Triggered Repetitive Control

Triggered Repetitive Control: Application to Mechanically Ventilated Patients

Joey Reinders, David Elshove, Bram Hunnekens, Nathan van de Wouw, *Fellow, IEEE*, and Tom Oomen, *Senior Member, IEEE*

***Abstract*— Asymptotic rejection of a periodic disturbance can be achieved using repetitive control (RC). The aim of this article is to develop a triggered RC (TRC) framework that can handle repeating tasks that are initiated by an external disturbance induced trigger on varying intervals, which clearly violates the periodicity assumption in RC. A design method for this TRC framework is presented with a stability guarantee. Finally, through an experimental use-case, it is shown that pressure tracking performance for mechanically ventilated patients is improved significantly.**

***Index Terms*— Mechanical ventilation, repetitive control (RC), respiratory systems, switching systems, target tracking, triggering.**

I. Introduction

MECHANICAL ventilators are essential equipment in intensive care units (ICUs) to assist patients who need support to breathe sufficiently. The main goals of mechanical ventilation are to ensure oxygenation and carbon dioxide elimination [1]. Especially during the flu season or a worldwide pandemic such as the COVID-19 pandemic [2], mechanical ventilation is a life saver for many patients around the world.

An important ventilation mode is pressure-controlled—assist control ventilation (PC-ACV). In PC-ACV, the ventilator synchronizes with the patient. This is achieved by triggering the ventilator when a spontaneous patient breath is detected. More specifically, when the patient starts an inspiration, the ventilator is triggered and increases the pressure near the patient's airway to ensure sufficient flow into the patient. Then, after a preset time, the ventilator decreases its pressure to allow an expiration flow of carbon-dioxide-rich air. After this expiration, the ventilator waits for the next patient-induced breath. In more generic terms, the controller should track a repeating reference profile (target pressure) which is triggered by an external disturbance (patient effort) at varying and a priori unknown intervals.

Accurate tracking of the pressure profile is essential to ensure sufficient patient support and enhance patient comfort. According to [3], improved pressure tracking prevents patient–ventilator asynchrony. In [4], patient–ventilator asynchrony is even associated with increased mortality rates. Furthermore, accurate tracking avoids undesired pressure peaks caused by overshoot. These peaks can possibly harm the patient's lungs and therewith prolong the ICU stay.

The challenging problem of pressure tracking in the presence of widely varying and uncertain patient parameters has spurred the development of a wide range of pressure control methodologies. In [5], an overview of modeling and control techniques for mechanical ventilation is presented. Examples of control methods for ventilation are variable-gain control [3], adaptive control [6], funnel-based control [7], model-based control [8], model-predictive control [9], and adaptive hose-compensation control [10], [11]. All these methods have shown to improve the tracking performance in ventilation. However, the repetitiveness of breathing is not used to further improve the performance.

Because of the repetitive nature of breathing, learning control strategies, such as iterative learning control (ILC) ([12], [13], [14], [15], [16]) and repetitive control (RC) ([16], [17], [18], [19], [20]), are particularly suitable for mechanical ventilation. ILC is a feedforward control strategy that computes the optimal feedforward signal based on errors made during previous tasks. In contrast, RC is a feedback control strategy for continuous repetitive processes, which do not have identical states at the start of a task. Recently, these learning control strategies have been applied to mechanical ventilation. For example, ILC is applied in [21], [22], and [23] and RC is applied in [24]. These methods achieve superior performance in case of a fully sedated patient, i.e., when the pressure reference is fully repetitive. However, in case a patient starts breathing spontaneously and triggers the ventilator at a priori unknown varying intervals (as in PC-ACV), the performance of these methods degrades significantly.

Manuscript received 13 April 2022; revised 8 September 2022; accepted 26 December 2022. Date of publication 24 January 2023; date of current version 22 June 2023. Recommended by Associate Editor K. Barton. *(Corresponding author: Joey Reinders.)*

Joey Reinders is with DEMCON Advanced Mechatronics, 5683 CR Best, The Netherlands, and also with the Department of Mechanical Engineering, Eindhoven University of Technology, 5612 AZ Eindhoven, The Netherlands (e-mail: joey.reinders@demcon.com).

David Elshove was with DEMCON Advanced Mechatronics, 5683 CR Best, The Netherlands, and also with the Department of Mechanical Engineering, Eindhoven University of Technology, 5612 AZ Eindhoven, The Netherlands. He is now with ALTEN Nederland, 5652 AH Eindhoven, The Netherlands (e-mail: david.elshove@alten.nl).

Bram Hunnekens is with DEMCON Advanced Mechatronics, 5683 CR Best, The Netherlands (e-mail: bram.hunnekens@demcon.com).

Nathan van de Wouw is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5652 AH Eindhoven, The Netherlands, and also with the Department of Civil, Environmental and Geo-Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: n.v.d.wouw@tue.nl).

Tom Oomen is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5652 AH Eindhoven, 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.2023.3237610.

Digital Object Identifier 10.1109/TCST.2023.3237610

Several approaches on learning control strategies with improved task-to-task flexibility are developed in literature. For example, ILC with basis functions [25], [26], basic task approach for ILC [27], and ILC with varying pass lengths [28], [29]. However, the main assumption for ILC is that the initial states are identical between subsequent tasks and independent of the output of the previous task. Because breathing is a continuous process, this assumption is not satisfied. Therefore, RC is the preferred learning control strategy for mechanical ventilation. Also, for RC several methods are developed that improve the task-to-task flexibility. For example, in [30], a robust RC approach for systems with uncertain period times is presented; however, only small variations are covered. In [31], delay-varying RC is developed that can handle repetitive disturbances that are repetitive with respect to a different variable than time. However, the example of triggered ventilation is not perfectly repetitive to any variable. Furthermore, in [32], an RC approach using the Gaussian processes is developed to suppress disturbances that are repetitive in the position domain instead of the time domain. This method does not allow unknown varying times in between tasks.

Although significant tracking performance improvements in mechanical ventilation have been obtained using learning control strategies, this superior tracking performance is not achieved for triggered ventilation modes such as PC-ACV, where the duration in between breaths is varying and a priori unknown. Therefore, the aim of this article is to develop a triggered learning control framework that learns from data of previous breaths to improve the tracking performance in case of PC-ACV. The developed framework could also be applied in other application fields with similar challenges, e.g., pick-and-place machines and dual-stage positioning systems.

The main contribution of this article is as follows.

1) A triggered RC framework that can improve the tracking performance for systems with perfectly repetitive tasks, where the start of a task is triggered at a priori unknown varying timing intervals.

Besides the main contribution, the article contains several subcontributions.

1) A method to guarantee stability of the closed-loop system with the proposed triggered RC framework.
2) A design methodology of the triggered RC framework for a mechanical ventilation system in PC-ACV.
3) An experimental case study showing the performance increase, compared with a state-of-the-art control strategy, in mechanical ventilation.

The outline of this article is as follows. In Section II, PC-ACV is explained and a general, not ventilation-specific, control problem is formulated. Then, in Section III, the triggered RC (TRC) framework is presented. In Section IV, the relevant system models for mechanical ventilation are described. Thereafter, in Section V, a TRC for mechanical ventilation is designed and its performance is experimentally tested. Finally, in Section VI, the main conclusions and future work are presented.

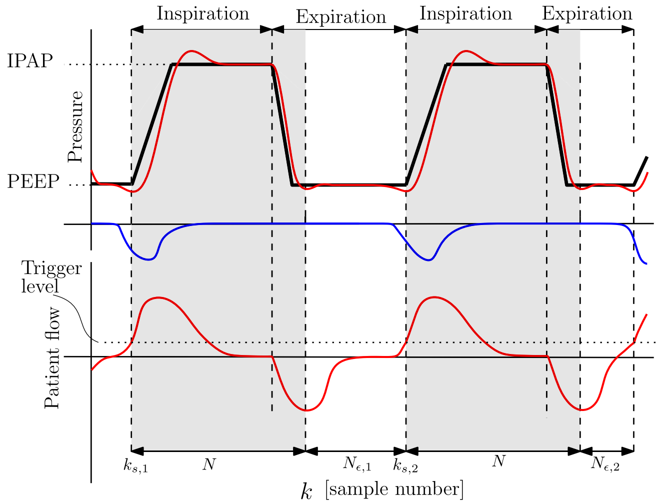

Fig. 1. Schematic example of PC-ACV with two breaths, where the inspiration is triggered by a flow trigger and the expiration is timed. The figure shows the target pressure (▬), the airway pressure and patient flow (▬), and the patient's breathing effort (▬).

II. Control Problem Formulation

In this section, the control problem of PC-ACV is reformulated to a general control problem. First, a description of PC-ACV of spontaneously breathing patients is given in Section II-A. Thereafter, in Section II-B, the control problem for mechanical ventilation in PC-ACV is stated and it is translated to a general TRC problem that is solved in this article. Next, in Section II-C, the main challenges when triggering conventional RC are explained and demonstrated through a simulation example.

A. PC-ACV of Spontaneously Breathing Patients

The goal in PC-ACV is to assist the patient's breathing when the spontaneous breathing effort is insufficient. The ventilator assists the patient to achieve the desired tidal volume. In such assisted ventilation modes, it is important to synchronize the ventilator strokes with the patient's spontaneous breathing effort for the comfort of the patient.

A schematic example of PC-ACV pressure and flow curves is depicted in Fig. 1. The figure shows that the patient is generating a negative pressure in the lung, resulting in a small positive patient flow. However, this effort is not enough to achieve the desired volumes. Therefore, when the patient flow exceeds a predefined trigger level, the ventilator increases the airway pressure, i.e., the pressure near the patient's airway, to the inspiratory positive airway pressure (IPAP) level, to increase the patient flow and volume. Then, after a predefined time, the ventilator pressure decreases again, to the positive end-expiratory pressure (PEEP) level, to allow for an expiration. After the ventilator breath, of N samples long, is finished, the ventilator waits until a new patient-induced trigger is detected. Then, it starts a new ventilator-assisted breath. The time in between breaths is unknown and varying since it is determined by the patient. During the idle phase, this controller aims at keeping the pressure at PEEP level.

B. Control Problem

As mentioned in the introduction, RC is a suitable control strategy to handle the variety of patients in ventilation. Therefore, a general control problem is formulated using Fig. 1 and the block diagram of a plant with a parallel structured repetitive controller in Fig. 2, i.e., the RC R is implemented in parallel with the feedback controller C. Now, we aim to describe a generic control problem here, but in the scope of PC-ACV, G represents the patient–hose–ventilator dynamics, y is the airway pressure, and r is the target pressure, see the black line in Fig. 1.

A plant G has to be controlled using a feedback controller C_i and RC R_i, consisting of a learning filter L_i and robustness filter Q_i which can be causal implementations of noncausal filters using the delay operators with p_l and p_q in Fig. 2, where z^{-1} represents a delay of one discrete time step. This control strategy should ensure that the output y tracks the reference r, i.e., the *control goal* is to achieve a tracking error $e := r - y$ of zero. The reference r, which is visualized in Fig. 1, has the following properties.

1) During the *active phase* $r(k_{s,j}+i) := r(k_{s,j+1}+i)\ \forall j \in \mathbb{N}^{+}$ and $\forall i \in [1, N]$ (with j a task counter), i.e., the reference r is exactly the same in shape and duration (N samples) as the reference r during the subsequent tasks.
2) During the *idle phase* $r(k_{s,j}+N+i) := r(k_{s,j}+N)\ \forall j \in \mathbb{N}^{+}$ and $\forall i \in [1, N_{\epsilon,j}]$, i.e., the reference r in the idle phase is constant and equal to the last value of the reference in the active phase.
3) The duration of the *idle phase*, $N_{\epsilon,j}$, is varying over j and a priori unknown. Switching to the idle phase is caused by the detection of an external disturbance d.[1]

Switching between these phases at varying intervals obstructs usage of the conventional RC, which is designed to suppress disturbances, i.e., track references, that are perfectly periodic with respect to time, see for example [24], [33]. Simply switching conventional RC on and off results in an undesired actuation spike at the end of a task. The performance issue when switching conventional RC is investigated in Section II-C by means of a simulation use-case.

C. Example: Toward Triggering Repetitive Control

In this section, the challenge of triggering a conventional RC is visualized through simulations. More specifically, it is shown that applying standard RC does not achieve satisfactory performance in an application with a repetitive reference on varying intervals. To show these challenges, first the simulation use-case is described where standard RC is used in a triggered application. Then, the time-domain simulation results and the resulting problem are presented. Finally, the challenges that should be addressed are described.

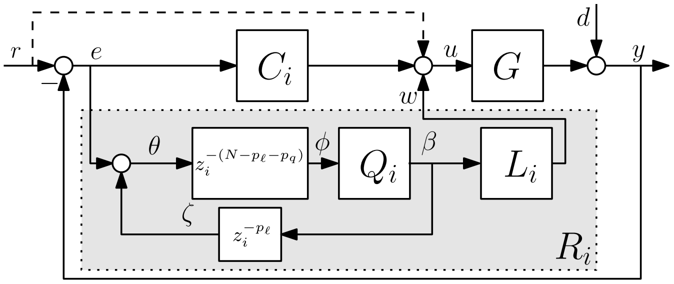

Fig. 2. Block diagram of the plant G, feedback controller C_i, and a parallel repetitive controller R_i. The index i indicates that in the triggered RC setting the controller blocks are potentially different in the active and idle phases. Switching between these states can have different causes; in mechanical ventilation this is the detection of the disturbance d. The dashed arrow is included in the ventilation example without loss of generality of the framework.

1) Simulation Use-Case: In this simulation, PC-ACV in the mechanical ventilation application is adopted as a use-case and a conventional parallel repetitive controller structure is adopted as presented in Fig. 2. The repetitive controller filters in this example are designed following the design methodology in [24]. However, in the current article a parallel structured RC is adopted, for which the design approach in [24] is appropriately modified. In the ventilation use-case, hose-compensation control is included in the plant G; as a consequence, the feedback controller C_i is set to zero. Therefore, plug-in RC as in [24], where the RC output is processed by the controller $C_i = 0$, cannot be used directly, since the RC output would be nullified. Therefore, parallel RC as visualized by the block diagram in Fig. 2 is adopted. To handle the varying times in between tasks, i.e., breaths, the RC is turned on at the start of the active phase and turned off at the end of the active phase. During the idle phase, the internal states of the repetitive controller are kept constant and its output is set to zero. This represents an intuitive RC design for such scenario.

2) Time-Domain Results: The time-domain results of the simulation are depicted in Fig. 3. The figures show three separate tasks in the simulations, i.e., the first, sixth, and 20th tasks. During these tasks, it shows the reference (or target pressure), the output (or measured airway pressure), the RC output w, and a scaled version of the disturbance (or patient effort). The figure shows that the tracking performance during the tasks significantly improves over subsequent tasks. However, at the transition from the active to the idle phase a problem is observed. The RC output in the bottom plot of all three figures increases at the end of the active breath. This results in an increasing undesired peak in the measured output at the start of the idle phase; see the blue line in the top figure of the sixth and 20th tasks. This spike in the measured output is seen in the idle phase because of the output delay, i.e., the output does not instantly respond to the control signal. This spike in the output deteriorates control performance at the start of the idle phase and is therefore undesired.

3) Challenges: The spike in the RC output is a result of the noncausal filters and errors at the beginning of the task. These noncausal filters allow a conventional RC implementation to start actuation $p_l + p_q$ samples before the end of task j. More specifically, if an error is entering the memory block, $z_i^{-(N-p_\ell-p_q)}$ in Fig. 2; at the beginning of the task, it propagates through the causal filters Q_i and L_i before the Nth time step,

[1] In the case of PC-ACV, d reflects the patient-induced disturbance on the airway pressure due to a spontaneous breathing action.

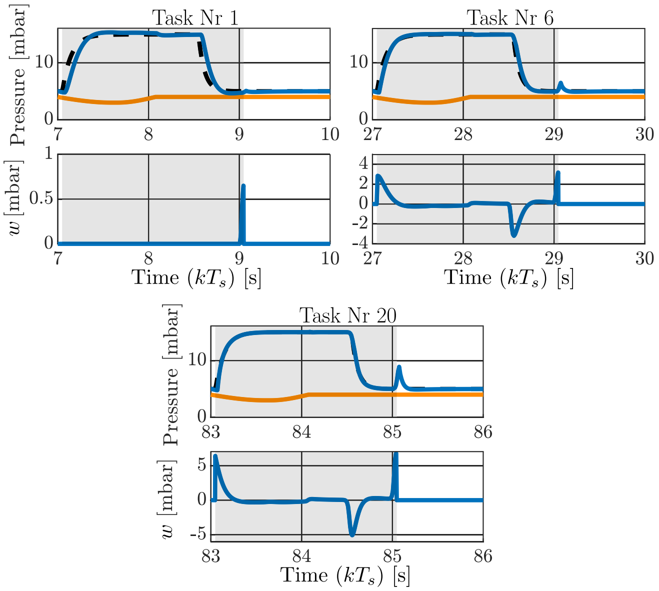

Fig. 3. Simulation results of PC-ACV showing conventional repetitive control that is turned on and off upon the trigger of the active phases, which is indicated by the gray area. The figures shows the reference r or p_{target} (- - -), the output y or $\tilde{p}_{\text{aw}}$ and RC output w (—), and a scaled disturbance d or p_{mus} (—). Showing that conventional RC in a triggered setting results in an undesired actuation spike at the end of the task.

i.e., before the end of the task. This allows the control output w to increase before the end of the task to compensate the error at the start of the subsequent task. This actuation can eliminate the tracking error at the start of task $j+1$ in the conventional RC. However, the task $j+1$ does not start immediately after the end of the active phase of task j due to the varying and uncertain length $N_{\epsilon,j}$ of the idle phase. Therefore, this actuation results in an undesired peak at the start of the idle phase, which is increasing over subsequent breaths.

Two distinct causes for the error at the beginning of a task, and therewith causes of the actuation spike, are identified. The first cause of the error at the start of the active phase is the immediate ramp-up of the reference r, i.e., the target pressure. Because the system output does not respond to this instantaneously, this results in an error at the start of the active phase. The second cause of the error at the start of the active phase is the disturbance d, which causes a slight dip in the output y. This disturbance and its timing are unknown and the resulting error cannot be compensated perfectly. To overcome these challenges, a triggered RC framework is developed, which is presented in Section III.

III. TRC Framework

In this section, the TRC framework is presented and its theoretical support is established. The main contribution of this article, i.e., the TRC framework, is presented in Section III-A. Thereafter, a stability proof for the controlled system is presented in Section III-B.

A. TRC Framework

In this section, the exact implementation of the TRC framework is explained. This means that for both the idle and active phases the content of the building blocks in R_i of Fig. 2 is explained. Note that this section does not contain the design of the feedback controller C_i and the exact TRC filter design method. For the mechanical ventilation use-case, this is explained in Section V-C. Next, the RC building blocks during the active phase, $i = 1$, and thereafter the building blocks in the idle phase, $i = 2$, are explained.

1) Active Phase: In the active phase, the learning filter L_1 and the robustness filter Q_1 are designed similar to the conventional parallel RC; see Section V-C for a detailed design methodology of L_1 and Q_1 for the mechanical ventilation use-case.

To avoid that the error at the start of active phase j causes an actuation spike at the end of the active phase j, this initial error should not be propagated through robustness and learning filters. To avoid that the immediate increase in the reference causes an error, the reference is delayed, such that it does not change immediately when the task is started. This allows the RC to respond to the change in reference at the start of the task, eliminating this error.

Furthermore, it should be avoided that the error caused by the disturbance d causes an actuation spike. This is achieved by altering the delay blocks, i.e., memory block $(z_i^{-(N-p_l-p_q)})$ and preview buffer $(z_i^{-p_l})$, in Fig. 2. The memory block is changed such that it behaves as a pure delay in the first $N - p_\ell - p_q$ samples in a task. Thereafter, it holds its last output but keeps updating its internal states, preventing that the error at the beginning of the task is propagated to the learning and robustness filter. To achieve this, the main memory block $z_1^{-(N-p_\ell-p_q)}$, with input $\theta(k)$, is replaced by the following switching state-space system during task j:

$$x_Z(k+1) = A_Z x_Z(k) + B_Z \theta(k)$$
$$\phi(k) = \begin{cases} C_Z x_Z(k) + D_Z \theta(k) \\ \qquad\qquad \text{for } k - k_{s,j} < N - p_\ell - p_q \\ \phi(k-1), \quad \text{for } N - p_\ell - p_q \le k - k_{s,j} \le N \end{cases} \tag{1}$$

where

$$A_Z = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ & & & \\ & I_{N-p_\ell-p_q-1} & & \vdots \\ & & & 0 \end{bmatrix}, \quad B_Z = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$
$$C_Z = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}, \quad D_Z = 0 \tag{2}$$

where $k_{s,j}$ is defined as the sample number k at the start of task j, i.e., $k_{s,j} = (j-1)N + \sum_{m=1}^{n-1} N_{\epsilon,m} + 1$.

In addition, the preview buffer $z_1^{-p_\ell}$ in Fig. 2 has a contribution to the actuation spike. Therefore, the preview buffer is altered such that it is a direct feedthrough at times $k - k_{s,j} \in [N - p_\ell, N]$, during every active phase. This means that at the end of the task both the learning and robustness filters are causal filters and no actuation spike will occur. This is implemented using the following switching state-space system for the preview buffer, with input $\beta(k)$, in the active phase during task j:

$$x_P(k+1) = A_P x_P(k) + B_P \beta(k)$$
$$\zeta(k) = \begin{cases} C_P x_P(k) + D_P \beta(k), & \text{for } k - k_{s,j} < N - p_\ell \\ \beta(k), & \text{for } N - p_\ell \leq k - k_{s,j} \leq N \end{cases} \tag{3}$$

where

$$A_P = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ & & & \vdots \\ & I_{N-p_\ell-1} & & \\ & & & 0 \end{bmatrix}, \quad B_P = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$
$$C_P = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}, \quad D_P = 0. \tag{4}$$

This preview buffer behaves as a pure delay during the first $N - p_\ell$ samples of a task. Thereafter, it behaves as a direct feedthrough and its internal states are updated.

2) Idle Phase: During the idle phase, i.e., $k - k_{s,j} \in [N+1, N+N_{\epsilon,j}]$, the design of the different building blocks of R_2 is straightforward. During this phase, it is desired to hold the internal states of all the blocks, such that the controller remembers what it learned during previous active phases. Therefore, the state-space models of the four building blocks of R_2 in Fig. 2 all have a system matrix A, i.e., A_Z, A_P, A_Q, and A_L, equal to an identity matrix of the proper size. All other system matrices of the four building blocks of R_2 in Fig. 2 contain only zeros in the idle phase. Using these state-space models, the states of the different blocks are held and no new information is going in or out of R_2.

B. Stability of the Closed-Loop System

In this section, a stability proof for the closed-loop controlled system with TRC is presented. Because the TRC strategy contains a robustness filter Q, the error dynamics will not converge exactly to zero for time-varying references. Hence, convergence of the error dynamics to zero cannot be shown for the time-varying references which are considered in RC. The stability proof in this section provides a method to guarantee asymptotic stability of the unforced system and it guarantees that the closed-loop system has (asymptotically) bounded states for bounded inputs, i.e., it exhibits the input-to-state stability (ISS) property. The stability proof follows three main steps.

1) The closed-loop system (with TRC) is written as a switching system.
2) This switching system is rewritten to a new switching system over successive breaths; this new switching system describes the relationship between the plant and controller states at the start of task j and those at the start of task $j+1$.
3) This new switching system representation is used to prove (input-to-state) stability using a common quadratic Lyapunov function (CQLF).

First, the closed-loop system within a single task is described by the block diagram in Fig. 2. For generality, the disturbance and the dashed arrow are omitted. This closed-loop system is written as a switching state-space model as follows:

$$\left.\begin{aligned} x(k+1) &= A_l x(k) + B_l r(k) \\ y(k) &= C_l x(k) + D_l r(k) \end{aligned}\right\}, \quad \text{for } l \in 1, 2, 3, 4 \tag{5}$$

where x represents the states of all different blocks in Fig. 2, i.e.,

$$x = \begin{bmatrix} x_G & x_L & x_Q & x_C & x_Z & x_P \end{bmatrix}^T \tag{6}$$

where the subscripts in the state vector represent the different building blocks, i.e., G represents the plant, L the learning filter, Q the robustness filter, C the linear feedback controller, Z the main buffer, and P the preview buffer. The corresponding state-space system matrices are retrieved by combining the state-space models of all the subsystems and are given in the Appendix . In the state-space system (5), $l = 1, 2, 3$ represent the system in the active phase and $l = 4$ represents the system in the idle phase. More specifically, during breath j,

$$\begin{aligned} l &= 1 \quad \forall\, k - k_{s,j} \in [1, N - p_\ell - p_q - 1] \\ l &= 2 \quad \forall\, k - k_{s,j} \in [N - p_\ell - p_q, N - p_\ell - 1] \\ l &= 3 \quad \forall\, k - k_{s,j} \in [N - p_\ell, N] \\ l &= 4 \quad \forall\, k - k_{s,j} \in [N+1, N+N_{\epsilon,j}]. \end{aligned}$$

Second, the closed-loop dynamics are written as a state-space model over multiple breaths. This state-space model represents the relationship between the system state $\gamma(j) := x(k_{s,j})$, i.e., system and controller states, at the start of active phase j, and the system state $\gamma(j+1) := x(k_{s,j+1})$, i.e., system and controller states, at the start of the next active phase $j+1$. The new closed-loop switching system is obtained using the knowledge that the system always switches in a fixed order and that the closed-loop system matrices in these distinct phases are known. This results in the following closed-loop state-space model:

$$\gamma(j+1) = A(N, N_{\epsilon,j})\gamma(j) + B(N, N_{\epsilon,j})\underline{r}(j) \tag{7}$$

where $\underline{r}(j) = [r(k_{s,j}) \cdots r(k_{s,j}+N+N_{\epsilon,j})]^T$ is the bounded input vector (concerning the reference) and the system matrices are defined in (8), as shown at the bottom of the next page. Note that the system matrices in (7) depend on the length of the idle phase, which in turn varies over j. Hence, a switching system is retrieved where the system matrices depend on the length of the idle phase, i.e., $N_{\epsilon,j}$.

Third, the switching state-space system on task level in (7) is used to prove stability of the closed-loop system. For this stability proof, it is assumed that the idle phase has a known maximum duration in samples $N_{\max}$, i.e., $N_{\epsilon,j} \in \{0, 1, \ldots, N_{\max}\}$. Furthermore, it is assumed that the value $N_{\epsilon,j}$ can change arbitrarily within this set from task to task. Using this, we can prove that the controlled system is (robustly) exponentially stable for $\underline{r}(j) = 0$ and it has bounded a state evolution for bounded inputs for bounded inputs $\underline{r}(j)$ by following Proposition 1. If the conditions of this proposition are satisfied, it also implies stability of (5); see Remark 1.

Proposition 1: Consider the closed-loop switching system (7) with arbitrary switching of $N_{\epsilon,j} \in \{0, 1, \ldots, N_{\max}\}$, for all

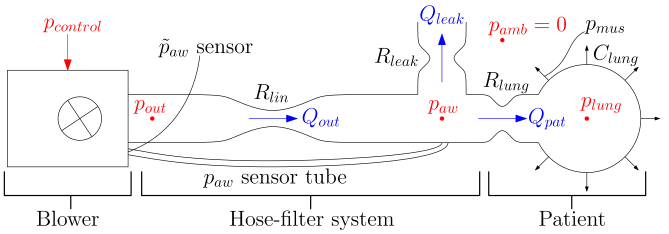

Fig. 4. Schematic representation of the blower–hose–patient system of the considered positive pressure ventilation system. Showing the resistances, lung compliance, flows, and pressures.

j, and the discrete-time Lyapunov equation ([34, Sec. 8.6]):

$$\underbrace{A(N, N_{\epsilon,j})^T P A(N, N_{\epsilon,j}) - P}_{:=D_\ell(N,N_{\epsilon,j})} \prec -\alpha P^T \quad \forall\ N_{\epsilon,j} \in \{0, 1, \ldots, N_{\max}\}. \tag{9}$$

If there exists a matrix $P = P^T \succ 0$ and an $\alpha > 0$ such that the linear matrix inequalities in (9) are satisfied, then the origin of (7) is globally exponentially stable for $\underline{r}(j) = 0$. In addition, for $\underline{r}(j) \neq 0$, (7) is input-to-state stable with respect to the input $\underline{r}$.

Proof: The proof follows standard line of reasoning for ISS stability analysis of (switching) discrete-time linear systems, such as, e.g., in [35] and [36]. □

Remark 1: Note that if the conditions in Proposition 1 are satisfied, this also implies as follows.

1) For $r = 0$, the states of (5) converge to zero.
2) For $r \neq 0$, the state evolution of (5) is bounded.

IV. Ventilation System and Patient Effort Modeling

In this section, models of the ventilation system and patient effort are presented. In Section IV-A, a high-level description of the ventilation system is presented. Then, in Section IV-B, models of the considered ventilation system are presented. First, a first principles model is derived that is used in the closed-loop stability analysis in Section V-D. Then, an experimental frequency response function (FRF) model is presented that is used for the TRC filter design in Section V-C. Thereafter, in Section IV-C, the considered patient effort model is given.

A. High-Level System Description

A schematic of the considered blower–patient–hose system, with the relevant parameters, is shown in Fig. 4. The main components are the blower, the hose-filter system, and the patient.

1) Blower: A centrifugal blower compresses ambient air to achieve the desired blower outlet pressure p_{out}. The blower system is modeled as a second-order low-pass filter with an output delay τ_b. This describes the relationship between the control output p_{control} and the outlet pressure p_{out}.

2) Hose-Filter System: The hose-filter system connects the blower to the patient. The difference between p_{out} and p_{aw} results in a flow through the hose Q_{out}. This pressure difference and flow relationship are modeled by a linear hose resistance R_{lin}. The change in airway pressure p_{aw} results in two flows, namely, the leak flow Q_{leak} and the patient flow Q_{pat}. The leak flow is used to flush exhaled CO$_2$-rich air from the hose. The patient flow is required to ventilate the patient. The airway pressure is measured using a pilot line attached to the module and the end of the hose. The pressure transfer through the hose and the pilot line results in delays in the system, caused by the finite propagation speed of the pressure waves. This delay is assumed to be a measurement delay of p_{aw} referred to as the hose delay τ_h. For simplicity, all the system delays are lumped as one output delay $\tau_d = \tau_b + \tau_h$ on the measurement of the airway pressure. The measured, i.e., delayed, airway pressure is denoted by $\tilde{p}_{\text{aw}}$.

3) Patient: The patient is modeled by a linear one-compartmental lung model as described in [37, pp. 37–60]. This model consists of a linear resistance R_{lung} and a linear compliance C_{lung}. The patient flow is a result of the lung resistance and the difference between the airway pressure and the lung pressure p_{lung}, i.e., the pressure inside the lungs. The patient flow results in a change in the lung pressure, and the relationship between patient flow and lung pressure is given by the lung compliance. Finally, in this article, a patient with spontaneous, muscle-induced, breathing effort $p_{\text{mus}}(t)$ is considered. The patient effort is an unknown disturbance introduced by the patient's spontaneous breathing effort that affects the lung pressure. This patient effort allows a patient to (partially) inhale and exhale by themselves.

B. Ventilation System Model With Hose Compensation

In this section, a linear state-space model of the ventilation system is presented. First, a model of the ventilation system is presented. Thereafter, a state-of-the-art control strategy, i.e., hose-compensation control [10], is included in the model to obtain the full mechanical ventilation model. This model can be used as the plant G in the stability analysis of Section V-D. Thereafter, experimental FRFs of this controlled plant are presented. These FRF models are used for design of the RC filters in Section V-C.

The ventilation system consists of a blower, hose-filter system, and the patient itself, as shown in Fig. 4. First, the blower is modeled as a second-order low-pass filter with cutoff

$$\begin{aligned} A(N, N_{\epsilon,j}) &= A_4^{N_{\epsilon,j}} A_3^{p_\ell} A_2^{p_q} A_1^{N-p_\ell-p_q} \\ B(N, N_{\epsilon,j}) &= \Big[A_4^{N_{\epsilon,j}} A_3^{p_\ell} A_2^{p_q} A_1^{N-p_\ell-p_q-1} B_1 \quad A_4^{N_{\epsilon,j}} A_3^{p_\ell} A_2^{p_q} A_1^{N-p_\ell-p_q-2} B_1 \quad \cdots \quad A_4^{N_{\epsilon,j}} A_3^{p_\ell} A_2^{p_q} B_1 \quad A_4^{N_{\epsilon,j}} A_3^{p_\ell} A_2^{p_q-1} B_2 \quad \cdots \\ &\qquad A_4^{N_{\epsilon,j}} A_3^{p_\ell} B_2 \quad A_4^{N_{\epsilon,j}} A_3^{p_\ell-1} B_3 \quad \cdots \quad A_4^{N_{\epsilon,j}} B_3 \quad A_4^{N_{\epsilon,j}-1} B_4 \quad \cdots \quad B_4 \Big] \end{aligned} \tag{8}$$

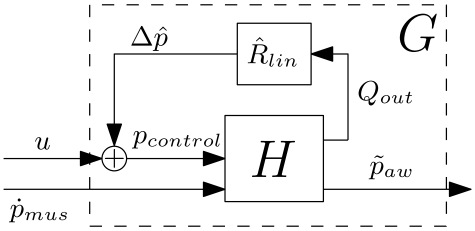

Fig. 5. Schematic overview of the open-loop ventilation system H combined with hose-compensation control strategy, combined this results in the plant G in Fig. 2.

frequency $\omega_n = 30$ Hz. This gives the following state-space model of the blower:

$$\begin{aligned}\dot{x}_b &= \mathbf{A_b} x_b + \mathbf{B_b} p_{\text{control}}\\ p_{\text{out}} &= \mathbf{C_b} x_b\end{aligned} \tag{10}$$

with

$$\mathbf{A_b} = \begin{bmatrix} -\omega_n & 0 \\ \omega_n & -\omega_n \end{bmatrix}, \quad \mathbf{B_b} = \begin{bmatrix} \omega_n \\ 0 \end{bmatrix}, \quad \mathbf{C_b} = \begin{bmatrix} 0 & 1 \end{bmatrix} \tag{11}$$

where p_{control} is the input for the blower, i.e., the total control signal, x_b are the blower states, and p_{out} is the blower outlet pressure.

Second, the hose-filter patient system is obtained by assuming that all the resistances are linear and that the patient is modeled by a linear one-compartment lung model. Detailed derivation of this model can be found in [10]. The different components, i.e., hose, leak, and patient, are connected by conservation of flow, i.e., $Q_{\text{out}} = Q_{\text{pat}} + Q_{\text{leak}}$. This results in the following linear state-space patient–hose model:

$$\begin{aligned}\dot{p}_{\text{lung}} &= \mathbf{A_p} p_{\text{lung}} + \mathbf{B_p} p_{\text{out}} + \mathbf{E_p} \dot{p}_{\text{mus}}\\ p_{\text{aw}} &= \mathbf{C_p} p_{\text{lung}} + \mathbf{D_p} p_{\text{out}}\end{aligned} \tag{12}$$

with

$$\begin{aligned}&\mathbf{A_p} = -\frac{R_{\text{lin}} + R_{\text{leak}}}{C_{\text{lung}} \bar{R}}, \quad \mathbf{B_p} = \frac{R_{\text{leak}}}{C_{\text{lung}} \bar{R}}, \quad \mathbf{C_p} = \frac{R_{\text{lin}} R_{\text{leak}}}{\bar{R}}\\ &\mathbf{D_p} = \frac{R_{\text{leak}} R_{\text{lung}}}{\bar{R}}, \quad \mathbf{E_p} = 1\\ &\text{and } \bar{R} = R_{\text{lin}} R_{\text{leak}} + R_{\text{lin}} R_{\text{lung}} + R_{\text{leak}} R_{\text{lung}}.\end{aligned} \tag{13}$$

Discrete-time models are used for RC design and the stability analysis. Therefore, the blower and the hose-filter patient system models are discretized using exact zero-order hold discretization with a sampling time $T_s = 2 \times 10^{-3}$ s. The associated discrete-time system matrices are denoted with an additional subscript d.

Third, a discrete-time model for the output delay τ_d is obtained. This model represents the delay between the actual airway pressure $p_{\text{aw}}(k)$ and the measured airway pressure $\tilde{p}_{\text{aw}}(k)$. For this model, the delay length is computed in terms of sampling time T_s. This gives the delay length in samples $d_s = (\tau_d / T_s)$. The following discrete-time state-space model represents the output delay z^{-d_s}:

$$\begin{aligned}x_{\tau_d,d}(k+1) &= \mathbf{A_{\tau_d,d}} x_{\tau_d,d}(k) + \mathbf{B_{\tau_d,d}} p_{\text{aw}}(k)\\ \tilde{p}_{\text{aw}}(k) &= \mathbf{C_{\tau_d,d}} x_{\tau_d,d}(k)\end{aligned} \tag{14}$$

with

$$\begin{aligned}&\mathbf{A_{\tau_d,d}} = \begin{bmatrix} 0 & \cdots & 0 \\ & \ddots & \vdots \\ I_{d_s-1} & & 0 \end{bmatrix}, \quad \mathbf{B_{\tau_d,d}} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}\\ &\mathbf{C_{\tau_d,d}} = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}.\end{aligned} \tag{15}$$

In this state-space model, the actual airway pressure $p_{\text{aw}}(k)$ first propagates for d_s samples through the state $x_{\tau_d,d}$ before it is seen in the output $\tilde{p}_{\text{aw}}$.

Next, the discretized models in (10)–(15) are combined. Using the combined state $x_{m,d} = \begin{bmatrix} x_{b,d}^T & p_{\text{lung},d} & x_{\tau_d,d}^T \end{bmatrix}^T$, gives the following discrete-time state-space model of the ventilation system H:

$$\begin{aligned}x_{m,d}(k+1) &= \mathbf{A_{m,d}} x_{m,d}(k) + \mathbf{B_{m,d}} p_{\text{control}}(k) + \mathbf{E_{m,d}} \dot{p}_{\text{mus}}(k)\\ \tilde{p}_{\text{aw}}(k) &= \mathbf{C_{m,d}} x_{m,d}(k)\end{aligned} \tag{16}$$

with

$$\begin{aligned}&\mathbf{A_{m,d}} = \begin{bmatrix} \mathbf{A_{b,d}} & 0 & 0 \\ \mathbf{B_{p,d}} \mathbf{C_{b,d}} & \mathbf{A_{p,d}} & 0 \\ \mathbf{B_{\tau_d,d}} \mathbf{D_{p,d}} \mathbf{C_{b,d}} & \mathbf{B_{\tau_d,d}} \mathbf{C_{p,d}} & \mathbf{A_{\tau_d,d}} \end{bmatrix}\\ &\mathbf{B_{m,d}} = \begin{bmatrix} \mathbf{B_{b,d}} \\ 0 \\ 0 \end{bmatrix}, \mathbf{C_{m,d}} = \begin{bmatrix} 0 & 0 & \mathbf{C_{\tau_d,d}} \end{bmatrix}, \quad \mathbf{E_{m,d}} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.\end{aligned} \tag{17}$$

Finally, linear hose compensation is added to the ventilation system to retrieve the desired plant model. The hose-compensation strategy in this article is a special case of the strategy in [10]. A schematic overview of the ventilation system H with hose compensation is depicted in Fig. 5. Hose-compensation control is used to compensate the pressure drop over the hose; in combination with unit feedforward, i.e., $u := p_{\text{target}}$ in Fig. 5, it theoretically results in zero tracking error according to [10]. Fig. 5 shows how hose compensation is added to the control signal u and included to the ventilation system H represented by (16) and (17). This results in the controlled plant G represented by (19) below. This compensation is implemented as follows:

$$p_{\text{control}} = u + \Delta\hat{p} = u + \hat{R}_{\text{lin}} Q_{\text{out}}. \tag{18}$$

To obtain the dynamics of G, Q_{out} is rewritten as $Q_{\text{out}} := ((p_{\text{out}} - p_{\text{aw}})/R_{\text{lin}})$, where $p_{\text{out}} = \mathbf{C_{b,d}} x_{b,d}$ and $p_{\text{aw}} = \mathbf{C_{p,d}} p_{\text{lung},d} + \mathbf{D_{p,d}} \mathbf{C_{b,d}} x_{b,d}$, and (18) is substituted in (16) and (17). This results in

$$\begin{aligned}x_{f,d}(k+1) &= \mathbf{A_{f,d}} x_{f,d}(k) + \mathbf{B_{f,d}} p_{\text{target}}(k) + \mathbf{E_{f,d}} \dot{p}_{\text{mus}}(k)\\ \tilde{p}_{\text{aw}}(k) &= \mathbf{C_{f,d}} x_{f,d}(k)\end{aligned} \tag{19}$$

where $x_{f,d} = x_{m,d}$

$$\begin{aligned}&\mathbf{A_{f,d}} = \mathbf{A_{m,d}} + \mathbf{B_{m,d}} \frac{\hat{R}_{\text{lin}}}{R_{\text{lin}}} \begin{bmatrix} \mathbf{C_{b,d}} - \mathbf{D_{p,d}} \mathbf{C_{b,d}} & -\mathbf{C_{p,d}} & 0 \end{bmatrix}\\ &\mathbf{B_{f,d}} = \mathbf{B_{m,d}}, \quad \mathbf{C_{f,d}} = \mathbf{C_{m,d}}, \quad \text{and } \mathbf{E_{f,d}} = \mathbf{E_{m,d}}.\end{aligned} \tag{20}$$

In the remainder of this article, the plant G, described by (19) and (20), is used as the open-loop plant. This is the ventilation model including this hose-compensation strategy. Note that the

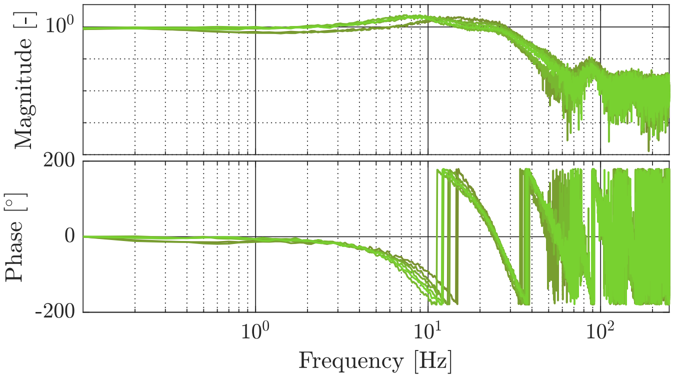

Fig. 6. Identified FRF models of the ventilation system with hose-compensation control from u to $\tilde{p}_{\text{aw}}$, i.e., the plant G. The different FRFs represent different patients from babies to adults at different pressure levels.

reference pressure p_{target} corresponds to the reference r in the triggered RC framework of Section III, i.e., $r := p_{\text{target}}$.

For the RC filter design, the experimental FRF models of the ventilation system with hose-compensation control are obtained. More specifically, FRFs from u to $\tilde{p}_{\text{aw}}$ for different patients are depicted in Fig. 6. In these experiments, the estimated hose resistance $\hat{R}_{\text{lin}}$ is retrieved by a calibration procedure. The figure shows FRFs of different patients from babies to adults. Furthermore, these FRFs are obtained at different pressure levels, to capture the effect of the absolute pressure on the FRF. The significant phase lag that is seen in the FRFs is caused by the output delay τ_d. These FRFs are used to design the RC filters in Section V-C.

C. Patient Effort Modeling

The patient's spontaneous effort p_{mus} enables a person, i.e., healthy or patient, to inhale and exhale air by themselves. Physically, the patient effort can be seen as a change in lung pressure caused by contractions and relaxation of the respiratory muscles, e.g., the diaphragm.

According to [38], a sinusoidal half-wave is a common and accurate model of the patient effort. An example of the sinusoidal half-wave is depicted in Fig. 7. The decrease in p_{mus} results in a decrease in the lung pressure, which results in flow into the patient lungs. In other words, a decrease in p_{mus} represents an inspiration. Thereafter, the increase in pressure results in an increase in the lung pressure and flow out of the lungs. In other words, an increase in p_{mus} represents an expiration.

In this article, it is assumed that the patient effort is an exogenous disturbance to the system. In practice, it is a result of the breathing behavior of the patient. Modeling of this behavior is out of scope for this article.

V. TRC Applied to Mechanical Ventilation

In this section, the proposed TRC framework of Section III is used to improve the pressure tracking performance of an experimental mechanical ventilation setup. First, in Section V-A, the experimental setup and use-cases are described. Thereafter, in Section V-B, the overall controller design is presented. Next, the RC filter designs are explained in Section V-C. Then, the stability of the closed-loop system is analyzed in Section V-D. Finally, in Section V-E, the experimental results are presented and compared with a benchmark control strategy.

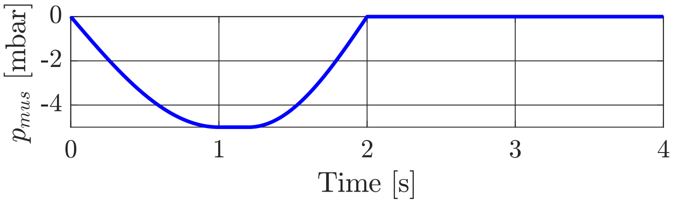

Fig. 7. Model of the muscle-induced patient effort $p_{\text{mus}}(t)$, defined by the sinusoidal half-wave.

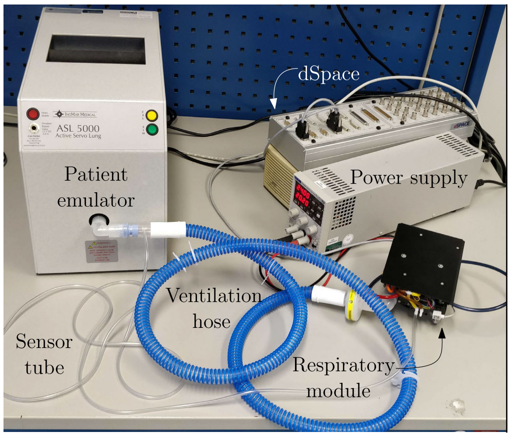

Fig. 8. Experimental setup consisting of the blower-driven ventilator, ASL 5000 breathing simulation, dSpace, and a hose.

A. Experimental Setup and Use-Cases

The main components of the experimental setup used in this case study are depicted in Fig. 8. The figure shows a Macawi blower-driven mechanical ventilation module (DEMCON Macawi respiratory systems, Best, The Netherlands). Furthermore, the ASL 5000 Breathing Simulator (IngMar Medical, Pittsburgh, PA) is shown in the figure. This breathing simulator is used to emulate a linear one-compartmental patient model as described in [37]. Furthermore, a typical hose-filter system for ventilation of a patient in a hospital setting is shown. The control and ventilation algorithms are implemented in a dSPACE system (dSPACE GmbH, Paderborn, Germany).

To design and evaluate TRC for mechanical ventilation, three different patients and ventilation scenarios are considered. The considered patient scenarios are a baby, pediatric, and adult scenario from the ISO standard for pressure-controlled mandatory ventilation obtained from Table 201.104 in NEN-EN-ISO 80601-2-12:2011 (NEN, Delft, The Netherlands). For these standardized scenarios, the patient parameters and the ventilator settings are given in Table I. Furthermore, the patient effort p_{mus} for all three scenarios is depicted in Fig. 9. The variations in this patient effort are exaggerated to analyze the control performance in a worst case scenario. As a benchmark controller, the hose

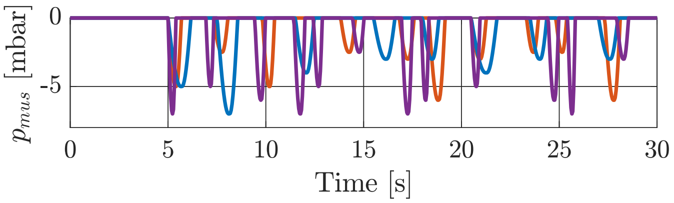

Fig. 9. Considered patient effort for all the cases during the first 30 s. Showing the breathing effort of the adult (▬), the pediatric (▬), and the baby (▬) use-cases.

TABLE I

Patient Parameters and Ventilation Settings Used for Filter Design and in the Experiments

Parameter	Adult	Pediatric	Baby	Unit
R_{lung}	5	50	50	mbar s / L
C_{lung}	50	10	3	mL/mbar
PEEP	5	5	10	mbar
IPAP	15	35	25	mbar
Inspiratory time	1.5	1	0.6	s
N	1000	650	450	samples

resistance compensation strategy, as discussed in Section IV, is considered. This corresponds to the first breath in every use-case, when the TRC has not yet learned from previous breaths.

B. Overall Controller Design

The overall controller design can be divided into four separate controllers, i.e., feedforward control, hose-compensation control, linear feedback control, and the triggered repetivie controller R. As a basis, the control scheme in Fig. 2 is considered. First, as a feedforward controller, unity feedforward is considered, i.e., the dashed arrow in Fig. 2. Next, the hose resistance compensation controller is implemented to achieve adequate performance from the start, i.e., before the triggered repetitive controller has learned from previous breaths. Implementation of the hose resistance controller results in the plant G described by (19). This control strategy ensures that the initial performance of the system is sufficient. The estimated hose resistance $\hat{R}_{\text{lin}}$ is obtained by an offline calibration of the hose prior to ventilation.

The linear feedback controller C_i in Fig. 2 is set to be 0 in both the active and idle phases, i.e., $C_1 = C_2 = 0$. This choice for C_i is made because the hose resistance compensation controller in G combined with unity feedforward already achieves sufficient performance during the first breath and during the idle phase.

Finally, the triggered repetivie controller R_i is designed using the framework presented in Section III. The exact learning and robustness filter design is discussed next.

C. Repetitive Control Filter Design

The filters for the triggered repetitive ctonroller during the active phase are designed following the methodology of [24]. More specifically, this means that the learning filter L_1 in Fig. 2 is based on an average FRF measurement of the plant G for several patients. Furthermore, the robustness filter Q_1 in Fig. 2 is designed such that it ensures stability of the conventional RC, i.e., *without switching*, using the separate FRFs of all the considered patients.

Before designing the RC filters for the mechanical ventilation system, a stability result and a design procedure are given. First, the sensitivity of the controlled system in Fig. 2, without the dashed arrow, during the active phase is defined as follows:

$$e = \underbrace{(1 + C_1 G)^{-1}}_{S_1} \underbrace{(1 + P_{S,1} R_1)^{-1}}_{S_{R,1}} (r - d) \tag{21}$$

where $P_{S,1} = (1 + C_1 G)^{-1} G$. Using this sensitivity, the following sufficient stability theorem is obtained for the system in the active phase, without switching to the idle phase. This theorem ensures that the transfer function $S_{R,1}$ is asymptotically stable, and hence, the sensitivity function in (21) is asymptotically stable.

Theorem 1 (Based on the MIMO Plug-In RC Stability Results in [19]): Assume that S_1 and $P_{S,1}$ are asymptotically stable. Then, $S_{R,1}$ is asymptotically stable for all N if

$$|Q_1(z)(1 - P_{S,1}(z) L_1(z))| < 1 \quad \forall z = e^{i\omega},\ \omega \in [0, 2\pi). \tag{22}$$

This stability theorem ensures that the entire loop in Fig. 2 is asymptotically stable if no switching to the idle state occurs.

Using the stability condition in Theorem 1, the following two-step design procedure is followed for the single-input single-output (SISO) RC systems; see [17], [30], [39], and [33].

Procedure 1 (Based on Frequency-Domain SISO RC Design, From [33] and [24]).

1) Given a parametric model of the "nominal" process sensitivity $P_{S,1}(z)$, construct $L_1(z)$ as an approximate, possibly noncausal, stable inverse of $P_{S,1}(z)$, i.e., $L_1(z) \approx P_{S,1}^{-1}(z)$.
2) Using nonparametric FRF models, $P_{S,1}^{p}(e^{i\omega})$, $p \in \{1, \dots, N_p\}$ with N_p the number of patient models, of different patients, design one $Q_1(z)$ such that Theorem 1 is satisfied for $P_{S,1}^{p}(e^{i\omega})$ $\forall p \in \{1, \dots, N_p\}$.

To design the RC filters for the active phase, Procedure 1 is followed. First, FRF measurements of the process sensitivity $P_{S,1}^{p}(z)$ for every patient are obtained; see Fig. 10. The average of these FRF measurements and a fourth-order fit with 12 samples delay are used to obtain a parametric model $P_{S,1}^{n}(z)$ of the "nominal" process sensitivity; see Fig. 10. This fit of the "nominal" process sensitivity is used to construct a noncausal learning filter $L_1(z) = z^{p_l} L_c(z)$ as an approximate stable inverse of $P_{S,1}^{n}(z)$. This inverse is obtained using zero phase error tracking control (ZPETC) [40].

For the robustness filter design, the second step in Procedure 1 is followed. First, the stability condition of the conventional repetitive controller is checked for $Q_1 = 1$. In the left-hand side of Fig. 11, it is clearly seen that the stability conditions for conventional RC are not guaranteed for all the patients. Therefore, a 20th-order noncausal zero-phase finite impulse response (FIR) filter with a cutoff frequency of 20 Hz is implemented as the robustness filter Q_1. This FIR filter is implemented by computing a causal symmetric FIR

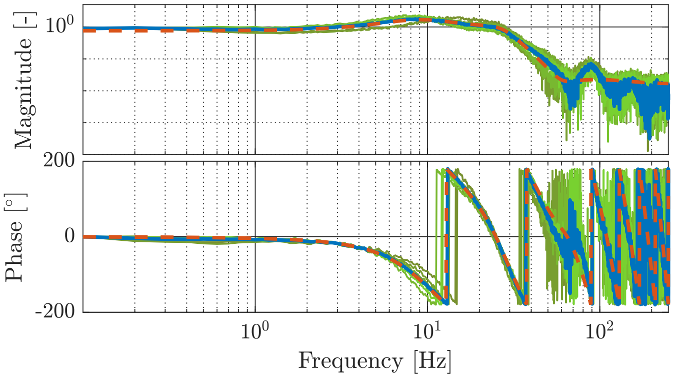

Fig. 10. FRF measurements of process sensitivity for the individual patients (━), the "nominal" (average) process sensitivity (━), and the fourth-order fit of the "nominal" process sensitivity (- - -).

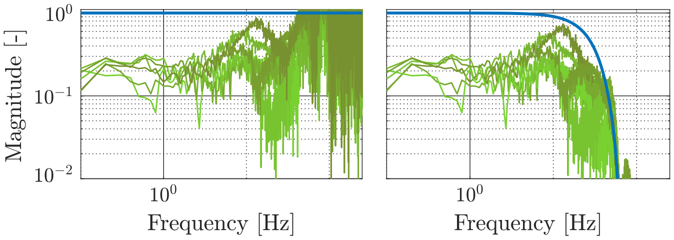

Fig. 11. Left: stability condition for all the patients with $Q_1 = 1$. Right: stability condition for all the patients with Q_1 a 20th-order FIR filter with cutoff frequency at 20 Hz. The figures show $(|Q_1(1 - P_{S,1}L_1)|)$ for every patient (━) and the corresponding Q filters (━).

filter $Q_{1,c}$ and applying a forward shift of z^{p_q} with p_q half the order of the FIR filter. This makes it a zero-phase FIR filter that is symmetric around zero lag, such that no phase lag is introduced by the filter. The forward shift is possible because of the memory loop, as long as $p_l + p_q \leq N$. With the implementation of this robustness filter, the conventional RC stability condition in Theorem 1 is ensured, as is shown in the right-hand side Fig. 11.

Eventually, to reduce the effect of the varying disturbances, e.g., breathing effort, in subsequent breaths a learning gain $\alpha \in [0, 1]$ is included by multiplying L_1 with α. The choice of the learning gain is a tradeoff between convergence speed and varying disturbance suppression, and this varying disturbance is mainly caused by the patient effort in the ventilation use-case. A high learning gain means that the controller converges quickly and responds abruptly to the changes in the patient's breathing effort. A low value for the learning gains results in a controller that converges slowly and tries to compensate the average of a set of patient breaths. Therefore, a learning gain of 0.2 is implemented, and this value is a result of extensive testing.

D. Stability Analysis

In Section V-C, stability of the closed-loop system is guaranteed in case the system does not switch to the idle phase. However, in the PC-ACV use-case, the system switches to the idle phase after every ventilator-assisted breath. Therefore, the results in Section III-B are used to ensure stability of the switching system controlled by TRC. In other words, the three-step procedure in Section III-B is followed.

First, the switching system in (5) is obtained, with

1) G the mechanical ventilation system with hose compensation in (19) for $l = 1, 2, 3, 4$.
2) Learning filter L_1 as designed in Section V-C for $l = 1, 2, 3$, and L_2 during the idle phase, i.e., for $l = 4$, designed such that the internal states are held, as described in Section III-A.
3) Robustness filter Q_1 as designed in Section V-C for $l = 1, 2, 3$, and Q_2 during the idle phase, i.e., for $l = 4$, designed such that the internal states are held, as described in Section III-A.
4) $C_1 = C_2 = 0$ in both the phases, i.e., for $l = 1, 2, 3, 4$.
5) The main memory block B_Z $(z_i^{-(N-p_\ell-p_q)})$ is designed following (1) in the active phase for $l = 1, 2, 3$, and during the idle phase, i.e., for $l = 4$, designed such that the internal states are held, as described in Section III-A.
6) The preview buffer B_P $(z_i^{-p_\ell})$ is designed following (3) in the active phase for $l = 1, 2, 3$, and during the idle phase, i.e., for $l = 4$, designed such that the internal states are held, as described in Section III-A.

Second, using this representation the system is rewritten as the switching state-space model over multiple breaths in (7). This model describes the system and controller state at the start of breath $j + 1$, using the states at the start of breath j and the reference $\underline{r}(j) = [r(k_{s,j}) \ \cdots \ r(k_{s,j} + N + N_{\epsilon,j})]^T$, with $r(k) := p_{\text{target}}(k)$ for the ventilation use-case.

Third, for this switching system a CQLF should be computed that satisfies Proposition 1. Doing this for the considered use-cases, a challenge occurs. Namely, the P matrices are very large in dimension, up to 1042 × 1042 for the adult case. This is caused by the size of the system matrix $A(N, N_{\epsilon,j})$, which is large because of the states that are introduced by the memory loop in the triggered repetitive controller. Therefore, when attempting to solve the system of linear matrix inequalities in MATLAB to compute a suitable P matrix, our system, an Intel Core i7-7700HQ, 2.8-GHz processor with 16-GB RAM, runs out of memory and is unable to find a P matrix that satisfies the linear matrix inequalities and guarantees stability. Furthermore, it is observed that the computation time and storage cost grow rapidly with an increasing value of N. This is a known limitation of such methods to prove stability; see [41] and [42], and solving this issue is out-of-scope for this article.

For smaller values of N, we are able to find P matrices that guarantee stability of the system. Therefore, we followed two approaches to analyze the system's stability properties for the baby use-case, namely,

1) Solve the LMIs for smaller values of N, i.e., a CQLF is found for breath lengths that are shorter than the actual breath length.
2) Solve the LMIs for the same system and controller which is discretized at a lower sample frequency, more specifically 50 Hz instead of 500 Hz. The reduced sample frequency results in a significantly smaller buffer length N, reducing the computational burden.

Using these two approaches to analyze the system's stability properties and extensive testing in simulations and

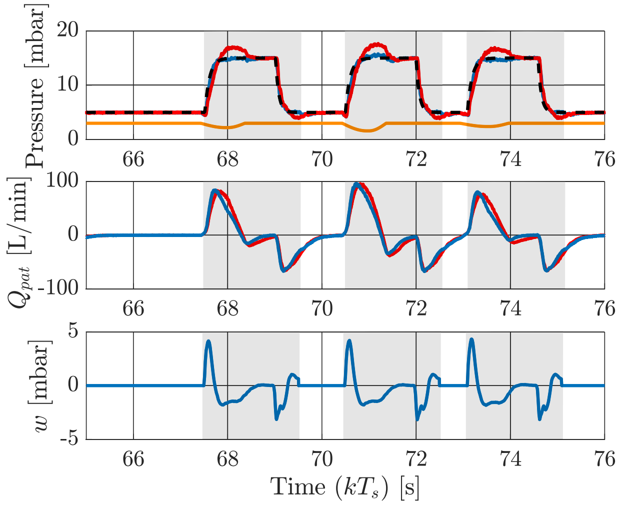

Fig. 12. Airway pressure, patient flow, and TRC output of the 19th–21th breath for the converged triggered repetivie controller (▬) and hose resistance compensation controller (▬) for the adult case, and the target pressure (- - -) and the scaled patient effort p_{mus} (▬). The triggered repetivie controller is in the active phase in the gray areas and in the idle phase in the white areas. The figure shows that the airway pressure tracking performance is improved.

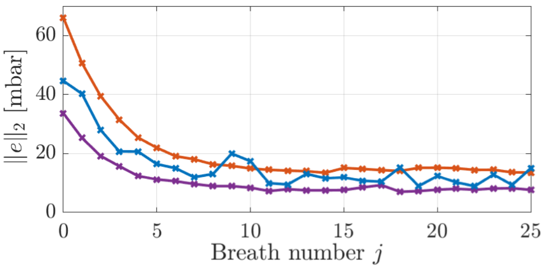

Fig. 13. Error two-norm per breath for every use-case; a single breath consists of the active phase and the subsequent idle phase. The first breath represents the system with hose-compensation control, since the RC output is zero. The figure shows the adult (▬), the pediatric (▬), and the baby (▬) use-cases.

experiments, we are confident that the system behaves safe and stable in practice.

These computational issues are a known issue for control strategies with large dimensions, e.g., model predictive control with large prediction horizons and solving the Ricatti equations for H_∞ control with large dimensional systems. Therefore, solving large-scale LMIs is considered out of the scope of this article. Also other approaches to reduce the buffer size could be considered in future work, e.g., using periodic basis functions to compute the RC signal.

E. Experimental Results

In this section, the results of the experiments are presented. First, the time-domain results of the adult use-case are shown and analyzed. Thereafter, all the use-cases are analyzed in terms of the pressure error two-norm on breath level.

The results of the 19th–21th breath of the adult use-case are shown in Fig. 12. The figure shows the airway pressure and patient flow for the system with TRC and for the system with just hose resistance compensation. It is clearly shown that the airway pressure tracking performance is improved significantly by TRC. Both the rise times and overshoot are significantly reduced. Some slight oscillations in the airway pressure with TRC are seen at the IPAP level, and these oscillations are caused by the varying patient efforts. Because these are varying over breaths, the triggered repetitive controller cannot compensate this perfectly. Furthermore, the actuation peak after the active phase, as observed in Section II-C, is prevented.

The error two-norm per breath for all the use-cases is shown in Fig. 13. The error two-norm per breath is defined as the two-norm of the tracking error, $e := p_{\text{target}} - \tilde{p}_{\text{aw}}$, during the active and subsequent idle phases. The first data point in Fig. 13 represents the error two-norm of the system with hose resistance compensation only, i.e., the output of the repetitive controller is zero. It is seen in the figure that for all the three use-cases the error converges in about ten breaths, and this is due to the particular choice of α. The figure shows that the tracking performance is improved with a factor 3.9–4.6. Upon convergence, some oscillations are seen, and these are caused by the varying lengths of the idle phase and the varying patient efforts. The adult use-case is especially sensitive to varying patient efforts because of the high compliance and low resistance.

Concluding, pressure tracking performance is improved significantly for a wide variety of spontaneously breathing patients by including TRC in the mechanical ventilation setup. Furthermore, the undesired actuation spike, as seen in Section II-C, is resolved.

VI. Conclusion and Future Work

The developed triggered repetitive control (RC) framework enables improved tracking performance improvement for systems with repeating tasks where the start of a task is triggered at varying intertask times, and this provides a breakthrough for applying repetitive control (RC) techniques to mechanical ventilation with possibly spontaneously breathing patients. The main challenge that is solved in this article is the fact that the period in between a task is not exactly the same for subsequent tasks, and it depends on the timing of an external disturbance, deteriorating the performance of traditional RC. This challenge is solved by activating and deactivating the RC when the repetitive task starts and ends, respectively. Furthermore, adjustments to the traditional RC filters are made to prevent undesired actuation peaks at the end of the repetitive task.

Furthermore, the TRC framework is theoretically supported by a stability analysis. Thereafter, a design procedure of TRC for mechanically ventilated patients is presented. Following this design procedure, TRC is implemented and tested in an experimental setup for ventilation. A performance analysis of these experiments shows a significant performance increase compared with other control strategies for ventilation. In the experimental case study, a reduction of the error two-norm up to a factor 4.6 has been achieved compared with a state-of-the-art control strategy.

Several recommendations are considered relevant for future extensions and improvements. First, a variable learning gain could be considered to decrease the effects of varying patient efforts on the tracking performance upon convergence while ensuring fast convergence. Second, an additional triggered

$$
\begin{aligned}
A_l &= \begin{bmatrix} a_{l,11} & a_{l,12} & a_{l,13} & a_{l,14} & a_{l,15} & 0 \\ 0 & A_{L,l} & B_{L,l}C_{Q,l} & 0 & B_{L,l}D_{Q,l}C_{Z,l} & 0 \\ 0 & 0 & A_{Q,l} & 0 & B_{Q,l}C_{Z,l} & 0 \\ -B_{C,l}C_G & 0 & 0 & A_{C,l} & 0 & 0 \\ a_{l,51} & a_{l,52} & a_{l,53} & a_{l,54} & a_{l,55} & B_{Z,l}C_{P,l} \\ 0 & 0 & B_{P,l}C_{Q,l} & 0 & B_{P,l}D_{Q,l}C_{Z,l} & A_{P,l} \end{bmatrix} \\
B_l &= \begin{bmatrix} B_G(1+D_{C,l}) & 0 & 0 & B_{C,l} & B_{Z,l} & 0 \end{bmatrix}^T \\
C_l &= \begin{bmatrix} C_G & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
D_l &= 0
\end{aligned} \tag{23}
$$

repetitive controller for the expiration should be added to enable triggered off-cycling of the mechanical ventilator. This allows ventilator strokes with varying lengths, improving synchronization between the patient and the ventilator even further. Third, a simple method to guarantee stability of the closed-loop system with TRC should be developed, to avoid the computational burden of solving the linear matrix inequalities. Using such stability guarantee, it might be possible to develop a design procedure that guarantees stability. Finally, input-to-state bounds can be used in future work to obtain a formal measure of the system's performance.

Appendix
Interconnected State-Space System

The full switching closed-loop system with TRC is described by the state-space system in (5). The corresponding state-space matrices are retrieved by working out the interconnections in Fig. 2 and considering the state vector in (6). This results in the following state-space system matrices: (23), as shown at the top of the page, where

$$
\begin{aligned}
a_{l,11} &= A_G - B_G D_{C,l} C_G, \quad a_{l,51} = -B_{Z,l} C_G \\
a_{l,12} &= B_G C_{L,l}, \quad a_{l,52} = 0 \\
a_{l,13} &= B_G D_{L,l} C_{Q,l}, \quad a_{l,53} = B_{Z,l} D_{P,l} C_{Q,l} \\
a_{l,14} &= B_G C_{C,l}, a_{l,54} = 0 \\
a_{l,15} &= B_G D_{L,l} D_{Q,l} C_{Z,l}, \quad a_{l,55} = A_{Z,l} + B_{Z,l} D_{P,l} D_{Q,l} C_{Z,l}.
\end{aligned} \tag{24}
$$

In these matrices, the first subscript defines to which subsystem in Fig. 2 the corresponding matrix belongs, and the second subscript, i.e., $l \in 1, 2, 3, 4$, refers to the phase it belongs to.

References

[1] M. A. Warner and B. Patel, "Mechanical ventilation," in *Benumof and Hagberg's Airway Management*. Amsterdam, The Netherlands: Elsevier, 2013, pp. 981–997.

[2] C. R. Wells et al., "Projecting the demand for ventilators at the peak of the COVID-19 outbreak in the USA," *Lancet Infectious Diseases*, vol. 20, no. 10, pp. 1123–1125, Oct. 2020.

[3] B. Hunnekens, S. Kamps, and N. Van De Wouw, "Variable-gain control for respiratory systems," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 1, pp. 163–171, Jan. 2020.

[4] L. Blanch et al., "Asynchronies during mechanical ventilation are associated with mortality," *Intensive Care Med.*, vol. 41, no. 4, pp. 633–641, 2015.

[5] M. Borrello, "Modeling and control of systems for critical care ventilation," in *Proc. Amer. Control Conf.*, Jun. 2005, pp. 2166–2180.

[6] M. A. Borrello, "Adaptive inverse model control of pressure based ventilation," in *Proc. Amer. Control Conf.*, Jun. 2001, pp. 1286–1291.

[7] A. Pomprapa, S. Weyer, S. Leonhardt, M. Walter, and B. Misgeld, "Periodic funnel-based control for peak inspiratory pressure," in *Proc. 54th IEEE Conf. Decis. Control (CDC)*, Dec. 2015, pp. 5617–5622.

[8] M. Scheel, T. Schauer, A. Berndt, and O. Simanski, "Model-based control approach for a CPAP-device considering patient's breathing effort," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9948–9953, Jul. 2017.

[9] H. Li and W. M. Haddad, "Model predictive control for a multicompartment respiratory system," in *Proc. Amer. Control Conf. (ACC)*, Jun. 2012, pp. 5574–5579.

[10] J. Reinders, B. Hunnekens, F. Heck, T. Oomen, and N. van de Wouw, "Adaptive control for mechanical ventilation for improved pressure support," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 1, pp. 180–193, Jan. 2021.

[11] J. Reinders, B. Hunnekens, F. Heck, T. Oomen, and N. van de Wouw, "Accurate pressure tracking to support mechanically ventilated patients using an estimated nonlinear hose model and delay compensation," *Control Eng. Pract.*, vol. 106, Jan. 2021, Art. no. 104660.

[12] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. Robot. Syst.*, vol. 1, no. 2, pp. 123–140, 1984.

[13] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, Jun. 2006.

[14] K. L. Moore, *Iterative Learning Control for Deterministic Systems*. London, U.K.: Springer-Verlag, 1993.

[15] C. T. Freeman, E. Rogers, A.-M. Hughes, J. H. Burridge, and K. L. Meadmore, "Iterative learning control in health care: Electrical stimulation and robotic-assisted upper-limb stroke rehabilitation," *IEEE Control Syst.*, vol. 32, no. 1, pp. 18–43, Feb. 2012.

[16] Y. Chen, K. L. Moore, J. Yu, and T. Zhang, "Iterative learning control and repetitive control in hard disk drive industry—A tutorial," *Int. J. Adapt. Control Signal Process.*, vol. 22, no. 4, pp. 325–343, 2008.

[17] 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.

[18] T. Inoue, M. Nakano, T. Kubo, S. Matsumoto, and H. Baba, "High accuracy control of a proton synchrotron magnet power supply," *IFAC Proc. Volumes*, vol. 14, no. 2, pp. 3137–3142, Aug. 1981.

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

[20] G. Pipeleers, B. Demeulenaere, and J. Swevers, *Optimal Linear Controller Design for Periodic Inputs*, vol. 394. London, U.K.: Springer, 2009.

[21] M. Scheel, A. Berndt, and O. Simanski, "Iterative learning control: An example for mechanical ventilated patients," *IFAC-PapersOnLine*, vol. 48, no. 20, pp. 523–527, 2015.

[22] A. F. de Castro and L. A. B. Tôrres, "Iterative learning control applied to a recently proposed mechanical ventilator topology," *IFAC-PapersOnLine*, vol. 52, no. 1, pp. 154–159, 2019.

[23] H. Hazarika and A. Swarup, "Improved performance of flow rate tracking in a ventilator using iterative learning control," in *Proc. Int. Conf. Electr. Electron. Eng. (ICE3)*, Feb. 2020, pp. 446–451.

[24] J. Reinders, R. Verkade, B. Hunnekens, N. van de Wouw, and T. Oomen, "Improving mechanical ventilation for patient care through repetitive control," in *Proc. IFAC 21st Triennial World Congr.* Berlin, Germany, Jul. 2020, pp. 1441–1446.

[25] F. Boeren, A. Bareja, T. Kok, and T. Oomen, "Frequency-domain ILC approach for repeating and varying tasks: With application to semiconductor bonding equipment," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 6, pp. 2716–2727, Dec. 2016.

[26] L. Blanken, G. Isil, S. Koekebakker, and T. Oomen, "Flexible ILC: Towards a convex approach for non-causal rational basis functions," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12107–12112, Jul. 2017.

[27] D. J. Hoelzle, A. G. Alleyne, and A. J. W. Johnson, "Basis task approach to iterative learning control with applications to micro-robotic deposition," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 5, pp. 1138–1148, Sep. 2011.

[28] J. Shi, J. Xu, J. Sun, and Y. Yang, "Iterative learning control for time-varying systems subject to variable pass lengths: Application to robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 67, no. 10, pp. 8629–8637, Oct. 2020.

[29] T. Seel, T. Schauer, and J. Raisch, "Monotonic convergence of iterative learning control systems with variable pass length," *Int. J. Control*, vol. 90, no. 3, pp. 393–406, Jun. 2016.

[30] M. Steinbuch, "Repetitive control for systems with uncertain period-time," *Automatica*, vol. 38, no. 12, pp. 2103–2109, 2002.

[31] R. J. E. Merry, D. J. Kessels, W. P. M. H. Heemels, M. J. G. van de Molengraft, and M. Steinbuch, "Delay-varying repetitive control with application to a walking piezo actuator," *Automatica*, vol. 47, no. 8, pp. 1737–1743, 2011.

[32] 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.

[33] L. Blanken, S. Koekebakker, and T. Oomen, "Multivariable repetitive control: Decentralized designs with application to continuous media flow printing," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 1, pp. 294–304, Feb. 2020.

[34] J. Hespanha, *Linear Systems Theory*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[35] D. Liberzon, *Switching in Systems and Control* (Systems & Control: Foundations & Applications). Boston, MA, USA: Birkhäuser, 2003.

[36] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, no. 6, pp. 857–869, Jun. 2001.

[37] J. H. T. Bates, *Lung Mechanics*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[38] E. Fresnel, J.-F. Muir, and C. Letellier, "Realistic human muscle pressure for driving a mechanical lung," *EPJ Nonlinear Biomed. Phys.*, vol. 2, no. 1, pp. 1–18, Aug. 2014.

[39] M. Tomizuka, T.-C. Tsao, and K.-K. Chew, "Analysis and synthesis of discrete-time repetitive controllers," *ASME J. Dyn. Syst., Meas., Control*, vol. 111, no. 3, pp. 353–358, Sep. 1989.

[40] 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.

[41] A. Yurtsever, J. A. Tropp, O. Fercoq, M. Udell, and V. Cevher, "Scalable semidefinite programming," *SIAM J. Math. Data Sci.*, vol. 3, no. 1, pp. 171–200, 2021.

[42] A. Majumdar, G. Hall, and A. A. Ahmadi, "Recent scalability improvements for semidefinite programming with applications in machine learning, control, and robotics," *Annu. Rev. Control, Robot., Auto. Syst.*, vol. 3, no. 1, pp. 331–360, May 2020.