Gaussian Process Repetitive Control for Suppressing Spatial Disturbances

Noud Mooren* Gert Witvoet *,** Tom Oomen *

* Control Systems Technology, Department of Mechanical Engineering,
Eindhoven University of Technology, Eindhoven, The Netherlands
(n.f.m.mooren@tue.nl, g.witvoet@tue.nl, t.a.e.oomen@tue.nl)
** TNO Technical Sciences, Optomechatronics Department, Delft, The
Netherlands.

Abstract: Motion systems are often subject to disturbances such as cogging, commutation errors, and imbalances, that vary with velocity and appear periodic in time for constant operating velocities. The aim of this paper is to develop a repetitive controller (RC) for disturbances that are not periodic in the time domain, yet occur due to an identical position-domain disturbance. A new spatial RC framework is developed, allowing to attenuate disturbances that are periodic in the position domain but manifest a-periodic in the time domain. A Gaussian process (GP) based memory is employed with a suitable periodic kernel that can effectively deal with the intermittent observations inherent to the position domain. A mechatronic example confirms the potential of the method.

Keywords: Repetitive Control, Position Domain, Gaussian Process.

1. INTRODUCTION

Motion systems performing repeating tasks often generate disturbances that are periodic in the position domain. Consider for example; a rotational system with bearings or imperfections leading to an imbalance, a motor/gear system where a tooth profile induces a disturbance, or a linear motor with cogging (Ahn et al., 2005). These are typical examples where disturbances are inherently spatially periodic, i.e., depending on (angular-) position. For constant operating conditions, i.e., a constant angular velocity or a system performing repeating tasks, these disturbances appear periodic in the time domain. However, if the operating conditions vary over time, the disturbance will appear a-periodic in the time domain, see e.g., Chen and Chiu (2008), Li (2015).

Repetitive control (RC) attenuates periodic disturbances with a fixed and known period, see e.g., Hara et al. (1988); Longman (2010); Wang et al. (2009). For constant operating conditions, position-domain disturbances manifest periodically in the time domain. Hence, only then RC can suppress the disturbance by capturing an internal model in a time-domain memory loop. Subsequently, by the internal model principle (Francis and Wonham, 1976) RC can asymptotically reject the periodic disturbance.

Traditional RC approaches lead to a degraded performance if the operating velocity, i.e., disturbance frequency, changes. Many practical applications require a continuously varying operation velocity, for example a printer moving over paper, a rotary system for tracking satellites (Saathof et al., 2019), or a wafer-scanner (Blanken et al.,

2017). In these cases, the disturbance is periodic in the position domain, however, due to speed variations it appears a-periodic in the time domain. This results in a situation where traditional RC is not effective.

Several repetitive control techniques have been developed to deal with these period variations. In Blanken and Oomen (2019), an extension is presented allowing to design multiple RCs for systems that operate at multiple fixed velocities. However, performance during velocity changes is not guaranteed. In Witvoet et al. (2019), an alternative is presented where the fundamental disturbance frequency is chosen as small as possible covering a wide range of disturbance frequencies, which is rather pragmatic and may lead to worse stability margins. In Steinbuch (2002), RC has been extended with robustness against small variations in the disturbance frequency, however large or continuous variations are not covered. Other existing approaches focus on defining the system in the spatial domain, resulting in a nonlinear system that requires an additional step of feedback linearization, see e.g., Chen and Yang (2007).

Although recent progress has been made to increase robustness for slightly varying disturbance frequencies, a solution for large or continuous variations, where the source of the disturbance is repeating in the position domain, is not yet established. In answer to this, the aim of this paper is to extend RC towards the spatial domain. The key idea is to use a spatial memory, instead of a time-domain memory, by means of a Gaussian Process (GP) with a suitable periodic kernel (Murphy, 2012; Williams and Rasmussen, 2006; Pillonetto et al., 2014; Jidling et al., 2018). The GP estimates a continuous function, allowing to deal effectively with the intermittent observations as they occur in the position domain. This paper contains the following sub-contributions;

(C1) spatial RC approach for systems with a-periodic and varying disturbance frequencies,

(C2) a new spatial memory with a GP and suitable periodic kernel allowing for intermittent observations, in combination with a time domain learning filter, and

(C3) a simulation case study confirming the potential.

This paper is outlined as follows. The disturbance rejection problem is defined in section 2. In Section 3, the spatial RC approach is introduced (C1). In Section 4, a position-domain memory loop by means of a GP is provided with a suitable periodic kernel (C2). In Section 5, a simulation example shows that spatial RC outperforms the standard RC in the considered setting (C3). Finally, conclusions are provided in Section 6.

2. PROBLEM DEFINITION

In this section, the problem setting and considered disturbances are defined. This shows that observations indeed are non-equidistant in position. Finally, the problem definition and contributions are formulated.

2.1 Problem setting

Consider the control setup depicted in Fig. 1, where C_{fb} is a stabilizing feedback controller, and P is a single-input single-output (SISO) linear time-invariant (LTI) plant. The signal $r(t)$ is an optional reference to be tracked, $y(t)$ is the output to be controlled, and $d(t)$ is an unknown exogenous disturbance that is collocated with the input signal $u_{\mathrm{fb}}(t)$. The disturbance $d(t)$ is assumed to be periodic in the spatial domain with some given spatial period p_{per}, i.e., it satisfies the following assumptions.

Assumption *2.1. Let $d(t)$ be an unknown exogenous disturbance that satisfies:*

(A1) $d(t)$ is composed of an unknown position-domain function $d_p(p)$ and the current position $p(t)$, i.e.,

$$d(t) = d_p(p(t)) \qquad (1)$$

which is schematically depicted in Fig. 1, and

(A2) $d_p(p)$ is spatially periodic with period p_{per}, i.e.,

$$d_p(p) = d_p(p + k \cdot p_{\mathrm{per}}) \text{ for } k \in \mathbb{N} \qquad (2)$$

where $p_{\mathrm{per}} \in \mathbb{R}$ is known and fixed.

The disturbance $d(t)$ depends on the position signal $p(t)$. If the position increases/decreases with a fixed rate, i.e., the speed $dp(t)/dt$ is constant, then $d(t)$ appears periodic in the time domain. If the velocity is time varying, then the disturbance is a-periodic in the time domain, while the underlying function generating the disturbance d_p remains periodic in the position domain. An example is shown in Fig. 2, where the velocity changes in the gray area leading to a-periodicity in the time domain.

The aim of this paper is to develop a spatial RC approach such that the following control objective is satisfied.
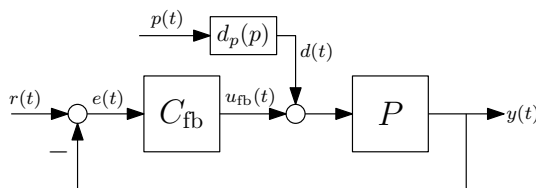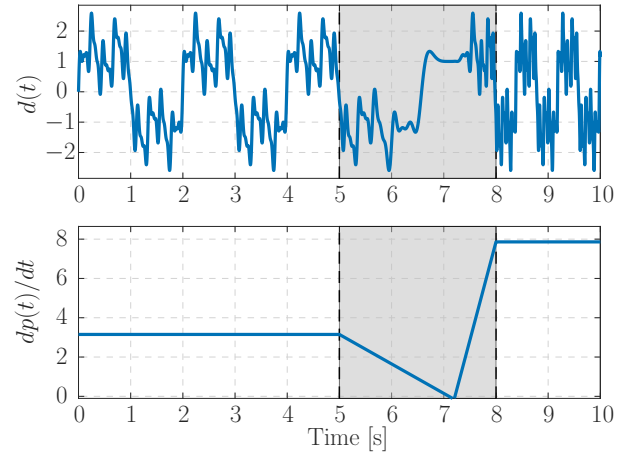

Fig. 1. Problem setting.

Fig. 2. Simulation example of the disturbance (top plot), with corresponding velocity (bottom plot) as function of time. Gray area indicates the velocity change causing a-periodicity in the time domain disturbance.

Definition 2.1. (Control Objective). *Minimize the effect of the potentially a-periodic disturbance $d(t)$ on the position error $e(t)$, independent of velocity variations.*

2.2 Traditional repetitive control

Traditional repetitive control is not effective for disturbances with varying period or a-periodic disturbances, see, e.g., Longman (2010); Hara et al. (1988). According to the well-known internal model principle, a model of the disturbance must be present in the feedback loop to attenuate it (Francis and Wonham, 1976). In traditional RC, this is captured in a time-domain memory loop based on previous error data. The size of the memory loop must coincide with the disturbance period. If the disturbance period changes, the pre-defined buffer size is not adequate anymore, resulting in degraded performance.

The key idea in this paper, is to specify a memory loop in the spatial domain, where the disturbance is assumed to be fixed and periodic. The main challenge arising in the spatial domain, is that observations are inherently non-equidistantly distributed due to speed variations. Hence, a fixed memory loop as in traditional RC is not suitable, see Mooren et al. (2020). Therefore, an alternative solution using a GP, essentially estimating a continuous memory loop, is presented here.

2.3 Problem definition

Disturbances that vary in the time domain, but yet have an underlying position-domain disturbance that is periodic, cannot be attenuation by traditional RC. This necessitates the construction of a memory loop in the position domain to deal with the non-equidistant nature of the observations. Thus, the aim of this paper is to develop a new spatial RC approach, in which a memory loop is constructed in the spatial domain by means of a Gaussian process with a suitable kernel. This enables suppression of disturbances satisfying (A1)-(A2) with varying frequency.

3. SPATIAL REPETITIVE CONTROL FRAMEWORK

In this section, the spatial RC framework is established, enabling attenuation of disturbances that are periodic in
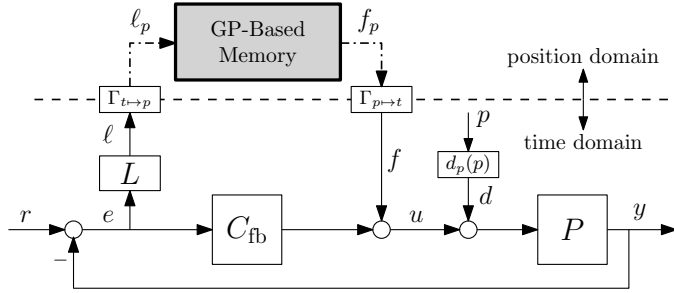
Fig. 3. Position-based repetitive control scheme, where the solid lines represent time-domain signals and the dashed lines represent position-domain signals.

the spatial domain. The subsequent section is devoted to the of a spatial memory loop using a GP.

3.1 Spatial repetitive control setting

The spatial RC framework is depicted in Fig. 3, where L is a learning filter, $\ell(t)$ is the learning signal, and $f(t)$ is the output of the RC that is injected in the feedback loop. An explicit distinction is made between time-domain signals and position-domain signals with subscript $(\cdot)_p$. The learning signal $\ell(t)$ is mapped to the position-domain signal $\ell_p(p)$, indicated by the mapping $\Gamma_{t\mapsto p}$. Subsequently, the signal $\ell_p(p)$ is used to construct the disturbance model in the spatial memory loop. The output of the memory loop is mapped back to the time domain by $\Gamma_{p\mapsto t}$, resulting in $f(t)$ that is injected in the feedback loop. Note that the learning filter L is located before the buffer, which will appear to be essential in this spatial approach.

The generated signal $f(t)$ exactly compensates for the disturbance $d(t)$. This is done by estimating the function $d_p(p)$, i.e., the underlying cause of the disturbance, in the memory loop by means of a GP, see Fig. 4. Details on the implementation of the GP in the memory loop are discussed in Section 4. To estimate d_p in the buffer, a specific choice of learning filter is required, which is analog to the L filter design in traditional RC.

Procedure 1. (Learning filter design).

(1) Identify a parametric model $\hat{P}$ of the system,
(2) compute an estimate of the process sensitivity

$$\widehat{PS} = \frac{\hat{P}}{1 + C_{\mathrm{fb}}\hat{P}}, \tag{3}$$

(3) invert the process sensitivity estimate to obtain L

$$L = \widehat{PS}^{-1}. \tag{4}$$

Remark *3.1.* If $\widehat{PS}$ contains non-minimum phase zeros, leading to an unstable inverse, several techniques exist to compute stable but non-causal inverses, see e.g., van Zundert and Oomen (2017); Tomizuka (1987).

Note that the learning filter is allowed to have finite preview, i.e., non-causality $L = z^{n_l}L_c$, where n_l is the number of samples of preview. How to include this in the memory loop is shown in Section 4.

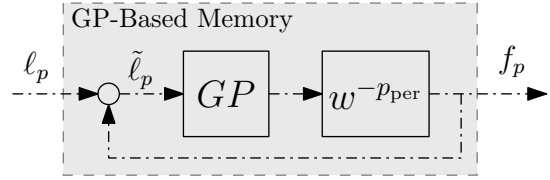The learning signal is mapped to the spatial domain to generate ℓ_p, used in the memory loop as shown in Fig. 4.

Fig. 4. Spatial memory loop with a Gaussian process.

Procedure 2. (Mapping $\ell(t)$ to $\ell_p(p)$).

Given the current position $p(t)$

(1) obtain $p^*(t) = \mathrm{mod}\,(p(t), p_{\mathrm{per}})$
(2) construct $\ell_p(p) = \ell(p) \circ p^*(t)$

By selecting the learning filter L as the inverse of the process sensitivity, the learning signal $\ell(t)$ becomes an approximation of the disturbance, i.e., $\hat{d}(t) = Le(t)$. Hence, $\ell_p(p)$ will approximate $d_p(p)$ that is stored in the GP. Note that a feedback loop is present in the memory loop. This allows the error and therefore also $\ell_p(p)$ to converge to zero while $\tilde{\ell}_p(p)$ remains equal to an estimate of $d_p(p)$. Furthermore, the output of the memory loop is delayed with exactly one period in the spatial domain similar to traditional RC, denoted with $w^{-p_{\mathrm{per}}}$.

Finally, the output of the memory loop $f_p(p)$ is mapped to the time domain which completes the RC framework.

Procedure 3. (Mapping $f_p(p)$ to $f(t)$).

Given the current position $p(t)$,

(1) obtain $p^* = \mathrm{mod}\,(p(t), p_{\mathrm{per}})$,
(2) evaluate the posterior mean of the GP's density function at p^* given observations, denoted as

$$u_{\mathrm{ff}}(t) = \mu_{\mathrm{post}}(p^*)$$

and further discussed in Section 4.

Several interesting observations can be made with respect to traditional RC. First of all, the learning filter L is placed before instead of after the memory loop. In the tradition RC, the L filter can be placed before or after the memory loop using the commutative property of linear systems. Because of the transformations from time to position domain this property does not hold anymore. Secondly, the summation point is moved to the output of the feedback controller instead of being at the input. This allows to extrapolate the learned function $d_p(p)$ to the time-domain, i.e., generate the exact opposite of $d(t)$. Finally, the mappings between time and position domain can be interpreted similar to using basis function in iterative learning control (ILC), see e.g., van de Wijdeven and Bosgra (2010). Allowing to learn the disturbance in the position domain, and extrapolate to the time domain for varying operating velocities.

In the following section, the implementation of GPs in the memory loop is discussed.

4. SPATIAL MEMORY LOOP USING GPS

A Gaussian Process (GP) regression model can be interpreted as a distribution over functions. Using inference on the basis of training data, i.e., observations, in combination with priors, represented by a kernel, one can determine

statistical properties of the underlying function, see e.g., Murphy (2012); Williams and Rasmussen (2006). In this section the GP is analyzed and a suitable periodic kernel is presented. Finally, a procedure is provided to integrate a GP in the spatial memory as shown in Fig. 4.

4.1 GP based spatial memory loop

The aim is to estimate a continuous function representing the true spatial disturbance $d_p(p)$, using observations $\ell_p(p)$ that are contaminated with noise, i.e.,

$$\tilde{\ell}_p(p) = d_p(p) + \epsilon, \text{ with } \epsilon \sim \mathcal{N}(0, \sigma_n^2) \qquad (5)$$

where ϵ is independent identically distributed zero-mean Gaussian noise with variance σ_n^2. Furthermore, a suitable kernel choice allows to deal deal with noisy observations and include desired properties of the estimated signal, such as periodicity and smoothness.

Next, assume that d_p and $\tilde{\ell}_p$ are random variables that have a joint Gaussian distribution denoted as follows,

$$\begin{bmatrix} d_p \\ \tilde{\ell}_p \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K + \sigma_n^2 I_N & K_* \\ K_*^\top & K_{**} \end{bmatrix} \right) \qquad (6)$$

where $K \in \mathbb{R}^{N \times N}$, $K_* \in \mathbb{R}^{N \times N^*}$ and $K_{**} \in \mathbb{R}^{N^* \times N^*}$ are kernels or covariance functions, with N the number of observations and N^* the number of test positions.

Remark 4.1. Note that the mean of the distribution is assumed to be zero for the ease of notation. This can easily be extended for a non-zero mean (Murphy, 2012).

The Gaussian process is completely determined by its mean and covariance function. Given observations $\tilde{\ell}_p$, the posterior distribution becomes,

$$\hat{d}_p | \tilde{\ell}_p \sim \mathcal{N}\left(\mu_{\text{post}}, P_N^{\text{post}} \right), \qquad (7)$$

with mean μ_{post} and covariance P_N matrix,

$$\mu_{\text{post}} = K_*^\top (K + \sigma_n^2 I_N)^{-1} \tilde{\ell}_p, \qquad (8)$$

$$P_N = K_{**} - K_*^\top (K + \sigma_n^2 I_N)^{-1} K_*. \qquad (9)$$

Since there is only one test point p at a time, the posterior mean can be computed efficiently as follows,

$$\mu_{\text{post}}(p) = \sum_{i=1}^N \alpha_i \kappa(p_i, p) \qquad (10)$$

where $\kappa(p_i, p)$ is the kernel evaluated at training point p_i and test point p, and

$$\alpha = (K + \sigma_n^2 I)^{-1} \tilde{\ell}_{\text{train}} = [\alpha_1 \ \alpha_2 \ \ldots \ \alpha_N] \qquad (11)$$

in which $\tilde{\ell}_{\text{train}}$ contains the observations at position p_i.

4.2 Periodic kernels

Next, is the selection of a suitable kernel function imposing priors on the estimate $\hat{d}_p$. Since the disturbance d_p is periodic this should be included in the kernel function. However, traditional kernels as often used in system identification approaches, see e.g., Pillonetto et al. (2014); Chen et al. (2012); Blanken and Oomen (2020), do not impose these type of priors. In this paper, the following kernel is utilized to impose periodicity and smoothness,

$$\kappa(p, p') = \sigma_f^2 \exp\left(\frac{-2\sin^2(\frac{p-p'}{2\lambda})}{l^2} \right) \qquad (12)$$

that is periodic due to the presence of the sine function, and σ_f, λ, ℓ are the hyper-parameters. An example of a
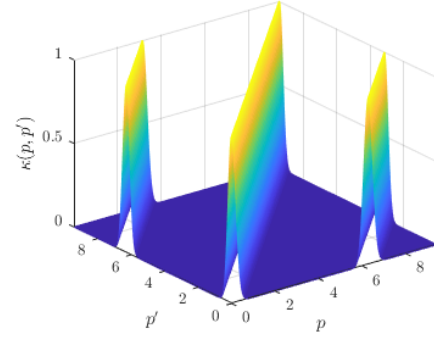

Fig. 5. Periodic kernel used for spatial RC, with $\lambda = 2\pi$, $l = 0.2$, $\sigma_f = 1$ and $\sigma_n = 10^{-6}$.

periodic kernel for a specific set of hyper-parameters is depicted in Fig. 5, where it can be seen that the kernel repeats every 2π. The parameter λ represents the period of the function d_p, which is given by p_{per} according to Assumption 2.1. This includes periodicity, i.e., an observation at $p = 0$ infers something about the position p_{per}. The parameter l imposes smoothness of the observations assuming that the underlying function d_p is smooth as well. Hence, an observation at some position p also provides information about positions close to that specific observation. This kernel allows to extrapolate beyond the data observations for fast learning.

Remark 4.2. An empirical Bayesian optimization can be used to optimize the hyper-parameters, see e.g., Snoek et al. (2012).

4.3 GP buffer procedure

In this section, a procedure to integrate GPs in the spatial RC framework is presented. Furthermore, non-causality of the learning filter is included in the memory loop.

Procedure 4. (Position-domain RC using GP).

I. Initialization and prior

(1) Set the kernel parameters σ_f, σ_n, λ and l.
(2) Obtain observation $\tilde{\ell}_p(p)$ at current position $p(k)$.
(3) Set counter $i = 1$

II. Every $\bar{N}^{\text{th}}$ sample:

if $p(k) \leq p_{\text{per}}$

(1) Store training observations:

$$p_{\text{train}}(i) = \tilde{p}, \quad \ell_{\text{train}}(i) = \tilde{\ell}_p$$

where $\tilde{\ell}_p = \ell_p$, and

$$\tilde{p} = \text{mod}(p(k) - n_l \frac{dp}{dt} T_s, p_{\text{per}}) \qquad (13)$$

with n_l is the number of preview samples in L and T_s is the sample time.
(2) Set $i = i + 1$

else

(1) In this case the memory loop is already filled, which is included in the computation of $\tilde{\ell}_p$. Again store the data,

$$p_{\text{train}}(i) = \tilde{p}, \quad \ell_{\text{train}}(i) = \tilde{\ell}_p$$

where $\tilde{p}$ is given by (13), and $\tilde{\ell}_p = \tilde{\ell} + \mu_{\text{prev}}(\tilde{p}(k))$ and μ_{prev} is the posterior mean obtained by evaluation the GP given the data p_{train} and ℓ_{train} corresponding to the previous spatial period.

(2) Set $i = i + 1$

III. Computing $u_{\text{ff}}(k)$, at every sample

(1) Given data p_{train} and ℓ_{train} from the previous period, compute the posterior mean of the GP using (10) and (11).

Note that every $\bar{N}^{\text{th}}$ observation is used to update the GP. This facilitates the computational aspects and can easily be implemented in the kernel by adapting the smoothness parameter l. By setting $\bar{N} = 1$ all observation are used.

This completes the spatial RC framework including a GP based memory loop. In the following section, spatial and traditional RC are compared in a simulation example.

5. SIMULATION STUDY

In this section, a simulation example is provided to confirm that the spatial RC approach outperforms traditional RC for changing disturbance frequencies.

5.1 System description and simulation setting

Consider a second order mass-spring-damper system,

$$P(s) = \frac{1}{Js^2 + ds + k} \tag{14}$$

with inertia $J = 1$ kg·m², damping $d = 1$ Nm/s, and stiffness $k = 10^4$ Nm. A position-dependent disturbance $d(t)$ acts on the system being controlled by a torque T, see Fig. 6. The plant is discretized by zero-order-hold discretization with sample frequency $f_s = 1000$ Hz. A stabilizing feedback controller is designed consisting of a gain, lead filter and a low-pass filter. The resulting bandwidth, i.e., 0 dB crossing of the open-loop, is 50 Hz.

5.2 Repetitive controller design

The learning filter is designed according to Procedure 1, i.e., equation (4). For traditional RC, the learning filter is designed as the inverse of the complementary sensitivity function, see Longman (2010).

The positioning signal $p(t)$ that drives the disturbance is generated in a second equivalent loop, this allows to separate the effect of the reference and the disturbance. The velocity profile for the reference is equivalent to Fig. 2, resulting in the position signal $p(t)$ in the bottom plot of Fig. 7. The disturbance mapping $d_p(p)$ is given by,

$$d_p(p) = 1.5\sin(p) + 0.8\sin(3p) + 0.6\sin(9p) + \dots$$
$$+ 0.4\sin(18p) + 0.2\sin(27p).$$

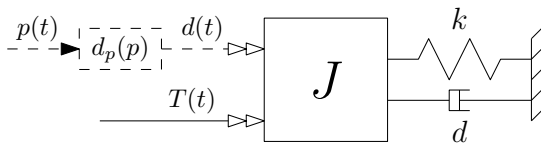Fig. 6. Simulation model with mass-spring-damper system, disturbance $d(t)$ and input torque $T(t)$.
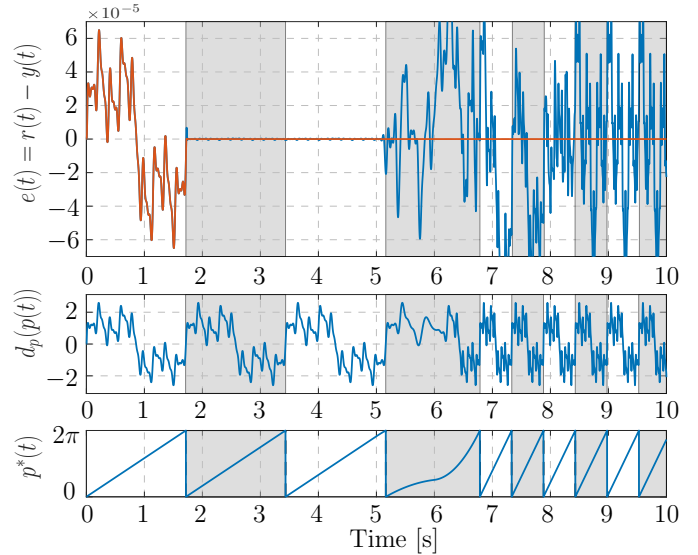

Fig. 7. Top: Resulting positioning error for the traditional RC (—) and the spatial RC (—). Middle: corresponding disturbance signal. Bottom: positioning signal generating the disturbance. The individual periods are indicated with the white and gray areas.

and the time-domain disturbance is generated as in (1), i.e., at sample k it is given by $d(k) = d_p(p(k))$, see Fig. 7. The period of the spatial RC is set to one revolution $p_{\text{per}} = 2\pi$ rad, and $\sigma_n = 10^{-6}$, $l = 0.1$, $\sigma_f = 1$. The memory loop size for traditional RC is chosen equal to the first disturbance period $N_{\text{conv}} = 1717$.

5.3 Simulation results

A simulation is conducted using both traditional RC and spatial RC. The resulting errors and the applied disturbance are shown in Fig. 7, where the individual periods are indicated with white and gray areas. It can be seen that the disturbance is periodic during the first 3 periods, after which a change in velocity results in a changing disturbance. To compare both methods, the 2-norm of each period j is normalized by the period length N_j, the result is depicted in Fig. 8. During the first period, the errors are equivalent since both RCs are not yet active. After one period it can be seen that both methods are able to significantly attenuate the disturbance leading to a small error.

Remark 5.1. Note that the traditional RC is not able to completely mitigate the disturbance. This occurs since the disturbance period time varies a little, due to its dependence on $p(t)$. Hence, it deviates slightly from the memory-loop size.

After 3 periods, the disturbance changes significantly leading to degraded performance in the traditional RC approach. It is not able to learn the disturbance since it is not compatible with the memory loop size. Hence, traditional RC requires re-tuning for changing velocities. The spatial RC is not affected by the changing velocity. Clearly it outperforms traditional RC, and continues to converge to a smaller error, see Fig. 8.

Finally, it is concluded that the developed spatial RC method outperforms traditional RC, by enabling attenua-
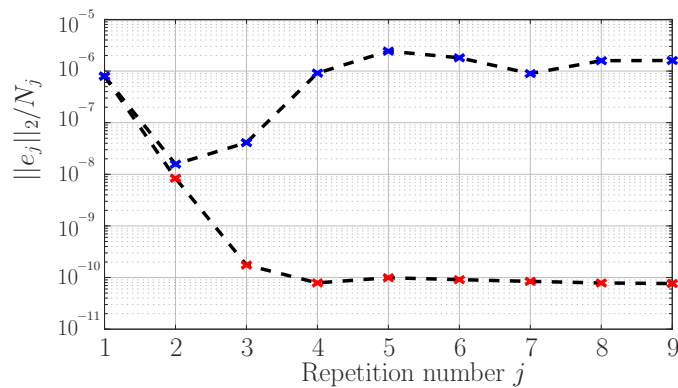
Fig. 8. 2-norm of the error normalized with the period length, for the traditional RC ($\times$), and the spatial RC ($\times$) as function of the repetition number.

tion of disturbances that are periodic in the spatial domain but potentially a-periodic in the time domain.

6. CONCLUSION

Disturbances that are driven by position, may appear a-periodic in the time domain, due to velocity changes, yet have a periodic position-domain equivalent. In this paper, a new spatial RC approach is developed that attenuates these disturbances independent of velocity variations. A memory loop in the spatial domain based on a GP with suitable periodic kernel is developed. This effectively deals with non-equidistant observations inherent to the spatial-domain. A simulation example compares spatial RC with traditional RC and confirms the performance benefit, i.e., that it can deal with position dependent disturbances such as cogging, imbalances, etc., as encountered in many practical applications.

REFERENCES

Ahn, H.S., Chen, Y., and Dou, H. (2005). State-periodic adaptive compensation of cogging and coulomb friction in permanent magnet linear motors. In *Proceedings of the American Control Conference, 2005.*, 3036–3041. IEEE.

Blanken, L., Boeren, F., Bruijnen, D., and Oomen, T. (2017). Batch-to-batch rational feedforward control: From iterative learning to identification approaches, with application to a wafer stage. *IEEE/ASME Transactions on Mechatronics*, 22(2), 826–837.

Blanken, L. and Oomen, T. (2019). Multivariable iterative learning control design procedures: From decentralized to centralized, illustrated on an industrial printer. *IEEE Transactions on Control Systems Technology*.

Blanken, L. and Oomen, T. (2020). Kernel-based identification of non-causal systems with application to inverse model control. *Automatica*, 114, 108830.

Chen, C.L. and Chiu, G.T.C. (2008). Spatially periodic disturbance rejection with spatially sampled robust repetitive control. *Journal of Dynamic Systems, Measurement, and Control*, 130(2), 021002.

Chen, C.L. and Yang, Y.H. (2007). Adaptive repetitive control for uncertain variable-speed rotational motion systems subject to spatially periodic disturbances. In *2007 American Control Conference*, 564–569. IEEE.

Chen, T., Ohlsson, H., and Ljung, L. (2012). On the estimation of transfer functions, regularizations and

gaussian processes revisited. *Automatica*, 48(8), 1525–1535.

Francis, B.A. and Wonham, W.M. (1976). The internal model principle of control theory. *Automatica*, 12(5), 457–465.

Hara, S., Yamamoto, Y., Omata, T., and Nakano, M. (1988). Repetitive control system: A new type servo system for periodic exogenous signals. *IEEE Transactions on automatic control*, 33(7), 659–668.

Jidling, C., Hendriks, J., Wahlström, N., Gregg, A., Schön, T.B., Wensrich, C., and Wills, A. (2018). Probabilistic modelling and reconstruction of strain. *Nuclear Instruments and Methods in Physics Research*, 436, 141–155.

Li, P.Y. (2015). Prototype angle-domain repetitive control-affine parameterization approach. *Journal of Dynamic Systems, Measurement, and Control*, 137(12).

Longman, R.W. (2010). On the theory and design of linear repetitive control systems. *European Journal of Control*, 16(5), 447–496.

Mooren, N., Witvoet, G., Açan, I., Kooijman, J., and Oomen, T. (2020). Suppressing position-dependent disturbances in repetitive control: With application to a substrate carrier system. In *International Workshop on Advanced Motion Control (Accepted for publication)*.

Murphy, K.P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Pillonetto, G., Dinuzzo, F., Chen, T., De Nicolao, G., and Ljung, L. (2014). Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3), 657–682.

Saathof, R., Crowcombe, W., Kuiper, S., van der Valk, N., Pettazzi, F., de Lange, D., Kerkhof, P., van Riel, M., de Man, H., Truyens, N., et al. (2019). Optical satellite communication space terminal technology at TNO. In *International Conference on Space Optics*, volume 11180. International Society for Optics and Photonics.

Snoek, J., Larochelle, H., and Adams, R.P. (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, 2951–2959.

Steinbuch, M. (2002). Repetitive control for systems with uncertain period-time. *Automatica*, 38(12), 2103–2109.

Tomizuka, M. (1987). Zero phase error tracking algorithm for digital control. *Journal of Dynamic Systems, Measurement, and Control*, 109(1), 65–68.

van de Wijdeven, J. and Bosgra, O. (2010). Using basis functions in iterative learning control: analysis and design theory. *Int. Journal of Control*, 83(4), 661–675.

van Zundert, J. and Oomen, T. (2017). On inversion-based approaches for feedforward and ILC. *Mechatronics*, 50, 282 – 291.

Wang, Y., Gao, F., and Doyle III, F.J. (2009). Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control*, 19(10), 1589–1600.

Williams, C.K. and Rasmussen, C.E. (2006). *Gaussian processes for machine learning*, volume 2. MIT Press Cambridge, MA.

Witvoet, G., Peters, J., Kuiper, S., and Oomen, T. (2019). Line-to-line repetitive control of a 6-DoF hexapod stage for overlay measurements using atomic force micropsy. In *Preceedings of American Control Conference (ACC)*, Philadelphia, PA, USA.