

Reinforcement Learning-Driven Adaptive Synchronization of Real-Time Biomedical Signal Streams [†]

Bogdan Nicușor Bindea ^{1,*} , Călin Cenan ¹, Gratiana Hermann ²  and Zeno Sparchez ³

¹ Department of Computer Science, Technical University of Cluj-Napoca, 400027 Cluj-Napoca, Romania; calin.cenan@cs.utcluj.ro

² Department 1—Pathological Anatomy, Morpho-Functional Sciences, Faculty of Medicine, “Iuliu Hațieganu” University of Medicine and Pharmacy, 400012 Cluj-Napoca, Romania; hermann.gratiana.meital@elearn.umfcluj.ro

³ Department 4—3rd Medical Clinic, Internal Medicine, Faculty of Medicine, “Iuliu Hațieganu” University of Medicine and Pharmacy, 400012 Cluj-Napoca, Romania; zsparchez@gmail.com

* Correspondence: bogdan.bindea@cs.utcluj.ro

[†] Presented at the International Conference on Electromagnetic Fields, Signals and BioMedical Engineering (ICEMS-BIOMED), Suceava, Romania, 7–9 May 2026.

Abstract

Real-time biomedical monitoring systems generate continuous streams of physiological data that must be integrated into analytical repositories with minimal latency. Traditional Extract–Transform–Load (ETL) pipelines used in healthcare infrastructures typically rely on static synchronization schedules that cannot adapt to bursty workloads or fluctuating computational resources. This often leads to increased latency, stale analytical views, or inefficient resource usage. This paper proposes a reinforcement learning-driven adaptive ETL orchestration framework designed for biomedical signal streams in intensive care environments. The proposed approach models synchronization scheduling as a sequential decision problem where an agent dynamically adjusts synchronization intervals and batch sizes based on real-time telemetry, including latency, backlog size, throughput, and system resource utilization. The architecture integrates change-data-capture (CDC) mechanisms and idempotent database operations to ensure incremental synchronization and consistency. Experimental evaluation using simulated ICU-style workloads demonstrates that the proposed approach improves data freshness and reduces synchronization latency compared with fixed-interval and heuristic baseline strategies. The results highlight the potential of reinforcement learning-based orchestration for improving the performance and scalability of biomedical data infrastructures, supporting more reliable real-time clinical monitoring and decision-support systems.

Keywords: reinforcement learning; biomedical signal processing; ETL orchestration; real-time data pipelines; ICU monitoring systems; adaptive synchronization



Academic Editors: L. Dan Milici,
Mihai Munteanu, Simona Miclăuș,
Paul Bechet and Radu Ciorap

Published: 30 June 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the [Creative Commons
Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Biomedical monitoring systems in modern hospitals continuously generate large volumes of physiological data, including electrocardiographic signals, blood pressure measurements, oxygen saturation levels, and respiratory parameters. These data streams are essential for patient monitoring, predictive analytics, and clinical decision-support systems. Ensuring timely integration of such signals into analytical repositories is therefore a critical requirement for modern healthcare infrastructures.

Recent advances in cloud-based healthcare data platforms and intelligent ETL pipelines have improved the scalability of biomedical data processing systems [1–3]. However, synchronization mechanisms between operational systems and analytical databases frequently rely on static Extract–Transform–Load scheduling strategies. Fixed synchronization intervals can lead to inefficient data pipelines when workloads vary significantly over time.

Under high data arrival rates, static ETL schedules may cause synchronization backlogs and increased latency, resulting in outdated analytical views. Conversely, during low workloads, frequent synchronization cycles may unnecessarily consume computational resources. These limitations motivate the development of adaptive synchronization strategies capable of dynamically adjusting pipeline behavior.

Reinforcement learning has emerged as an effective technique for adaptive decision-making in dynamic environments. Applications of reinforcement learning in resource allocation, scheduling, and pipeline optimization have demonstrated significant performance improvements in distributed systems [4–6]. Additionally, recent research has explored the use of artificial intelligence for optimizing ETL pipelines and real-time healthcare data processing [7–11].

Despite these advances, limited research has investigated the application of reinforcement learning for adaptive synchronization of biomedical signal streams. This paper addresses this gap by proposing an RL-driven ETL orchestration framework designed for real-time biomedical data environments.

The main objective of this work is to design and evaluate an adaptive synchronization mechanism capable of improving data freshness and reducing synchronization latency in biomedical monitoring systems.

2. Adaptive ETL Architecture for Biomedical Data Streams

The proposed architecture integrates reinforcement learning into the orchestration layer of a biomedical ETL pipeline. The system consists of the following main components:

1. Monitoring and telemetry module;
2. Reinforcement learning controller;
3. Change-data-capture extraction module;
4. Transformation and loading component;
5. Analytical database repository.

The monitoring module continuously collects system telemetry, including synchronization latency, backlog size, throughput, and CPU utilization. These metrics provide the state representation used by the reinforcement learning agent.

The reinforcement learning controller observes the current system state and selects synchronization parameters, including synchronization interval and batch size. These parameters determine how frequently the ETL process executes and how many records are processed during each cycle.

The system integrates ICU monitoring devices, a CDC-based data ingestion layer, an ETL orchestrator, and an analytical database. A monitoring and telemetry module continuously collects system-level metrics, including synchronization latency, backlog size, throughput, and resource utilization, and provides this information to a reinforcement learning controller.

The reinforcement learning controller dynamically adjusts synchronization parameters, such as interval and batch size, based on the observed system state. This enables adaptive control of the ETL process in response to changing workload conditions.

The overall architecture of the proposed system is illustrated in Figure 1.

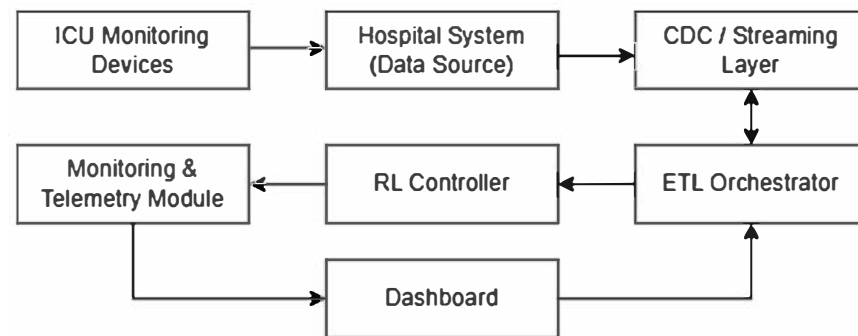


Figure 1. Reinforcement learning-driven adaptive ETL architecture for real-time biomedical signal synchronization.

The data extraction layer employs change-data-capture techniques to retrieve only newly generated or modified records from the source systems, thereby reducing unnecessary data transfers and improving synchronization efficiency [2]. During the transformation stage, incoming biomedical records are standardized and validated before being loaded into the analytical database.

The loading process uses idempotent UPSERT operations to ensure data consistency and prevent duplication during synchronization retries. The analytical repository, implemented using PostgreSQL, provides transactional consistency and reliable storage for biomedical data.

Overall, the architecture forms a closed-loop control system in which telemetry data continuously informs synchronization decisions, enabling efficient, adaptive, and low-latency data integration.

3. Reinforcement Learning Formulation

The synchronization problem is modeled as a Markov Decision Process (MDP), following the standard reinforcement learning formulation [12]. At each decision step, the reinforcement learning agent observes the system state and selects a synchronization strategy.

3.1. State Representation

The system state consists of several telemetry metrics:

- Synchronization latency;
- Data backlog size;
- Data throughput rate;
- CPU utilization;
- Memory utilization.

These variables capture both pipeline performance and computational resource consumption.

3.2. Action Space

The action space defines the synchronization parameters controlled by the agent. Each action corresponds to a pair:

- Synchronization interval;
- Synchronization batch size;

These parameters determine the frequency and size of ETL synchronization operations.

3.3. Reward Function

The reward function balances multiple objectives including latency reduction, data freshness, and resource efficiency. It can be expressed as:

$$R = -\alpha \cdot \text{Latency} - \beta \cdot \text{Backlog} - \gamma \cdot \text{CPU} - \delta \cdot \text{ConsistencyError}$$

where α , β , γ , and δ are weighting coefficients.

The weighting coefficients (α , β , γ , δ) were selected empirically to balance competing system objectives. In real-time biomedical monitoring scenarios, synchronization latency and backlog size have a direct impact on data freshness and clinical usability; therefore, higher weights were assigned to latency (α) and backlog (β).

CPU utilization (γ) was assigned a moderate weight to prevent excessive resource consumption without compromising responsiveness. The consistency error term (δ) was given a high penalty to ensure data integrity, as duplicate or missing records may negatively impact downstream clinical analytics.

A sensitivity analysis was conducted by varying the coefficients within predefined ranges. The results indicated that prioritizing latency and backlog reduction leads to the most stable and clinically relevant performance, while overly penalizing CPU usage resulted in delayed synchronization under high workloads.

This formulation encourages synchronization strategies that minimize latency and backlog while maintaining efficient resource usage.

3.4. Learning Algorithm

Deep Q-Learning (DQN) is used as the baseline reinforcement learning algorithm [13]. DQN has been widely used in adaptive decision-making problems due to its ability to learn optimal policies from high-dimensional state spaces.

Alternative algorithms such as Proximal Policy Optimization (PPO) [14] or Actor-Critic methods [15] could also be applied for continuous action spaces.

4. Experimental Setup

To evaluate the proposed approach, experiments were conducted using simulated biomedical data streams representing intensive care monitoring scenarios.

The experimental environment consisted of a Linux server running PostgreSQL 15 and a reinforcement learning implementation developed in Python 3.11, using standard scientific computing libraries. Biomedical data streams were simulated using ICU-style datasets containing vital signs and physiological measurements. Data arrival patterns were designed to replicate real-world monitoring workloads, including bursty updates and variable data generation rates.

- Two baseline synchronization strategies were implemented for comparison:
- Fixed-interval synchronization (periodic scheduling);
- Threshold-based synchronization triggered by backlog size.

The following performance metrics were evaluated:

- Mean synchronization latency;
- Data freshness ratio;
- CPU utilization;
- Consistency errors;
- Convergence behavior of the reinforcement learning policy.

Each experiment simulated several hours of continuous data streaming to ensure stable evaluation results.

5. Results and Discussion

The reinforcement learning-based synchronization framework demonstrated improved performance compared with static scheduling strategies.

The adaptive controller dynamically adjusted synchronization intervals in response to workload changes. During high data arrival periods, the agent reduced synchronization intervals to minimize backlog growth and maintain low latency. During lower workloads, synchronization intervals increased, reducing unnecessary computational overhead.

Compared with fixed scheduling strategies, the RL-based approach achieved the following improvements:

- Reduced average synchronization latency;
- Increased data freshness;
- Lower variance in CPU utilization;
- Stable convergence after a short learning period.

These results align with previous studies that demonstrate the effectiveness of reinforcement learning for pipeline optimization and resource scheduling [5,7,9].

In biomedical monitoring systems, improved synchronization latency directly translates into more timely clinical data availability, which may enhance the performance of predictive models and decision-support tools.

Table 1 summarizes the performance of the evaluated synchronization strategies. The reinforcement learning-based approach achieves the lowest average latency, reducing it by approximately 34% compared to the threshold-based strategy and over 50% compared to fixed scheduling.

Table 1. Performance comparison of synchronization strategies.

Strategy	Avg. Latency (ms)	Data Freshness (%)	CPU Utilization (%)
Fixed Interval	850	78	65
Threshold-Based	620	85	72
RL-Based (Proposed)	410	93	60

In addition, the proposed method improves data freshness, reaching over 90%, which indicates that analytical views are updated more frequently with recent data. CPU utilization is also more stable and slightly reduced compared to heuristic approaches, demonstrating improved resource efficiency.

These results confirm that adaptive synchronization driven by reinforcement learning provides a better trade-off between latency, freshness, and computational cost.

6. Conclusions

This study presented a reinforcement learning-driven adaptive ETL orchestration framework for real-time biomedical signal synchronization. The proposed approach integrates telemetry monitoring, CDC-based incremental extraction, and idempotent database loading within a closed-loop control architecture.

Experimental results indicate that reinforcement learning can significantly improve synchronization efficiency compared with static ETL scheduling strategies. The adaptive controller reduces latency and improves data freshness while maintaining efficient resource utilization.

These findings suggest that intelligent orchestration mechanisms can play a key role in next-generation biomedical data infrastructures, particularly in environments requiring real-time data integration such as intensive care monitoring systems.

Future research will focus on extending the framework to distributed data processing architectures and exploring multi-agent reinforcement learning approaches for large-scale healthcare data environments.

The main contributions of this paper are: (1) the design of a reinforcement learning-based adaptive ETL orchestration framework for biomedical data streams, (2) the formulation of synchronization as a Markov Decision Process, and (3) an experimental evaluation demonstrating improved latency and data freshness compared to traditional scheduling strategies.

Author Contributions: Conceptualization, B.N.B. and C.C.; methodology, B.N.B.; software, B.N.B.; validation, B.N.B., C.C., G.H. and Z.S.; formal analysis, B.N.B.; investigation, B.N.B.; resources, G.H. and Z.S.; data curation, B.N.B.; writing—original draft preparation, B.N.B.; writing—review and editing, C.C., G.H. and Z.S.; visualization, B.N.B.; supervision, C.C.; project administration, C.C.; funding acquisition, C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No real patient data were used in this study. The experimental evaluation was conducted using simulated ICU-style biomedical data streams generated for the purpose of evaluating the proposed framework; no publicly archived dataset is associated with this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Salehi, S.S.; Saadatfar, H.; Oyelere, S.S.; Hussain, S.; Joloudari, J.H.; Ledari, M.T.; Arslan, E.; Barzegar, B. Enhancing healthcare outcome with scalable processing and predictive analytics via cloud healthcare API. *Front. Digit. Health* **2026**, *7*, 1687131. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Amaresam, R.K. AI-Enhanced Cloud ETL for Real-Time Health Data Processing: Transforming Healthcare through Intelligent Data Management. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2025**, *11*, 1761–1781. [\[CrossRef\]](#)
3. Urblik, L.; Kajati, E.; Papcun, P.; Zolotova, I. A Modular Framework for Data Processing at the Edge: Design and Implementation. *Sensors* **2023**, *23*, 7662. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Pedratscher, S.; Samani, Z.N.; Poveda, J.A.; Fahringer, T.; Etheredge, M.; Younesi, A.; Barrionuevo, J.J.D.; Thoman, P. STREAMLINE: Dynamic and Resource-Efficient Auto-Tuning of Stream Processing Data Pipeline Ensembles. *Internet Things* **2025**, *34*, 101731. [\[CrossRef\]](#)
5. Li, T.; Xu, Z.; Tang, J.; Wang, Y. Model-Free Control for Distributed Stream Data Processing Using Deep Reinforcement Learning. *arXiv* **2018**, arXiv:1803.01016.
6. Ni, X.; Li, J.; Yu, M.; Zhou, W.; Wu, K.-L. Generalizable Resource Allocation in Stream Processing via Deep Reinforcement Learning. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 857–864. [\[CrossRef\]](#)
7. Chirumamilla, K.R. Reinforcement Learning to Optimize ETL Pipelines. *Eastasouth J. Inf. Syst. Comput. Sci.* **2023**, *1*, 171–183. [\[CrossRef\]](#)
8. Vemulapalli, V.K.C. Adaptive Reinforcement Learning Framework for Enterprise Data Integration in LLM Training. *Sarcouncil J. Eng. Comput. Sci.* **2025**, *4*, 90–109.
9. Annam, D. AI-powered real-time data pipeline optimization using deep reinforcement learning. *World J. Adv. Res. Rev.* **2025**, *26*, 2647–2653. [\[CrossRef\]](#)
10. Kumar G., S.S.; Kumar M., R. Towards Autonomous Data Transformation: A Hybrid Deep Reinforcement Learning and Transformer Framework for ETL Automation. *Int. J. Intell. Eng. Syst.* **2025**, *18*, 241–263. [\[CrossRef\]](#)
11. Nayak, U. AI-Powered Data Pipelines: Leveraging Machine Learning for ETL Optimization. *J. Softw. Eng. Simul.* **2025**, *11*, 134–136. [\[CrossRef\]](#)
12. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2018.
13. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [\[CrossRef\]](#) [\[PubMed\]](#)

14. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal Policy Optimization Algorithms. *arXiv* **2017**, arXiv:1707.06347.
15. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous Control with Deep Reinforcement Learning. *arXiv* **2015**, arXiv:1509.02971.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.