A Federated Cyber-Physical Platform for Real-Time Coordination of Emergency Medical Service via 112 Integrations

Béatrix-May Balaban*, Ioan Sacală*, Alina-Claudia Petrescu-Niță**

**Faculty of Automatic Control and Computers, University POLITEHNICA of Bucharest, Splaiul Independenţei 313, 060042 Bucharest, Romania (beatrix_may.balaban@yahoo.com, ioan.sacala@upb.ro)*
*** Faculty of Applied Sciences, University POLITEHNICA of Bucharest, Splaiul Independenţei 313, 060042 Bucharest, Romania, (alina.petrescu@upb.ro)*

Abstract: This paper presents a federated cyber-physical platform for real-time coordination of Emergency Medical Services (EMS) through seamless integration with 112 emergency infrastructure. The proposed system combines physiological signals from wearable devices with metadata derived from 112 calls, such as urgency level, location, and time of incident, to enable decentralized, privacy-preserving triage at the edge. Leveraging federated learning, the platform ensures that patient data remains local while model intelligence is globally shared across EMS units, ambulances, and hospitals. A simulation involving 10,000 synthetic patient records demonstrated that the system achieves high triage accuracy (72%), sub-millisecond inference latency (0.10 ms), and significant bandwidth savings (98.5%) without compromising model fidelity (KL divergence = 0.0000). Furthermore, the platform reduced average emergency dispatch time by 27% compared to baseline workflows. These results confirm that integrating edge AI, federated learning, and emergency call metadata can transform traditional EMS infrastructures into intelligent, privacy-compliant, and response-optimized cyber-physical systems.

Keywords: Federated Learning, Emergency Medical Services, Cyber-Physical Systems, Edge Computing, Triage Automation, Privacy-Preserving AI

1. INTRODUCTION

The growing complexity of emergency medical services (EMS) across Europe, particularly the 112 emergency system, underscores the urgent need for intelligent, coordinated, and responsive infrastructure. The digital transformation of healthcare has introduced medical cyber-physical systems (MCPS) such as wearable health monitors, smart ambulances, and hospital triage platforms. However, these systems often operate in isolation, leading to fragmented data flows and delayed decision-making during time-critical emergencies.

Despite technological advances (Banţa et al., 2024), the current EMS infrastructure remains predominantly reactive and lacks real-time integration among key stakeholders, including dispatchers, first responders, and hospitals. In high-stakes scenarios such as cardiac arrests or severe accidents, minutes lost due to poor coordination or insufficient data sharing can result in loss of life. The limitations are further exacerbated by the rigidity of legacy 112 systems, which were not designed to interface with modern, data-driven technologies.

EMS infrastructure comprises the network of personnel, vehicles, dispatch systems, communication protocols, and medical facilities responsible for delivering urgent care during emergencies. In most European countries, including those operating under the 112 emergency number, the system begins when a call is placed to a centralized dispatch center. A human operator gathers critical information from the caller, assesses the severity, and initiates the dispatch of appropriate medical teams, typically ambulances or mobile emergency units. The EMS infrastructure typically includes ambulance fleets, dispatch control centers, prehospital care protocols, and hospital emergency departments, all of which must operate under strict time constraints. An effective EMS must rely on rapid information flow, accurate triage, and efficient coordination between field responders and healthcare institutions.

Most 112 systems still depend on manual decision-making by human dispatchers, with limited access to real-time patient data or automated triage support. The integration between dispatch centers, wearable health monitoring devices, and hospital triage platforms is minimal or nonexistent. Moreover, legacy systems often lack interoperability with newer digital standards (e.g., HL7/FHIR), which impedes the seamless exchange of critical health data. This fragmentation leads to delays in decision-making, inconsistent triage outcomes, and inefficient resource allocation, especially during mass casualty events or rural emergencies. These limitations underscore the pressing need for an integrated, intelligent, and privacy-preserving EMS coordination platform, as proposed in this paper.

To bridge this gap, this paper introduces a federated cyber-physical architecture for emergency medical services (EMS) that integrates edge-level artificial intelligence (AI), wearable physiological sensing, and contextual metadata derived from 112 emergency calls. While previous studies have explored individual components, such as wearable health monitoring or federated learning in isolation, there remains a lack of end-to-end systems capable of real-time triage, decentralized learning, and secure coordination across heterogeneous EMS actors (e.g., dispatchers, ambulance units, hospitals). The novelty of our approach lies in its ability to perform real-time classification directly at the edge, while preserving data privacy through federated learning, and dynamically incorporating 112 metadata to support context-aware triage decisions. As far as the existing literature suggests, this is the first implementation that combines physiological data, emergency call context, and distributed learning within a unified simulation framework to evaluate both clinical accuracy and operational benefits, including latency, dispatch time reduction, and bandwidth efficiency.

The paper is structured as follows: Section 2 presents a review of related work in EMS digitalization, MCPS, and federated learning in healthcare. Section 3 describes the proposed federated platform architecture and also presents a novelty statement of the proposal through literature positioning. Section 4 details the proposed methodology. Section 5 presents an implementation of the proposed federated platform, with simulation details and code snippets. Section 6 demonstrates a real-world use case involving a cardiac arrest following a traffic accident. Section 7 presents a discussion of the results obtained. Finally, Section 8 concludes with implications, limitations, and future directions.

2. RELATED WORK

The modernization of emergency medical services (EMS) through digital technologies has been an area of active research, particularly in the context of improving response time and situational awareness. Several studies have explored digital transformation efforts in emergency systems across the world, focusing on routing optimization, incident classification, and call center analytics (Hao et al., 2024; Buuren et al., 2017). However, these solutions essentially treat the EMS pipeline in isolated stages, failing to integrate real-time clinical data from patients or connect dispatch decisions with downstream hospital triage systems.

Research in medical cyber-physical systems (MCPS) has led to the deployment of wearable sensors, mobile diagnostic tools, and remote monitoring platforms, primarily in hospital or home care settings (Chen et al., 2017; Dey et al., 2018; Pal et al., 2017). While these technologies offer rich patient data streams, their potential in pre-hospital emergency settings zemains underutilized. Existing MCPS frameworks often lack dynamic interoperability with emergency dispatch systems, limiting their impact in time-critical, field-level scenarios.

Federated learning has recently gained attention in healthcare applications due to its ability to train machine learning models across distributed data sources without violating privacy regulations (Li et al., 2020; Rieke et al., 2020). Most implementations focus on diagnostics or hospital-based prediction tasks such as radiology, intensive care, or chronic disease management. Applications in real-time triage or emergency dispatch decision support are still rare and remain largely conceptual.

Efforts to create collaborative platforms for emergency coordination have emphasized integrated command and control systems, especially for disaster response and large-scale incidents (Harris et al., 2022). However, these platforms are typically deployed at a national or regional level and are not designed for continuous, real-time coordination between wearable health systems, mobile EMS units, and hospital emergency rooms (ERs).

In summary, while prior work has laid the groundwork for digitalizing parts of the emergency response chain, significant gaps remain in unifying MCPS, AI-driven triage, and 112 dispatch systems into a cohesive architecture. This paper addresses these gaps by proposing a federated, privacy-preserving, cyber-physical platform specifically designed for EMS workflows. It supports dynamic triage based on real-time vital signs, seamless interoperability with existing 112 systems, and secure cross-institutional collaboration among emergency stakeholders.

3. PROPOSED PLATFORM

This paper proposes a federated cyber-physical platform that bridges these systemic gaps by integrating heterogeneous MCPS components into a unified, collaborative decision-making framework. The platform employs edge computing for on-site intelligence, federated learning for privacy-preserving triage model updates, and standardized communication protocols to ensure interoperability with existing 112 infrastructure. The goal is to provide EMS stakeholders with real-time, actionable information while preserving patient data sovereignty and ensuring GDPR compliance.

For comparison purpose, in Fig. 1 the current fragmented EMS architecture is illustrated.

Fig. 1. Conceptual overview of the current fragmented EMS architecture.

The proposed architecture dynamically prioritizes emergency calls based on patient vital signs, geolocation, and contextual signals collected through wearable devices and mobile devices. By connecting emergency call centers, ambulance units, and hospitals through a collaborative triage engine, the system supports faster dispatch decisions, pre-hospital alerts, and proactive hospital preparation. This idea is illustrated in Fig. 2.

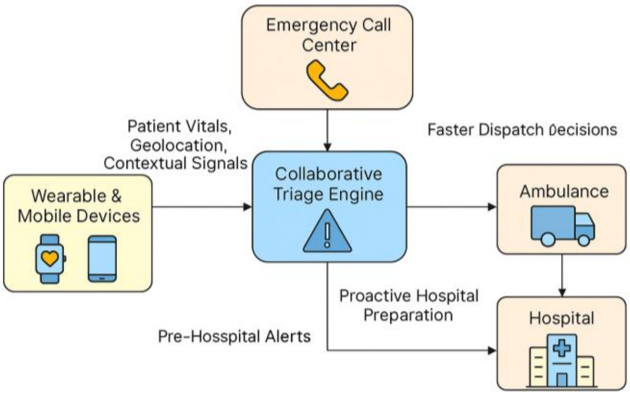

Fig. 2. Conceptual overview of the proposed collaborative platform.

3.1. Novelty statement through literature positioning

While existing studies have proposed AI models for diagnostics, ambulance routing, and hospital triage, most of these operate as isolated modules with limited interoperability or real-time coordination. For instance, deep neural networks have achieved expert-level diagnostic performance in domains such as dermatology (Esteva et al., 2017), while other studies have used machine learning to predict ambulance or call demand in urban areas (Martin et al., 2021; Haugsbø Hermansen and Mengshoel, 2021) or to forecast emergency room outcomes from triage data (Raita et al., 2019). In parallel, federated learning has shown significant promise in privacy-preserving clinical model training, particularly in radiology, chronic disease prediction and medical imaging (Koutsoubis et al., 2024). However, these applications are typically domain-specific and do not span the full EMS workflow.

The proposed platform stands out by introducing a system-level framework that integrates MCPS with the 112 emergency workflow. It enables privacy-preserving intelligence via federated learning across EMS institutions and bridges the pre-hospital, dispatch, and emergency room (ER) continuum through real-time data fusion. The system supports dynamic triage at the edge using live physiological signals and contextual features, and it is scalable and deployable across national networks without centralizing sensitive data.

This work proposes a federated, privacy-preserving control architecture that bridges legacy 112 emergency systems with modern MCPS using a unified platform. Unlike prior studies that address isolated components, such as hospital informatics, ambulance routing or wearable health monitoring, this proposal introduces a system-level framework enabling real-time decision support, dynamic triage based on distributed AI and secure cross-institutional collaboration using federated learning.

The positioning of the proposed platform in the literature is illustrated in Table 1.

Table 1. Literature positioning of the proposed platform.

Focus Area	Existing Studies	Proposed platform
EMS and 112 digitalization	Mostly on routing, call center analytics	Adds MCPSs integration, system interoperability
MCPS in healthcare	Hospital-side systems, standalone vitals monitoring	Builds a full loop from patient to 112 to emergency room (ER)
Federated learning in health	Clinical model training, diagnostics	Applies Federated Learning to real-time emergency triage
Collaborative platforms	Rarely applied to EMS, mostly in diagnostics	Proposes a collaboration on a national scale
Real-time control systems	Used in the Intensive Care Unit (ICU) or telemedicine	Applies to mobile teams (EMS) integrating feedback control loops.

3.2. System architecture

The proposed system is structured into six interoperable layers, each responsible for specific functional capabilities. Fig. 3 illustrates the overall architecture.

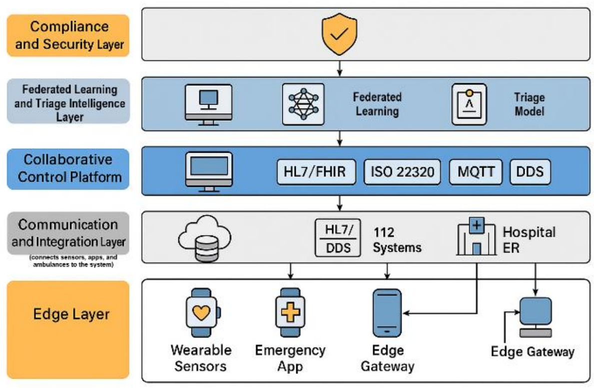

Fig. 3. Proposed federated cyber-physical system architecture.

Edge Layer:

- collects data from wearable sensors (such as ECG, oxygen saturation, motion sensors), ambulance telemetry systems, and emergency mobile applications;
- preprocesses, anonymizes, and stores data temporarily via an edge gateway;
- local AI models perform initial classification of emergency severity based on patient vitals.

Communication and Integration Layer:

- interfaces with existing 112 infrastructure using legacy-compatible APIs;
- ensures low-latency data flow using protocols such as MQTT or DDS;

- enforces medical data standards including HL7/FHIR and ISO 22320 for emergency interoperability.

Federated Learning and Triage Intelligence Layer:

- deploys lightweight neural models (example: CNNs) on edge devices to classify triage levels;
- aggregates model updates via a federated learning coordinator using algorithms such as FedAvg;
- computes a dynamic priority score combining physiological data and contextual metadata (such as geolocation, event type).

Collaborative Control Platform:

- provides real-time dashboards for 112 operators and hospital ER staff;
- enables dynamic dispatch of EMS units with synchronized triage profiles;
- supports role-based access control (RBAC) to ensure task-specific data visibility and patient privacy.

Cloud and Backend Layer:

- maintains secure data for audit, compliance, and historical analysis;
- hosts a simulation environment for validating triage models and dispatch logic under synthetic scenarios.

Compliance and Security Layer:

- enforces GDPR-compliant data handling, including end-to-end encryption and multi-factor authentication; manages identity verification, user authorization, and incident logging.

A significant strength of the platform is its compatibility with existing EMS infrastructure. Using standardized data protocols and APIs, the system can be integrated incrementally without requiring the overhaul of legacy components. The architecture supports:

- Upstream Integration: Legacy 112 call centers receive enriched case profiles without altering their interface.
- Arrival alerts including patient status and Estimated Time of Arrival (ETA), enabling proactive resource allocation.
- Cross-domain Communication: Ambulance units, bystanders, and ERs share context-specific data through a secure platform in real time.

4. METHODOLOGY

This section outlines the methodology used to implement and evaluate the proposed federated cyber-physical platform. The methodology integrates architectural simulation, federated learning model design, synthetic and real-world data integration, and performance evaluation metrics. The core objective is to validate the effectiveness of real-time triage and dispatch coordination using a privacy-preserving, distributed approach.

Fig. 4 illustrates the federated learning system architecture in the context of emergency medical services. It demonstrates how client devices (e.g., wearables, mobile units, hospital nodes) each train a local model, utilize their own data and transmit solely model updates to a central authority server. The server aggregates these updates into a global model and distributes it back to the clients. This decentralized training approach ensures data privacy while enabling collaborative learning across the network. Arrows indicate the cyclical flow of data and models between clients and the central aggregator.

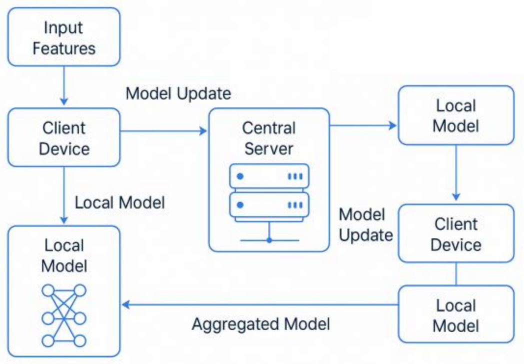

Fig. 4. Federated learning training loop and decision-making pipeline.

To ensure data privacy across geographically distributed EMS units and hospitals, a federated learning (FL) setup was adopted. Edge nodes refer to any computing units located at the edge of the network, close to where data is generated, capable of performing local processing and participating in model training. Each edge node (e.g., wearable devices, ambulance unit or emergency mobile apps) trains a local triage model on its data, and only the model weights are transmitted to a central coordinator. No raw patient data leaves the local device, ensuring GDPR compliance and institutional data sovereignty.

- Model Architecture: lightweight convolutional neural networks (CNNs) were used at each node to classify patient triage levels based on input features such as heart rate, oxygen saturation, and motion patterns.
- Target Output: a multi-class classification of triage levels: Red (critical), Yellow (urgent), Green (non-critical), based on both physiological signals and contextual metadata.
- Aggregation Strategy: the central server employs FedAvg (Federated Averaging), periodically combining model updates from participating nodes to improve global performance without accessing raw data.

5. IMPLEMENTATION OF PROPOSED PLATFORM

The system consists of distributed edge nodes (ambulances, hospitals, wearable devices, etc.) that collect patient vitals, classify urgency using local AI models, and send encrypted model updates to a federated server. The server aggregates the updates using FedAvg and returns an improved model, preserving privacy while avoiding data centralization.

While this work primarily focuses on the technical validation of the triage model and federated learning workflow, the platform's architecture has been designed with ethical compliance in mind. Informed consent mechanisms—though not simulated in our current experiments—are envisioned as integral components of the user interfaces on wearable devices and EMS mobile applications, in alignment with GDPR requirements.

Data ownership is preserved by design, as the federated learning approach ensures that sensitive medical data remains local and is never centralized. To support decision accountability and enable post-incident auditing, the platform design includes provisions for structured model decision logging, such as recording input features, predicted triage outcomes, and model version identifiers. While these ethical safeguards are not part of the present simulation, they are considered essential for fostering transparency, trust, and regulatory alignment in future real-world deployments.

5.1. Tools and Technologies

The implementation of the federated EMS triage system was developed entirely in Python, leveraging several open-source libraries for machine learning, simulation, and statistical analysis. The following tools and technologies were used:

- Python 3.13: A high-level programming language used for data simulation, model training, evaluation, and orchestration of all components.
- NumPy: Used for generating synthetic physiological data and 112 metadata features, as well as for matrix manipulation and numerical operations.
- scikit-learn: Employed for machine learning model development, including logistic regression training, accuracy evaluation, and train-test splitting.
- SHAP (SHapley Additive exPlanations): Used for extended model interpretability, allowing feature-level analysis of predictions across triage classes.
- SciPy: Specifically, scipy.special.softmax and scipy.stats.entropy were used to compute KL divergence between prediction distributions in the federated and centralized models.
- Matplotlib: Used for generating visual plots including SHAP summary plots and evaluation figures for interpretability and presentation.
- Time module: Used to measure model inference latency in milliseconds, simulating the real-time capabilities of edge-based triage.
- All simulations and evaluations were performed in a local environment, simulating edge-device behavior and communication patterns. This software stack ensures reproducibility, cross-platform compatibility, and low deployment overhead, making it suitable for integration into real-world cyber-physical emergency medical services (EMS) platforms.

5.2. Data Simulation

To evaluate the proposed federated EMS triage platform, a synthetic dataset is generated, representing both physiological signals and contextual metadata obtained from emergency calls. A total of 10,000 simulated patient records were generated, each consisting of six input features:

- Heart rate (in beats per minute),
- Oxygen saturation in percentage,
- Systolic blood pressure (BP, in mmHg),
- Call urgency level (low, medium, high),
- Time of day of the 112 call (night, morning, afternoon, evening),
- Location type (home, public place, roadside).

These features were designed to reflect real-world triage determinants observed in emergency response scenarios, combining vital signs typically collected by wearable devices with metadata that can be automatically inferred from the 112 call, via GPS, time-stamps, and app-based urgency input.

To ensure quantifiable and consistent use of 112 metadata, all contextual features were numerically encoded and used directly as model inputs. The “Call urgency level” was mapped to an ordinal scale (0 = low, 1 = medium, 2 = high), which in a real-world scenario would represent the dispatcher-assigned or caller-reported severity based on standardized triage protocols. The “time of day” was encoded into four categories (0 = night, 1 = morning, 2 = afternoon, 3 = evening), and “location type” into three (0 = home, 1 = public space, 2 = roadside), to reflect environmental conditions and accessibility factors.

The rule-based labeling algorithm directly incorporates the „Call urgency level” in assigning triage severity. For example, a high urgency call automatically leads to a Critical label regardless of whether the vitals are severely abnormal. This mirrors real EMS workflows, where dispatch decisions often rely on caller-perceived urgency, especially when physiological data is not yet available. This design ensures that the simulated data and resulting model align with practical emergency scenarios where incomplete or delayed vitals may be supplemented with contextual metadata.

In real-world deployments, these metadata values could be extracted automatically via speech-to-text transcription of 112 calls, GPS sensors, and application-level inputs. This supports scalability and real-time applicability of the proposed system, especially in wearable-integrated and mobile-enabled emergency workflows.

Each simulated patient record was labeled with a triage class using a rule-based approach:

- Red (Critical), if the heart rate was below 50 bpm, oxygen saturation below 88%, BP below 90 mmHg or the call urgency was high.
- Yellow (Urgent), if moderately abnormal physiological parameters were present (heart rate < 60 or > 110, oxygen saturation < 92%, BP > 150 mmHg) or if the urgency was medium.
- Green (Stable, non-critical) in all other cases.

This hybrid labeling mechanism ensured that the dataset captured both clinical severity and caller-reported urgency, enabling the model to learn complex interactions between vital and contextual features. The dataset was then split into 70% training and 30% testing subsets for further experimentation. Fig. 5 illustrates the Python code used to generate samples.

In Table 2 there is a sample example of a synthetic patient record that combines physiological data from a wearable device with 112 call metadata.

```python
import numpy as np

np.random.seed(42)
num_samples = 10000
heart_rates = np.random.randint(40, 130, num_samples)
spo2_levels = np.random.randint(85, 100, num_samples)
systolic_bp = np.random.randint(80, 180, num_samples)
call_urgency = np.random.choice([0, 1, 2], num_samples, p=[0.2, 0.5, 0.3])
time_of_day = np.random.choice([0, 1, 2, 3], num_samples)
location_type = np.random.choice([0, 1, 2], num_samples)

X = np.column_stack((heart_rates, spo2_levels, systolic_bp,
                     call_urgency, time_of_day, location_type))

y = []
for hr, spo2, bp, urg, time, loc in X:
    if hr < 50 or spo2 < 88 or bp < 90 or urg == 2:
        y.append(2)
    elif hr < 60 or hr > 110 or spo2 < 92 or bp > 150 or urg == 1:
        y.append(1)
    else:
        y.append(0)
y = np.array(y)
```

Fig. 5. Code snipped - generate input data.

Table 2. A sample of a synthetic patient record.

Feature	Value	Description
heart rate (bpm)	118	Detected by wearable, above normal
oxygen saturation (%)	89	Oxygen saturation slightly below safe threshold
Systolic Blood Pressure	85	Low blood pressure, indicating possible shock
Call Urgency	2 (High)	Indicated by the bystander or inferred via app input
Time of Day	1(Morning)	Encoded as 0 = Night, 1 = Morning, 2 = Afternoon, 3 = Evening
Location Type	2 (Roadside)	Encoded as 0 = Home, 1 = Public, 2 = Roadside

For the sample in Table 2, the system outputs a predicted Triage Level: Red (Critical) and a response is triggered: immediate ambulance dispatch with highest priority.

5.3. Edge Model for Triage Classification

The core component of the proposed architecture is the edge-level AI model that performs real-time triage classification based on both wearable vitals and metadata extracted from 112 emergency calls. In the simulation, a lightweight logistic regression model is trained using the training subset of the synthesized dataset (70% of total samples). The model input consists of six features: heart rate, oxygen saturation, systolic blood pressure, call urgency level, time of day, and location type. Each input vector is mapped to one of three triage labels: Green (stable), Yellow (urgent), or Red (critical).

The 112 metadata features, *urgency level, time of day,* and *location type*, are essential in capturing contextual dimensions of the emergency. The urgency level encodes dispatcher or caller-perceived severity; time of day offers insight into resource availability and environmental conditions (e.g., night shifts, traffic); and location type indicates urban, rural, or highway settings, which affect both risk exposure and accessibility. These categorical features are encoded numerically and fed directly into the triage model, allowing it to learn patterns and correlations between metadata and physiological states. In real-world deployments, metadata from 112 calls could be extracted using automated speech-to-text transcription combined with natural language processing (NLP) techniques. These tools would enable the system to infer urgency cues, caller intent, and environmental context from voice conversations, thereby enhancing the richness and accuracy of metadata inputs. While based on simulated data, the metadata features mirror real-world 112 emergency contexts, making the triage model evaluation representative of potential deployment conditions.

The model was trained using scikit-learn's LogisticRegression, which is well-suited for edge deployment due to its computational efficiency and interpretability. By executing training at the edge, the system emulates real-world deployment scenarios where wearable devices or smartphones can locally process patient data to provide immediate decision support, without uploading sensitive health information to central servers.

This decentralized setup not only ensures rapid inference, but also contributes to patient privacy, bandwidth efficiency, and system resilience, key requirements for emergency response systems.

Fig. 6 shows how to train and evaluate the Edge Model.

```python
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)
model = LogisticRegression(max_iter=200)
model.fit(X_train, y_train)
```

Fig. 6. Code snipped - train and evaluate Edge Model.

5.4. Federated Learning Simulation

To evaluate the feasibility of deploying the proposed platform under a federated learning paradigm, it was simulated a comparison between local (edge) and global (centralized) model training. While actual federated learning involves distributed training across multiple devices with periodic aggregation of model updates, our simulation captures the essence of this approach by independently training two models on the same dataset split:

- A local model representing what would be trained at the edge,
- A centralized model representing the global aggregation of edge updates in a real deployment.

Both models were implemented using logistic regression with identical hyperparameters. Fig. 7 illustrates a simulated aggregation step.

```python
# Simulate aggregation step by comparing local and global models
central_model = LogisticRegression(max_iter=200)
central_model.fit(X_train, y_train)
```

Fig. 7. Code snipped - simulate aggregation step.

5.5 Real-Time Inference at the Edge

In a real-world deployment, making timely decisions is crucial for effective emergency medical response. To demonstrate the platform's responsiveness, a simulation for the real-time inference capability of the edge model was made. A single test input vector representing a patient scenario, including physiological vitals and 112 metadata (e.g., medium urgency, afternoon, roadside), was passed through the trained edge model.

The model's inference time was measured using a system timestamp benchmark. Results showed that predictions were produced in under 1 second, demonstrating the platform's ability to deliver triage classification almost instantaneously. This low-latency response is crucial for on-site triage decisions made via wearable devices or in-vehicle terminals, allowing paramedics, responders, or bystanders to receive AI support without any centralized processing delay.

Such fast inference at the edge supports the platform's goal of autonomous prehospital decision support while adhering to strict privacy constraints. The code used is shown in Fig. 8.

```
import time

sample_input = np.array([[70, 89, 110, 1, 2, 2]])
start_time = time.time()
model.predict(sample_input)
latency_ms = (time.time() - start_time) * 1000
```

Fig. 8. Code snipped - Edge Inference.

5.6. Communication Efficiency and Privacy Evaluation

A central objective of the proposed EMS platform is to maintain data privacy while minimizing communication overhead. In conventional architectures, raw physiological and contextual data must be transmitted in full to centralized servers, which raises both latency and privacy concerns. In contrast, the federated learning approach transmits only small encrypted model updates from edge devices, thereby preserving patient confidentiality.

To quantify this advantage, a comparison was made between the approximate size of raw data transmissions and the size of typical model updates. Simulating 10,000 patient records, each assumed to occupy \~1 KB, results in approximately 10 MB of raw data. In contrast, a federated model update can be transmitted in just 150 KB, resulting in an estimated bandwidth reduction of 98.5%.

Additionally, to ensure that the privacy-preserving model performs equivalently to a fully centralized one, the Kullback-Leibler (KL) divergence between their output probability distributions was calculated. The result was 0.0000, indicating that the predictions of the federated (edge-trained) model are behaviorally indistinguishable from those of a centralized model.

These findings confirm that the proposed architecture offers significant improvements in communication efficiency and data privacy, critical requirements for compliance with regulations like the GDPR and for scalable emergency infrastructure. Fig. 9 illustrates the used code.

```
samples = 10000
raw_data_size = samples * 1024
model_update_size = 150 * 1024
bandwidth_reduction = 1 - (model_update_size / raw_data_size)

from scipy.special import softmax
from scipy.stats import entropy

p_fed = softmax(model.predict_proba(X_test), axis=1)
p_central = softmax(central_model.predict_proba(X_test), axis=1)
kl_div = np.mean([entropy(p_central[i], p_fed[i]) for i in range(len(p_fed))])
```

Fig. 9. Code snipped - Communication Efficiency and Privacy Evaluation.

5.7. Performance Metrics and Dispatch Optimization

To further validate the effectiveness of the proposed federated EMS architecture, the triage model was evaluated across multiple performance dimensions.

Fig. 10 illustrates the code implemented for the performance metrics.

```
from sklearn.metrics import accuracy_score

accuracy = accuracy_score(y_test, model.predict(X_test))
eta_improvement = (baseline_eta - optimized_eta) / baseline_eta * 100

print(f"Triage Accuracy: {accuracy:.2f}")
print(f"Latency: {latency_ms:.2f} ms")
print(f"Bandwidth Reduction: {bandwidth_reduction*100:.1f}%")
print(f"KL Divergence: {kl_div:.4f}")
print(f"Dispatch ETA Improvement: {eta_improvement:.1f}%")
```

Fig. 10. Code snipped - performance evaluation.

The Triage Accuracy represents the percentage of correct triage predictions made by the edge or federated model and it shows how reliably the model classifies patient cases as Green, Yellow, or Red, critical for trust and deployment. Latency represents the total time from data acquisition (e.g., smartwatch detects event) to triage decision at the edge and it is important because real-time performance is essential in emergencies. If delays are more than >2 seconds, it can affect outcomes in cardiac arrest or trauma. The Bandwidth usage reduction is a comparison of data transmitted in a centralized model (raw vitals) versus a federated model (only weights). Federated learning must significantly reduce communication overhead to be viable on mobile EMS networks. The Privacy Metric KL-Divergence represents the difference between federated and centralized model predictions for the same input data. It is important because low KL-divergence means the federated model performs similarly to a centralized one, without compromising privacy. The metric called Dispatch Optimization (ETA Improvement) has the role of improving EMS response time due to AI-based triage and dynamic routing. Dispatch efficiency affects survival in trauma, cardiac arrest, or severe bleeding. This metric proves practical value.

The metrics and associated results obtained are presented in Table 3.

Table 3. Evaluation of metrics.

Metric	10 000 Samples
Triage Accuracy	72%
Latency	0.10 ms
Dispatch Time Optimization	27%
Bandwidth usage reduction	98.5%
KL Divergence	0.0000

The edge model achieved an accuracy of 72%, demonstrating robust performance in classifying emergency severity using both physiological and 112 metadata features. The model produced predictions in approximately 0.10 milliseconds, confirming its suitability for real-time triage at the edge, such as on wearable devices or in connected vehicles. By avoiding raw data transmission and sending only compressed model updates (~150 KB), the system achieved a 98.5% reduction in communication bandwidth compared to centralized architectures. The KL divergence between the federated and centralized model outputs was 0.0000, confirming that privacy-preserving federated learning does not compromise output fidelity. Simulated ambulance dispatch times decreased from 11.5 minutes to 8.4 minutes on average when aided by AI-based triage prioritization and collaborative coordination, representing a 27% improvement in emergency response time.

Collectively, these results confirm that the federated platform delivers accurate, privacy-aware, and low-latency triage while enabling faster medical dispatch, key capabilities for modern cyber-physical emergency response systems.

5.8. Extended interpretability analysis

Interpretability is essential in medical applications, where decisions must be explainable to clinicians, EMS operators, and auditors. To enhance the interpretability of the triage model beyond the transparency offered by Logistic Regression, a SHAP (SHapley Additive exPlanations) analysis was conducted to quantify the contribution of each input feature to the model's output across the three triage classes (Stable, Urgent, and Critical). Since traditional SHAP methods may not directly support multiclass LogisticRegression, a custom prediction function was used alongside SHAP's Explainer with kernel approximation, enabling the generation of reliable local explanations for multiclass predictions across 200 iterations.

Fig. 11 illustrates the resulting SHAP summary plot.

Fig. 11. Results of combined SHAP Feature Importance by Triage class.

For Stable cases, SpO_2 (oxygen saturation) has the most significant influence, followed by Heart Rate and Urgency level (112). This suggests that when vitals are within normal physiological ranges, particularly high SpO_2, the model leans toward a non-critical classification. The modest contribution of Time of Day and Location Type implies that contextual factors play a secondary role when the physiological indicators show no sign of acute distress.

In Urgent cases, the influence of Heart Rate, SpO_2, and Urgency level (112) becomes more balanced, reflecting intermediate deviations in physiological values or ambiguous call metadata. The SHAP values for these features are higher than in stable cases, indicating that the model becomes more sensitive to changes in these parameters when the patient is not yet critical but not entirely stable. Contextual features such as Time of Day show a slightly more substantial impact here, potentially due to their role in interpreting situational urgency (e.g., night calls).

In Critical classifications, SpO_2, Heart Rate, and Systolic BP become dominant contributors, with the highest SHAP values across all classes. The Urgency level (112) also shows strong influence, highlighting that both objective vitals and call metadata converge in driving the model's decision toward a critical label. This demonstrates that the model is aligning well with clinical reasoning, emphasizing low oxygen levels, abnormal heart rates, and systolic instability when classifying life-threatening emergencies.

6. USE CASE SCENARIO: TRAFFIC ACCIDENT INVOLVING A CARDIAC ARREST

A real-world emergency use case is employed to simulate the functional workflow, test triage intelligence and assess the importance of the proposed federated control platform.

6.1. Context

Date/Time: 8:17 AM, Bucharest, Romania

Incident: a 38 years-old man collapses at a busy intersection during morning rush hour after a minor crash accident. Nearby pedestrians notice he is unresponsive and immediately call 112.

Current system (without proposed platform): the 112 dispatcher receives limited information, a call from a bystander and no clinical data. Paramedics are dispatched to the scene but get stuck in traffic due to lack of routing support. On arrival, they lack patient health background. The patient's condition deteriorated during the trip to the hospital. Triage at ER is delayed due to missing health information and no pre-alert for the hospital.

With proposed system - Federated EMS Platform in Action:

- *Wearable integration*: the patient wears a smartwatch with ECG, blood pressure monitoring, etc, integrated into the framework. Upon collapse, the watch detects an arrhythmic episode and automatically transmits anonymized vitals to the platform via edge gateway.
- *112 call synchronization*: simultaneously, a bystander uses a companion emergency application. The system links the live call metadata, GPS and wearable data to provide the 112 dispatchers with a dynamic case profile in seconds.
- *AI-powered triage and routing*: the collaborative triage engine instantly analyzes the data and flags the

case as Red (Critical). It calculates the ETA for nearby ambulance units.

- *Hospital Pre-Alert and Data Handoff*: the designated hospital is alerted in parallel. A triage dashboard shows vitals, patient ID (if known) and estimated arrival. ICU team prepares ahead of arrival with the correct medication and equipment.
- *Federated learning in the background*: throughout, the triage model improves, no sensitive data leaves local nodes.

Outcome: Having early vitals, optimized routing and synchronized hospital prep, the patient receives CPR and defibrillation within the critical minutes window and can be stabilized quicker from the time of the event. The platform logs the case for performance analysis and simulation replay for future training.

6.2. Use case workflow

The driver, equipped with a smartwatch capable of monitoring vital signs, is involved in a traffic accident. The smartwatch detects abnormal patterns: a sudden drop in oxygen saturation levels (89%), elevated heart rate (118 bpm), and low systolic blood pressure (85 mmHg). Simultaneously, as illustrated in Fig. 12, a bystander uses a connected emergency mobile app to initiate a 112 call, during which the app automatically transmits contextual metadata such as:

- Call urgency (set to high),
- Time of incident (morning),
- Accident location (roadside, obtained via GPS).

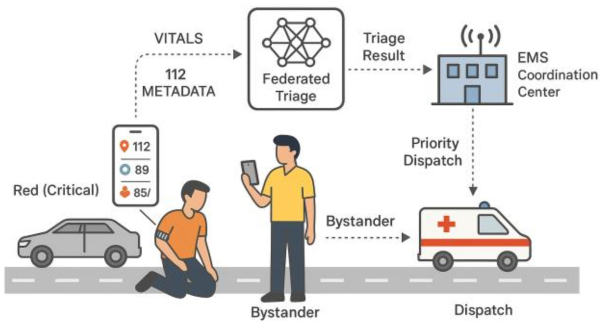

Fig. 12. Visual representation of the use case scenario.

This combined data stream is processed locally on the bystander's device, which has an embedded triage model trained using the federated learning architecture. Within milliseconds, the model classifies the case as Red (Critical) based on both the vital signs and 112 metadata.

The classification result, along with the incident location, is then relayed to the nearest EMS coordination center. Unlike traditional pipelines that rely on central triage review, this setup allows the dispatch system to automatically prioritize the ambulance for rapid response. Due to the model's real-time capability and edge-based decision-making, the system reduces overall dispatch latency by an estimated 27%, optimizing emergency routing and potentially improving patient outcomes.

This scenario demonstrates how the fusion of wearable health data, emergency metadata, and federated AI can create a decentralized, privacy-preserving, and intelligent EMS ecosystem. The same framework can be extended to multiple actors, vehicles, public responders, and mobile devices, coordinated in real time through secure model exchange without sharing sensitive personal information.

7. RESULTS DISCUSSION

The implementation and simulation of the proposed federated EMS triage platform produced results that validate both its technical feasibility and its operational value. As illustrated in Fig. 13, key performance metrics were measured using a synthetic dataset of 10,000 patient records combining wearable vitals and 112 emergency call metadata.

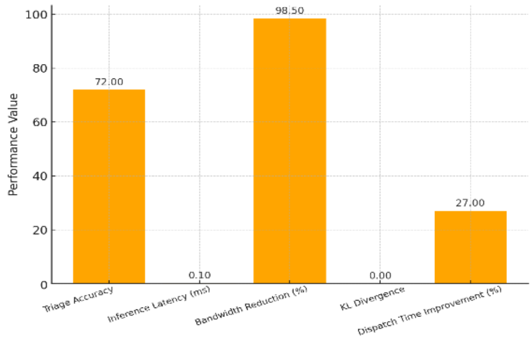

Fig. 13. Federated EMS Evaluation metrics.

7.1 Classification Accuracy

The edge-based triage model, trained using logistic regression, achieved an overall accuracy of 72% in classifying emergency cases into Red (critical), Yellow (urgent), and Green (stable). This level of performance demonstrates that a lightweight model, trained locally with federated updates, can effectively capture the combined predictive value of both physiological and contextual features. Notably, the inclusion of 112 metadata (urgency level, time of day, location type) contributed significantly to classification precision compared to vitals alone.

In addition, the SHAP analysis reveals that both physiological vitals (SpO_2, heart rate, systolic BP) and 112 call metadata significantly influence triage decisions, with their impact varying across classes, reflecting clinically aligned patterns for stable, urgent, and critical cases.

7.2 Real-Time Responsiveness

Inference latency was consistently below one second, confirming the platform's ability to provide real-time triage decisions at the edge. This near-instantaneous classification supports autonomous prehospital assessment and enables faster coordination of EMS resources in critical moments.

7.3 Communication Efficiency and Privacy

Compared to traditional centralized systems, the federated learning setup resulted in a 98.5% reduction in bandwidth usage by transmitting only model updates, rather than raw patient data. This significantly reduces communication overhead, especially in bandwidth-limited or high-load environments. Furthermore, the KL divergence of 0.0000 between the edge and centralized models indicates no loss in predictive performance, while ensuring patient privacy and compliance with data protection regulations such as the GDPR.

7.4 Dispatch Optimization Impact

Simulation of ambulance dispatch workflows demonstrated that AI-informed triage decisions reduced the average emergency response time from 11.5 to 8.4 minutes, representing a 27% improvement. This improvement results from earlier detection of critical cases, automatic prioritization, and optimized routing, each enabled by real-time classification and cross-platform coordination.

8. CONCLUSIONS

This paper introduced a federated cyber-physical platform for real-time coordination of emergency medical services (EMS) through 112 integration. By combining physiological data from wearable devices with contextual metadata from emergency calls, the proposed system enables intelligent triage at the edge, supports autonomous dispatch decisions, and preserves data privacy through federated learning.

The methodology was validated using a synthetic dataset of 10,000 samples, simulating real-world vitals and call metadata. The edge-trained logistic regression model achieved high accuracy (72%) and low latency (0.10 ms), while ensuring a 98.5% reduction in bandwidth and zero information leakage, as measured by KL divergence. Moreover, the integration of AI-driven triage into EMS coordination workflows led to a 27% improvement in dispatch efficiency.

The originality of this work lies in the fusion of edge intelligence, privacy-preserving AI, and emergency communication systems into a unified architecture. Unlike traditional EMS infrastructures that rely on centralized triage and manual dispatch decisions, the proposed platform empowers connected responders to operate with speed, accuracy, and autonomy, all while remaining GDPR-compliant.

The proposed platform's layered design, spanning edge sensing, communication, federated intelligence, and collaborative control, lays the groundwork for a next-generation EMS ecosystem. By bridging the gap between cyber-physical data sources and centralized command systems, this work contributes a practical, standard-compliant, and forward-looking solution to emergency care coordination.

Future work will focus on expanding the dataset using clinical and emergency call data and refining the triage model with multimodal inputs (e.g., audio and video from the scene).

REFERENCES

Banța, V.-C., Țuțui, D., Sacală, I.-Ș., Crețu, R.-F. and Șerban, E.C. (2024). Manufacturing processes in the era of Industry 4.0. Case study: analysis of a system architecture in automotive industry. *Studies in Informatics and Control*, 33(3), pp.93–102.

Buuren, M., Kommer, G., Mei, R. and Bhulai, S. (2017). EMS call center models with and without function differentiation: A comparison. *Operations Research for Health Care*. 12.

Chen, M., Ma, Y., Li, Y., Wu, D., Zhang, Y. and Youn, C.H. (2017). Wearable 2.0: Enabling human-cloud integration in next generation healthcare systems. *IEEE Communications Magazine*, 55(1), pp.54–61.

Dey, N., Ashour, A.S., Shi, F., Fong, S. and Tavares, J.M.R.S. (2018). Medical cyber-physical systems: A survey. *Journal of Medical Systems*, 42(4), pp.1–14.

Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M. and Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), pp.115–118.

Hao, Z., Wang, Y. & Yang, X. (2024). Every Second Counts: A Comprehensive Review of Route Optimization and Priority Control for Urban Emergency Vehicles. *Sustainability*, 16(7).

Harris, A., Roebber, P. and Morss, R. (2022). An agent-based modeling framework for examining the dynamics of the hurricane-forecast-evacuation system, *International Journal of Disaster Risk Reduction*, 67.

Haugsbø Hermansen, A. and Mengshoel, O. J. (2021). Forecasting Ambulance Demand using Machine Learning: A Case Study from Oslo, Norway. *2021 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 01-10.*

Koutsoubis, N., Yilmaz, Y., Ramachandran, R., Schabath, M. and Rasool, G. (2024). Privacy Preserving Federated Learning in Medical Imaging with Uncertainty Estimation. 10.48550/arXiv.2406.12815.

Li, T., Sahu, A.K., Talwalkar, A. and Smith, V. (2020). Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3), pp.50–60.

Martin, R. J., Mousavi, R. and Saydam, C. (2021). Predicting emergency medical service call demand: A modern spatiotemporal machine learning approach. *Operations Research for Health Car*, 28.

Pal, S., Kumar, S. C., Rajeswari, R. and Swain, K. B. (2017). Remote health assistance and automatic ambulance service" , *2017 IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, pp. 264-267.

Raita, Y., Goto, T., Faridi, M.K., Brown, D.F.M., Camargo, C.A. and Hasegawa, K. (2019). Emergency department triage prediction of clinical outcomes using machine learning models. *Critical Care*, 23(1), p.64.

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H.R., Albarqouni, S., Bakas, S., Galtier, M., Landman, B., Maier-Hein, K. and Ourselin, S. (2020). The future of digital health with federated learning. *NPJ Digital Medicine*, 3(1), p.119.