TU Delft

Delft University of Technology

Automatic patient-ventilator asynchrony detection framework using objective asynchrony definitions

van de Kamp, Lars; Reinders, Joey; Hunnekens, Bram; Oomen, Tom; van de Wouw, Nathan

DOI
10.1016/j.ifacsc.2023.100236

Publication date
2024

Document Version
Final published version

Published in
IFAC Journal of Systems and Control

Citation (APA)
van de Kamp, L., Reinders, J., Hunnekens, B., Oomen, T., & van de Wouw, N. (2024). Automatic patient-ventilator asynchrony detection framework using objective asynchrony definitions. *IFAC Journal of Systems and Control*, *27*, Article 100236. https://doi.org/10.1016/j.ifacsc.2023.100236

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

IFAC Journal of Systems and Control 27 (2024) 100236

Contents lists available at ScienceDirect

IFAC Journal of Systems and Control

journal homepage: www.elsevier.com/locate/ifacsc

Automatic patient-ventilator asynchrony detection framework using objective asynchrony definitions

Lars van de Kamp [a,b,*], Joey Reinders [a], Bram Hunnekens [a], Tom Oomen [b,c], Nathan van de Wouw [b]

[a] Demcon Life Sciences and Health, Kanaaldijk 29, Best, 5683 CR, Noord-Brabant, The Netherlands
[b] Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, 5612 AZ, Noord-Brabant, The Netherlands
[c] Delft Center for Systems and Control, Delft university of Technology, Mekelweg 5, Delft, 2628 CD, Zuid-Holland, The Netherlands

ARTICLE INFO

Article history:
Received 14 October 2023
Received in revised form 30 November 2023
Accepted 27 December 2023
Available online 7 January 2024

Keywords:
Mechanical ventilation
Patient-ventilator asynchrony
Detection
Classification
Supervised learning
Recurrent neural networks

ABSTRACT

Patient-ventilator asynchrony is one of the largest challenges in mechanical ventilation and is associated with prolonged ICU stay and increased mortality. The aim of this paper is to automatically detect and classify the different types of patient-ventilator asynchronies during a patient's breath using the typically available data on commercially available ventilators. This is achieved by a detection and classification framework using an objective definition of asynchrony and a supervised learning approach. The achieved detection performance of the near-real time framework on a clinical dataset is a significant improvement over current clinical practice, therewith and, this framework has the potential to significantly improve the patient comfort and treatment outcomes.

© 2024 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

Mechanical ventilation is a life-saving therapy used 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 (Warner & Patel, 2013). Especially during the flu season or a world-wide pandemic such as the COVID-19 pandemic Wells et al. (2020) and Borrello (2021), mechanical ventilation is a life saver for many patients around the world.

A common mode of ventilation is Pressure Support Ventilation (PSV). In PSV, the ventilator supports a spontaneously breathing patient who is unable to breathe sufficiently by itself. When the ventilator detects an inspiration by the patient, the ventilator increases the pressure levels to increase the air flow and assist the patient's breath. Then, when the patient starts its expiration, the ventilator should lower the pressure to allow the patient's expiration. To maximize the patient's comfort, recovery, and safety, it is important that the ventilator support is synchronized with the patient's breathing. In other words, the ventilator's inspiration and expiration start times should be synchronized with the patient's inspiration and expiration start times. Synchronization of the ventilator's and patient's timing is one of largest challenges in PSV. A mismatch between these timings is called Patient-Ventilator Asynchrony (PVA). Severe levels of PVA are observed in many ventilated patients, ranging from 24% in Thille, Rodriguez, Cabello, Lellouche, and Brochard (2006) to 43% in Vignaux et al. (2009). According to Blanch et al. (2015), Epstein (2011), Pham, Telias, Piraino, Yoshida, and Brochard (2018), Thille et al. (2006), PVA is associated with prolonged ICU stay and even increased mortality.

Further improvement of ventilation outcomes therefore hinges on preventing these asynchronies. A first step towards preventing asynchronies is detecting them. However, detecting asynchronies, using the available flow and pressure waveforms, is highly time-consuming and challenging for clinicians (Colombo et al., 2011). Additionally, the workload for medical staff is already very high and expected to further increase in the future (Angus, Kelley, Schmitz, White, Popovich, 2000). Therefore, it is necessary to detect and classify PVA automatically and reliably.

For this research, a set of requirements for the automatic detection approach is defined, with the goal to develop an approach that can be implemented on ventilators. These requirements include the following:

1. The detection approach must be able to detect asynchronies near real-time, i.e., during a breath, to be able to visualize and present it in a timely fashion to the clinician.
2. An objective asynchrony definition for all asynchronies within pressure support ventilation must be found.

* Corresponding author at: Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, 5612 AZ, Noord-Brabant, The Netherlands.
E-mail address: l.g.j.v.d.kamp@tue.nl (L. van de Kamp).

https://doi.org/10.1016/j.ifacsc.2023.100236
2468-6018/© 2024 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

3. The detection approach must be able to detect all different asynchrony types in PSV.
4. The detection approach should only use currently available non-invasive measured ventilator signals.

In recent years, substantial research has been conducted to develop algorithms that can detect and classify different types of asynchronies. In Adams et al. (2017) and Blanch et al. (2012), an algorithm based on bedside clinical rules is proposed that is able to detect only one asynchrony type, and hence conflicts with requirement 3. In Mulqueeny et al. (2009), a naive Bayes algorithm based on 21 features from the pressure and flow waveforms is proposed. This algorithm is able to distinguish ineffective efforts from normal breaths. In Gholami et al. (2018), a random forest network is used that is able to automatically detect premature cycling and delayed cycling asynchronies from extracted features of the pressure and flow curves. The above two methods conflict with requirements 1 and 3. In Ng et al. (2021), Rehm et al. (2020), advances are made towards real-time monitoring of PVA in practical applications; however, not all different asynchrony types are detected. Additional advances are necessary to fulfill requirement 3.

In Zhang et al. (2020), a long-short term memory network is used that is able to classify double trigger and ineffective efforts also based on the pressure and flow curves. Although near real-time detection is possible, not all asynchrony types can be detected, which conflicts with requirement 3. In Bakkes, Montree, Mischi, Mojoli, and Turco (2020), a convolutional neural network is used that is able to identify the inspiration and expiration start times of the patient and the ventilator. Subsequently, these timings are translated to a specific asynchrony type. In van Diepen et al. (2021), the performance of the algorithm of Bakkes et al. (2020) is evaluated on simulated PVA data. In Pan et al. (2021), a convolutional neural network is proposed that classifies a breath based on the corresponding PVA type and it gives an indication which part of the breath is important for the classification process. The last two methods opt for a convolutional network where an entire breath is needed as an input. This makes near real-time detection during a breath impossible. As a result, requirement 1 is not satisfied.

Although existing literature shows that rule-based algorithms and machine learning algorithms are promising solutions to detect and classify PVA, the presented approaches do not satisfy the requirements as specified above. Additionally, the labeling process using expert knowledge is challenging, which results in poor data quality (Colombo et al., 2011). The aim of this paper is to develop a detection and classification framework that can detect all types of asynchrony in PSV using the real-time measured data that is typically available in commercial ventilators. This is achieved by first presenting an objective characterization of PVA. Subsequently, this definition is used to generate labeled ventilation data based on patient and ventilator timings. Thereafter, a recurrent neural network is designed and trained enabling us to recognize these asynchronies in real-time. Finally, performance of this algorithm is analyzed using clinical data.

The main contribution of this paper is formulated as follows:

- the design of a framework to automatically detect and classify different types of patient-ventilator asynchrony for real-time bedside monitoring.

Besides the main contribution, this paper contains several sub-contributions:

- a new objective characterization of many relevant patient-ventilator asynchrony types;
- a performance analysis of the developed algorithm, using clinical data;
- an analysis of how much data is required to train a highly performing detection network.

The outline of this paper is as follows. In Section 2, the main components of the PVA detection and classification framework are described. Thereafter, in Section 3, the relevant types of PVA are characterized. In Section 4, the clinical data is introduced and the overall recurrent neural network structure is presented. Subsequently, in Section 5, the performance of the network is evaluated using a clinical dataset and the generalization capability is analyzed. Finally, in Section 6, the main conclusions and recommendations for future work are formulated.

2. PVA detection and classification framework

To develop the Patient-Ventilator Asynchrony (PVA) detection and classification algorithm, a framework is designed that can utilize different supervised learning algorithms. In supervised learning, a mapping from the available input (measured data) to the output (PVA type) is learned based on a known set of input–output pairs (Murphy, 2012). To achieve this, a dataset, a model structure, and a fit criterion are required. The dataset is used to train and eventually validate the algorithm. The model structure is a function that can describe the relationship between the input and the output data. Finally, the fit criterion is used to obtain the optimal parameters of the model structure.

A supervised learning approach is used, because the available clinical dataset offers annotated pressure, flow, and volume curves making it possible to obtain a highly accurate labeled dataset. Furthermore, supervised learning approaches have been successfully used in literature to classify PVA, as shown in the introduction. Next, the three different components of the framework, i.e., the *data*, the *model structure*, and the *fit criterion*, are described.

The *data* used in this PVA detection and classification problem is labeled ventilation data. The inputs of the detection and classification model are the measured airway pressure p_{aw}, the patient flow Q_{pat}, and the patient's lung volume V_{pat}. These signals are available in commercial ventilation systems, see Fig. 1 . The desired output of the algorithm is the type of PVA that is occurring in every breath. The data used to construct the model could be either measured data that is labeled by an expert or synthetic data from a simulation environment. In this paper, we opt to use clinical patient data. Labeling clinical data based on already available ventilator data is a labor intensive and error-prone task. Therefore, the data is annotated using an additional invasive measurement, the oesophagus pressure, resulting in more accurate labels.

The *model structure* is used to obtain a function, or mapping, from the input data (measured data) to the output data (PVA type). Different model structures can be used in the proposed framework for PVA detection and classification. In this paper, a Recurrent Neural Network (RNN) structure (Goodfellow, Bengio, & Courville, 2016) is considered, because RNNs can use their internal state to process input sequences of variable length. This makes them particularly suitable for time-series classification problems. Furthermore, the RNN model structure has proven to successfully classify PVA, see Zhang et al. (2020), and has proven to be useful in many other medical time-series classification problems, such as Drumond, Marques, Vasconcelos, and Clua (2018) and Andersen, Peimankar, and Puthusserypady (2019).

The *fit criterion* is used to obtain the optimal model structure and parameters, i.e., weights and biases of the RNN, such that the model optimally describes the relation between the input and output data. A suitable choice for the fit criterion is the cross-entropy loss function, as defined in Bishop (1995). According to Simard, Steinkraus, and Platt (2003), this loss function results

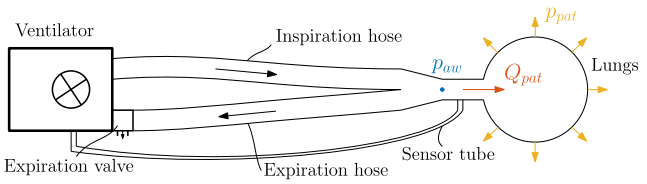


Fig. 1. A schematic overview of a ventilator connected to the patient with a dual-hose system. The airway pressure, p_{aw}, Q_{pat}, and the patient effort p_{pat} are important signals for detection of patient-ventilator asynchrony.

in faster training and improved generalization compared to the other fit criteria in classification problems. Of course, alternative criteria directly fit in the proposed framework, which can easily be adapted to this end. Minimizing the loss function of choice with respect to the model structure parameters is referred to as the training process.

3. Characterizing patient-ventilator asynchrony

In this section, a new objective characterization of many relevant patient-ventilator asynchrony types is defined, which is the first sub-contribution of this paper. First, in Section 3.1, Pressure Controlled Ventilation (PSV) is explained in detail. Thereafter, in Section 3.2, patient-ventilator asynchrony is explained and defined. Note that the proposed PVA definitions are not exclusive to PSV, they can be applied to all modes of mechanical ventilation with minor adjustments.

3.1. Pressure support ventilation

A mechanical ventilation system as in Fig. 1 contain a hose set-up that connects the patient to the ventilator. Signals that are important during ventilation are the airway pressure, p_{aw}, the flow into the patient Q_{pat}, and the patient effort p_{pat}. A ventilation mode typically used when patients are spontaneously breathing ($p_{pat}(t) \neq 0$) is pressure support ventilation. A schematic example of PSV pressure and flow curves is depicted in Fig. 2. The figure shows that the patient starts an inspiration at the black asterisk. During its inspiration, the patient is generating a negative pressure in its lungs, resulting in a small positive patient flow. When this patient flow exceeds a predefined trigger level indicated by the black cross, the ventilator increases the airway pressure, i.e., the pressure near the patient's airway, to the Inspiratory Positive Airway Pressure (IPAP) level. Then, after some time the patient starts its expiration, indicated by the black plus sign. During its expiration, the patient is increasing the lung pressure again. Then, when the patient flow is under a certain preset percentage of its peak flow, indicated by the black dot, the ventilator expiration starts. The airway pressure is lowered to the Positive End-Expiratory Pressure (PEEP) level to allow the expiration. After such breathing sequence is completed the ventilator waits until it detects the next patient inspiration. A mismatch in these timings results in PVA which is explained and defined in the next section.

3.2. Patient-ventilator asynchrony

PVA can be divided into two main categories, namely *timing asynchronies* and *severe asynchronies*. The timing asynchronies are related to a pair of a patient and ventilator breath that have a mismatch in timing. The severe asynchronies are related to patient breaths that are not clearly linked to a single ventilator stroke or vice versa. In the remainder of this section, the different

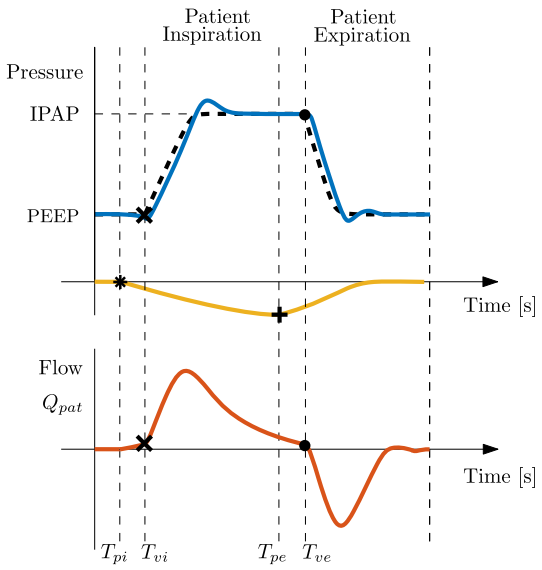


Fig. 2. A schematic PSV breathing cycle with a spontaneously breathing patient, showing the patient's spontaneous breathing effort p_{pat} (—), the target pressure p_{target} (- - -), the patient flow Q_{pat} (—), and airway pressure p_{aw} (—). Furthermore, the ventilator inspiration trigger (×), the ventilator expiration trigger (●), the patient's inspiration start (✱), and the patient's expiration start (✚), and their respective timings (T_{vi}, T_{ve}, T_{pi}, and T_{pe}) are depicted.

asynchrony types are described physically and defined mathematically. For the mathematical PVA definitions, the patient's inspiration $T_{pi}(j)$ and expiration $T_{pe}(j)$ timing and the ventilators inspiration $T_{vi}(k)$ and expiration timing $T_{ve}(k)$ are used, which are visualized in Fig. 2. The start times of the patient's inspiration and expiration can be derived from the oesophageal pressure or EAdi monitoring. The variable for the patient breath counter is denoted by $j \in \{1, 2, \ldots, n\}$, with n the number of patient breaths, and the ventilator stroke counter is denoted by $k \in \{1, 2, \ldots, m\}$, with m the number of ventilator strokes.

It should be noted that these timings cannot be defined randomly; they obey two important physical assumptions. Firstly, the patient and the ventilator are always switching between an inspiration and expiration. For example, it is not allowed to have two patient inspirations without a patient expiration in between. Secondly, the timings are ordered in chronological order. This means that the patient and ventilator timings obey the following inequalities:

$$T_{pi}(j) < T_{pe}(j) < T_{pi}(j+1), \ \forall j \in \{1, 2, \ldots, n\} \quad (1)$$

and

$$T_{vi}(k) < T_{ve}(k) < T_{vi}(k+1), \ \forall k \in \{1, 2, \ldots, m\}. \quad (2)$$

In the remainder of this section, the timing asynchronies and the severe asynchronies are described separately.

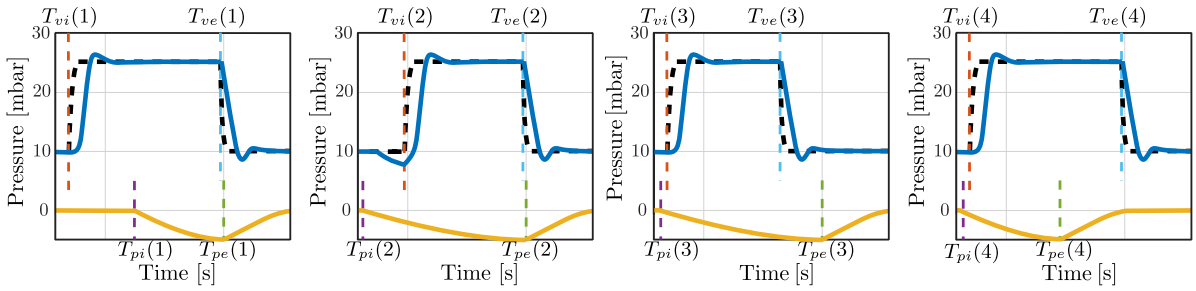
3.2.1. Timing asynchronies

Timing asynchronies are defined for breaths consisting of a single spontaneous breath by the patient and a related single ventilator stroke, as depicted in Fig. 2. To validate that a patient breath j and a ventilator stroke k are related and should be analyzed for a timing asynchrony, the following inequalities should be satisfied for a pair (j, k):

$$T_{pe}(j-1) < T_{vi}(k) < T_{pe}(j)$$

and

$$T_{ve}(k-1) < T_{pi}(j) < T_{ve}(k). \quad (3)$$



(a) Premature trigger. (b) Delayed trigger. (c) Premature cycling. (d) Delayed cycling.

Fig. 3. Visualizations of the different timing asynchronies during PSV. The figure shows the spontaneous breathing effort p_{pat} (——), the target pressure p_{target} (- - -), and the airway pressure p_{aw} (——).

If these inequalities hold for a pair (j, k), then the pair (j, k) is associated to either a synchronized breath or a timing asynchrony. The first inequality in (3) ensures that the start of a ventilator stroke k happens between the start of the patient's expiration $j-1$ and j. The second inequality in (3) ensures that the patient inspiration j has to start after ventilator expiration $k-1$ and before ventilator expiration k.

A pair of (j, k) that satisfies (3) can either be a synchronized breath or it can contain a so-called inspiration asynchrony, a cycling asynchrony, or both asynchronies. The different inspiration and cycling asynchronies are visualized in Fig. 3. The inspiration and cycling asynchronies are defined using the inspiration delay $\Delta t_{insp} := T_{vi}(k) - T_{pi}(j)$ and expiration delay $\Delta t_{exp} := T_{ve}(k) - T_{pe}(j)$, respectively. Using the inspiration and expiration delay, the following (a)synchronies are defined for a pair (j, k) that satisfy (3) (the timing asynchrony definitions, including the specific values, are based on Bakkes et al. (2020)):

- **Synchronized Inspiration** (SI): $0 \leq \Delta t_{insp} \leq 0.3$ s, the ventilator's inspiration (×) does not start before or 0.3 s after the patient's inspiration (∗), i.e., the ventilator's inspiration is synchronized with the patient's inspiration;
- **Premature Trigger** (PT): $\Delta t_{insp} < 0.0$ s, the ventilator's inspiration (×) starts before the patient's inspiration (∗), i.e., the ventilator is triggered before the start of the patient's inspiration;
- **Delayed Trigger** (DT): $\Delta t_{insp} > 0.3$ s, the ventilator's inspiration (×) starts more than 0.3 s after the patient's inspiration (∗) start, i.e., the ventilator is triggered too late;
- **Synchronized Cycling** (SC): $-0.2 \leq \Delta t_{exp} \leq 0.2$ s, the ventilator's expiration (•) starts within 0.2 s before or after the patient's expiration (×), i.e., the ventilator's expiration is synchronized with the patient's expiration;
- **Premature Cycling** (PC): $\Delta t_{exp} < -0.2$ s, the ventilator's expiration (•) starts more than 0.2 s before the patient's expiration (×), i.e., the ventilator cycles off prematurely;
- **Delayed Cycling** (DC): $\Delta t_{exp} > 0.2$ s, the ventilator's expiration (•) starts more than 0.2 s after the patient's expiration (×), i.e., the ventilator cycles off too late.

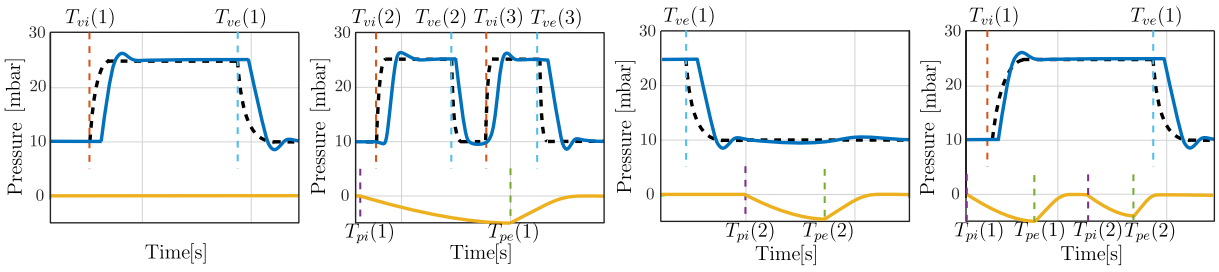
If a pair (j, k) satisfies the inequalities in (3), its PVA type is defined by a single Δt_{insp} and Δt_{exp}. Hence, every pair (j, k) that satisfies (3) can be classified as one of these timing (a)synchronies.

3.2.2. Severe asynchronies

In case a single patient breath is not clearly related to a single ventilator stroke or vice versa, a severe asynchrony is occurring. Three different types of severe asynchrony are considered: auto triggers, double triggers, and ineffective efforts. These asynchronies are visualized in Fig. 4. The auto trigger shows a ventilator stroke in absence of a patient breath. The double trigger consists of two ventilator strokes in the presence of only one patient breath. An ineffective effort is defined as a patient breath without a ventilator stroke, which can be subdivided into an inspiratory and an expiratory ineffective effort.

Based on inspiration and expiration start times of the patient and ventilator, these severe asynchronies can be defined mathematically as well:

- **Auto Trigger** (AT): If there exists a combination of k and j such that
$$T_{pe}(j-1) < T_{vi}(k) < T_{pe}(j) \quad \text{and} \quad T_{ve}(k) \leq T_{pi}(j) \tag{4}$$
hold, then the ventilator stroke k is an auto trigger. The combination of these conditions ensures that there are no patient breaths during ventilator stroke k. In Fig. 4(a) with $j = 1$ and $k = 1$ the inequalities are satisfied and $k = 1$ is identified as an auto trigger.
- **Double Trigger** (DbT): If there exists a combination of k and j such that
$$T_{pe}(j-1) < T_{vi}(k) < T_{pe}(j) \quad \text{and} \quad T_{ve}(k-1) \geq T_{pi}(j) \tag{5}$$
hold, then the ventilator stroke k is a double trigger during the jth patient breath. The combination of these conditions ensures that the start of ventilator stroke k is the second ventilator stroke during patient breath j. In Fig. 4(b) with $j = 1$ and $k = 3$ the inequalities are satisfied and $k = 3$ is identified as a double trigger during patient breath $j = 1$.
- **Expiratory Ineffective Effort** (IEe): If there exists a combination of k and j such that
$$T_{vi}(k) \geq T_{pe}(j) \quad \text{and} \quad T_{ve}(k-1) < T_{pi}(j) < T_{vi}(k) \tag{6}$$
hold, then the patient breath j is an expiratory ineffective effort. The combination of these conditions ensures that no ventilator stroke occurs during patient breath j. In Fig. 4(c), the inequalities are satisfied with $j = 2$ and $k = 2$. The patient breath $j = 2$ is identified as an expiratory ineffective effort.
- **Inspiratory Ineffective Effort** (IEi): If there exists a combination of k and j such that
$$T_{pe}(j-1) \geq T_{vi}(k) < T_{pe}(j) \quad \text{and} \quad T_{ve}(k-1) < T_{pi}(j) < T_{ve}(k) \tag{7}$$



(a) Auto trigger. (b) Double trigger. (c) Expiratory Ineffective effort. (d) Inspiratory Ineffective effort.

Fig. 4. Visualizations of the different types of severe asynchronies during PSV. The figure shows the spontaneous breathing effort p_{pat} (—), the target pressure p_{target} (- - -), and the airway pressure p_{aw} (—).

hold, then the patient breath j is an inspiratory ineffective effort. The combination of these conditions ensures that the second patient breath j occurs during an inspiration cycle of the ventilator. In Fig. 4(d), the inequalities are satisfied with $j = 2$ and $k = 1$. The patient breath $j = 2$ is identified as an inspiratory ineffective effort.

The defined combinations of the timing asynchronies and the severe asynchronies, describe the clinically relevant asynchrony types mathematically.

4. Detection and classification model

In this section, the presented framework is used to develop an algorithm to detect and classify PVA from ventilation data. First, the clinical data set with its statistics is introduced in Section 4.1. Subsequently, the overall model structure is defined in Section 4.2. Then, the loss function is explained in Section 4.3. Thereafter, the training of the network is explained in Section 4.4. Finally, in Section 4.5, the performance evaluation criteria are discussed.

4.1. Introduction to the clinical data

The clinical data consists of 15 patients from the Fondazione I.R.C.C.S. Policlinco San Matteo (Pavia, Italy) (Mojoli et al., 2022). The patients are subjected to different levels of pressure support ventilation and were connected to either a GE healthcare Engstrom (Madison(WI), USA) or a Hamilton G5 ventilator (Bonaduz, Switzerland). The airway pressure, patient flow, and volume are measured over time, together with the esophageal pressure to obtain a more accurate measurement of the patient respiratory effort. Experts in the field labeled the data by indicating the inspiration and expiration start times of the patient according to the method explained in Mojoli et al. (2022). Leveraging the objective asynchrony definitions introduced in Section 3, we construct a multi-variate time-series that contain the asynchrony labels. This labeling approach overcomes the drawbacks of original human labeling since more information is available and exact rules are leveraged. In Fig. 5, a preview of the clinical data with the labels in the form of a multi-variate time-series is shown.

In Table 1, the incidence of each asynchrony type is shown. The majority of the labels are synchronous, namely, 51.8%. The most prevalent asynchrony types in this data set are delayed cycling, expiratory ineffective efforts, and delayed triggers, respectively. The timing (a)synchronies are defined based on inspiratory and expiratory timings with the objective rules defined in Section 3.2. The distribution of the timing asynchronies in the space of the inspiration and expiration delays is shown in Fig. 6. Here, it can be seen that a lot of asynchronies are located at the boundaries, which might make it challenging to distinguish them during detection.

Table 1
Incidence of the different asynchrony types in the clinical dataset.

(A)synchrony type	Amount
Premature triggering (PT)	1
Synchronized inspiration (SI)	2623
Delayed triggering (DT)	682
Premature cycling (PC)	492
Synchronized expiration (SE)	1306
Delayed cycling (DC)	1495
Auto triggering (AT)	5
Double triggering (DbT)	0
Expiratory ineffective effort (IEe)	965
Inspiratory ineffective effort (IEi)	13

4.2. Overall model structure

The overall model structure is defined by the inputs, outputs, and the model structure itself, which is schematically visualized in Fig. 7.

4.2.1. Inputs

The input $x(t) = [p_{aw}(t), Q_{pat}(t), V_{pat}(t)]^\top$ of the classification network is a multivariate time-series sampled at 50 Hz, which includes the airway pressure, patient flow, and the volume in the patient's lungs. The choice for using these signals is based on practical implications, as they are readily available on almost all mechanical ventilators.

4.2.2. Outputs

The output of the network $\hat{y}(t)$ should predict the ground-truth labels $y(t)$, as closely as possible. The output is also a multivariate time-series that contains eleven separate time-series, one for each PVA type and one for the zero-label, i.e., $\hat{y}(t) = [\hat{y}^1(t), \ldots, \hat{y}^{n_c}(t)]^\top$ with n_c the number of classes in the vector. If a particular PVA type occurs, the value of that class in the ground-truth time-series is one, otherwise the value of the time-series equals zero, as shown in Fig. 5.

4.2.3. Model structure

To obtain a mapping from the inputs to the desired outputs, a Recurrent Neural Network (RNN) is considered. In this paper an RNN with Long Short-Term Memory (LSTM) cells is used. Note that other network structures can be used as well in the proposed framework. However, RNN structures with LSTM cells are effective sequence models used in practical applications (Goodfellow et al., 2016). This model structure allows a mapping that can handle varying sequence lengths and updates every sample. Therefore, asynchronies can be detected and classified in near real-time by feeding new data to the network as it is measured. Standard RNNs have problems with exploding/vanishing gradients when classifying long time-series (Pascanu, Mikolov,

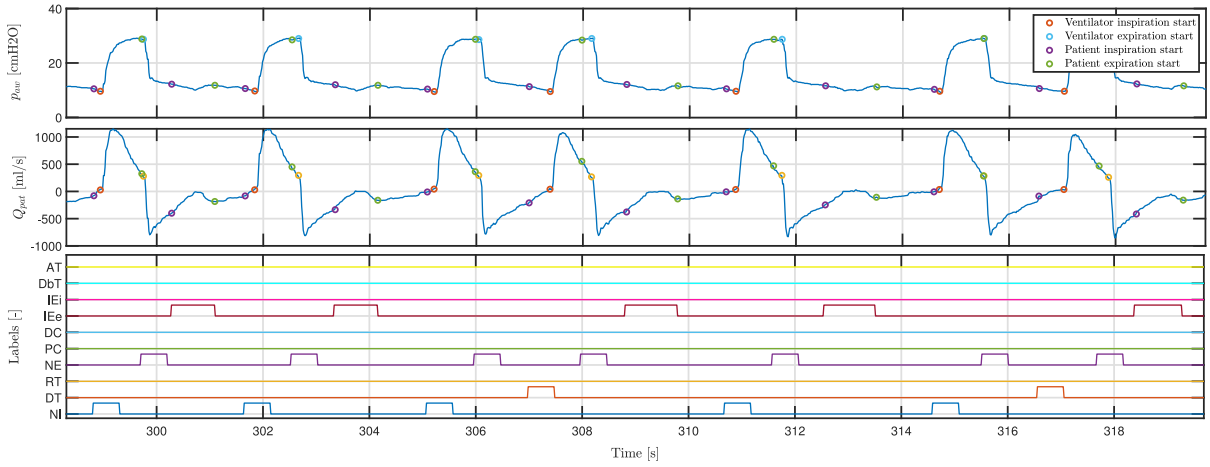


Fig. 5. A subset of the clinical dataset, where the airway pressure, patient flow, the patient and ventilator timings, and the time-series asynchrony labels are displayed.

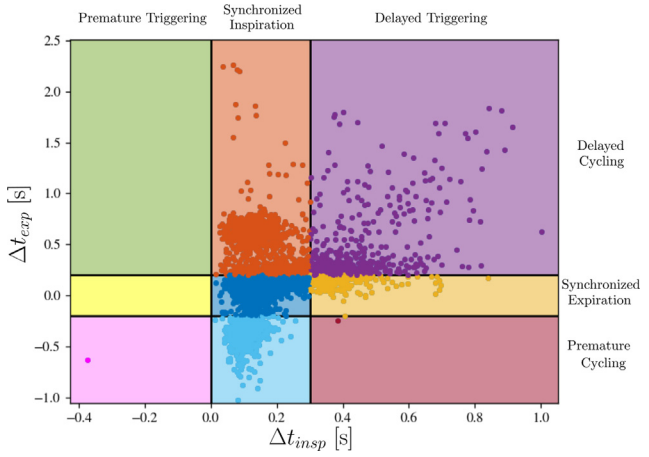


Fig. 6. Distribution of the timing asynchronies based on their inspiration and expiration delays. Each colored area represents a different combination of inspiration and expiration (a)synchrony.

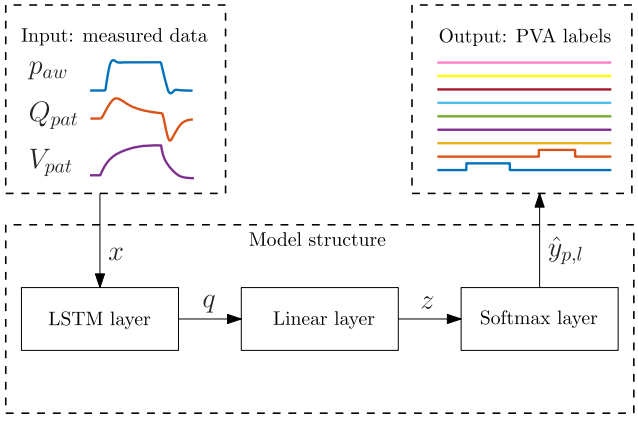


Fig. 7. Definition of the model structure that is used for patient-ventilator asynchrony detection. Showing the inputs, the overall model structure, and the desired output. The LSTM layer contains 256 cells.

& Bengio, 2013). The problem of exploding/vanishing gradients is tackled by using LSTM cells in the RNN. These cells contain multiplicative gates that are capable of extracting and storing information over longer periods of time (Graves, Liwicki, Fernández, Bertolami, Bunke, Schmidhuber, 2009).

The overall model structure is schematically depicted in Fig. 7. The model consists of an LSTM layer with l cells, a linear layer, and a softmax layer. The LSTM layer maps the inputs into l different LSTM outputs q. Those LSTM outputs q are combined by a fully connected linear layer into an output vector $z \in \mathbb{R}^{n_c \times 1}$, where n_c denotes the number of output classes of the algorithm. These outputs are re-scaled between zero and one with the softmax function. The output of the softmax layer $\hat{y}_{p,l}(t, \theta)$ is a vector with n_c values between 0 and 1, which gives an indication of the probability that a particular class is occurring at time step t. θ represents the model parameters, e.g., weights and biases. To determine which PVA type is predicted by the model, $\hat{y}_{p,l}(t, \theta)$ is transformed to a one-hot output vector as follows:

$$\hat{y}^c(t) := \begin{cases} 1 & \text{if } \hat{y}^c_{p,l}(t) \geq \hat{y}^m_{p,l}(t)\ \forall m \in \{1, \ldots, n_c\} \\ 0 & \text{otherwise} \end{cases}, \tag{8}$$

where c denotes the class number in the vector. In words, $\hat{y}(t)$ is a vector with a one at the index where $\hat{y}_{p,l}$ is the highest, i.e., the PVA type that the algorithm expects to be present and zero everywhere else.

4.3. Loss function and optimization algorithm

The optimal values of the weights and biases θ are determined to obtain the optimal mapping from the inputs x to the ground-truth labels with the proposed model structure. This is achieved by minimizing a loss function. The loss function, or fit criterion, considered in this paper is the cross-entropy loss function. Other loss functions can be used as well in the proposed framework. However, the cross-entropy loss function leads to faster training and improved generalization compared to the sum-of-squares in classification problems (Simard et al., 2003). The cross-entropy loss function is defined as

$$L_{CE}(\theta) = -\sum_{i=1}^{N}\sum_{t=1}^{n} y^i(t)^\top \cdot \log(\hat{y}^i_{p,l}(t, \theta)), \tag{9}$$

where N is the number of ventilation use cases, n the sequence length of the ventilation use case i in samples, $y^i(t) \in \mathbb{R}^{n_c \times 1}$ the ground truth one-hot label vector of case i at time t, and $\hat{y}^i_{p,l}(t, \theta) \in \mathbb{R}^{n_c \times 1}$ the predicted label vector of case i at time sample t.

To obtain the weights and biases θ that minimize this loss function, different optimization algorithms can be used. In this example, particularly good results have been obtained with the Adam solver (Kingma & Ba, 2015). Adam is a gradient descent optimization algorithm that is widely used in the field of deep learning (Soydaner, 2020).

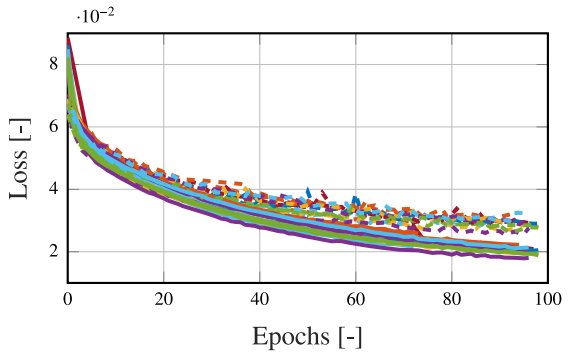


Fig. 8. Training (solid lines) and validation (dashed lines) loss of the training process with slightly different training sets.

4.4. Training of the network

For the training of the network we opt to use 14 patient use cases for training and validation and leave one patient use case as a test case. Of those 14 use cases, 10% of the time-series and labels across the 14 patients is used for validation. The training set is divided into batches of 256 samples and their sequence is shuffled to obtain better training results. The amount of epochs during training is around 100, because the validation loss does not decrease further after cycling through the full training data that many times. In Fig. 8, the training and validation losses for different training sets are shown.

4.5. Performance evaluation metrics

A fair evaluation of the detection performance of the trained network is crucial for future implementation in practice. The output of the model is a multi-variate time-series; therefore, it makes sense to use metrics like accuracy or root mean square errors. These metrics are indeed used during the training process. However, for the performance evaluation, we convert the time-series labels to breath-level labels. Labels on breath-level are more intuitive (especially for interpretation by clinicians) and give us the opportunity to use typical machine learning metrics: precision, recall, and F1-score.

To translate the predicted sample-based labels $\hat{y}(t)$ to breath-by-breath labels and a breath-by-breath performance evaluation criterion, the one-hot output vector $\hat{y}(t)$ of the algorithm is first filtered to eliminate instances where a certain label is classified very briefly. This is achieved by counting which class is classified the most in a predefined window and defining this as the new output $\hat{y}_f(t)$. Thereafter, these filtered labels $\hat{y}_f(t)$ are connected to a breath in the breath-domain based on timing. Eventually, these predicted breath-domain labels are used to compute precision, recall, and F1-score for the different asynchronies. The procedure to obtain the breath-domain accuracy from the predicted time-domain labels $\hat{y}(t)$ is summarized as follows.

Procedure.

1. *The one-hot output vector of the algorithm $\hat{y}(t)$ is filtered with a moving counting filter (MCF) as follows:*

$$a(t) := \left(\sum_{\tau=t-w_s/2}^{t+w_s/2} \hat{y}(\tau) \right), \tag{10}$$

where $a(t)$ is the sum of all vectors in the specified window around sample t with w_s the window size in samples. The output vector of the filter is computed from $a(t)$ with:

$$\hat{y}_f^c(t) := \begin{cases} 1 & \text{if } a^c(t) \geq a^m(t) \forall m \in \{1, \ldots, n_c\}, \\ 0 & \text{otherwise}, \end{cases}$$

where $\hat{y}_f^c(t)$ is the output of the filter at index $c \in \{1, 2, \ldots, n_c\}$ and time sample t, such that $\hat{y}_f(t) = \begin{bmatrix} \hat{y}_f^1(t) & \ldots & \hat{y}_f^{nc}(t) \end{bmatrix}^T$.

2. *The labels $\hat{y}_f(t)$ at a specific time step t are connected to the closest breath phase, i.e., an inspiration or expiration. This gives new breath domain labels $\hat{Y}$. These labels indicate per breath phase which PVA type is detected within that breath phase.*
3. *Using these new labels on breath level, the precision, recall, and F1-score are computed for all the asynchrony types within PSV.*

The breath-by-breath labeling enables us to define the following performance metrics. The precision or Positive predictive value (PPV) of a asynchrony type, reflects how many of the detected asynchronies of that type are true positives, i.e.,

$$PPV = \frac{TP}{TP + FP}, \tag{11}$$

where TP are the true positives, and FP the false positives. The recall or true positive rate (TPR) of an asynchrony type reflects on the ratio between the amount of detected asynchronies and the true amount of asynchronies of that type, i.e.,

$$TPR = \frac{TP}{TP + FN}, \tag{12}$$

where FN are the false negatives. Both of these metrics, highlight the classification performance for a single perspective. Therefore, we also introduce the $F1$-score, which is the harmonic mean of the recall and precision, i.e.,

$$F1 = \frac{2 \cdot PPV \cdot TPR}{PPV + TPR}. \tag{13}$$

5. Performance evaluation

In this section, the performance of the final algorithm is evaluated, which is the third sub-contribution of this paper. First, the performance on a clinical test set is evaluated in Section 5.1. Thereafter, in Section 5.2, an experiment is conducted to analyze the amount of training data that is necessary to train a network with high detection performance and generalization capabilities.

5.1. Detection performance on clinical data

In this section, the performance of the detection algorithm is evaluated. Unlike traditional time-series metrics, which are not practical for clinicians, our focus is on providing insights on a per-breath basis, as this is more relevant in real-world healthcare scenarios. Therefore, a breath-by-breath performance measure is proposed to properly analyze the performance of the algorithm. Specifically, we examine precision (PPV), recall (TPR), and the F1-score for various asynchrony types. In the training process, data of fourteen patients are used for training and validation and the data of the remaining patient are used as a test set. Rotating the test set results in fifteen different trained networks. The upcoming results show the combined results of all the networks together. Besides the precision, recall and F1-score, we also examine the confusion matrix and delay matrix for a more in-depth analysis.

For near real-time detection, it is crucial that the inference step has a low computational cost. On the current hardware, Intel Core i7-9750H CPU, it takes 17.7 ms to predict the asynchrony labels for 1 s of waveform data. Thus, detection of asynchronies during a breath is indeed possible.

In Table 2, the precision (PPV), the recall (TPR), and the F1-score are displayed per asynchrony type. On average the F1-score across all asynchrony types is 0.87. Synchronized inspiration, premature cycling, and auto-triggers are even detected with higher F1-scores compared to the average. Combining the (a)synchrony types into two categories, the synchronous breaths and asynchronous breaths, leads to F1-scores of 0.86 and 0.81, respectively. The challenging asynchrony types to detect are premature triggering, delayed triggering, and both ineffective efforts.

The premature triggering has an F1-score of zero, because in the entire dataset there is only one incidence of this asynchrony type (see the left column of Fig. 6). This incidence is included in the test set which means that the model had no exposure to it during training. Additional clinical data is necessary to train the model on PT asynchronies. The similarity of PT waveforms with other asynchrony types is little, hence, it is expected that training the model on more PT data does not reduce detection performance of other asynchronies.

A more in-depth analysis regarding the delayed triggering is made based on Fig. 9, where the timing asynchronies are displayed based on their inspiration and expiration delays. In this figure, a breath, represented by a data point, is detected correctly if it has the same color as the background of the rectangle it is located in. A hypothesis why detection performance of certain asynchrony types is lower is that those asynchronies are located close to the border of two asynchrony types, making it harder to distinguish (and hence detect) them. However, as can be seen in the figure, this is not entirely the case for the delayed trigger asynchronies (right part in Fig. 9, $\Delta t_{insp} > 0.3$ s). Miss-classifications are located far from the border with normal inspiration, from which we can conclude that these miss-classifications are not due to the asynchrony definition but due to specific patient test cases that are hard to detect for the network.

Additionally, the confusion matrix as presented in Table 3 helps analyzing the miss-classifications further. Delayed triggers are often miss-classified as synchronized inspiration or expiratory ineffective efforts, which are the asynchrony types that have similar pressure and flow curves (Fig. 5). This makes detection by the human eye very challenging. The detection performance of the expiratory ineffective efforts are relatively low due to a lot of false positive. These false positives are a result of cardiac oscillations that are visible in the flow and are similar to ineffective effort waveforms. Another factor that is possibly contributing to the misclassification is the imbalance in the training dataset. Synchronized inspirations (SI) are dominantly present in the training data compared to delayed trigger (DT) asynchronies, which might result in a model that is biased towards detecting SI over DT.

Overall, the detection performance is an improvement compared to clinical practice, where asynchronies remain undetected. Compared to other state-of-the-art, the detection performance is slightly lower for some asynchrony types. On the other hand, the benefit of the proposed detection method is that all asynchrony types within pressure support ventilation can potentially be detected because of the objective PVA rules.

5.2. Training set size and generalization

In this section, the results regarding the training set size and the generalization performance are discussed. Implementation on a ventilator also requires the algorithm to generalize to patients outside of the training set. Therefore, an experiment is designed to test the variation within the dataset together with the generalization ability of the network. Below, we introduce the experimental approach, which is followed by the results and an interpretation of the results.

In the experiment, the clinical data is divided into two groups as shown in Fig. 10. Three use-cases have the same asynchrony types and are similar upon visual inspection. These use cases are defined as A, B, and C. The remaining use cases are defined into one use case named Z. The use case C is used as test set while the other use-cases function as training sets. Two approaches to research the generalization of the network together with the amount of necessary training data are defined: training the LSTM network with and without the Z use-case. In Fig. 10, the different approaches are visually displayed.

In Fig. 11, the performance results obtained after training various networks on the training sets as explained in Fig. 10 are shown. The performance is assessed based on the average F1-score of these networks.

First, the amount of training data that is required to achieve a high detection performance is analyzed. In Fig. 11, it can be

Table 2
Detection results of the combined networks, which show the precision recall and incidence for each of the asynchrony types detected.

Label	PPV	TPR	F1
Premature triggering	0	0	0
Synchronized inspiration	0.87	0.91	0.89
Delayed triggering	0.62	0.44	0.51
Premature cycling	0.96	0.90	0.93
Synchronized expiration	0.77	0.82	0.79
Delayed cycling	0.85	0.82	0.83
Auto triggering	1.00	0.80	0.89
Expiratory ineffective effort	0.69	0.86	0.76
Inspiratory ineffective effort	0.75	0.46	0.57

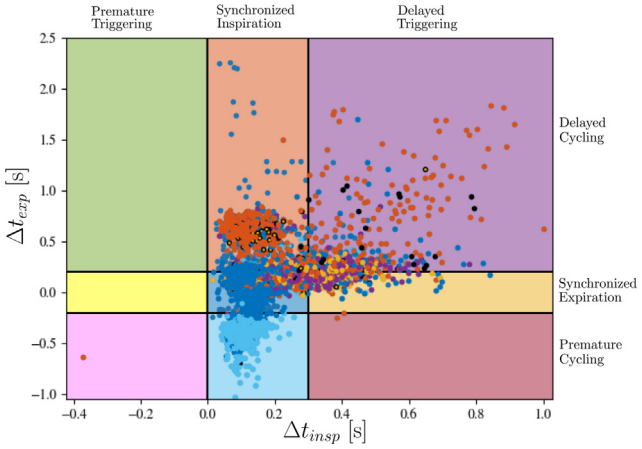


Fig. 9. Results of the detection performance visualized in the delay matrix. A breath is detected correctly if the color of the data point corresponds with the background (background is the true PVA type). The fill and edge of a data point are colored black if, respectively, the inspiration and expiration remain undetected.

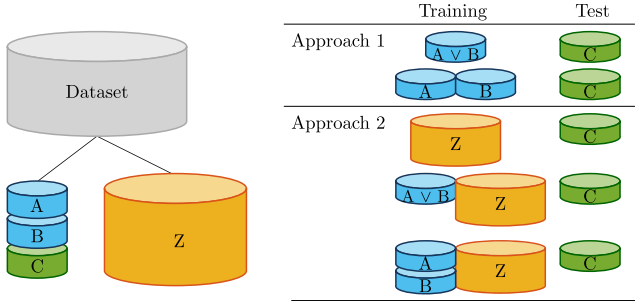


Fig. 10. Visualization of the dataset split into A, B, C, and Z together with the experiment design. Adjusting the size and composition of the training set and using the same test set, results in information about the generalization capabilities of the network.

Table 3
Total confusion matrix of the detected asynchronies.

Labels	Predicted labels										
	0.0	PT	SI	DT	PC	SE	DC	AT	DbT	IEe	IEi
0.0	–	0	118	42	4	35	46	0	0	254	0
PT	0	0	1	0	0	0	0	0	0	0	0
SI	100	0	2378	99	0	0	0	0	0	34	0
DT	83	0	207	296	0	0	0	0	0	93	0
PC	0	0	0	0	445	47	0	0	0	0	0
SE	59	0	0	0	13	1073	159	0	0	0	0
DC	28	0	0	0	0	246	1219	0	0	0	2
AT	1	0	0	0	0	0	0	4	0	0	0
DbT	0	0	0	0	0	0	0	0	–	0	0
IEe	69	0	19	44	0	1	0	0	0	831	0
IEi	0	0	0	0	0	0	7	0	0	0	6

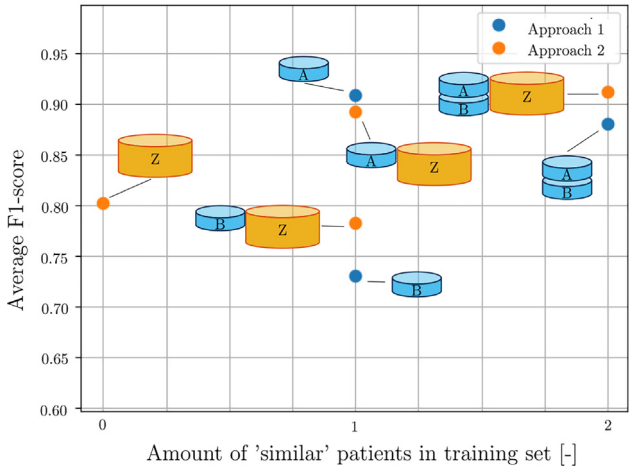


Fig. 11. Average F1-score of the networks trained with different training sizes and compositions. A single similar data in the training set results in a high average F1-score over the different asynchrony types.

seen that one similar set in training already results in a high detection performance, even higher than training on the other 12 cases from the dataset. It is shown that the average F1-score on the test set is much better when training solely on dataset A compared to training solely on set B. Hence, the distribution of set A is more similar to distribution of set C. This highlights that visual waveform similarities might not be a reliable measure for similarity in the distributions across different sets.

Secondly, we are able to analyze the generalization ability of the network. Training the network on dataset Z, results in a F1-score of 0.80, this is much lower compared to training solely on dataset A, while set Z contains approximately 12 times the amount of data points. This shows that the network finds it challenging to generalize to never seen before patient use cases. Therefore, increasing the variability of the training set is important. However, it is crucial to find a balance as excessive variability possibly leads to performance deterioration, as evidenced by the F1-scores of training set Z and B-Z.

In conclusion, the distribution of the training set emerges as a critical factor influencing detection performance. Hence, further research is necessary to determine if multiple networks, one for each patient group, results in superior detection performance.

6. Conclusions and recommendations

The developed Patient-Ventilator Asynchrony (PVA) detection and classification framework enables real-time detection and classification of the relevant types of asynchrony between a patient and the mechanical ventilator using measured data that is currently available on commercial ventilators. This information about PVA can be used by the clinician to prevent/mitigate PVA and therewith reduce ventilation times. With clinical data, it is shown that the developed algorithm is able to detect the different asynchrony types with an average F1-score of 0.87 for the given use cases. The F1-score is a measure of accuracy, where perfect asynchrony detection is achieved if the F1-score is 1. The highest performance is obtained for premature cycling asynchronies with an F1-score of 0.93. The most challenging asynchrony to detect is delayed triggering with an F1-score of 0.51. Investigation on the synchrony versus asynchrony level shows that the network is able to detect asynchronies with an overall F1-score of 0.81. This shows that the algorithm is able to successfully detect a large majority of the asynchronies. Therewith, the automatic detection can support clinicians to improve asynchrony significantly.

It is shown that the algorithm can successfully detect and classify asynchronies. Several recommendations for future improvements are considered. First, it must be noted that the accuracy of detection is highly dependent on the objective PVA definitions. Therefore, further validation is necessary to show objective PVA definitions are in-line with the PVA definitions in clinical practice. The framework as presented in this paper is valid if other objective definitions are used, but requires the model to be trained from the ground up. Furthermore, the objective PVA definitions can be extended to other ventilator modes as future work. This also enables us to extend the PVA detection model to other ventilator modes. Secondly, more knowledge should be used in the current PVA definition, e.g., if the patient's spontaneous breathing effort is small, then it is more important to achieve the desired tidal volume than to prevent asynchrony. Thirdly, the proposed network can be further optimized by changing the network structure to improve the detection performance. Thereafter, as future work, a quantitative analysis can be done to compare the different state-of-the-art detection methods. Fourth, the clinical dataset could be expanded with simulated (synthetic) patient data to resolve the class imbalance problem; hereby, a detection model can be trained that maintains impartiality towards different asynchrony types. Fifth, the algorithm should be implemented in an actual ventilation system, such that it can be used in practice. Finally, we foresee a future system that alerts the clinician if a specific type of asynchrony occurs together with an advice for improved ventilator settings. This is the first step towards patient-in-the-loop control.

CRediT authorship contribution statement

Lars van de Kamp: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation,

Visualization, Writing – original draft, Writing – review & editing. **Joey Reinders:** Conceptualization, Formal analysis, Investigation, Methodology, Supervision, Visualization, Writing – original draft, Writing – review & editing, Validation. **Bram Hunnekens:** Conceptualization, Methodology, Supervision, Writing – review & editing, Validation. **Tom Oomen:** Conceptualization, Methodology, Supervision, Writing – review & editing, Validation. **Nathan van de Wouw:** Conceptualization, Methodology, Supervision, Writing – review & editing, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

The authors wish to thank Francesco Mojoli and Tom Bakkes for giving access to the dataset of 15 patients from the Fondazione I.R.C.C.S. Policlinco San Matteo (reference number 41223).

References

Adams, J. Y., Lieng, M. K., Kuhn, B. T., Rehm, G. B., Guo, E. C., Taylor, S. L., et al. (2017). Development and validation of a multi-algorithm analytic platform to detect off-target mechanical ventilation. *Scientific Reports*, *7*(1), http://dx.doi.org/10.1038/s41598-017-15052-x.

Andersen, R. S., Peimankar, A., & Puthusserypady, S. (2019). A deep learning approach for real-time detection of atrial fibrillation. *Expert Systems with Applications*, *115*, 465–473.

Angus, D. C., Kelley, M. A., Schmitz, R. J., White, A., & Popovich, J. (2000). Current and projected workforce requirements for care of the critically ill and patients with pulmonary disease. *Jama*, *284*(21), 2762–2770.

Bakkes, T., Montree, R., Mischi, M., Mojoli, F., & Turco, S. (2020). A machine learning method for automatic detection and classification of patient-ventilator asynchrony. In *2020 42nd annual international conference of the IEEE engineering in medicine & biology society*. IEEE, http://dx.doi.org/10.1109/embc44109.2020.9175796.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. New York: Oxford University Press Inc.

Blanch, L., Sales, B., Montanya, J., Lucangelo, U., Garcia-Esquirol, O., Villagra, A., et al. (2012). Validation of the better care® system to detect ineffective efforts during expiration in mechanically ventilated patients: a pilot study. *Intensive Care Medicine*, *38*(5), 772–780. http://dx.doi.org/10.1007/s00134-012-2493-4.

Blanch, L., Villagra, A., Sales, B., Montanya, J., Lucangelo, U., Luján, M., et al. (2015). Asynchronies during mechanical ventilation are associated with mortality. *Intensive Care Medicine*, *41*(4), 633–641. http://dx.doi.org/10.1007/s00134-015-3692-6.

Borrello, M. (2021). The application of controls in critical care ventilation. In *2021 IEEE conference on control technology and applications* (pp. 701–718). http://dx.doi.org/10.1109/CCTA48906.2021.9659125.

Colombo, D., Cammarota, G., Alemani, M., Carenzo, L., Barra, F. L., Vaschetto, R., et al. (2011). Efficacy of ventilator waveforms observation in detecting patient-ventilator asynchrony.. *Critical Care Medicine*, *39*, 2452–2457. http://dx.doi.org/10.1097/CCM.0b013e318225753c.

Drumond, R. R., Marques, B. A., Vasconcelos, C. N., & Clua, E. (2018). PEEK: An LSTM recurrent network for motion classification from sparse data. In *Proceedings of the international joint conference on computer vision, imaging and computer graphics theory and applications 1* (pp. 215–222). http://dx.doi.org/10.5220/0006585202150222.

Epstein, S. K. (2011). How often does patient-ventilator asynchrony occur and what are the consequences? *Respiratory Care*, *56*, 25–38. http://dx.doi.org/10.4187/respcare.01009.

Gholami, B., Phan, T. S., Haddad, W. M., Cason, A., Mullis, J., Price, L., et al. (2018). Replicating human expertise of mechanical ventilation waveform analysis in detecting patient-ventilator cycling asynchrony using machine learning. *Computers in Biology and Medicine*, *97*, 137–144. http://dx.doi.org/10.1016/j.compbiomed.2018.04.016.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2009). A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(5), 855–868.

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd international conference on learning representations, ICLR 2015 - conference track proceedings* (pp. 1–15).

Mojoli, F., Pozzi, M., Orlando, A., Bianchi, I. M., Arisi, E., Iotti, G. A., et al. (2022). Timing of inspiratory muscle activity detected from airway pressure and flow during pressure supper ventilation: the waveform method. *Critical Care*, *26*(1), http://dx.doi.org/10.1186/s13054-022-03895-4.

Mulqueeny, Q., Redmond, S., Tassaux, D., Vignaux, L., Jolliet, P., Ceriana, P., et al. (2009). Automated detection of asynchrony in patient-ventilator interaction. In *2009 Annual international conference of the IEEE engineering in medicine and biology society*. IEEE, http://dx.doi.org/10.1109/iembs.2009.5332684.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. Cambridge, Massachusetts: The MIT Press, http://dx.doi.org/10.1007/978-94-011-3532-0_2.

Ng, Q. A., Chiew, Y. S., Wang, X., Tan, C. P., Nor, M. B. M., Damanhuri, N. S., et al. (2021). Network data acquisition and monitoring system for intensive care mechanical ventilation treatment. *IEEE Access*, *9*, 91859–91873.

Pan, Q., Zhang, L., Jia, M., Pan, J., Gong, Q., Lu, Y., et al. (2021). An interpretable 1d convolutional neural network for detecting patient-ventilator asynchrony in mechanical ventilation. *Computer Methods and Programs in Biomedicine*, *204*, Article 106057. http://dx.doi.org/10.1016/j.cmpb.2021.106057.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *Proceedings of the international conference on machine learning* (pp. 1310–1318).

Pham, T., Telias, I., Piraino, T., Yoshida, T., & Brochard, L. J. (2018). Asynchrony consequences and management. *Critical Care Clinics*, *34*(3), 325–341. http://dx.doi.org/10.1016/j.ccc.2018.03.008.

Rehm, G. B., Woo, S. H., Chen, X. L., Kuhn, B. T., Cortes-Puch, I., Anderson, N. R., et al. (2020). Leveraging IoTs and machine learning for patient diagnosis and ventilation management in the intensive care unit. *IEEE Pervasive Computing*, *19*(3), 68–78.

Simard, P. Y., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. In *Proceedings of the international conference on document analysis and recognition* (pp. 958–963). http://dx.doi.org/10.1109/ICDAR.2003.1227801.

Soydaner, D. (2020). A comparison of optimization algorithms for deep learning. *International Journal of Pattern Recognition and Artificial Intelligence*, *34*(13).

Thille, A. W., Rodriguez, P., Cabello, B., Lellouche, F., & Brochard, L. (2006). Patient-ventilator asynchrony during assisted mechanical ventilation. *Intensive Care Medicine*, *32*(10), 1515–1522. http://dx.doi.org/10.1007/s00134-006-0301-8.

van Diepen, A., Bakkes, T. H. G. F., De Bie, A. J. R., Turco, S., Bouwman, R. A., Woerlee, P. H., et al. (2021). A model-based approach to synthetic data set generation for patient-ventilator waveforms for machine learning and educational use. arXiv:2103.15684.

Vignaux, L., Vargas, F., Roeseler, J., Tassaux, D., Thille, A. W., Kossowsky, M. P., et al. (2009). Patient–ventilator asynchrony during non-invasive ventilation for acute respiratory failure: a multicenter study. *Intensive Care Medicine*, *35*(5), http://dx.doi.org/10.1007/s00134-009-1416-5.

Warner, M. A., & Patel, B. (2013). Mechanical ventilation. In *Benumof and Hagberg's airway management* (pp. 981–997). Elsevier, http://dx.doi.org/10.1016/b978-1-4377-2764-7.00048-8.

Wells, C. R., Fitzpatrick, M. C., Sah, P., Shoukat, A., Pandey, A., El-Sayed, A. M., et al. (2020). Projecting the demand for ventilators at the peak of covid-19 outbreak in the usa. *The Lancet Infectious Diseases*, *20*(10), 1123–1125.

Zhang, L., Mao, K., Duan, K., Fang, S., Lu, Y., Gong, Q., et al. (2020). Detection of patient-ventilator asynchrony from mechanical ventilation waveforms using a two-layer long short-term memory neural network. *Computers in Biology and Medicine*, *120*, Article 103721. http://dx.doi.org/10.1016/j.compbiomed.2020.103721.