mathcal{O}(M^2 N)$.

### B. Sparse GP Regression for RC

To outline the sparse GP approximation, first, define a sparse training set $\bar{\mathcal{D}}(X_m, \bar{\mathbf{f}}_m)$ consisting of $M$ predefined inducing points $X_m$ with function values $\bar{\mathbf{f}}_m = \bar{f}(X_m)$. This set is used to support the full training dataset $\mathcal{D}$.

Next, the joint prior (9) is written explicitly as function of the inducing points function values $\bar{\mathbf{f}}_m$, that is,

$$p(\bar{y}_d, \bar{f}_*) = \int p(\bar{y}_d, \bar{f}_* | \bar{\mathbf{f}}_m) p(\bar{\mathbf{f}}_m)\, \mathrm{d}\bar{\mathbf{f}}_m \tag{39}$$

where $\bar{f}_m$ is marginalized out, see [23]. Next, a key assumption in many GP approximations is that $\bar{\mathbf{y}}_d$ and $\bar{f}_*$ are conditionally independent and only connected through the inducing points $\bar{\mathbf{f}}_m$, i.e., $\bar{\mathbf{f}} \perp \bar{f}_* | \bar{\mathbf{f}}_m$. As a result, the joint probability distribution of two function values is equal to the product of the individual probabilities, i.e., the joint prior (39) now becomes

$$q(\bar{\mathbf{y}}_d, \bar{f}_*) = \int q(\bar{\mathbf{y}}_d | \bar{\mathbf{f}}_m) q(\bar{f}_* | \bar{\mathbf{f}}_m) p(\bar{\mathbf{f}}_m) \mathrm{d}\bar{\mathbf{f}}_m \tag{40}$$

where $q(\bar{\mathbf{y}}_d | \bar{\mathbf{f}}_m)$ and $q(\bar{f}_* | \bar{\mathbf{f}}_m)$ are referred to as the training and test conditionals, respectively, and the prior on the inducing points remains exact, i.e., $p(\bar{\mathbf{f}}_m) = \mathcal{N}(0, K_{UU})$, see [39]–[41] for a derivation. Here, the matrix $K_{UU} \in \mathbb{R}^{M \times M}$ is the covariance function $\kappa$ evaluated at all combinations of inducing points $X_m$. The FITC algorithm approximates specifically the training conditional as follows:

$$q(\bar{y}_d | \bar{f}_m) = \mathcal{N}\left(K_{XU} K_{UU}^{-1} \bar{f}_m, \Lambda + \sigma_n^2 I_N\right) \tag{41}$$

with $\Lambda = \text{diag}[K_{XX} - Q_{XX}]$ being a diagonal approximation of the kernel matrix $K_{XX}$ and $Q_{AB} = K_{AU} K_{UU}^{-1} K_{UB}$, whereas the test conditional remains exact. The posterior distribution for a new test point $p_*$ is given by

$$p(\bar{f}_* | p_*, X_m, \mathcal{D}) = \mathcal{N}\left(\mu_{FITC}, P_{\text{post}}^{\text{FITC}}\right) \tag{42}$$

where the mean and covariance are

$$\mu_{\text{FITC}} = K_{*U} \Sigma^{-1} K_{UX} \left(\Lambda + \sigma_n^2 I_N\right)^{-1} \bar{\mathbf{y}}_d \tag{43a}$$

$$P_{\text{post}}^{\text{FITC}} = K_{**} - K_{U*}^{\top} \left(K_{UU}^{-1} - \Sigma^{-1}\right) K_{U*} \tag{43b}$$

$$\Sigma = K_{UU} + K_{UX} \left(\Lambda + \sigma_n^2 I_N\right)^{-1} K_{XU}. \tag{43c}$$

In line with the full GP, a single test point in the FITC algorithm is also of the form (12) with

$$\alpha_{\text{FITC}} = \Sigma^{-1} K_{UX} \left(\Lambda + \sigma_n^2 I_N\right)^{-1} \bar{\mathbf{y}}_d. \tag{44}$$

In the posterior distribution (43a), the inversion of the matrix $\Sigma$ is now $M \times M$ instead of $N \times N$. In addition, the inversion of $\Lambda \in \mathbb{R}^{N \times N}$ is inexpensive since it is a diagonal matrix. This reduces the computational effort to $\mathcal{O}(M^2 N)$.

*Remark 12:* The sparse GP scales linearly with the number of data points $N$, instead of cubically for the full GP, significantly reducing the computational complexity. In a practical application, $N$ should be limited, e.g., by using a subset of the data containing the most recent $N$ observations, which is an alternative simple sparse GP approximation. Note that many results for efficient GP computation [40] can be used in the context of spatial GP-based RC.

The sparse FITC GP is implemented in Procedure 1 by additionally selecting the number of inducing points $M$ and their positions $X_m$ in step (A). Then, in step (B), replace $\alpha$ with $\alpha_{\text{FITC}}$ to compute the sparse mean with (43b).

### C. Hyperparameters and Inducing Points Optimization

The location of the inducing points, as well as the hyperparameters, can be chosen manually. Alternatively, these can be optimize by maximizing the log marginal likelihood

$$\log p(\bar{\mathbf{y}}_d | X_m) = -\frac{1}{2} \log |Q| - \frac{1}{2} \bar{\mathbf{y}}_d^{\top} (Q)^{-1} \bar{\mathbf{y}}_d \tag{45}$$

where $Q = K_{UU} + \Lambda$ with respect to the inducing points $X_m$ and optionally hyperparameters. This can be performed offline to initialize the inducing point locations and hyperparameters in one step, see [39].

## VI. Simulation Case Study

In this section, a simulation case study is carried out to show that spatial RC, with full and sparse GP, can effectively compensate for spatially periodic disturbances that are nonperiodic in the time domain. To evaluate the obtainable performance, a comparison is made with traditional RC, which can obviously not deal with the velocity variations.

### A. System and Disturbance

The setting in Fig. 3 is considered, where $P$ replicates the dominant dynamics in the $\phi_z$ direction of the industrial substrate carrier. It is modeled as a second order mass-spring-damper system

$$P(s) = \frac{1}{Js^2 + ds + k} \tag{46}$$

with inertia $J = 1$ kg·m$^2$, damping $d = 1$ Nm/s, and stiffness $k = 10^4$ N/m and discretized by zero-order-hold with sampling frequency $f_s = 1000$ Hz. A stabilizing feedback controller is designed as given by

$$C(z) = \frac{67539(z+1)(z-0.9196)}{(z-0.4524)(z+0.1137)}. \tag{47}$$

Furthermore, a spatially periodic disturbance $d(t)$ acts on the system, collocated with the control input torque $T$. The torque $T$ in the substrate carrier corresponds to the sum of the lateral forces generated by the roller segments, multiplied by the length from the roller to the PoI on the belt, i.e., the effective torque at the PoI in $r_z$. The spatial disturbance is given by

$$\bar{d}(p) = 1.5\sin(p) + 0.8\sin(3p) + 0.6\sin(9p) + 0.4\sin(18p) + 0.2\sin(27p)$$

where the velocity of $p(t)$ is $2\pi$ rad/s at the start of the simulation, resulting in a time-domain disturbance period of exactly 1 s. In the second phase, the velocity varies over time, resulting in a nonperiodic time-domain disturbance. Finally, the velocity is constant at 10.9 rad/s such that the time-domain disturbance is periodic, and the period is not an integer multiple of 1 s. The time-domain disturbance is shown in Fig. 4. The reference signal is set to zero, i.e., a pure disturbance attenuation problem is considered.

### B. Design of GP-RC, Sparse GP-RC, and Temporal RC

*1) Spatial GP RC:* The learning filter is designed according to Procedure 1, resulting in $n_l = 1$ samples of preview. The marginal likelihood (45) is optimized with respect to the hyperparameters to obtain $\sigma_f = 2.7$, $l = 0.13$, and $p_{\text{per}} = 2\pi$. The noise variance used throughout the simulation is $\sigma_n = 10^{-4}$, which is additive white noise on the disturbance. To reduce the computational load, every tenth sample is added to $\mathcal{D}$, and subsequently, the GP is trained.

*2) Sparse Spatial GP RC:* For the sparse GP RC, the number of inducing points $M$ is set to 100, which are equidistantly distributed in the range $[0, p_{\text{per}})$. The same hyperparameters are used as in the full GP case; every tenth sample is used to train the sparse GP.

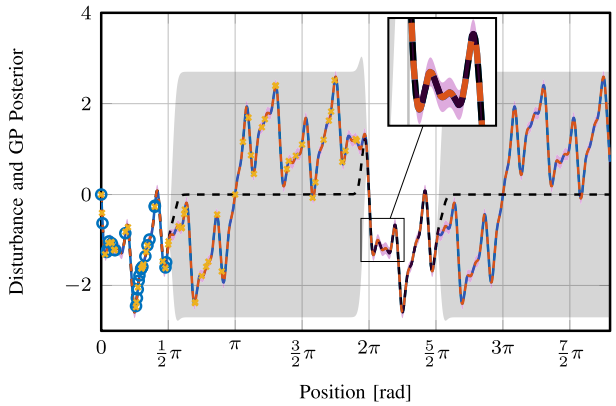


Fig. 12. True spatial disturbance (—) and the estimated GP posterior distribution evaluated on position $[0, 4\pi]$ depicted for two training sets. First, training points (○) in the interval $[0, (\pi/2)]$ resulting in the mean (- - -) and the standard deviation (▒), which shows that the estimate extrapolates beyond the data due to periodicity, and the mean becomes zero where no information is available. Second, the full training set with $N = 1600$ points (×) on $[0, 2\pi]$ resulting in the mean (- - -) and variance (▒), showing that a good estimate of the disturbance is obtained in the entire range $[0, 4\pi]$ based on data in the first spatial period.

*3) Temporal RC:* For traditional RC, the learning filter is designed as the inverse of the complementary sensitivity function $T = 1 - S$, see [6]. The memory loop size is equal to the disturbance period at the start of the simulation, i.e., $N_{\text{RC}} = 1000$ samples. Traditional RC provides a measure for the obtainable performance in the first constant velocity phase.

### C. Results: GP-Based Disturbance Model

First, the obtained GP-based disturbance model that is captured in the spatial memory loop is further investigated. In Fig. 12, a snapshot of the GP-based memory loop is shown. The actual disturbance and GP posterior are given for two training cases: 1) where $N = 1600$ observations in the interval $[0, \pi/2]$ are used and 2) where only the training values on $[0, 2\pi]$ are used to train the GP. As a result, a good estimate of the disturbance is obtained over the entire range. Due to the periodicity and smoothness in the kernel, i.e., a sample at the current position $p$ is connected to its neighboring samples and samples at $p - n \cdot p_{\text{per}}$, the identified disturbance model clearly extends beyond the training points. Where no data are present, the GP mean tends to zero, and the variance grows.

To investigate the model quality of the sparse GP, Fig. 13 shows the root mean squared (rms) estimation error for the full GP and the sparse GP as a function of the number of inducing points $M$. The inducing points are equidistantly distributed on $(0, 2\pi]$. This shows that, from $M = 80$ onward, the sparse GP obtains an equivalently good estimate of the disturbance compared to the full GP with 1600 training points. This indicates that a small number of inducing points, in combination with the periodicity of the kernel, are sufficient to support the full dataset.

### D. Results: Performance Comparison

The spatial RC and traditional RC error responses are shown in Fig. 14, where the gray area indicates where the disturbance

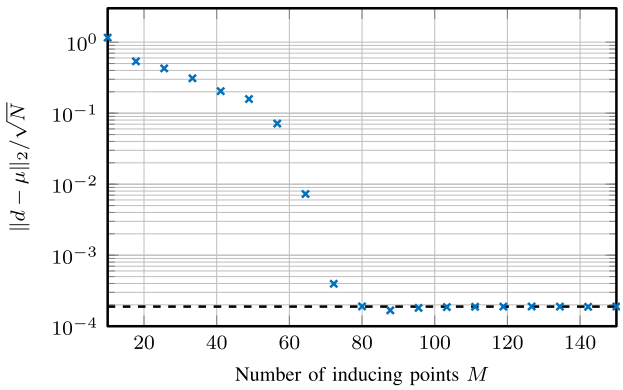


Fig. 13. RMS estimation error of the full GP (- - -) with $N = 1600$ training points and the sparse GP rms estimation error (×) as function of the number of inducing points $M$, indicating that a small number ($M \approx 80 \ll N$) of equidistantly distributed inducing points in $[0, 2\pi]$ sufficiently support the full dataset.

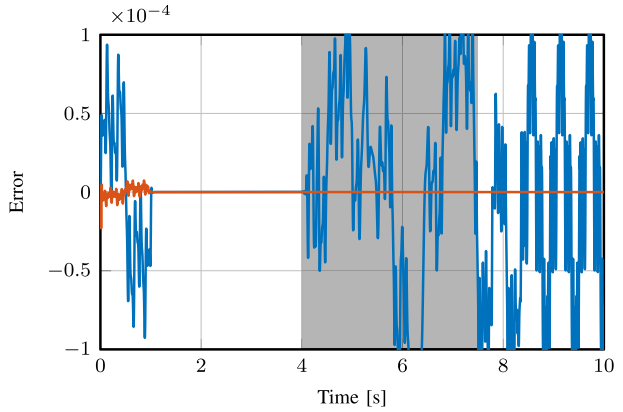


Fig. 14. Simulated positioning error for traditional RC (—) and the spatial RC (—). Spatial RC is invariant under the velocity change, as indicated by (—), whereas traditional RC leads to performance degradation in case of velocity change.

is nonperiodic in time. In addition, the two-norm of the error for each spatial period $j$ normalized by its length $N_j$ is given in Fig. 15. The following observations are made.

1) The time response shows that spatial RC (—) significantly reduces the error from sample $n_l = 1$ onward, i.e., suppression is obtained in the first period, after which the error is reduced when more training data are gathered. Note that traditional RC does require one full period before compensation.
2) The error norm shows that spatial RC (×) and sparse spatial RC (×) are not influenced by the change in velocity and maintain to have good performance. Of course, traditional RC performance (×) decreases due to the inadequate buffer size. The temporal buffer size can be adapted for each velocity change; however, this introduces additional transients, requires interpolation to use the learned compensation signal at the other velocity, and cannot cope with noninteger buffer sizes and continuous velocity changes.
3) The performance of spatial RC, utilizing only 10% of the data, is equal to the traditional RC performance in the first constant velocity part. Hence, it uses the data very efficiently due to the suitable prior knowledge.

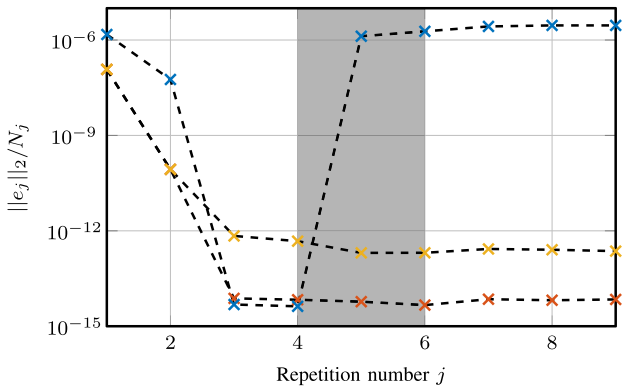


Fig. 15. Simulated error 2-norm normalized with the period length $N_j$ as function of the repetition number $j$ for traditional RC (×), spatial RC with full GP (×), and with sparse GP for $M = 100$ inducing points (×). This indicates that spatial GP-RC obtains highly similar performance compared to traditional RC while being able to maintain performance during velocity changes, indicated by (×) as in Fig. 4, where traditional RC degrades significantly.

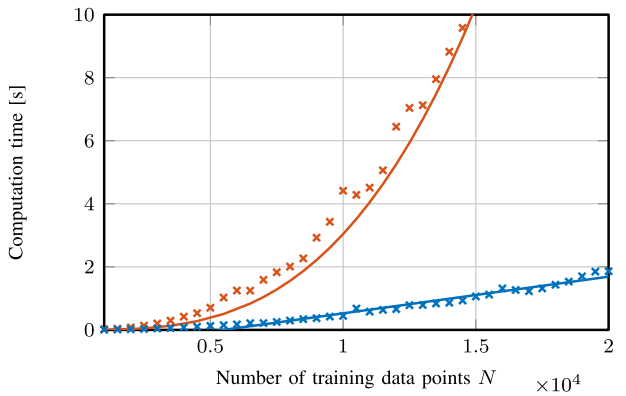


Fig. 16. Computation time of the full GP with a cubic fit $\mathcal{O}(N^3)$ (—) and a sparse FITC GP for $M = 50$ inducing points with a linear fit $\mathcal{O}(M^2N)$ (—) as function of the training data $N$. The implementation for this example is not yet optimized; therefore, a large buffer size is chosen to avoid the overhead of the implementation, yielding relatively large computation times.

Indeed, it follows that spatial RC is not influenced by the changing disturbance. The sparse GP approaches the full GP performance relatively well, while, at the same time, reducing the computational load.

### *E. Results: Computation Time*

To show the benefit of the FITC GP in contrast to a full GP in terms of computation time, the computation time is measured as a function of the number of training data points, see Fig. 16. The number of inducing points for the FITC GP is $M = 50$ points. It can be seen that, indeed, the full GP computational complexity grows with $\mathcal{O}(N^3)$, as indicated by the fit. The sparse GP with computational complexity $\mathcal{O}(M^2N)$ scales linearly with $N$, as shown by the linear fit. Furthermore, it can be observed that the computation gain starts to become significant for $N > 500$ data points, which is easily reached in mechatronic applications. Note that these results are obtained on a regular PC for illustration purposes, and efficient employment on dedicated hardware can significantly reduce the computation time.

## VII. Application to a Substrate Carrier System

In this section, spatial GP-RC is applied to the industrial substrate carrier in Fig. 1 introduced in Section II-A, which is subject to spatially periodic roller disturbances. The aim is to validate that spatial RC can reject the spatial disturbance-invariant under variations in the roller velocity.

### A. Substrate Carrier Setup

As mentioned in Section II-A, the control goal of the substrate carrier is to accurately position a medium, e.g., paper or plastic, which is fixated to the steel belt by means of a vacuum. The steel belt is steered using two segmented rollers that control the $r_x$-, $r_y$-, and $\phi_z$-directions of the PoI, see Fig. 2. In this experiment, the aim is to keep $\phi_z$ equal to zero; hence, this is a pure disturbance rejection problem.

### B. Spatial Disturbances in the Substrate Carrier

Due to imperfections in the rollers and segments, a spatially periodic disturbance appears in the error that repeats every roller rotation. To show this, a single experiment is performed, which consists of three parts as follows:

1) constant roller velocity of 9 [rad/s];
2) deceleration of 5 [rad/s$^2$] for 0.8 s, which is approximately two roller rotations;
3) constant roller velocity of 5 [rad/s].

The error is measured with a baseline proportional derivative (PD) controller, see (—) in Fig. 17, where a periodic component is visible, which changes frequency during the different parts. The power spectral density (PSD) and the cumulative power spectrum (CPS) of the error without offset with velocity 9 [rad/s] (—) and velocity 5 [rad/s] (- -) are given in Fig. 19 as function of the spatial frequency [1/rotation], i.e., the frequency [Hz] scaled by the rotational velocity of the rollers [rotation/s]. This shows that, especially at 1 and 3 [1/rotation], the error contains clear contributions that are repeating in the roller-position domain-invariant under velocity variations. The first part corresponds to one roller revolution, and the third harmonic is most likely induced by having three segments in each roller that does not perfectly align.

During the normal operation, the belt runs at several operating velocities; this leads to a situation where the disturbance becomes nonperiodic in time. Traditional RC can only attenuate this disturbance for constant velocities and may even amplify the disturbance when the velocity changes. In the remainder of this section, it is shown that spatial GP-based RC attenuates the spatial disturbance-invariant under velocity variations.

### C. Spatial GP-RC Design

To implement spatial RC, a parametric model of the substrate carrier has been identified, and the learning filter is constructed according to Procedure 1. The zero phase error tracking control (ZPETC) algorithm is used to obtain a noncausal but stable learning filter with $n_l = 2$ samples of preview.

The periodic kernel hyperparameters are tuned such that the prior represents the actual spatial disturbance function. Tuning

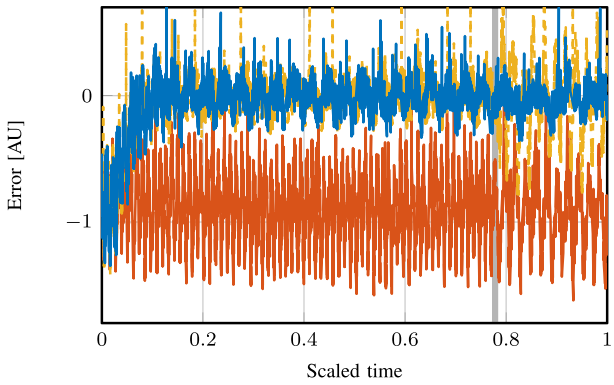


Fig. 17. Error response as function of scaled time for the PD controller (—), traditional RC (- -), and spatial RC (—). The gray area (▒) indicates where the roller velocity is changing. This shows that the periodic components and the dc offset are removed by the spatial RC controller independent of velocity changes. Here, AU is the arbitrary unit.

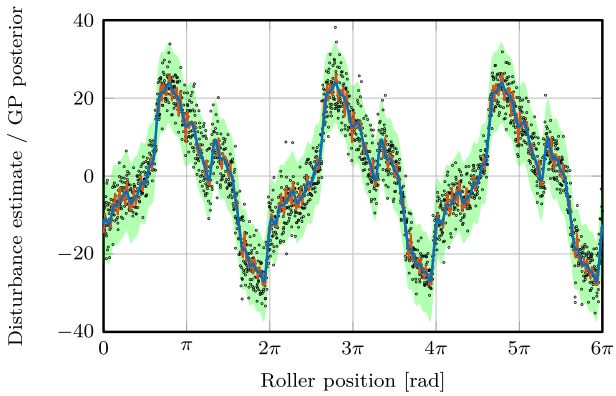


Fig. 18. Experimental error data filtered with the learning filter $L$ (●) as function of the roller position used as an estimate of the disturbance. The GP standard deviation $6\sigma$ (▒) and mean (—) with hyperparameters $\sigma_f = 27$, $\sigma_n = 1.7$, $\lambda = 2\pi$, and $l = 0.2$. To illustrate the effect of varying the length scale, the GP posterior mean is also computed for $l = 0.05$ (—), resulting in a much more erratic function.

is performed with the measured error that is filtered by the learning filter; this results in an estimate of the spatial disturbance, as shown in Fig. 18. Representing the filtered error as a function of position enables to tune the hyperparameters as follows.

1) The spatial disturbance period is known and equal to one roller rotation, i.e., $\lambda = 2\pi$.
2) The gain $\sigma_f = 27$ is an estimate of the deviation around the mean of the disturbance estimate.
3) The GP estimate given the data is shown with $l = 0.05$ (—) and $l = 0.2$ (—) in Fig. 18; this shows that a shorter length scale yields more high-frequency content in the estimate. To be more robust to noise and high-frequency modeling errors, a length scale $l = 0.2$ is preferred.
4) $\sigma_n = 1.7$ is an estimate of the standard deviation of the noise, yielding the confidence bound (▒) in Fig. 18.

Note that a very short length scale $l$ allows for more high-frequency components in the GP-RC output, i.e., the length scale acts as a low-pass filter similar to a robustness filter in traditional RC. Because the smoothness $l$ influences the GP input–output behavior, there is a direct connection

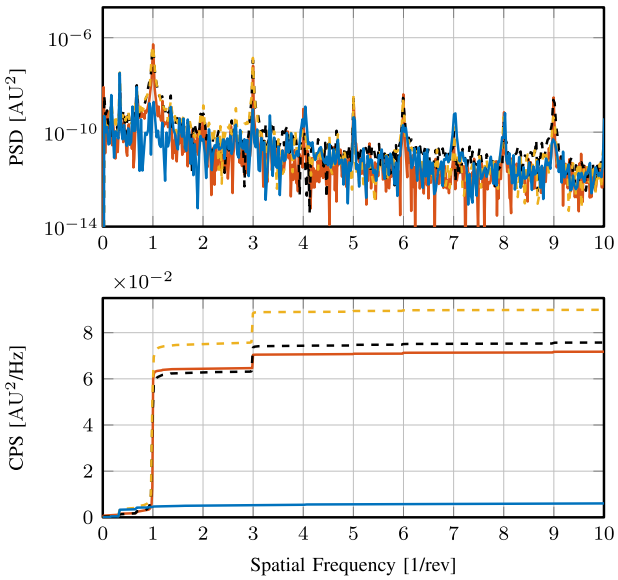


Fig. 19. PSD (top plot) and CPS (bottom plot) as a function of the spatial frequency for the error with PD controller at two different velocities (- - -) (9 [rad/s]) and (——) (5 [rad/s]). The disturbance is most prominent at the fundamental frequency [1/rev] and its second harmonic [3/rev] where the CPS shows significant increases in power independent of the velocity change. The error with spatial RC (——) and classical RC (- - -) after deceleration is also shown, indicating that spatial RC is not affected by the velocity change, whereas classical RC performance is severely degraded. Here, AU is the arbitrary unit.

between smoothness and stability through Assumption 2 and Theorem 1. Finally, to further reduce computational load, the data used to train a full GP are limited to 50 samples that are distributed equidistantly over the last spatial period. The Sherman–Morrison update is used for GP regression to further improve the computational load for practical implementation, see Remark 4. The experimental setup runs at a sampling frequency of 4000 Hz.

### D. Implementation Aspects

In the general case for motion systems, the gain of $(PS)^{-1}$ is large for high frequencies, leading to the amplification of high-frequency noise by the learning filter. Therefore, as a pragmatic solution, an additional zero-phase low-pass filter is placed in series with the learning filter that mitigates the effect of noise in $\ell$. This does result in high-frequent modeling errors, but it improves the convergence of the disturbance model in the relevant frequency range, i.e., where a good disturbance model is most relevant. Note that the low-pass filter is designed such that the stability condition in Theorem 1 is still satisfied.

In addition, traditional RC is implemented as a comparison, with a buffer size that is equal to the disturbance period in the first constant velocity part of the experiment.

### E. Results

The experiment outlined in Section VII-B is carried out to analyze the performance of spatial RC in comparison to the baseline PD controller and the traditional RC. The error responses are also shown in Fig. 17. The converged error after deceleration is analyzed in the spatial frequency domain, i.e., Fig. 19 shows in addition to the baseline PD error (——), the spatial RC error (——), and the traditional RC error (- - -), the corresponding CPS are also given. Note that a significant offset is present in the PD controlled error and, this, is removed from the data before computing the PSD and CPS such that the harmonics are better visible.

To analyze performance during velocity change, i.e., where the disturbance is nonperiodic, the two-norm of the error for each rotation $j$, normalized by the period length $N_j$, is given in Fig. 20. Note that all experimental results have been normalized for confidentially, and the unit is denoted by the arbitrary unit (AU). The following observations can be made.

1) The dominant components in the error CPS, indicated by the increases in power at 1 and 3 [1/rotation] in the CPS in (——) in Fig. 19, are completely suppressed by the spatial RC (——). In addition, Fig. 17 shows that an offset is present in the PD controlled error (——) due to the lack of an integrator, which is learned and compensated for by the spatial RC (——). The overall performance improvement, including the offset, is a factor 12 on the rms error, where a factor 3.5 is attributed to the suppression of the harmonic components.
2) The PSD in Fig. 19 shows that the error mainly at the fundamental frequency 1 [1/rotation] and the second harmonic 3 [1/rotation] are significantly reduced by the spatial RC (——) compared to the baseline error (——) that shows significant increases in the CPS. Note that only the first and third contribute significantly to the CPS.
3) The error-norm in Fig. 20 shows that spatial RC (×) and traditional RC (×) have similar performance in the first constant velocity part, i.e., where the traditional RC buffer size ($N = 2787$) is compatible with the disturbance period, both resulting in an overall performance improvement of approximately a factor 12. This shows that spatial GP RC, with a low number of training points ($N = 50$), obtains equal performance for constant velocities.
4) During deceleration, indicated by the gray area in Figs. 20 and 21, the spatial RC error (×) is unaffected. This implies that the approach is robust with respect to velocity variations by using the spatial model. As a comparison, traditional RC (×) shows a performance degradation if the disturbance frequency changes. Due to inadequate buffer size, the traditional RC is not able to converge to a small error in the second constant velocity part.
5) The PSD in Fig. 19 shows that a periodic component at spatial frequency 0.3 [1/rev] is present, which most likely originates from a belt disturbance, which is amplified by the GP-RC (——). This can be explained since, in general, nonperiodic components will be amplified by any RC as they do not align with the buffer size, see [42].

From these observations, it follows that, indeed, the spatial GP-RC approach suppresses the spatial disturbance and is

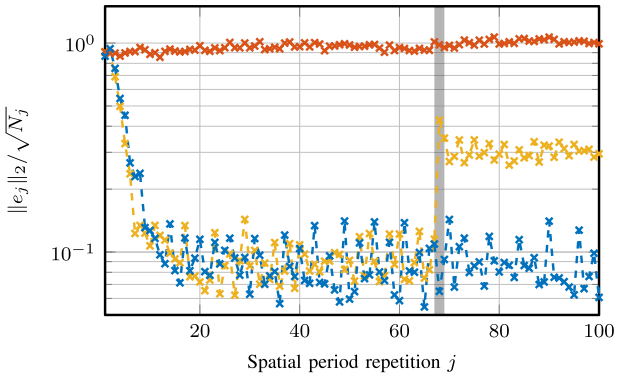


Fig. 20. Two-norm of the error scaled with $\sqrt{N}$ for the PD controller (×), spatial RC (×), and traditional RC (×). The gray area (▒) indicates where the velocity changes, showing the benefit of spatial RC where the performance is unaffected, whereas the traditional RC performance degrades due to inadequate buffer size after the velocity changes.

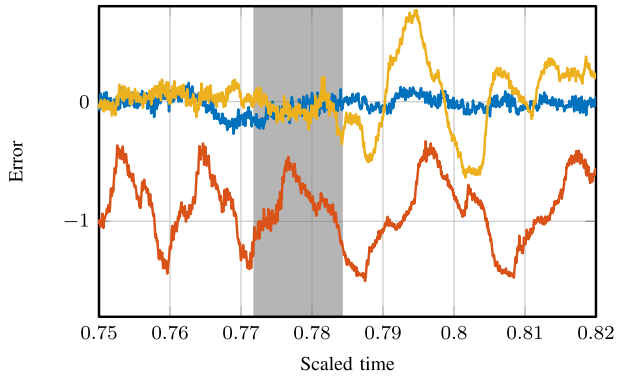


Fig. 21. Zoomed-in-view of the error as a function of time with PD only (—), traditional RC (—), and GP-based spatial RC (—) during the velocity change phase. This shows that indeed GP-based RC maintains good performance during and after the velocity changes. Traditional RC obviously degrades the performance due to the changing disturbance frequency that does not comply with the buffer size.

not affected by changes in velocity, resulting in a major performance increase of a factor of 12 of the rms error.

## VIII. Conclusion

Position-domain disturbances that appear nonperiodic in the time domain but are repeating in the position domain can be completely rejected by spatial RC, where a key enabler is the introduction of a GP-based spatial memory loop, as presented in this article. The approach uses a memory loop in the position domain together with a suitable learning filter to learn a spatial disturbance model. In contrast to existing RC approaches, a spatial memory loop is established by means of a GP with a suitable periodic kernel, thereby efficiently dealing with nonequidistant observations. The disturbance is modeled as a stochastic process, i.e., a collection of random variables in space, which is estimated from data and suitable prior knowledge. The resulting distribution, in particular, its mean, is a continuous function that is utilized in a spatial memory loop. The approach is validated in simulation and on an industrial substrate carrier. Experimental results show a performance improvement of a factor of 12 compared to currently implemented PD control, i.e., automatically learning and suppressing roller disturbances for arbitrary operating velocity in the industrial substrate carrier.

Ongoing work focuses on utilizing the full GP posterior distribution, i.e., also including the GP variance as a confidence measure on the disturbance model, see [43], which can act as a learning gain similar to traditional RC. In addition, extend the approach to cover multiperiod spatial disturbances with different spatial periods [11], [15], e.g., to suppress roller and belt disturbances that have different (spatial) periods in the considered example or to automatically learn commutation functions for brushless motors that minimize torque ripples.

## Acknowledgment

The authors would like to thank Tom van de Laar, Lennart Blanken, Ibrahim Açan, and Joep Kooijman for their contributions to this work and the collaboration with Sioux Technologies.

## References

[1] X. Huo, M. Wang, K.-Z. Liu, and X. Tong, "Attenuation of position-dependent periodic disturbance for rotary machines by improved spatial repetitive control with frequency alignment," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 1, pp. 339–348, Feb. 2020.

[2] C.-L. Chen and G. T.-C. Chiu, "Spatially periodic disturbance rejection with spatially sampled robust repetitive control," *J. Dyn. Syst., Meas., Control*, vol. 130, no. 2, pp. 021002-1–021002-11, Mar. 2008.

[3] P. Y. Li, "Prototype angle-domain repetitive control-affine parameterization approach," *J. Dyn. Syst., Meas., Control*, vol. 137, no. 12, pp. 121009-1–121009-9, Dec. 2015.

[4] Z. Sun, "Tracking or rejecting rotational-angle dependent signals using time varying repetitive control," in *Proc. Amer. Control Conf.*, vol. 1, 2004, pp. 144–149.

[5] B. A. Francis and W. M. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, no. 5, pp. 457–465, 1976.

[6] R. W. Longman, "On the theory and design of linear repetitive control systems," *Eur. J. Control*, vol. 16, no. 5, pp. 447–496, 2010.

[7] S. Hara, Y. Yamamoto, T. Omata, and M. Nakano, "Repetitive control system: A new type servo system for periodic exogenous signals," *IEEE Trans. Autom. Control*, vol. AC-33, no. 7, pp. 659–668, Jul. 1988.

[8] M. Tomizuka, "Dealing with periodic disturbances in controls of mechanical systems," *Annu. Rev. Control*, vol. 32, no. 2, pp. 193–199, Dec. 2008.

[9] M. Steinbuch, S. Weiland, and T. Singh, "Design of noise and period-time robust high-order repetitive control, with application to optical storage," *Automatica*, vol. 43, no. 12, pp. 2086–2095, Dec. 2007.

[10] G. Pipeleers, B. Demeulenaere, J. De Schutter, and J. Swevers, "Robust high-order repetitive control: Optimal performance trade-offs," *Automatica*, vol. 44, no. 10, pp. 2628–2634, 2008.

[11] L. Blanken, P. Bevers, S. Koekebakker, and T. Oomen, "Sequential multiperiod repetitive control design with application to industrial wide-format printing," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 2, pp. 770–778, Apr. 2020.

[12] G. Hillerstrom, "Adaptive suppression of vibrations—A repetitive control approach," *IEEE Trans. Control Syst. Technol.*, vol. 4, no. 1, pp. 72–78, Jan. 1996.

[13] E. Kurniawan, Z. Cao, and Z. Man, "Design of robust repetitive control with time-varying sampling periods," *IEEE Trans. Ind. Electron.*, vol. 61, no. 6, pp. 2834–2841, Jun. 2014.

[14] T. J. Manayathara, T.-C. Tsao, and J. Bentsman, "Rejection of unknown periodic load disturbances in continuous steel casting process using learning repetitive control approach," *IEEE Trans. Control Syst. Technol.*, vol. 4, no. 3, pp. 259–265, May 1996.

[15] X. Chen and M. Tomizuka, "A minimum parameter adaptive approach for rejecting multiple narrow-band disturbances with application to hard disk drives," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 2, pp. 408–415, Mar. 2012.

[16] C.-L. Chen and Y.-H. Yang, "Adaptive repetitive control for uncertain variable-speed rotational motion systems subject to spatially periodic disturbances," in *Proc. Amer. Control Conf.*, Jul. 2007, pp. 564–569.

[17] N. Mooren, G. Witvoet, I. Açan, J. Kooijman, and T. Oomen, "Suppressing position-dependent disturbances in repetitive control: With application to a substrate carrier system," in *Proc. IEEE 16th Int. Workshop Adv. Motion Control (AMC)*, Sep. 2020, pp. 331–336.

[18] W.-S. Yao, M.-C. Tsai, and Y. Yamamoto, "Implementation of repetitive controller for rejection of position-based periodic disturbances," *Control Eng. Pract.*, vol. 21, no. 9, pp. 1226–1237, 2013.
[19] Z. Cao and G. F. Ledwich, "Adaptive repetitive control to track variable periodic signals with fixed sampling rate," *IEEE/ASME Trans. Mechatronics*, vol. 7, no. 3, pp. 378–384, Sep. 2002.
[20] C. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning*, vol. 2, no. 3. Cambridge, MA, USA: MIT Press, 2006.
[21] K. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.
[22] H. Bijl, J.-W. van Wingerden, T. B. Schön, and M. Verhaegen, "Online sparse Gaussian process regression using FITC and PITC approximations," *IFAC-PapersOnLine*, vol. 48, no. 28, pp. 703–708, 2015.
[23] E. Snelson and Z. Ghahramani, "Local and global sparse Gaussian process approximations," in *Proc. Artif. Intell. Statist.*, 2007, pp. 524–531.
[24] N. Mooren, G. Witvoet, and T. Oomen, "Gaussian process repetitive control for suppressing spatial disturbances," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 1487–1492, 2020.
[25] A.-J. Beltman, R. Plak, J. Hazenberg, R. Pulles, B. Brals, and G. van Ooik, "Improved ink registration through advanced steel belt steering," in *Proc. NIP Digit. Fabr. Conf.*, 2012, pp. 222–225.
[26] A.-J. J. Beltman, A. Brals, R. Plak, and J. A. F. M. Simons, "Belt conveyor with an actuator for moving the belt in a lateral direction," U.S. Patent 8 807 331, Aug. 19, 2014.
[27] T. Oomen, "Control for precision mechatronics," in *Encyclopedia of Systems and Control*. Cham, Switzerland: Springer, 2019.
[28] Z. Sun, Z. Zhang, and T.-C. Tsao, "Trajectory tracking and disturbance rejection for linear time-varying systems: Input/output representation," *Syst. Control Lett.*, vol. 58, no. 6, pp. 452–460, Jun. 2009.
[29] H. Bijl, T. B. Schon, J.-W. Wingerden, and M. Verhaegen, "System identification through online sparse Gaussian process regression with input noise," *IFAC J. Syst. Control*, vol. 2, pp. 1–11, Dec. 2016.
[30] A. McHutchon and C. Rasmussen, "Gaussian process training with input noise," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 1341–1349.
[31] C. A. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties*. Philadelphia, PA, USA: SIAM, 2009.
[32] L. Blanken, T. Hazelaar, S. Koekebakker, and T. Oomen, "Multivariable repetitive control design framework applied to flatbed printing with continuous media flow," in *Proc. IEEE 56th Annu. Conf. Decis. Control (CDC)*, Dec. 2017, pp. 4727–4732.
[33] J. V. Zundert and T. Oomen, "On inversion-based approaches for feedforward and ILC," *IFAC Mechatronics*, vol. 50, pp. 282–291, Apr. 2018.
[34] M. Tomizuka, "Zero phase error tracking algorithm for digital control," *ASME Trans. J. Dyn. Syst. Meas. Control*, vol. 109, no. 1, pp. 65–68, Mar. 1987.
[35] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, Mar. 2014.
[36] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations and Gaussian processes—Revisited," *Automatica*, vol. 48, no. 8, pp. 1525–1535, Aug. 2012.
[37] C. Jidling *et al.*, "Probabilistic modelling and reconstruction of strain," *Nucl. Instrum. Methods Phys. Res. B, Beam Interact. Mater. At.*, vol. 436, pp. 141–155, Dec. 2018.
[38] R. Frigola, F. Lindsten, T. B. Schön, and C. E. Rasmussen, "Bayesian inference and learning in Gaussian process state-space models with particle MCMC," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3156–3164.
[39] E. Snelson and Z. Ghahramani, "Sparse Gaussian processes using pseudo-inputs," in *Proc. Adv. neural Inf. Process. Syst.*, 2006, pp. 1257–1264.
[40] J. Quiñonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate Gaussian process regression," *J. Mach. Learn. Res.*, vol. 6, pp. 1939–1959, Dec. 2005.
[41] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, "When Gaussian process meets big data: A review of scalable GPs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4405–4423, Nov. 2020.
[42] X. Chen and M. Tomizuka, "New repetitive control with improved steady-state performance and accelerated transient," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 2, pp. 664–675, Mar. 2014.
[43] S. Devasia, "Iterative machine learning for output tracking," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 2, pp. 516–526, Mar. 2017.

**Noud Mooren** (Member, IEEE) received the B.Sc. degree from the Fontys University of Applied Sciences, Eindhoven, The Netherlands, and the M.Sc. *(cum laude)* and Ph.D. degrees from the Eindhoven University of Technology, Eindhoven, in 2017 and 2022, respectively.

His research interest includes motion control and learning control techniques for applications in mechatronic systems.

Dr. Mooren was a recipient of the 2020 AMC Best Paper Award.

**Gert Witvoet** (Member, IEEE) received the M.Sc. *(cum laude)* and Ph.D. degrees from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2007 and 2011, respectively.

He is currently a Senior Dynamics and Control Specialist at the Netherlands Organisation for Applied Scientific Research (TNO), Delft, The Netherlands, and a part-time Assistant Professor with the Mechanical Engineering Department, Eindhoven University of Technology. His research interest includes the application of advanced motion control techniques on high-tech instruments and applications in the semiconductor, astronomy, and space markets.

Dr. Witvoet is a recipient of the Unilever Research Prize and several best master teacher awards.

**Tom Oomen** (Senior Member, IEEE) received the M.Sc. *(cum laude)* and Ph.D. degrees from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2005 and 2010, respectively.

He held visiting positions at KTH, Stockholm, Sweden, and The University of Newcastle, Callaghan, NSW, Australia. He is currently a Professor with the Department of Mechanical Engineering, Eindhoven University of Technology. He is also a part-time Full Professor with the Delft University of Technology, Delft, The Netherlands. His research interest includes data-driven modeling, learning, and control, with applications in precision mechatronics.

Dr. Oomen is a member of the Eindhoven Young Academy of Engineering. He was a recipient of the 7th Grand Nagamori Award, the Corus Young Talent Graduation Award, the IFAC 2019 TC 4.2 Mechatronics Young Research Award, the 2015 IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY Outstanding Paper Award, the 2017 IFAC Mechatronics Best Paper Award, the 2019 *IEEJ Journal of Industry Applications* Best Paper Award, and a Veni and Vidi Personal Grant. He is an Associate Editor of the IEEE CONTROL SYSTEMS LETTERS (L-CSS), *IFAC Mechatronics*, and IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY.