Edge-Cloud Collaborative Framework for Smart IoT Applications

Supervisor:

Dr. Praveen Kumar Donta

Prepared by:

Safir Jameel Manghat
sama9668@student.su.se

Department of Computer and Systems Sciences

Master's Thesis 30hp

Computer and Systems Sciences

Stockholm University

VT2025

Date: 04/02/2025
Version: v1.1

Table of Contents

Abstract—Over the last few years, numerous research studies have been conducted to examine the energy consumption for large amounts of delay-sensitive applications that brings serious challenges with the continuous development and diversity of Industrial Internet of Things (IIoT) applications in fog networks. In addition, conventional cloud technology cannot adhere to the delay requirement of sensitive IIoT applications due to long-distance data travel. To address this bottleneck, (Hazra et al., 2023) designed a novel energy–delay optimization framework called transmission scheduling and computation offloading (TSCO), while maintaining energy and delay constraints in the fog environment. To achieve this objective, a heuristic-based transmission scheduling strategy to transfer IIoT-generated tasks based on their importance. In Addition, researchers developed a graph-based task-offloading strategy using constrained-restricted mixed linear programming to handle high traffic in rush-hour scenarios. (Hazra et al., 2023) demonstrated extensive simulation results illustrate that the proposed TSCO approach significantly optimizes energy consumption and delay up to 12%–17% during computation and communication over the traditional baseline algorithms.

The goal of TSCO is to cooperatively schedule transmission and offload computation data through a hierarchical fog network. This involves determining an efficient transmission scheduling strategy for delay-sensitive IIoT applications and defining an optimal task-device-matching strategy in the fog networks.

TSCO aims to design a "green" IIoT system by handling various emergency tasks in the fog environment. By using the TSCO strategy, an even better strategy of fog association, transmission scheduling, and computation offloading not only minimizes the energy consumption rate but also increases the QoS for delay-critical IIoT applications.

Keywords: Energy efficiency, fog computing, Industrial Internet of Things (IIoT), mixed linear programming, task offloading, Systematic Literature Review (SLR), Transmission Scheduling and Computation Offloading (TSCO), Design Science Research (DSR)

1 Introduction

This chapter introduces the area of IIOT along with the key factors influencing their performance, such as latency, energy consumption and scalability. This paper primarily focuses on developing an intelligent framework that enhances latency reduction and energy consumption in IIOT applications. By leveraging a hierarchical fog-cloud computing model, the proposed framework aims to optimize task allocation and minimize energy consumption. Through a comprehensive evaluation this research aims to contribute to the development of efficient, scalable IIOT systems, addressing critical challenges in modern industrial automation and smart environments.

The Internet of Things (IoT) has rapidly transformed various industries, driving advancements in smart cities, healthcare, transportation, and industrial automation. IoT refers to a network of interconnected devices that collect, transmit, and process data to enable real-time decision-making. Among the various segments of IoT, the Industrial Internet of Things (IIoT) has gained significant attention due to its potential to revolutionize manufacturing, supply chain management, and industrial processes. IIoT extends the benefits of IoT to industrial applications, enabling automation, predictive maintenance, and improved operational efficiency. The demand for IIoT solutions is driven by the need for increased productivity, reduced downtime, and optimized resource utilization in industries such as manufacturing, energy, and logistics.

The increasing complexity of IoT applications has highlighted the need for intelligent solutions that would address many challenges hinder its widespread adoption. Key challenges include high latency in data processing, energy constraints in connected industrial devices, security vulnerabilities, and interoperability issues among heterogeneous systems. Traditional cloud-based architecture struggles to handle the massive volume of data generated by IIoT devices, leading to increased network congestion and delays in decision-making. To address these challenges, researchers have been exploring edge and fog computing solutions that distribute computational tasks closer to data sources, thereby enhancing real-time processing capabilities.

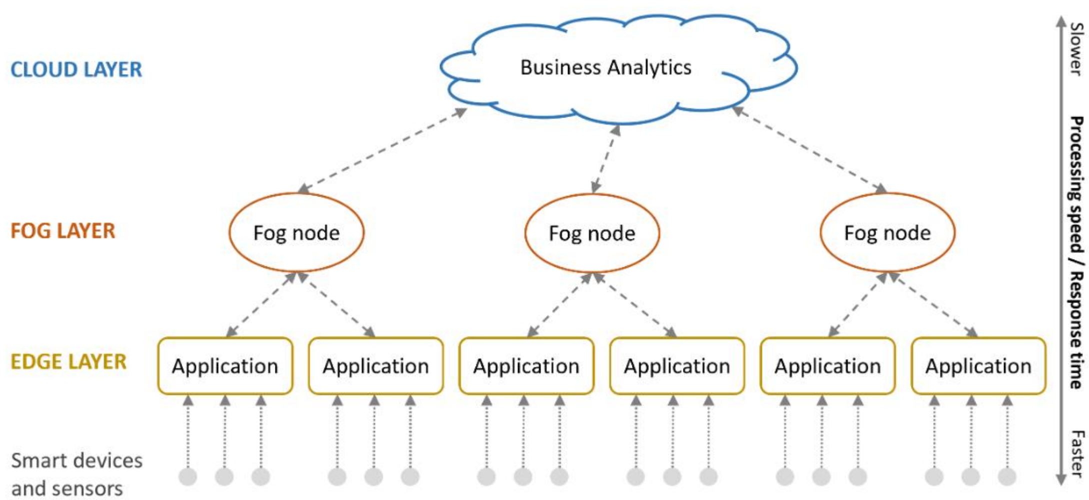

Fig. 1 Edge-Cloud hierarchy and its processing/response time

In the context of smart IoT, data processing and decision making are distributed across multiple layers of a hierarchical structure shown in Fig. 1. This hierarchy typically consists of IoT devices, edge nodes, fog nodes and cloud servers. Each layer plays a critical role in processing data and reducing response time which is critical for real-time IoT applications.

1.1 Background

The traditional cloud-centric IIoT model involves transmitting all sensor data to cloud servers for processing, storage, and analysis. While this model works well for applications that are not time-sensitive, it is impractical for real-time industrial applications such as predictive maintenance, robotics, and smart grid management (Badidi, E. 2019). The delay caused by the physical distance between IIoT devices and cloud servers can be detrimental in scenarios where immediate responses are required. In Addition, the continuous transmission of large volumes of data places a strain on network bandwidth and increases energy consumption.

A more efficient alternative is a hierarchical fog-cloud environment. In this model, fog computing provides intermediate processing between the IIoT devices and the cloud. Fog nodes, which can be routers, gateways, dedicated edge devices, process latency-sensitive tasks near the data source, while computationally intensive tasks are offloaded to the cloud. This hybrid approach minimizes latency, reduces energy consumption, and optimizes network resource utilization.

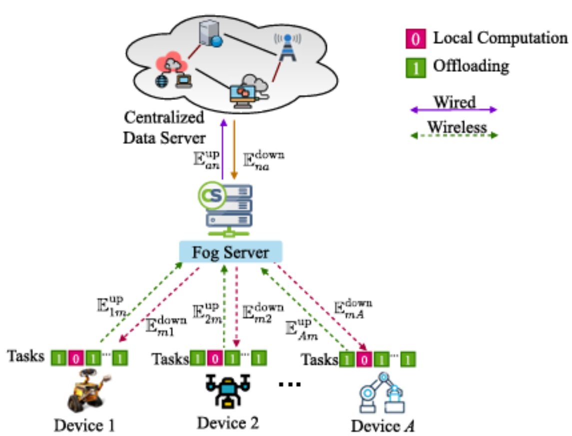

Fig 2. Hierarchical IIOT-fog-cloud architecture (Hazra et al. (2023))

Several key concepts are central to understanding the development of an intelligent edge-cloud collaborative framework:

Computation Offloading: This technique involves transferring processing tasks from resource-limited IIoT devices to more powerful computing entities, such as fog nodes or cloud servers (**Wang et al. 2019**). Offloading reduces the energy consumption of IIoT devices and enhances processing efficiency. Fig. 2 shows a hierarchical fog-cloud architecture where task scheduling and computational offloading have been carried out.

Transmission Scheduling: Prioritizing data transmission from IIoT devices based on factors such as network conditions and data importance helps manage network congestion and ensures timely delivery of critical information (Hazra et al. 2023).

Task-Device Matching: Efficiently allocating tasks to the most suitable fog nodes or cloud servers optimizes resource utilization and enhances system performance (Avasalcai et al. 2021).

Fog Nodes: These are intermediate computing devices that provide localized processing and storage, bridging the gap between IIoT devices and cloud servers (Badidi, E. 2019).

1.2 Literature Review

To better understand the current state of research gap, a Systematic Literature Review was

conducted, The SLR analyzed and compares article published between 2015 and 2024, focusing on existing framework for for-cloud task offloading and highlights the advancement in (Hazra et al. 2023). While prior work addresses specific challenges, for instance latency, overhead and energy-delay trade-offs, Hazra's unified framework integrates into a scalable and efficient solution.

To compare TSCO framework with existing frameworks for task offloading and computation resource allocation, few comparison questions are formulated:

How do existing framework address latency and computation overhead?

What are the strengths and weaknesses of each framework?

How does TSCO framework improve upon prior work?

Existing frameworks primarily focus on either partial offloading or binary offloading, where tasks are either fully executed locally or offloaded to fog/cloud devices. While fog-based offloading reduces latency, it often leads to higher energy consumption due to limited computational resources at the edge. Conversely, cloud-based offloading minimizes energy consumption but introduces significant latency due to long-distance data transmission (Hazra et al., 2023; Sarkar et al., 2022).

Task prioritization is critical for delay-sensitive IIoT applications. However, most existing scheduling algorithms, such as priority-based and deadline-aware strategies, fail to account for dynamic network conditions, leading to suboptimal performance in high-traffic scenarios (Li et al., 2018; Wang et al., 2020).

Joint optimization of energy consumption and latency remains a significant challenge. While some frameworks propose energy-efficient offloading strategies, they often neglect latency constraints, and vice versa. This trade-off is particularly pronounced in heterogeneous fog-cloud environments, where resource availability varies dynamically (Yadav et al., 2020).

Efficient device-matching strategies are essential for optimizing resource utilization in fog-cloud environments. However, existing approaches often lack adaptability to dynamic network conditions, resulting in inefficient task allocation and resource underutilization (Hazra et al., 2023).

The Literature review highlights several gaps in the existing literature:

Lack of Holistic Frameworks: Most approaches focus on either energy efficiency or latency optimization, but not both simultaneously.

Dynamic Adaptation: Limited research on frameworks that adapt to dynamic network conditions and varying traffic loads.

Task Prioritization: Insufficient mechanisms for prioritizing tasks based on their importance and deadlines.

Scalability: Few frameworks address the scalability challenges of large-scale IIoT deployments.

Real-World Validation: Most studies rely on simulations, with limited real-world validation of proposed frameworks.

These gaps underscore the need for a holistic framework that jointly optimizes energy consumption and latency while addressing the limitations of existing approaches. The proposed Cooperative Transmission Scheduling and Computation Offloading (TSCO) framework aims to fill this gap by introducing innovative strategies for task prioritization, transmission scheduling, and computation offloading in fog-cloud environments.

1.3 Problem Statement

Despite the advancements in fog and cloud computing, several challenges remain unaddressed in the context of IIoT applications:

Existing frameworks often optimize either energy consumption or latency, but not both simultaneously. This leads to suboptimal performance in industrial environments where both metrics are critical.

Current approaches lack efficient mechanisms to prioritize tasks based on their importance and deadlines, resulting in increased processing delays for critical applications.

The absence of an optimal device-matching strategy leads to inefficient resource utilization, as tasks are not always assigned to the most suitable computing devices (fog or cloud).

Industrial environments are characterized by dynamic network conditions, such as varying traffic loads and resource availability. Existing frameworks often fail to adapt to these changes, leading to performance degradation.

These challenges highlight the need for a holistic framework that jointly optimizes energy consumption and latency while addressing the limitations of existing approaches. The proposed TSCO framework aims to fill this gap by introducing innovative strategies for task prioritization, transmission scheduling, and computation offloading in fog-cloud environments.

1.4 Research Question

While existing research has made significant advancements in edge-cloud collaboration, several gaps remain that require further investigation:

Latency Optimization: Many existing frameworks focus on reducing latency but do not provide comprehensive mechanisms for dynamically adjusting based on real-time network conditions.

Although computation offloading reduces device energy consumption, optimal strategies for balancing workload distribution between fog nodes and the cloud remain underexplored.

Ensuring secure data transmission and processing within a heterogeneous fog-cloud environment presents a significant challenge.

The lack of universally accepted standards hinders seamless integration and deployment across various IIoT ecosystems.

Based on these gaps, the following research questions emerge:

"**How *can a Cooperative Task Scheduling and Computation Offloading (TSCO) framework optimize energy consumption in IIoT systems by effectively leveraging the collaboration between fog and cloud resources?*** *"*

1.5 Delimitations

This research focuses on developing an AI-driven edge-cloud collaborative framework for IoT applications that require real-time processing and energy-efficient task scheduling. The study is limited to synthetic data simulations and evaluates the framework based on average delay, total energy consumption, and task completion rate. The framework is compared with Random Computation Offloading (RCO) and Priority-Based Computation Offloading (PBCO). Other aspects of IoT systems, such as hardware design, network protocols, and security, are beyond the scope of this study. Additionally, the research does not address region-specific or time-bound IoT applications, nor does it consider dynamic scaling of resources

2 Extended background

This chapter aims to provide an extended theoretical background to the study and introduce related research. The first sections present the foundational concepts of edge-cloud computing, fog networks, and the challenges of energy consumption and latency in Industrial Internet of Things (IIoT) applications. Based on a review of existing literature, an overview of current knowledge in the field is provided, highlighting the limitations of existing frameworks such as Random Computation Offloading (RCO) and Priority-Based Computation Offloading (PBCO). Finally, the chapter discusses the machine learning algorithms and mathematical modeling techniques used in the study to optimize task scheduling and computation offloading in the proposed Transmission Scheduling and Computation Offloading (TSCO) framework.

2.1 Overview: Introduce IoT and edge-cloud computing, focusing on IIoT applications.
2.2 Challenges: Discuss energy consumption, latency, and task scheduling in IIoT systems.
2.3 Existing Solutions: Review RCO, PBCO, DDPT, and EECO, highlighting their limitations.
2.4 Need for TSCO: Explain how TSCO addresses these limitations by optimizing energy and delay.
2.5 Theoretical Foundations: Describe the IBTS and DMO strategies used in TSCO.
2.6 Mathematical Modeling: Explain how TSCO uses mathematical modeling to optimize task allocation.
2.7 Performance: Highlight the advantages of TSCO over existing algorithms.
2.8 Relevance: Emphasize TSCO's contributions to edge-cloud computing and related fields.
2.9 Green IIoT: Discuss how TSCO promotes energy efficiency and sustainability in IIoT systems.
2.10

3 Methodology

In this chapter, a detailed discussion is presented in the stages of DSR development for producing the artefact in the form of TSCO framework. The results provided in this chapter show the most suitable research strategies and the knowledge base needed during the development of artefact in every stage in Fig. 3, starting from state of explicated problem, defining requirements, designing and developing the artefact, demonstrating it and evaluating it.

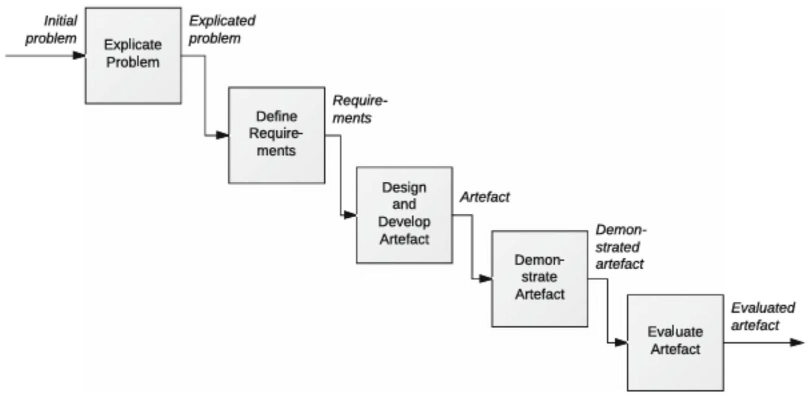

Fig. 3. The stages for conducting design science research using Johannesson & Perjons [9]

The TSCO (Transmission Scheduling and Computation Offloading) framework is designed to address energy consumption challenges in delay-sensitive Industrial Internet of Things (IIoT) applications within fog networks. The framework is developed using the Design Science Research Methodology (DSR), which consists of five key activities: Explicate Problem, Define Requirements, Design and Develop Artefact, Demonstrate Artefact, and Evaluate Artefact.

3.1 The stage of explicate problem

The core problem identified in this step is the high energy consumption associated with delay-sensitive IIoT applications in fog networks. This issue is particularly critical in scenarios involving emergency tasks, where efficient resource utilization and timely processing are essential. The significance of this problem lies in the need to balance energy efficiency and low-latency requirements in fog environments.

3.2 Define Requirements

This step involves identifying the type of artifacts needed and eliciting both functional and non-functional requirements. Functional requirements may include efficient task offloading and transmission scheduling, while non-functional requirements could focus on scalability, reliability, and energy efficiency.

3.3 **Design** and Development

Based on the defined requirements, the TSCO strategy is developed. This involves:

Designing a transmission scheduling policy that considers factors such as data size and transmission rate.

Developing a device-matching strategy to ensure efficient task offloading between fog nodes and IIoT devices.

The goal is to create a framework that optimizes both energy consumption and delay for IIoT applications.

3.4 Demonstrate Artefact

In this stage, the TSCO framework is simulated to evaluate its performance. Key performance metrics are analyzed to demonstrate how TSCO enhances the energy-delay tradeoff compared to existing frameworks. Metrics such as energy consumption, latency, and task completion time are used to showcase the framework's effectiveness.

3.5 Evaluate Artefact

The final step involves evaluating how well the TSCO framework meets the specified requirements. This includes assessing:

Processing time: The time taken to complete tasks.

Energy consumption: The total energy used by the system during task execution.

The evaluation provides insights into the framework's performance and identifies areas for further improvement.

4 References

--

[1] Hazra, A., Donta, P. K., Amgoth, T., & Dustdar, S. (2023). Cooperative Transmission Scheduling and Computation Offloading With Collaboration of Fog and Cloud for Industrial IoT Applications. *IEEE Internet of Things Journal*. *Doi:10.1109/JIOT.2022.3150070*

[2] Sarkar, I., Adhikari, M., Kumar, N., & Kumar, S. (2022). Dynamic Task Placement for Deadline-Aware IoT Applications in Federated Fog Networks. *IEEE Internet of Things Journal*. Doi: 10.1109/JIOT.2021.3088227.

[3] Li, G., Wu, J., Li, L., Wang, K., & Ye, T. (2018). Service Popularity-Based Smart Resources Partitioning for Fog Computing-Enabled Industrial Internet of Things. *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2018.2845844.

[4] Wang, J., Liu, K., Li, B., Liu, T., Li, R., & Han, Z. (2020). Delay-Sensitive Multi-Period Computation Offloading With Reliability Guarantees in Fog Networks. *IEEE Transactions on Mobile Computing*. Doi: 10.1109/TMC.2019.2918773.

[5] Yadav, R., Zhang, W., Kaiwartya, O., Song, H., & Yu, S. (2020). Energy-Latency Tradeoff for Dynamic Computation Offloading in Vehicular Fog Computing. *IEEE Transactions on Vehicular Technology*. DOI:10.1109/TVT.2020.3040596

[6] Q. Wang, S. Guo, J. Liu, and Y. Yang, "Energy-efficient computation offloading and resource allocation for delay-sensitive mobile edge computing," Sustain. Comput. Informat. Syst., vol. 21, pp. 154–164, Mar. 2019.

[7] C. Avasalcai, B. Zarrin and S. Dustdar, "EdgeFlow—Developing and Deploying Latency-Sensitive IoT Edge Applications," in *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3877-3888, 1 March1, 2022, doi: 10.1109/JIOT.2021.3101449.

[8] Badidi, E. (2019). A Fog Node Architecture for Real-Time Processing of Urban IoT Data Streams. *Computer Science On-line Conference*. DOI:10.1007/978-3-030-19807-7_32

[9] Johannesson P, Perjons E. An introduction to design science vol. 10. Springer; 2014.