PRESSURE PREDICTION IN MECHANICAL VENTILATORS

GROUP - 13
Joel David Pérez Arroyave - 20242020017
Cristian David Parroquiano Jimenez - 20222020192
Juan Gonzalez - 20222020200
Santiago Chavarro - 20231020219

contents

1. introduction
2. System Architecture Overview
3. System components and behavior
4. Chaos and system sensibility
5. Technologies and implementation

introduction

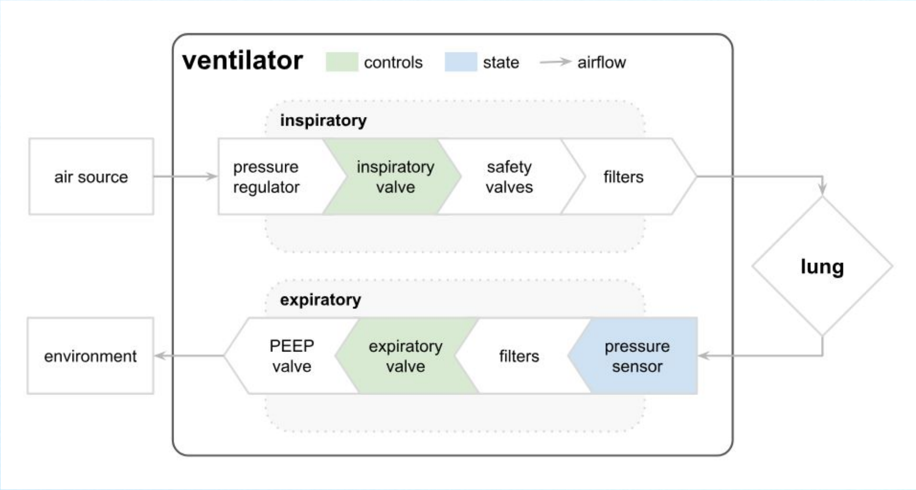
This project presents the analysis and design of an intelligent system for pressure prediction in mechanical ventilators using neural networks. The work aims to optimize the accuracy and adaptability of respiratory simulators, overcoming the limitations of traditional PID controllers. Through two integrated workshops, a hybrid approach is developed that combines physical models with LSTM networks, incorporating safety mechanisms, sensitivity control, and chaos mitigation. The result is a robust and adaptable architecture designed to support the development of medical ventilation systems capable of dynamically adjusting to each patient’s pulmonary characteristics.

02

SYSTEM ARCHITECTURE AND OVERVIEW

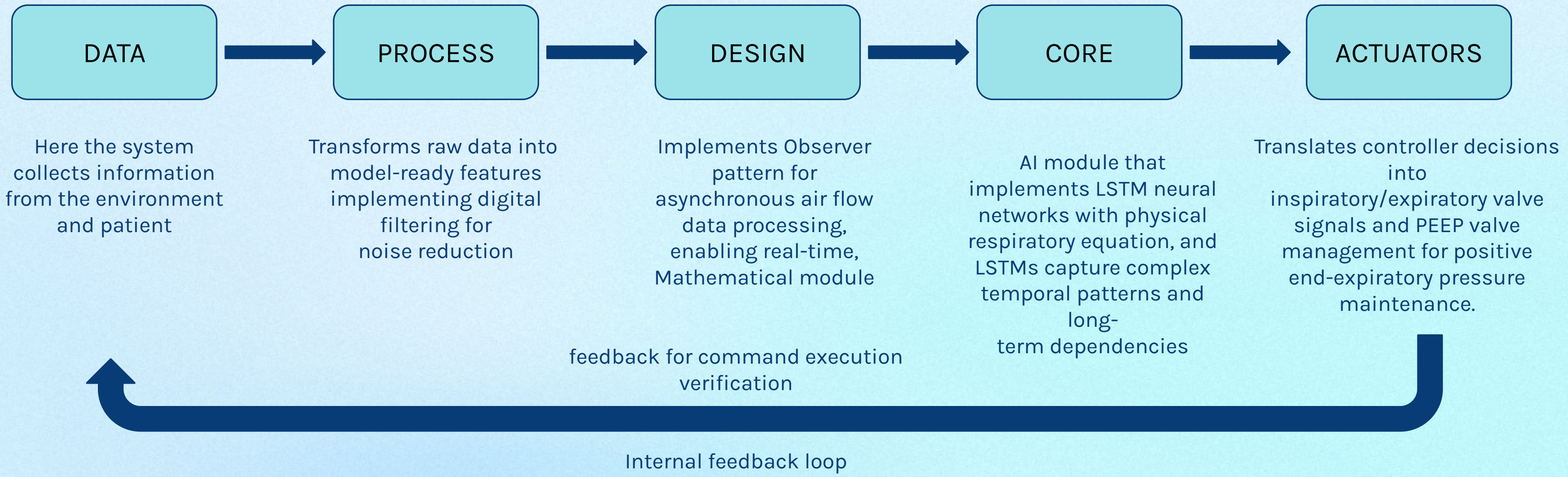
SYSTEM ARCHITECTURE AND OVERVIEW

consists of two main interconnected components, the mechanical ventilator simulating patient lungs and an artificial test lung, connected via respiratory circuitry.
The system processes air flow through multiple stages including pressure regulation, inspiratory valves, safety mechanisms, filtration systems, and feedback loops.



SYSTEM ARCHITECTURE AND OVERVIEW

The system architecture organizes into five coordinated layers achieving design objectives through integrated functionality, to guarantee robustness and real-time performance



03

SYSTEM COMPONENTS AND BEHAVIOR

DATA STRUCTURE

Main inputs and attributes of the system

Air

High-sensitivity input (0.9) requiring precise quality control

Particles

Environmental contaminants managed through filtration

Manual operation

Physician inputs introducing variability

Patient attributes

Anatomical characteristics influencing pressure requirements and illnesses variability

SYSTEM INTERNAL DATA

01
PRESSURE

02
INSPIRATORY VALVE

03
SAFETY VALVE

04
FILTERS

ENVIRONMENT

05
PRESSURE SENSOR

06
EXPIRATORY VALVE

07
PEEP VALVE

R (input)

Inspiratory
Valve opening
Int 0-100

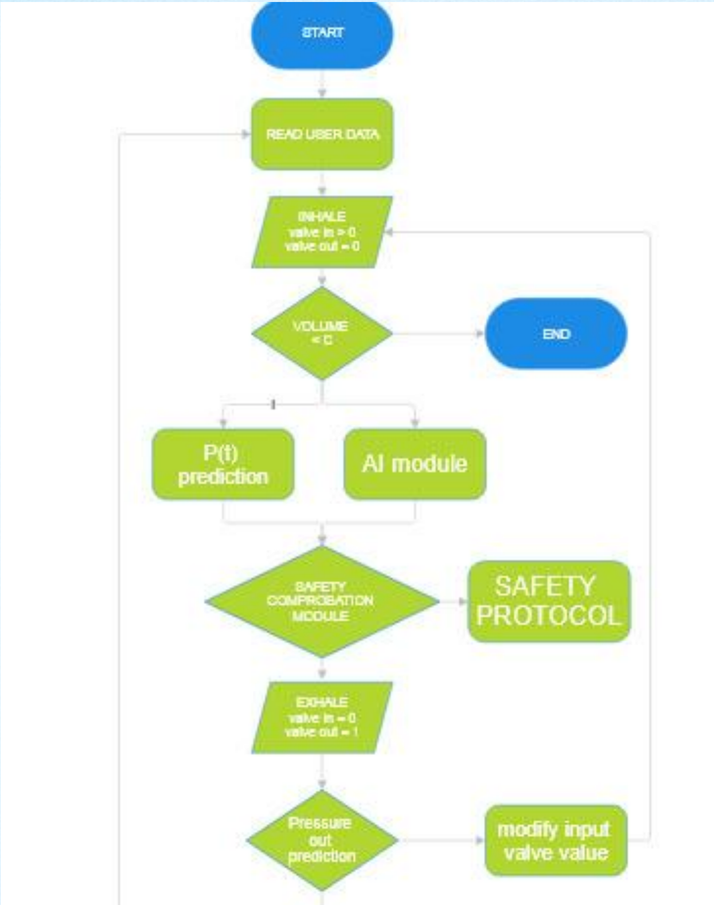
R (output)

expiratory
Valve opening
Boolean

C

Lung air capacity

SYSTEM BEHAVIOR



04

CHAOS AND SENSIBILITY

SENSITIVITY AND COMPLEXITY

The system shows medium-high complexity with strong environmental interactions and feedback.

Air quality and patient traits have the highest sensitivity (0.9) due to their direct effect on survival.

Medical conditions show medium sensitivity (0.8), while operational and environmental factors are lower (0.6) because of system compensation.

INPUT	TYPE	SENSITIVITY
AIR	INTEGER	0.9
PARTICLES	LIST	0.6
MANUAL OPERATION	LIST	0.6
PATIENT ALLERGIES	LIST	0.8
PATIENS ILLNESSES	LIST	0.8
PATIENT ATTRIBUTES	LIST	0.9

System Elements

Pressure

Inspiratory Valve

Filters

Safety Valve

Pressure Sensor

Expiratory Valve

PEEP Valve

Chaotics Dynamics

The system shows non-linear, chaotic behavior, especially during respiratory phase transitions, requiring advanced control strategies.

The simplified R–C model (Resistance–Compliance) omits key factors like viscoelasticity, chest wall effects, and lung heterogeneity, creating gaps between simulation and clinical reality.

COMPLEXITY

01 **Air Quality Variations:** Impact on filtration and pressure requirements

02 **Lung Attribute Diversity:** Anatomical and physiological variations

03 **Pathological Conditions:** Disease-specific ventilation requirements

04 **Manual Operation Variability:** Human factor introduction

05 **Patient Response Heterogeneity:** Individual treatment reactions

05

Technologies and implementation

Architecture Overview

The system is built around five coordinated layers:

- Data Ingestion Layer
- Preprocessing Module
- Core Machine Learning Module
- Safety Controller
- Actuation and Monitoring Layer

Core Technology Selection

01 Programming Language: Python (with C extensions for critical performance parts)

02 Machine Learning Framework: TensorFlow Serving for efficient inference and one alternative Machine Learning Model PyTorch (for flexible experimental models)

03 Communication Framework: ROS2 (Robot Operating System 2) enabling asynchronous publisher-subscriber messaging

04 Data Infrastructure: Apache Kafka for sensor data streams

05 Database Systems: TimescaleDB (for time-series data) and PostgreSQL (for patient records)

06 Numerical Computation: NumPy for mathematical operations

Implementation Methodology

Iteration 1

Build and validate data pipeline + baseline LSTM model.

Iteration 2

Integrate physical models and safety mechanisms.

Iteration 3

Finalize patient-specific adaptations and medical user interface.

Deployment and Integration

Containerization

Docker containers orchestrated by Kubernetes to enable rolling updates and scalable deployment.

Hardware Integration

Uses protocol adapters to translate medical device formats into open standards:

Validation

Strict verification of each hardware interface to ensure operational safety and medical certification readiness.

FUNCTIONS AND FEATURES

Requirements specification

Pressure prediction

Accurate anticipatory pressure calculation

Air quality sustainability

Continuous air purity maintenance

Mathematical modeling

Air flow dynamics understanding through differential equations

Physical sensing

Comprehensive patient behavioral data compilation

$$P(t) = R \cdot Q(t) + 1/C\, V(t) + L\, dQ/dt$$

CONSTRAINTS

- Power Resilience
- Medical certification
- Emergences Mitigation
- Capital Capacity
- Patient Safety

Implementation of sensitivity and Chaos Management

The implemented tool directly addresses the critical challenges identified in the analysis:

Sensitivity Mitigation

Multi-layer validation and real-time sensors monitor air quality, while adaptive filters adjust thresholds to patient-specific data.

Chaos Mitigation

Finite state machines and hysteresis algorithms control non-linear behaviors, with constraints and LSTM models ensuring stable operation and learning of complex dynamics