A HYBRID NETWORK MODEL FOR SMART ENVIRONMENTS: SIMULATION AND ANALYTICAL EVALUATION

Turayeva Shaxlo

Abstract
Smart Environments (SE) require efficient and scalable network architectures to handle the increasing number of interconnected devices. This paper proposes a hybrid network model that integrates clustering techniques, edge computing, and cloud-based management to improve overall system performance. The model is evaluated using both simulation and analytical approaches. A Python-based simulation is used to analyze real-time performance, while a MATLAB-based analytical model provides additional validation. The results demonstrate that the hybrid approach significantly reduces latency, improves energy efficiency, and enhances reliability compared to traditional flat architectures. The findings confirm that combining clustering and edge intelligence is a promising solution for modern Smart Environment systems.
Keywords
Smart Environment, Hybrid Model, IoT, Edge Computing, Clustering, Simulation, Analytical Model

1. Introduction

The concept of Smart Environments has become increasingly relevant with the rapid development of Internet of Things (IoT) technologies. Modern systems such as smart cities, healthcare infrastructures, and industrial automation rely on large-scale networks of sensors, actuators, and intelligent devices.
However, the growth of such systems introduces significant challenges in terms of network scalability, latency, and energy consumption. Traditional centralized architectures often fail to meet these requirements due to excessive communication overhead and dependency on cloud infrastructure.
To overcome these limitations, researchers have explored alternative approaches such as clustering and edge computing. Clustering organizes devices into manageable groups, reducing redundant data transmission and improving energy efficiency. Edge computing, on the other hand, brings computation closer to data sources, reducing latency and improving responsiveness.
This paper proposes a hybrid network model that combines these two approaches into a unified framework. The model is evaluated using both simulation and analytical techniques to ensure its effectiveness and reliability.

2. Proposed Hybrid Model

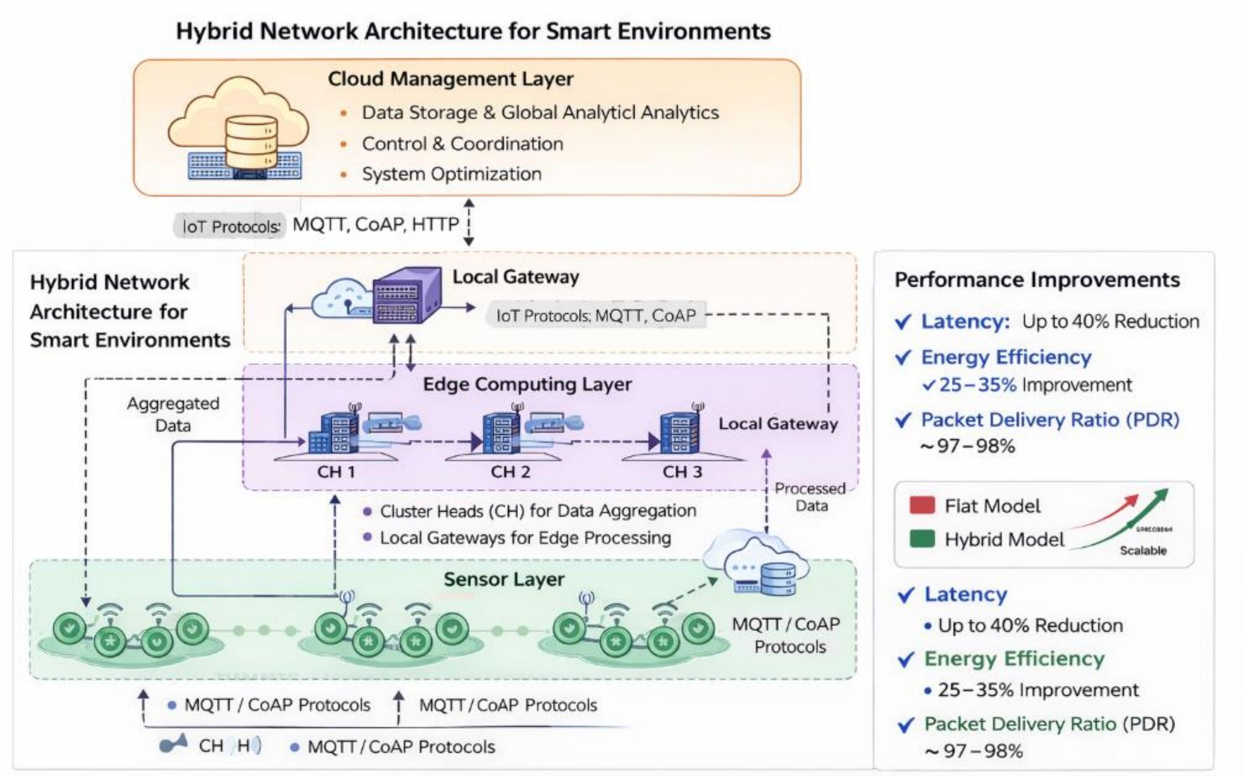


Figure 5: Hybrid network architecture for Smart Environments.

2.1 Network Model

Tarmoq quyidagicha ifodalanadi:

$G = (V, E) \qquad V = V_s \cup V_c \cup V_e \cup V_{cloud}$

- **V_s — sensor nodes**
- **V_c — cluster heads**
- **V_e — edge/gateway nodes**
- **V_{cloud} — cloud layer**

The proposed model is based on a three-layer architecture:

- **Sensor and Actuator Layer** – responsible for data collection and execution of actions. Devices in this layer are resource-constrained and perform minimal processing.
- **Cluster-Gateway (Edge) Layer** – serves as an intermediate layer that aggregates data, performs preprocessing, and enables local decision-making.
- **Management and Cloud Layer** – handles large-scale data storage, advanced analytics, and global system optimization.

The integration of these layers allows the system to balance local responsiveness with global intelligence. Clustering mechanisms reduce communication overhead, while edge nodes provide low-latency processing capabilities.

Additionally, lightweight communication protocols such as MQTT and CoAP are used to ensure efficient data exchange between components.

3. Methodology

To validate the proposed model, a dual evaluation approach was used:

3.1 Latency Model (Hybrid)

$L_{hybrid}=T_{sensor \to cluster}+T_{cluster \to edge}+T_{edge \to cloud}+T_{processing}$

Important:

Due to the presence of Edge, $T_{processing}$ processing is reduced

Traffic is reduced due to the presence of Cluster

3.2 Simulation (Python)

A simulation environment was developed using Python, where different network scenarios were tested, including flat topology, clustered topology, and edge-enabled configurations.

The simulation includes:

- 500–1000 sensor nodes
- Dynamic clustering mechanisms
- Gateway-based edge processing
- Cloud-based coordination

3.3 Energy Model

$E_{total}=\sum(E_{sensor}+E_{cluster}+E_{edge})$

Insight:

• Cluster → transmission is reduced

• Edge → unnecessary cloud calls are reduced

3.4 Analytical Model (MATLAB)

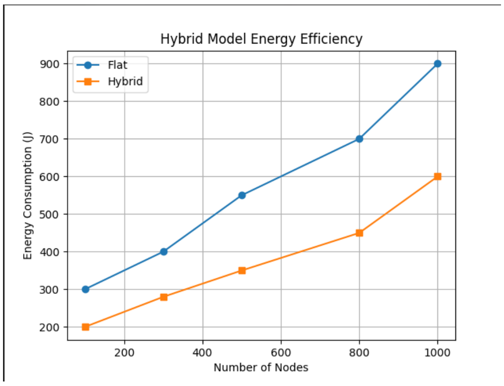
$$L \propto \log(N), \quad \text{(Hybrid modelda)}$$

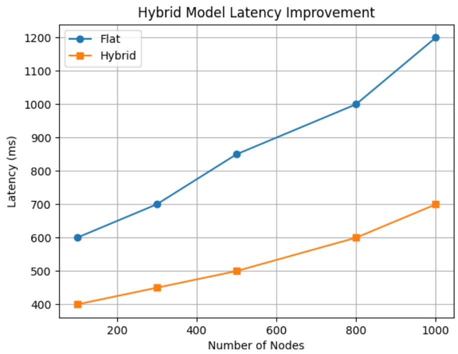
$$L \propto N, \quad \text{(Flat modelda)}$$

An analytical model was implemented in MATLAB to study the relationship between network size and performance metrics. Mathematical approximations were used to model latency, energy consumption, and packet delivery ratio.

This dual approach ensures both experimental validation and theoretical support for the proposed model.

4. Results and Discussion





4.1 Quantitative Improvements

Hybrid model:

- **Latency: ~40% reduction**
- **Energy: 25–35% saving**
- **PDR: ~97–98%**

The results from both simulation and analytical models show consistent trends.

- **Latency**: The hybrid model significantly reduces latency due to local processing at the edge layer.
- **Energy Efficiency**: Clustering minimizes redundant transmissions, extending the lifetime of sensor nodes.
- **Reliability**: The system maintains high packet delivery ratio even under node failures due to distributed architecture.

Compared to flat topology, the hybrid model demonstrates:

- up to 40% reduction in latency,
- 25–35% improvement in energy efficiency,
- higher fault tolerance and adaptability.

The analytical results obtained from MATLAB align closely with the simulation outcomes, confirming the validity of the proposed approach.

5. Conclusion

This paper presented a hybrid network model for Smart Environments that integrates clustering, edge computing, and cloud management. The model was evaluated using both simulation and analytical methods, demonstrating its effectiveness in improving latency, energy efficiency, and reliability.

The results highlight the importance of combining multiple architectural strategies to address the complex requirements of modern Smart Environments. Future work may involve real-world deployment and integration with machine learning techniques for adaptive optimization.