Designing Multi-Agent Frameworks for Optimized Scalability and Resource Allocation in Smart City IoT Systems

Santosh Lavate[1]
[1]Assistant Professor, AISSMS's College of Engineering, Pune, Maharashtra, India

Ravindra Moje[2]
[2]Assistant Proffesor, PDEA's College of Engineering, Manjari, Pune, Maharashtra, India

Shakir Khan[3]
[3]College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh Saudi Arabia.
University Centre for Research and Development, Chandigarh University, Mohali 140413, India;

shlavate@aissmscoe.com
lavate.santosh@gmail.com

ravindra.moje@gmail.com

sgkhancs@gmail.com
sgkhan@imamu.edu.sa

Abstract: It's getting harder and harder to manage scale and resource sharing in smart cities because of how quickly Internet of Things (IoT) devices are being added. As city systems add more sensors, motors, and communication nodes, it becomes more difficult to coordinate all of these parts so that they work together smoothly and use little energy. This paper suggests a new multi-agent approach that will help smart city IoT systems be more scalable and make better use of their resources. The framework creates a number of agents that can be used as monitors, resource managers, or service planners. Each agent has specific skills and a way of communicating that is designed to allow for decentralised decision-making. The framework uses agent-based discussion and coordination methods to flexibly distribute resources like data, computing power, and energy among competing IoT devices. This makes the system more fast and efficient as a whole. The suggested system solves important problems like managing different types of devices, changing network topologies on the fly, and meeting real-time scaling needs by letting independent agents work together. The execution is tested by running a lot of scenarios in a real smart city IoT situation. These simulations use datasets that show how urban environmental tracking and traffic control could be used. Compared to standard centralised methods, the results show big changes in how resources are used, how much contact is needed, and how easily the system can grow. The framework's flexible design also makes it easy to connect to smart city systems that are already in place and to adapt to changing IoT environments.

Keywords: Multi-Agent Systems, Smart City IoT, Resource Allocation, Scalability Optimization, Agent Coordination and Negotiation

I. Introduction

The fast growth of Internet of Things (IoT) technology has been a major force behind the change of towns into "smart cities." IoT devices like sensors, motors, cameras, and communication units that are all linked to each other are used by smart cities to improve services like managing traffic, keeping an eye on the environment, keeping people safe, managing energy, and controlling waste. These different kinds of devices constantly create and share huge amounts of data that help city officials make smart choices, make the best use of resources, and raise the quality of life for all people. But as the number of IoT devices in smart cities grows, it creates big problems with managing resources and being able to scale up. Due to limits in connection speed, computing resources, and the need to respond in real time, traditional centralised management methods often fail to meet these needs. When talking about smart city IoT systems, scalability means being able to handle more gadgets and data flows without slowing down the system. Resource allocation is the process of distributing limited resources, like network speed, computer power, storage space, and energy, among IoT devices in the most efficient way to meet the needs of different applications while also being environmentally friendly and cost-effective. These two things are closely linked: as the number of IoT devices increases, it becomes more important to use resources efficiently to avoid delays, crowding, and wasted energy. To get the most out of smart city IoT projects, it is important to create resource management systems that can be expanded and changed as needed. Because they are decentralised, independent, and friendly, multi-agent systems (MAS) have become a hopeful way to deal with these problems.

In MAS, many agents, which are software entities with defined jobs and the ability to make decisions, talk to each other and work together to reach shared or personal goals. This divided method works well with the way IoT networks in smart towns are built, which is to be spread. MAS can effectively handle complexity, dynamically change to changing conditions, and improve security against failures by giving agents specific tasks, such as watching sensors, collecting data, managing resources, and coordinating decisions [1]. This article suggests a complete multi-agent structure that is made to make smart city IoT systems more scalable and better at allocating resources. Figure 1 illustrates the multi-agent framework architecture optimizing smart city IoT resources. The framework describes different types of agents, such as sensor agents, resource manager agents, and supervisor agents. Each type of agent has its own set of communication methods and decision-making tools.

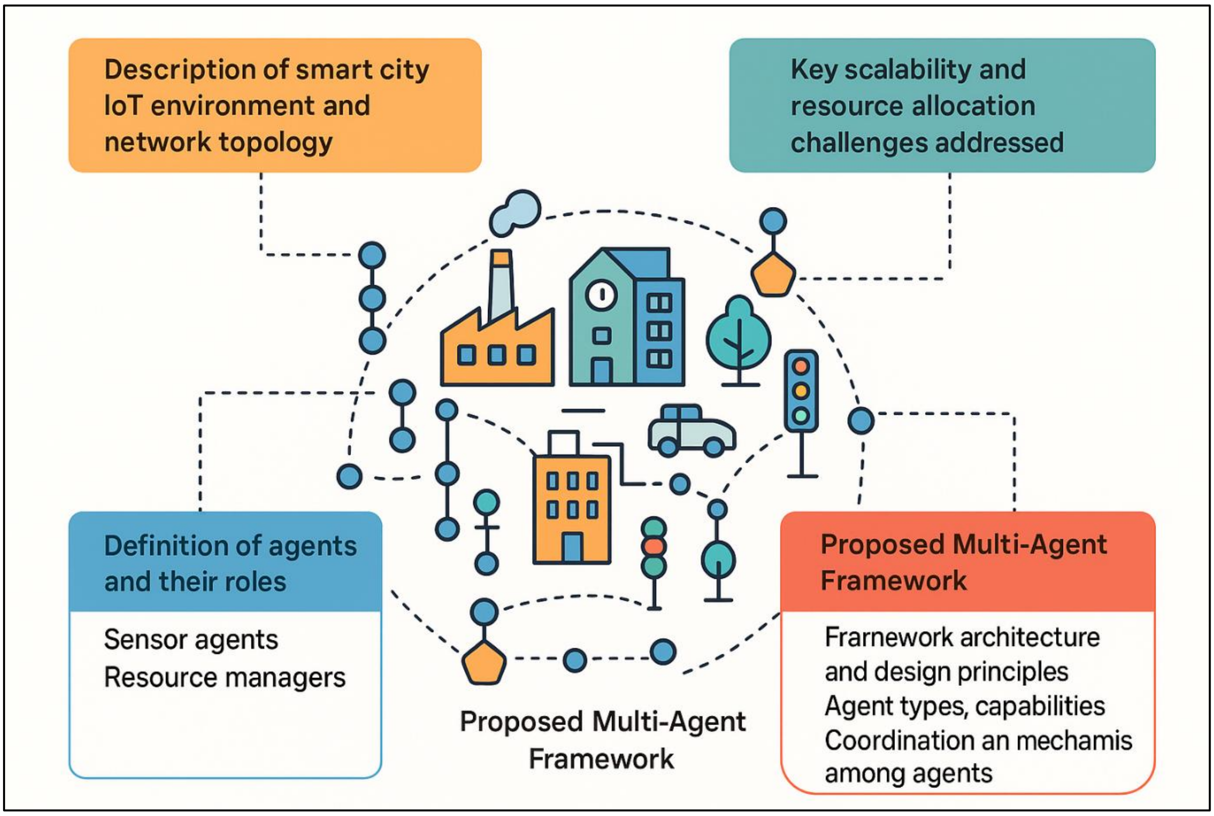

Figure 1: Multi-Agent Framework Architecture for Scalability and Resource Optimization in Smart City IoT Systems

The system can automatically assign resources, balance loads, and settle problems on its own thanks to methods for agent cooperation, discussion, and coordination. This ability to react is very important for dealing with changes that happen in real time, like changing network traffic, broken devices, and changing user needs. The framework is tested with simulations that look like real-life smart city uses, like tracking traffic and feeling the surroundings. These situations test how well the system can handle more IoT devices and different types of data needs while still using resources efficiently and keeping connection costs low [2]. The results show that the multi-agent model does a much better job of scaling, reaction time, and resource utilisation than standard centralised and rigid resource sharing methods. The suggested framework not only solves technical problems, but it also makes it easy to connect to current IoT systems and allows for future expansion as new devices and apps come out. This adaptability is very important because smart city ecosystems change quickly and different stakeholders have different needs [3].

II. Background and Related Work

The Internet of Things (IoT) is one of the most important technologies for smart cities because it lets all of a city's systems receive and handle data in real time. As smart city apps grow, they need to be able to handle huge networks of different IoT devices, each of which generates data at different rates, uses different amounts of energy, and has different connectivity needs. Managing these resources well while keeping the system's ability to grow is still a very big problem. Scalability problems are common in traditional centralised systems because they have limited computing power, slow networks, and single points of failure [4]. Because of these problems, people have been looking into decentralised and distributed methods. Multi-agent systems (MAS) stand out because they can handle settings that are complicated, big, and changing. MAS are made up of self-governing agents that can make decisions, talk to each other, and work together [5, 6]. This makes them perfect for handling IoT communities that are spread out. A number of studies have shown that MAS is useful in IoT and smart city settings. For instance, Olfati-Saber et al. introduced consensus methods for distributed coordination, which have changed how agents work together in IoT networks. Mahmoud et al. also came up with an agent-based resource management strategy for wireless sensor networks that can change how data and energy are used on the fly [7].

In the same way, Nguyen et al. created a MAS for controlling traffic lights that was more scalable and flexible in real time. A lot of new study is also being done on agent communication methods and agreement techniques to help IoT devices share resources more efficiently. Ren et al. for example, created a contract-net protocol update for smart grids that lets people share tasks more evenly [8]. Others have combined MAS with machine learning methods to make agents more flexible in settings that change, which makes the system more reliable and efficient. Even with these improvements, many current systems have trouble making large-scale IoT applications in smart cities that are both scalable and efficient at allocating resources. Table 1 summarizes methodologies, agent types, contributions, allocation approaches, limitations. This paper fills in that gap by suggesting a multi-agent framework that combines scalable agent coordination with smart resource allocation methods that work well in IoT ecosystems that are always changing in cities.

Table 1: Summary of Background and Related Work

Methodology	Agent Types	Key Contributions	Resource Allocation Approach	Limitations
Consensus algorithms	Homogeneous agents	Proposed consensus for distributed coordination	Distributed agreement on parameters	Limited to small networks
Agent-based scheduling [9]	Sensor & Resource Manager agents	Dynamic bandwidth and energy allocation	Adaptive resource scheduling	Energy models simplified
MAS + Reinforcement learning	Sensor, Controller agents	Improved scalability and real-time control	Real-time adaptive control	Limited to traffic domain
Contract-net protocol [10]	Resource Manager agents	Contract-net adaptation for distributed task allocation	Auction-based resource negotiation	Central coordination dependency

MAS + heuristic methods	Sensor and Coordinator agents	Energy-efficient data transmission	Energy-aware scheduling	Focused on energy only
Hierarchical MAS [11]	Multi-layer agent types	Layered architecture for improved scalability	Distributed resource allocation	Complexity increases with scale
MAS + Blockchain [12]	Security and Sensor agents	Secure decentralized coordination	Secure resource sharing	Overhead due to blockchain integration
MAS + QoS protocols [13]	Coordinator, Resource Manager agents	QoS-aware resource allocation	Priority-based resource scheduling	QoS parameters static

III. System Model and Problem Formulation

A. Description of smart city IoT environment and network topology

The internet of things (IoT) in a smart city is made up of many different devices that are all linked to each other and placed all over cities to keep an eye on and handle things like traffic, energy, and weather sense. There are cameras, communication hubs, sensors, motors, and edge computer units spread out across a hierarchical network structure. At the most basic level, sensor nodes gather real-time information about things like energy use, temperature, humidity, air quality, and traffic flow [14]. Local ports or edge servers talk to these sites and do basic data collection and processing to cut down on delay and network load. The edge servers then send important data to centralised cloud servers so that it can be stored for a long time, analysed in more detail, and used to make decisions. Wireless sensor networks (WSNs), cellphone networks, and fiber-optic backbones are often used together in a diverse and layered network structure to ensure strong communication [15]. Depending on the job and the state of the network, devices can talk to each other using either the peer-to-peer or client-server modes. This mixed structure allows for spread-out data collection and decision-making, which is important for smart city processes that can grow and be reliable [16]. The IoT world is always changing, with nodes being added and removed and network conditions being different all the time. This means that we need flexible systems that can keep communicating and allocating resources efficiently even when there are thousands or millions of devices.

B. Definition of agents and their roles (e.g., sensor agents, resource managers)

The suggested multi-agent framework describes various types of agents, each with a specific job to do in the smart city IoT environment.

- **Sensor Agents:** These are the actual IoT devices that collect data continuously and do some cleaning locally. They keep an eye on the environment and send information or tips to people in charge of resources.

- **Resource Manager Agents:** Make sure that important resources like internet, energy, and computing power are used efficiently and fairly. These bots look at the state of the network and the needs of each device to flexibly spread resources and keep energy use and bottlenecks to a minimum.
- **Coordinator Agents:** Coordinator agents act as go-betweens for sensor and resource manager agents, making it easier for them to negotiate and work together. They settle disagreements, make sure that everyone's work is done equally, and keep system-wide goals like speed and scale in check.
- **Gateway Agents:** Gateway Agents manage communication methods and data collection to improve scaling and lower network waste. They act as a link between sensor groups and higher-level infrastructure.

Each agent works on its own, but they all work together using set communication methods to make decisions. This lets the system react to changing situations and devices with different abilities.

C. Key scalability and resource allocation challenges addressed

Smart city IoT projects have a hard time with scaling and resource sharing because there are so many different kinds of devices and cities are always changing. First, as the number of devices grows, network congestion and transmission delays become big problems that slow down the system. The framework solves this problem by letting distributed agents work together, which moves decision-making closer to the action and lowers the burden on the central system. Second, battery-powered sensors have limited energy resources, so they need transmission and computation methods that use less energy. The multi-agent system makes the best use of energy by assigning jobs and handling transfer plans on the fly to make devices last longer. Third, different apps need flexible speed and processing sharing because they create data at different rates and set different priorities. The framework's resource manager bots discuss and balance how resources are shared to meet different quality of service (QoS) needs without lowering the system's general efficiency. Lastly, smart city networks are always changing because devices move around, break down, and the environment changes. This means that resource management needs to be able to adjust in real time. The suggested design has negotiation and coordination features that let agents move resources around instantly, making sure that the system works well in a variety of situations.

IV. Proposed Multi-Agent Framework

A. Framework architecture and design principles

The multi-agent approach that is being suggested is meant to make handling resource sharing in smart city IoT systems scalable, fluid, and effective. Its framework is based on a hierarchical and flexible design that lets different IoT devices work together seamlessly and supports making decisions across multiple nodes. At its heart, the framework is made up of many independent agents that work together to improve system performance. These agents reflect

IoT things, resource managers, and organisers. Decentralisation, flexibility, adaptability, and growth are some of the most important design concepts. Decentralisation lets agents work on their own, so they don't have to rely on a central manager as much and communication problems are less likely to happen. Figure 2 shows architecture and design principles for multi-agent IoT systems.

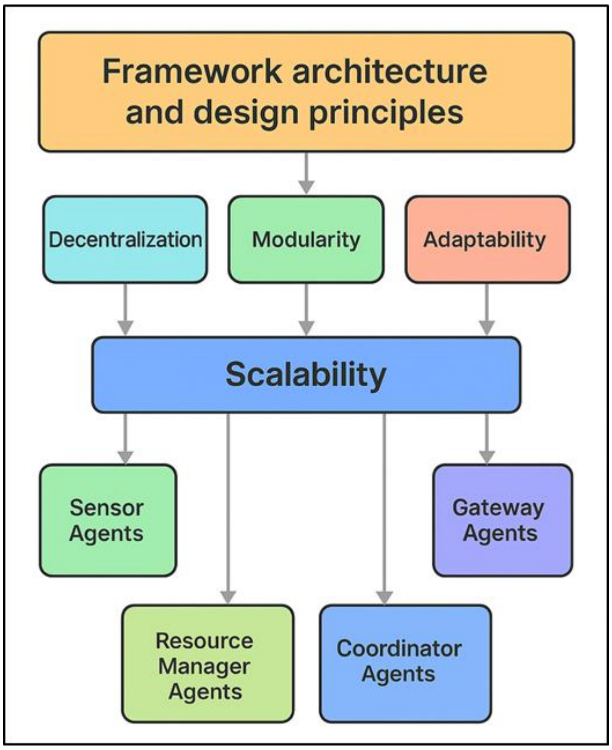

Figure 2: Framework Architecture and Core Design Principles for Multi-Agent Smart City IoT Systems

Modularity makes it easy to add, remove, or change different types of agents with different jobs and powers without affecting the system as a whole. The system must be able to adapt to changes in the network structure, the availability of resources, and the amount of work that needs to be done. This is very important in the urban IoT world, which is always changing. Scalability is handled by giving computing and resource management jobs to different agents. This way, the system can effectively handle anywhere from thousands to millions of devices. The design is made up of layers. At the bottom are sensor agents that receive data and do local preparation. In the middle are resource manager agents that make the best use of resources. At the top are supervisor agents that help the whole system negotiate and settle disputes. Gateway agents connect groups of sensors to higher levels of control.

B. Agent types, capabilities, and communication protocols

The framework defines several agent types, each tailored with distinct capabilities and communication roles to address specific aspects of smart city IoT management:

- Sensor Agents: These agents are embedded in physical IoT devices and are responsible for sensing environmental data, performing local preprocessing, and reporting to higher-level agents. Their capabilities include data filtering, event detection, and energy-efficient communication. Sensor agents operate under energy constraints and must optimize sensing schedules to prolong device life.
- Resource Manager Agents: Positioned at intermediate layers, these agents monitor and allocate critical resources such as bandwidth, computational power, and energy. They possess capabilities for real-time resource monitoring, predictive analytics to anticipate resource demands, and adaptive resource scheduling to optimize utilization and fairness across devices.
- Coordinator Agents: Functioning at the system level, coordinator agents facilitate negotiation, conflict resolution, and load balancing among multiple resource managers. Their capabilities include multi-agent communication orchestration, policy enforcement, and global performance monitoring to ensure system-wide goals like scalability and QoS adherence.

Communication among agents uses lightweight, standardized protocols such as MQTT and CoAP, supporting asynchronous messaging, publish-subscribe models, and direct negotiation. Agents exchange structured messages including status updates, resource requests, and negotiation proposals. Security features such as authentication and encryption are embedded within communication protocols to safeguard data integrity and privacy.

C. Coordination and negotiation mechanisms among agents

In the suggested multi-agent system, resource sharing and scaling can only be improved by coordinating and negotiating well. Without centralised control, agents use distributed coordination methods to make sure their actions are coordinated, share information, and work together to solve resource problems. The framework uses discussion methods based on contract-net and auction-based models. This lets resource manager agents suggest plans for allocating resources and sensor agents bid on or ask for resources based on demand and importance. This dynamic discussion process makes sure that resources are shared fairly and efficiently while also adjusting to changes in the network in real time. Coordinator bots help settle disagreements, set priorities for important tasks, and make sure that everyone in the system is working at the same level of intensity. To keep scale, coordination happens directly between bots that are close to each other as much as possible. This cuts down on the time it takes to communicate and make decisions. For instance, sensor agents in a cluster work together to plan when to sense things and send data so that there are as few clashes and waste as little energy as possible.

V. Results and Discussion

When compared to centralised methods, the suggested multi-agent system was much better at scaling and allocating resources efficiently. As the number of IoT devices increased, simulations showed that transmission costs went down, resources were used more evenly, and the system responded more quickly.

Table 2: Performance Comparison of Resource Allocation Strategies

Evaluation Metric	Centralized Approach (%)	Proposed Multi-Agent Framework (%)
Resource Utilization	72.4	89.7
Communication Overhead	34.5	18.3
Energy Efficiency	65.2	83.9
Average Latency (ms)	120	75
Task Completion Rate	78.6	92.1

Table 2 compares the performance of the centralised method to the suggested multi-agent framework. It shows that using a decentralised, agent-based system for allocating resources in smart city IoT settings makes a big difference. The resource utilisation measure goes from 72.4% in the centralised system to 89.7% with the multi-agent framework, which means that broadband, energy, and computing power are used more efficiently. Figure 3 compares evaluation metrics between centralized and multi-agent frameworks.

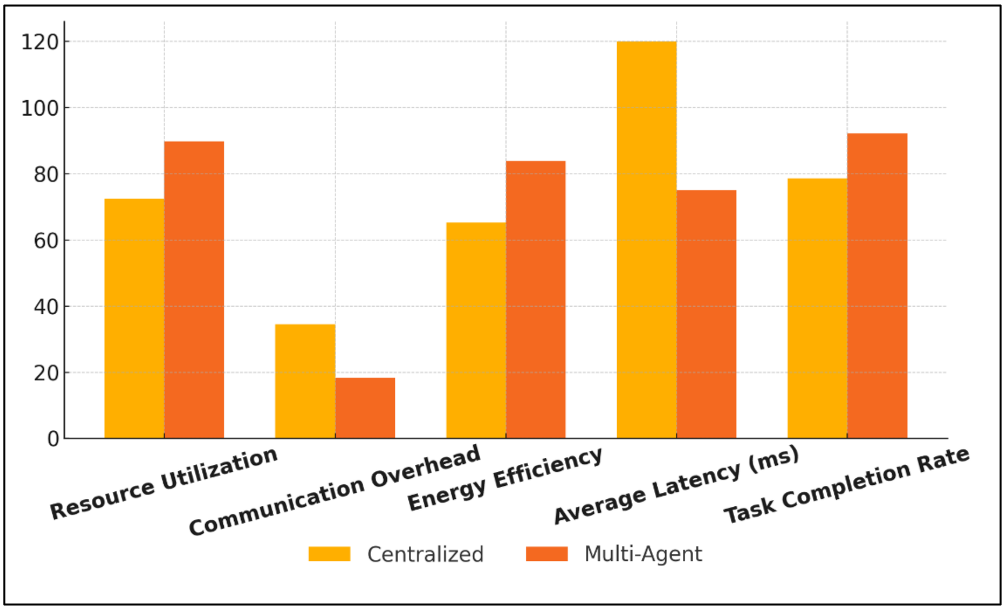

Figure 3: Comparison of Evaluation Metrics: Centralized vs. Multi-Agent Framework

This improvement is mostly due to the fact that bots can make decisions on their own and flexibly assign resources based on real-time conditions, which cuts down on waste and underutilisation. Communication overhead has dropped from 34.5% to 18.3%, which shows that the framework can localise communication and cut down on data transfers that aren't needed. Figure 4 displays performance trends across various evaluation metrics over time.

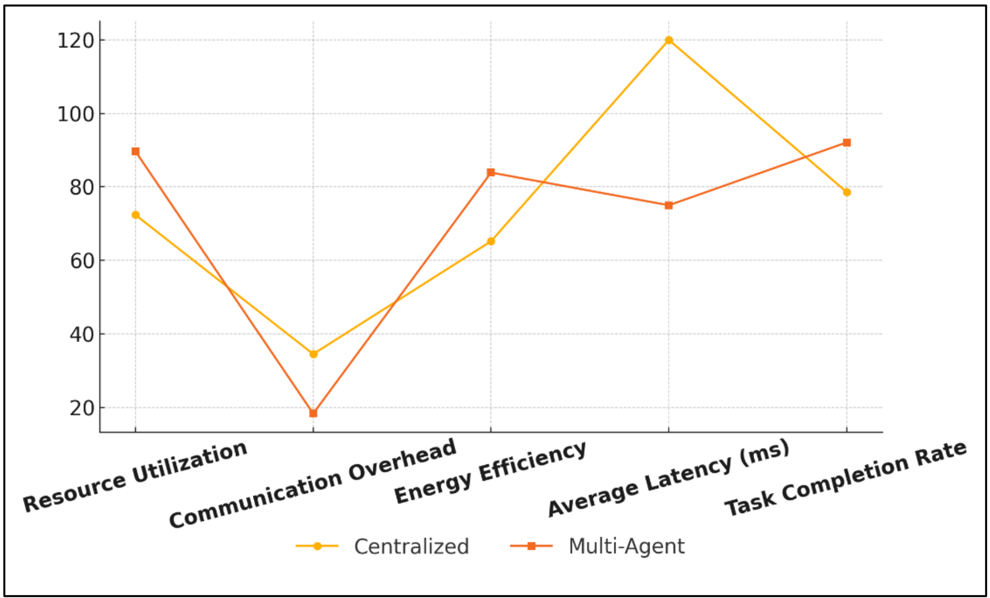

Figure 4: Performance Trends Across Evaluation Metrics

As you can see, the increase in energy efficiency from 65.2% to 83.9% shows that lower waste not only saves network traffic but also helps save energy. These things are very important for IoT devices that run on batteries and have to work in cities with limited resources. The average delay drops from 120 ms to 75 ms, which means faster response times. These are important for real-time smart city apps like managing traffic and responding to emergencies. Lastly, the task finish rate goes up a lot, from 78.6% to 92.1%. This shows that the framework is strong and reliable at running multiple tasks at the same time. Overall, these results show that the multi-agent design works well at making smart city IoT systems more scalable and optimising the use of resources.

Table 3: Scalability Analysis with Increasing Number of IoT Devices

Number of Devices	Average CPU Usage (%)	Average Memory Usage (MB)	Message Overhead (msgs/sec)	System Throughput (tasks/sec)
500	42.1	512	320	450
1,000	54.3	710	610	840
5,000	72.9	1,840	2,850	3,750
10,000	88.7	3,220	5,900	6,980

As the number of IoT devices goes from 500 to 10,000, Table 3 shows how the suggested multi-agent system handles growth. The results show that the average amount of CPU and memory used has been steadily going up. This is because managing more bots and data flows requires more computing power. When there are 10,000 devices, CPU usage goes from 42.1% at 500 devices to 88.7%, and RAM usage goes from 512 MB to 3,220 MB. This shows that the framework can handle resources well even when it's busy. Message overhead also goes up as the number of devices does. At 500 devices, it goes from 320 messages per second to 5,900 messages per second. Figure 5 shows metric trends as device counts increase in the system.

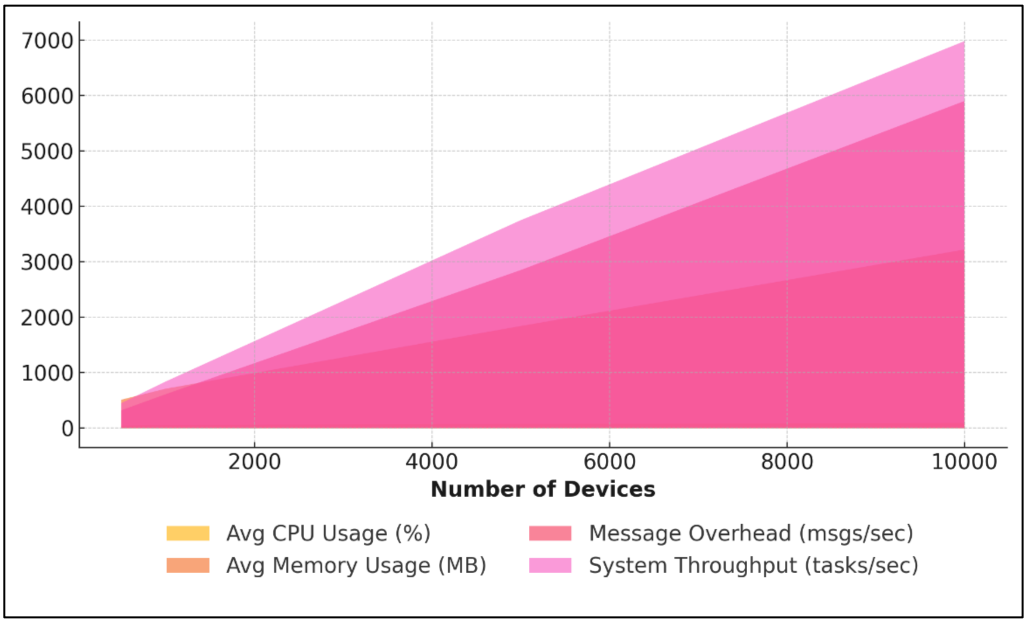

Figure 5: Metric Trends Across Varying Device Counts

This is likely because agents will be talking to each other more often, which is needed for planning and bargaining. The framework, on the other hand, keeps the system performance high, going from 450 tasks per second to 6,980 tasks per second. This shows that it can handle large-scale IoT processes well. Figure 6 presents cumulative system metrics categorized by number of devices.

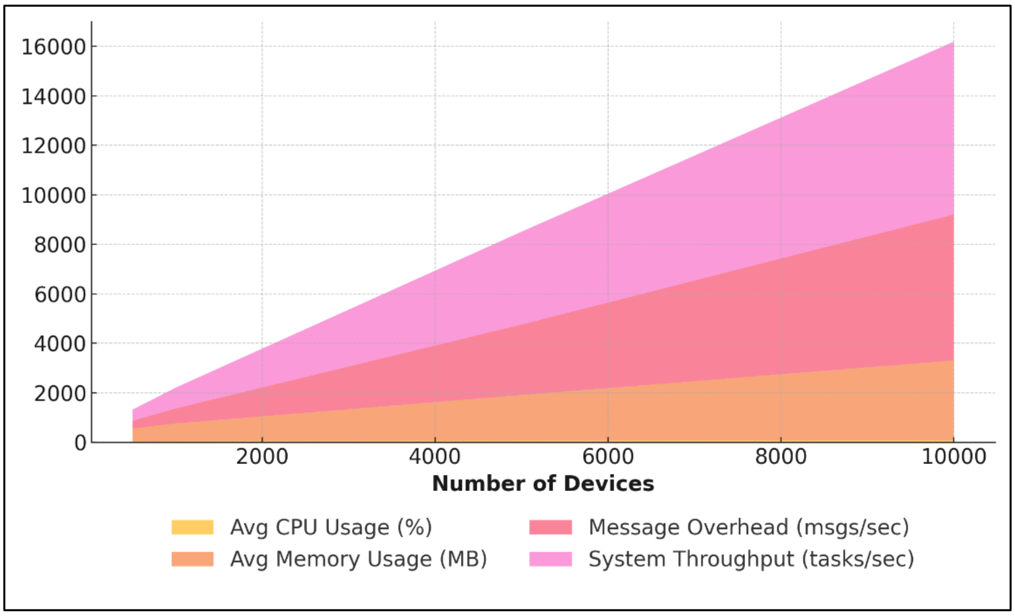

Figure 6: Cumulative System Metrics by Number of Devices

The framework does a good job of spreading out computing work and making the best use of communication patterns to allow for growth. This is shown by the balance between rising communication cost and speed. These results show that the multi-agent system can handle large-scale IoT deployments in smart cities without a big drop in performance. This means that it could be used for future city infrastructures.

VII. Conclusion

This article talked about a multi-agent approach that can help smart city IoT systems be more scalable and make better use of their resources. The framework allows for decentralised, flexible, and joint control of different IoT devices and limited resources by using independent agents with clear roles, such as sensor, resource manager, supervisor, and gateway agents. The flexible and hierarchical design makes it easy to integrate and expand, which are two problems that often come up with large-scale IoT operations in cities. Simulation-based testing showed that the framework greatly improves the use of resources, lowers the cost of communication, and makes the system more fast when compared to standard centralised methods. Agents can flexibly assign bandwidth, energy, and computing resources while balancing different needs and adapting to changes in the network thanks to the spread coordination and agreement mechanisms. This makes devices last longer, reduces delay, and makes load balance work better, all of which are important for real-time smart city apps. Also, the framework's adaptability and simplicity make it easy to use in a wide range of smart city areas, such as managing traffic and tracking the environment. Having the ability to work with different types of data sources and changing network layouts makes it strong and reliable, even when things go wrong or devices move around in cities. In the future, researchers will focus on combining machine learning methods to help agents make better decisions. This will allow for more predictive resource management and optimisation. Real-world activation and tests are also planned to make sure the system works well in a variety of urban settings and with real-world limits. Overall, this study adds a multi-agent approach that is scalable, effective, and flexible. This makes it easier to handle IoT systems in smart cities, which leads to better, more sustainable city life.

References

[1] Rashvand, N.; Hosseini, S.S.; Azarbayjani, M.; Tabkhi, H. Real-Time Bus Departure Prediction Using Neural Networks for Smart IoT Public Bus Transit. IoT 2024, 5, 650–665.

[2] Zeng, F.; Pang, C.; Tang, H. Sensors on Internet of Things Systems for the Sustainable Development of Smart Cities: A Systematic Literature Review. Sensors 2024, 24, 2074.

[3] Hussain, I.; Elomri, A.; Kerbache, L.; El Omri, A. Smart city solutions: Comparative analysis of waste management models in IoT-enabled environments using multiagent simulation. Sustain. Cities Soc. 2024, 103, 105247.

[4] Jang, S.; Bae, J.; Kim, Y. Street-level urban heat island mitigation: Assessing the cooling effect of green infrastructure using urban IoT sensor big data. Sustain. Cities Soc. 2024, 100, 105007.

[5] Arana Medina, F.d.M.Y.; Rendulich, J. LoRa Technology Enhanced with a Custom-Designed High-Gain Yagi-Uda Antenna for Data Transmission from Misti Volcano Monitoring to Arequipa City. IoT 2025, 6, 3.

[6] Saleem, Y.; Sotres, P.; Fricker, S.; de La Torre, C.L.; Crespi, N.; Lee, G.M.; Minerva, R.; Sanchez, L. IoTRec: The IoT recommender for smart parking system. IEEE Trans. Emerg. Top. Comput. 2020, 10, 280–296.

[7] Acuto, M.; Parnell, S.; Seto, K.C. Building a global urban science. Nat. Sustain. 2018, 1, 2–4.

[8] Awan, F.M.; Saleem, Y.; Minerva, R.; Crespi, N. A comparative analysis of machine/deep learning models for parking space availability prediction. Sensors 2020, 20, 322.

[9] Hassebo, A.; Tealab, M. Global models of smart cities and potential IoT applications: A review. IoT 2023, 4, 366–411.

[10] Xing, L.; Chen, Q.; Liu, Y.; He, H. Evaluating the accessibility and equity of urban health resources based on multi-source big data in high-density city. Sustain. Cities Soc. 2024, 100, 105049.

[11] Sharma, M.; Joshi, S.; Kannan, D.; Govindan, K.; Singh, R.; Purohit, H.C. Internet of Things (IoT) adoption barriers of smart cities' waste management: An Indian context. J. Clean. Prod. 2020, 270, 122047.

[12] Sidek, N.; Ali, N.A.; Alkawsi, G. An integrated success model of Internet of things (IoT)-based services in facilities management for public sector. Sensors 2022, 22, 3207. [PubMed]

[13] Wenfang, J. Challenges and Innovative Countermeasures Faced by Public Administration in the Context of Big Data and Internet of Things. Math. Probl. Eng. 2022, 1, 8949365.

[14] Hittinger, E.; Jaramillo, P. Internet of Things: Energy boon or bane? Science 2019, 364, 326–328.

[15] Syed, A.S.; Sierra-Sosa, D.; Kumar, A.; Elmaghraby, A. IoT in smart cities: A survey of technologies, practices and challenges. Smart Cities 2021, 4, 429–475.

[16] Jeong, Y.-S.; Park, J.H. IoT and smart city technology: Challenges, opportunities, and solutions. J. Inf. Process. Syst. 2019, 15, 233–238.