frontiers | Frontiers in Sustainable Cities

TYPE Original Research
PUBLISHED 09 September 2025
DOI 10.3389/frsc.2025.1623412

Check for updates

OPEN ACCESS

EDITED BY
Federico Cugurullo,
Trinity College Dublin, Ireland

REVIEWED BY
Katarína Vitálišová,
Matej Bel University, Slovakia
Ying Xu,
Central University of Finance and Economics, China

*CORRESPONDENCE
William Villegas-Ch
william.villegas@udla.edu.ec

†These authors have contributed equally to this work

RECEIVED 05 May 2025
ACCEPTED 20 August 2025
PUBLISHED 09 September 2025

CITATION
Villegas-Ch W, Gutierrez R, Govea J and Palacios P (2025) Design and evaluation of a decentralized urban governance system with embedded AI and blockchain-enabled IoT. *Front. Sustain. Cities* 7:1623412. doi: 10.3389/frsc.2025.1623412

Design and evaluation of a decentralized urban governance system with embedded AI and blockchain-enabled IoT

William Villegas-Ch[1]*, Rommel Gutierrez[1†], Jaime Govea[1†] and Pablo Palacios[2†]

[1]Escuela de Ingeniería en Ciberseguridad, Facultad de Ingeniería y Ciencias Aplicadas, Universidad de Las Américas, Quito, Ecuador, [2]Escuela de Informática y Telecomunicaciones, Universidad Diego Portales, Santiago, Chile

The increasing complexity of smart urban environments demands governance systems capable of autonomous decision-making, secure validation, and transparent traceability. This work presents the design and functional validation of a decentralized computational architecture for urban governance, focusing exclusively on its technical feasibility without involving real-world policy implementations. The proposed system integrates Internet of Things (IoT) devices, lightweight embedded Artificial Intelligence (AI) models, and blockchain-based smart contracts to enable automated, auditable, and fault-tolerant decision-making. The evaluation is carried out in a controlled environment equipped with Raspberry Pi boards, physical sensors, and actuators representing key urban infrastructure. Governance actions–such as adaptive lighting and traffic signal control–are inferred locally through embedded AI, validated through blockchain smart contracts, and executed only when validation logic is satisfied. Five operational scenarios of increasing complexity were conducted. The system consistently achieves response times between 230–360 ms, AI inference accuracy from 95% to 81%, and blockchain validation rates above 83%. It also recovers from hardware or connectivity failures in under 10 s. This study confirms the technical viability of distributed governance architectures powered by embedded intelligence and decentralized validation mechanisms.

KEYWORDS

decentralized urban governance, embedded artificial intelligence, blockchain-based validation, resilient IoT infrastructure, artificial intelligence

1 Introduction

The current dynamics of smart cities require governance mechanisms capable of processing large volumes of sensory information in real time, making autonomous decisions tailored to the context, and ensuring the traceability and validity of each action taken (Grossi and Welinder, 2024). Expanding connected devices in urban environments, such as environmental sensors, mobility systems, lighting control, and surveillance, has generated a dense and distributed cyber-physical infrastructure, whose effective management requires architectures beyond traditional centralized models. However, most currently implemented solutions rely on top-down approaches with single points of failure, hierarchical dependencies, and no formal mechanisms for autonomous and auditable decision validation. This technical limitation compromises smart urban systems' scalability, operational reliability, and transparency.

In this scenario, the convergence of three key technologies, the Internet of Things (IoT), distributed Artificial Intelligence (AI), and Blockchain, offers a unique opportunity to redefine urban governance based on principles of autonomy, resilience, and decentralization (Mahbub, 2021). The coordinated integration of these technologies enables edge devices to collect data and make real-time decisions, validate their consistency using contractual logic in blockchain environments, and act on the environment without requiring direct human intervention. However, this integration poses multiple technical challenges: synchronization between distributed nodes, energy consumption under computational constraints, tolerance to sensory or connectivity faults, and secure and efficient validation of decisions inferred by learning algorithms.

This study presents the design, implementation, and evaluation of a decentralized urban governance system based on the operational integration of physical IoT nodes, embedded artificial intelligence models, and smart contracts in a functional blockchain network (Montakhabi et al., 2023). Unlike previous proposals that explore these technologies in isolation or under virtual simulations, this study implements a physical experimental environment using Raspberry Pi devices connected to absolute sensors and actuators, integrated with a local inference module, and validated using smart contracts deployed on the Ethereum testnet. This implementation enables accurate operational testing of intelligent urban infrastructure behavior of an intelligent urban infrastructure under heterogeneous operating conditions and induced faults. The developed system is evaluated in five more complex scenarios, simulating an urban intersection with environmental variability, pedestrian flow, artificial weather conditions, and controlled failures. Each scenario generates a set of sensory events interpreted by adaptive inference models, whose output triggers autonomous decisions such as turning on lighting, controlling traffic lights, or activating energy-saving modes. These decisions are immediately validated by smart contracts that evaluate criteria of context, temporal consistency, and compliance with programmed policies. If the decision meets the contractual conditions, it is executed and recorded on the blockchain; otherwise, it is discarded or deferred until contextual validation is achieved.

The results obtained confirm the viability of this approach. In the most controlled scenario (Scenario 1), the system maintains an average response time of 230 milliseconds, an AI model accuracy of 95%, and a blockchain validation rate of 97%. As operational complexity increases (up to Scenario 5), the system continues to operate with average response times of 360 ms, an accuracy of 81%, and a validation rate of 83%, without compromising structural stability or functional consistency. In terms of resilience, the system recovers from critical failures, such as sensor loss, network interruption, or inference node failure, with recovery times between 4.5 and 10 s, implementing automatic compensation strategies such as logical fallback, state persistence, distributed inference, and lazy validation.

Furthermore, the system maintains controlled total power consumption, ranging between 61 and 86 mAh, even in the highest load and operational stress scenarios. This energy efficiency is achieved by implementing optimized inference models, asynchronous contract execution, and dynamic context-aware actuator control.

Compared to proposals that implement only partial components of this architecture, such as those that integrate AI-based inference in centralized environments without distributed auditing mechanisms, as described by (Miglani and Kumar, 2021), or those that rely exclusively on digital simulations without verification in physical environments, as in the work of Liu et al. (2019). The system proposed here demonstrates complete operational integration. It has been validated in a controlled physical environment, incorporates edge inference and distributed consensus mechanisms using blockchain, and guarantees traceability and resilience to changing conditions. Governance is decentralized, computationally verifiable, and auditable, which represents an advance over previous fragmented solutions.

The structure of the article is as follows. Section 2 presents a literature review, analyzing recent work addressing the integration of distributed technologies in urban contexts and identifying the operational and technical gaps that motivate this research. Section 3 describes the system architecture, the materials used, the physical implementation environment, the embedded artificial intelligence model, the blockchain validation logic, the urban simulation scenario, and the experimental procedure. Section 4 presents the results obtained regarding performance, accuracy, energy consumption, distributed validation, and operational resilience metrics, including a comparative analysis with representative proposals from the literature. Section 5 discusses the findings, their technical implications, the system's unique contributions, and the limitations affecting its scalability and applicability. Finally, Section 6 presents the study's conclusions and proposes specific lines of future work.

2 Literature review

A literature review was conducted based on studies published in high-relevance, open-access scientific databases, applying inclusion criteria such as: (i) the articulation between emerging technologies applied to urban environments, (ii) proposals for decentralized planning or governance models, and (iii) evidence of practical or simulated implementation.

The selected studies have been reorganized into two thematic blocks: (1) governance architectures and their technological enablers, and (2) practical deployments and validated scenarios. This organization responds to the need to clarify the structure, reduce fragmentation, and highlight the practical relevance of the works considered.

2.1 Governance architectures and enabling technologies

The evolution of smart cities has prompted a reconfiguration of traditional governance paradigms. Lustenberger et al. (2025) propose a model based on Decentralized Autonomous Organizations (DAOs), combining local-global structures with

TABLE 1 Classified review of studies by category and contribution.

Reference	Category	Technologies	Main contribution
Sharma et al. (2021)	Practical deployment	AI, IoT, blockchain	Decentralized architecture on Raspberry Pi
Zakaie Far et al. (2025)	Architecture	AI (Reinf.), blockchain, IoT	Collaborative multi-agent learning system
Lustenberger et al. (2025)	Governance model	Blockchain, DAO	Polycentric urban governance with voting mechanisms
Rasoulzadeh Aghdam et al. (2024)	Participatory design	AI, Blockchain	Bibliometric study on civic participation
Gracias et al. (2023)	Conceptual framework	Cross-platform	Structured model of digital governance systems
Cai and Hong (2024)	Sustainability	Blockchain, Neural nets	Autonomous management of digital economic impacts
Tan and Taeihagh (2020)	Impact assessment	Governance, ethics	Framework focused on transparency, inclusion, and algorithmic legitimacy
Mrabet and Sliti (2024)	Energy management	AI (contrastive nets), IoT	Urban energy demand forecasting and optimization
Lubis et al. (2025)	Bibliometric gap	AI, Blockchain	Identifies lack of empirical studies combining AI and blockchain for governance
Montori et al. (2023)	Industrial IoT	Toolchains, IoT	Architecture for condition monitoring and integration of legacy systems
Adreani et al. (2024)	Smart city platform	Digital Twins, ML, 3D IoT	Real-time 3D urban modeling integrated in Snap4City platform

smart contracts and holographic voting to redistribute power in public space management. This organizational and technological decentralization forms a conceptual baseline for distributed decision-making systems.

In a complementary approach, Rasoulzadeh Aghdam et al. (2024) conduct a bibliometric review of participatory governance, revealing the growing use of blockchain for traceable decisions and AI for algorithmic assistance. Gracias et al. (2023) contribute a theoretical synthesis of digital governance based on interdependent systems connecting institutions, citizens, and data platforms.

Zakaie Far et al. (2025) extend this notion by integrating reinforcement learning with blockchain to develop multi-agent systems that optimize urban resources autonomously. Cai and Hong (2024) highlight the relevance of explainable and auditable mechanisms using neural networks and blockchain for sustainable digital economies.

Adding a macro-perspective, Lubis et al. (2025) identify through bibliometric analysis the limited empirical deployment of integrated AI and blockchain architectures in governance. This work reinforces the research gap our system seeks to fill, namely, the lack of physical implementations and the operationalization of such models.

2.2 Practical deployments and urban experimentation

Sharma et al. (2021) demonstrate a distributed architecture that integrates IoT, AI, and blockchain on Raspberry Pi devices, although under simulated conditions. Mrabet and Sliti (2024) focus on predictive energy demand modeling using contrastive neural networks, targeting optimization in smart infrastructures.

Montori et al. (2023) present a condition monitoring toolchain based on abstraction layers and modular engineering pipelines, implemented and validated in the Arrowhead Tools project. The architecture simplifies management of heterogeneous IoT infrastructures, a critical factor for replicable and scalable systems.

Adreani et al. (2024) apply integrated digital twins and 3D modeling within the Snap4City platform, incorporating thousands of real-time IoT sources in the city of Florence. Their validation in operational environments shows the potential for data-driven decision-making supported by visual and algorithmic interfaces.

These studies provide essential precedents of technical feasibility and deployment strategies. While our system differs in its decentralized validation and embedded AI logic, it shares their commitment to operational realism, automation, and multi-source integration.

Table 1 summarizes the articles by thematic category, technologies, and primary contributions.

Despite progress in integrating technologies for urban governance, gaps remain regarding (i) empirical validation, (ii) seamless interoperability, and (iii) ethical-operational traceability. This work addresses these gaps with a fully functional prototype based on AI, IoT, and blockchain, under realistic urban dynamics in a controlled experimental setting. The system incorporates explainability through lightweight AI models, auditability via smart contracts, and local autonomy aligned with governance principles.

3 Materials and methods

The proposed system is grounded in a multi-level conceptual framework derived from the literature reviewed, which combines organizational, technological, and evaluative components. At the managerial level, the system assumes a decentralized governance paradigm where nodes (devices, institutions, or citizens) interact through autonomous decision-making protocols. Technologically, it integrates distributed IoT devices, local AI-based control mechanisms, and blockchain validation layers to ensure traceability and resilience. Evaluatively, the system is designed to measure operational performance (latency, autonomy, energy efficiency) and governance-aligned properties (transparency, auditability,

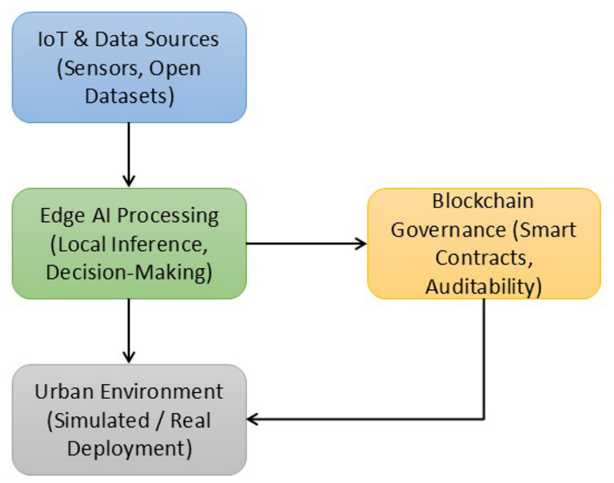

FIGURE 1
Multi-level conceptual framework integrating IoT sensing, embedded AI decision-making, and blockchain-based governance, connected to the urban environment.

and fairness). This architecture is not merely theoretical but materialized through a laboratory-scale deployment that emulates realistic urban scenarios. Thus, the methodology bridges conceptual principles and empirical implementation, directly addressing the gaps identified in the literature.

Figure 1 illustrates the multi-level conceptual framework that integrates the organizational, technological, and evaluative components of the proposed system. This figure synthesizes the layered architecture by showing how IoT data sources, edge AI inference, and blockchain-based validation interact within a decentralized governance paradigm, ultimately connected to the urban environment where decisions are executed.

3.1 General description of the proposed system

The architecture developed in this study implements a decentralized system for smart urban governance by integrating physical IoT nodes, embedded artificial intelligence modules, and a blockchain-based validation and auditing mechanism (Singh et al., 2020). The system was deployed and tested in a controlled experimental environment equipped with physical devices, enabling real-time execution of decision-making processes applicable to critical urban services such as traffic management, adaptive lighting, and energy distribution.

The system starts by acquiring contextual data through a network of physical IoT nodes consisting of sensors and actuators. These components capture real-time information from the test environment, including temperature, light intensity, and vibration levels. Sensory data is transmitted using lightweight communication protocols, specifically MQTT, enabling reliable, low-latency communication to the edge processing node (Alasmari and Alhogail, 2024). This node consolidates incoming data streams and connects to the embedded intelligence layer. During processing, it executes a lightweight AI model pre-trained for adaptive decision-making based on the input data. The model, optimized for resource-constrained devices, was trained on a synthetic dataset reflecting diverse urban scenarios. The resulting decisions include activating actuators, such as switching lights or modifying traffic signals.

After inference, the decision, contextual sensory data, and a timestamp are transmitted to a blockchain platform for decentralized validation. This step is executed using smart contracts deployed on the Ethereum testnet, which verify that the decision complies with pre-established governance rules (Ravikumar et al., 2022). Validated actions are recorded immutably on the blockchain ledger, ensuring traceability and transparency of system behavior. This validation layer prevents unauthorized manipulation and reinforces system integrity. Validated actions also interact with the physical actuators, closing the operational loop between sensing, inference, validation, and action. Simultaneously, the blockchain record serves as a persistent audit trail to support performance assessment, anomaly tracking, and policy evaluation.

Figure 2 illustrates the complete system architecture, detailing the data flow from sensory acquisition to blockchain validation and feedback. Color-coded modules represent each subsystem, while arrows indicate the logical sequence and data exchange between components.

3.2 Environment and equipment

The system was deployed and evaluated in a controlled experimental environment designed to reflect real-time urban operating conditions. This setup enabled the validation of a decentralized governance architecture composed of physical IoT nodes, embedded AI modules, and a blockchain-based auditing mechanism. The environment was configured according to criteria such as energy efficiency, interoperability of heterogeneous technologies, and support for real-time traceability. The implementation included both physical hardware and software components, selected based on a technical comparative analysis.

At the sensing and actuation layer, Arduino UNO R3 microcontrollers were used as terminal nodes to capture environmental data. The system included DHT22 sensors for temperature and humidity, LDRs for ambient light detection, and PIR modules for motion sensing. These signals served as contextual inputs for autonomous decision-making. Actuation was performed via solid-state relays, LED arrays, and SG90 microservos, enabling the system to replicate behaviors of urban infrastructure such as adaptive lighting and traffic control. Nodes were physically connected to edge processors and communicated asynchronously using the MQTT protocol.

Raspberry Pi 4 Model B boards with 4 GB RAM served as edge nodes, executing optimized AI models using TensorFlow Lite. These devices consolidated sensor data, applied contextual logic, and generated actions based on the operational conditions. The units ran Raspbian OS with a customized stack including SSH, NTP synchronization, and relevant libraries such as Web3.py and paho-mqtt (Lee, 2023; Pawar et al., 2023).

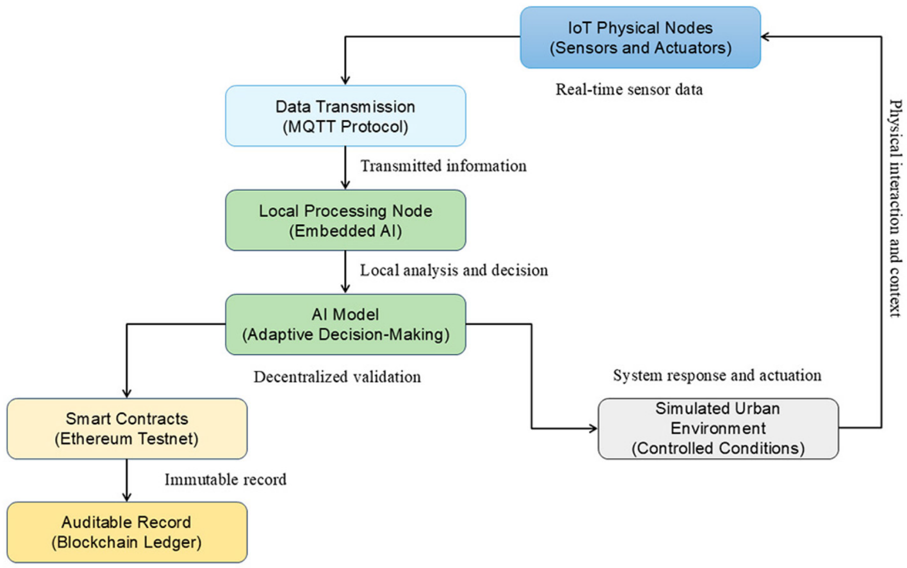

FIGURE 2
Functional architecture of the decentralized urban governance system with integration of IoT, AI, and blockchain.

TABLE 2 Technical comparison of candidate technologies and justification for selection.

A	B	C	D	E
Microcontroller	Raspberry Pi 4, Jetson nano, ESP32	Computing power, AI compatibility, power use, integration	Raspberry Pi 4 (4 GB)	Balanced processing power, low power use, TensorFlow Lite support, and direct blockchain connectivity
Sensor gateway	Arduino UNO, ESP32, STM32	Accuracy, sensor compatibility, programming	Arduino UNO R3	High compatibility with standard sensors, low cost, and easy integration via USB/I2C
IoT protocol	MQTT, CoAP, HTTP, AMQP	Protocol weight, latency, overhead, embedded support	MQTT (mosquitto)	Lightweight, asynchronous, publish/subscribe ready, and widely embedded-compatible
Blockchain platform	Ethereum testnet, hyperledger fabric, tezos	Deployment, smart contracts, tools, transparency	Ethereum Sepolia	Supports Web3.py, MetaMask, large community, realistic simulation environment
Logic validation	Solidity, go, Michelson	Language maturity, tooling, platform support	Solidity	Native to ethereum, supported by Remix/truffle, extensive audit-friendly documentation
AI framework	TensorFlow lite, PyTorch mobile, edge impulse, ONNX runtime	Compatibility, inference speed, conversion tools	TensorFlow Lite	Optimized for low-power real-time inference on Raspberry Pi, minimal accuracy loss
Visual orchestrator	Node-RED, ThingsBoard, Blynk	GUI integration, device control, customization	Node-RED	Enables rapid prototyping, MQTT-ready, and supports custom flows for test automation

A: Component; **B**: Evaluated alternatives; **C**: Comparative criteria; **D**: Selected Technology; **E**: Technical Justification.

Communications among system components occurred over a private Wi-Fi network with static IP addressing and a centralized MQTT broker (Mosquitto) running on one Raspberry Pi. This ensured low latency and efficient message routing.

The auditing and validation layer was implemented on the Ethereum Sepolia testnet. Smart contracts written in Solidity were used to validate AI-generated decisions against predefined governance logic (Tavakoli et al., 2024). Each decision, along with its sensor input and timestamp, was recorded immutably on the blockchain. Tools such as Remix IDE, MetaMask, and automated scripts in Web3.py (Kaur et al., 2024) supported contract deployment and validation.

Node-RED was initially used for orchestration and debugging, enabling real-time visualization of data flow. Once validated, the system was fully migrated to native Python code.Table 2 presents a structured comparison of evaluated technologies, justifying the final selections implemented in the operational architecture.

The selection of these components responds to a design strategy geared toward operational efficiency and scientific replicability. The resulting system combines algorithmic autonomy, decentralized traceability, and distributed sensing, validating its applicability in simulated urban contexts as a preliminary step toward deployment in real-world scenarios.

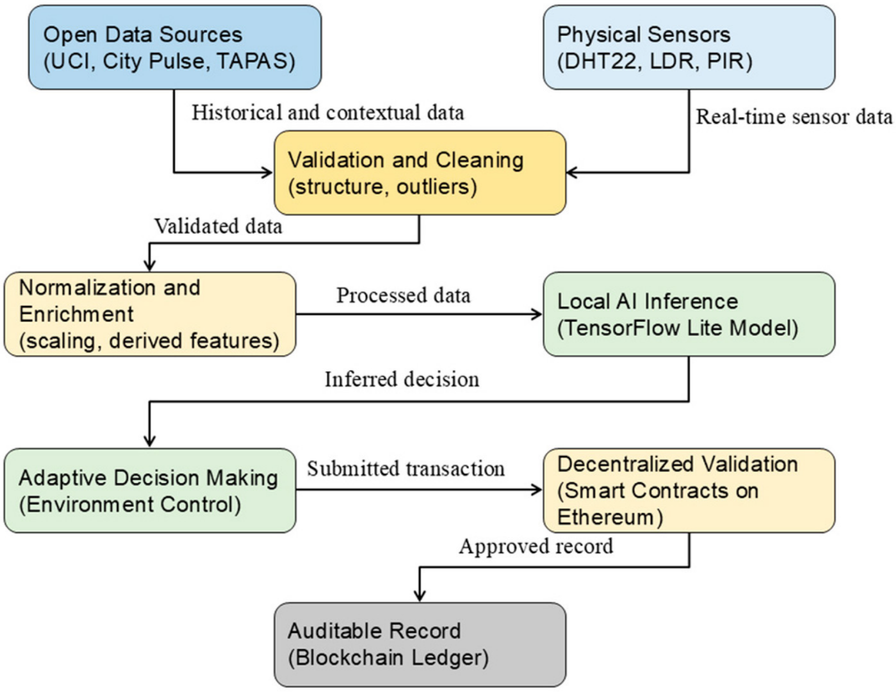

FIGURE 3
Data acquisition, processing, and integration flow in the decentralized urban governance system.

3.3 Data source and processing

The system integrates data from both public datasets and physical sensors deployed in the experimental environment, enabling the validation of decision-making processes under conditions that combine real-time inputs with historical urban patterns. This hybrid data strategy enhances the generalization capacity of the embedded AI model and ensures contextual relevance across evaluation scenarios.

The public datasets used for training and fine-tuning include:

- UCI Air quality dataset, which provides real-world environmental readings such as temperature, humidity, and pollutant concentrations from electronic sensors in urban areas (Kumar et al., 2023).
- City pulse dataset, which offers data on traffic, pollution, and citizen behavior across European smart cities (Daneshvar et al., 2023).
- TAPAS cologne dataset, a high-resolution synthetic dataset simulating urban vehicular traffic, including density, speed, and intersection occupancy metrics (Zhu et al., 2018).

These datasets supported the training of the lightweight AI model implemented in the system, providing representative patterns used to optimize its decision-making under diverse urban conditions.

Complementarily, the experimental setup continuously generates real-time data through physical sensors integrated with Arduino UNO microcontrollers. These include temperature and humidity (DHT22), ambient light (LDR), and motion detection (PIR). Sensor readings are transmitted via MQTT to an edge processing node (Raspberry Pi 4), where the data is locally stored and preprocessed.

The preprocessing pipeline includes:

- Structural validation to verify record completeness and detect malformed entries.
- Data cleaning using interquartile range (IQR) analysis and outlier removal based on historical distributions.
- Min-max normalization adapted to the scale of each sensory variable.
- Feature engineering to derive new variables such as rate of change or composite environmental indices.

The processed data feeds a TensorFlow Lite model deployed on the edge node, which performs real-time inference to determine adaptive actions. Each inference result, along with its input context, is encoded into a structured transaction and submitted to the Ethereum testnet, where a smart contract verifies compliance with predefined governance rules. Upon approval, the decision is immutably recorded on the blockchain.

Figure 3 illustrates the end-to-end data flow, from acquisition through inference and validation. It highlights the convergence of two primary data sources: open datasets used for model training and real-time physical sensor data used during system execution. The processing chain includes validation, normalization, inference, and decentralized auditing, ensuring operational transparency and reinforcing the system's integrity.

Subsequently, the validated data is normalized and enriched using scaling techniques and derived variable generation, which allows for uniform scales across sensors and optimizes the input for the inference models. The next stage corresponds to local inference using AI, in which a lightweight TensorFlow Lite model determines the optimal action to execute. The inferred decision branches into two paths: one to the adaptive environment control module (e.g., activating actuators or changing the state of the simulated system) and another to decentralized validation through smart contracts deployed on the Ethereum testnet (Tsuyuguchi and Wang, 2024). The decision is converted into a structured transaction that is evaluated by the smart contract. If approved, the action is immutably recorded in the blockchain ledger, guaranteeing its traceability and reinforcing the principle of transparent auditing in the system architecture.

3.4 Implemented artificial intelligence model

The AI module implemented in the system is responsible for automating decisions within the decentralized governance architecture. This section details the neural network architecture deployed, the input variables used during inference, the training methodology, and its integration with the smart contract validation layer.

3.4.1 Model type

Given the computational limitations of embedded hardware and the need for low-latency inference, a lightweight Multilayer Perceptron (MLP) model was selected, trained, and optimized offline. The finalized model was exported to TensorFlow Lite format and deployed directly on the Raspberry Pi edge nodes.

The model addresses a multi-class classification problem, predicting the optimal governance action based on contextual sensor inputs. The network consists of two fully connected hidden layers with ReLU activation functions, followed by an output layer using softmax activation. The inference function is defined as:

$$\hat{\mathbf{y}} = \text{softmax}\left(\mathbf{W}_2 \cdot \phi(\mathbf{W}_1 \cdot \mathbf{X} + \mathbf{b}_1) + \mathbf{b}_2\right) \tag{1}$$

Here, ϕ denotes the ReLU activation, $\mathbf{X} \in \mathbb{R}^5$ represents the input vector comprising sensory variables, and $\hat{\mathbf{y}} \in \mathbb{R}^c$ is the resulting probability distribution over the action classes. Parameters $\mathbf{W}_i$ and $\mathbf{b}_i$ correspond to the weights and biases learned during training.

The deployed model operates directly on the edge nodes and produces decision outputs in real time, which are subsequently evaluated through the blockchain-based smart contract layer to ensure compliance with governance policies before execution.

3.4.2 Input variables

The input to the AI model consists of five contextual features extracted from real-time sensor readings and derived from system-level processing during operation. These variables were selected to represent relevant aspects of the urban environment and support reliable decision-making under varying conditions.

The input vector $\mathbf{X}$ includes the following components:

- x_1: Normalized ambient temperature [°C]
- x_2: Relative illumination level [%]
- x_3: Presence detection (binary) [0,1]
- x_4: Time of day encoded as $\sin(\theta_t)$ to preserve periodicity
- x_5: Urban occupancy index, derived from aggregated sensor signals

The complete input vector is defined as:

$$\mathbf{X} = [x_1, x_2, x_3, x_4, x_5]^\top \tag{2}$$

Before inference, each feature was subjected to structural validation and Min-Max normalization based on expected operational ranges. Derived features, such as the circular time representation and occupancy index, were computed to enhance the model's ability to capture both temporal and spatial dynamics. These input variables were consistently used across training, validation, and deployment phases of the system.

3.4.3 Model training

The AI model was trained offline using a hybrid dataset composed of synthetic records from open sources (UCI, City Pulse, TAPAS) and real-world data acquired from the physical testbed. A balanced and labeled dataset was constructed to reflect expected urban responses under diverse sensory conditions.

Training was performed using the categorical cross-entropy loss function and the Adam optimizer, with a learning rate of 0.001. To ensure model generalization, L2 regularization and cross-validation techniques were applied, mitigating the risk of overfitting. After training, the model was converted into TensorFlow Lite (.tflite) format using post-training quantization. This process significantly reduced model size while preserving accuracy, enabling deployment on ARM-based devices with limited computational resources.

3.4.4 Model output and integration

The output of the model is a probability vector $\hat{\mathbf{y}}$, where each element $\hat{y}_i$ represents the predicted likelihood for each governance action. The optimal decision is selected using the following rule:

$$\text{action}_{\text{opt}} = \arg\max_i \hat{y}_i \tag{3}$$

The output classes correspond to predefined policy actions in the urban control framework:

- Class 0: turn off public lighting
- Class 1: activate low-intensity lighting
- Class 2: activate high-intensity lighting
- Class 3: change the traffic light to red
- Class 4: change the traffic light to green
- Class 5: trigger energy-saving mode
- Class 6: maintain current state

Each decision inferred by the AI model is packaged into a structured transaction that includes the selected action, the associated sensory context, and the digital signature of the originating node. This transaction is transmitted to the blockchain validation layer, where a smart contract verifies its consistency and compliance with predefined governance rules. Upon approval, the action is executed within the controlled experimental environment and immutably recorded on the blockchain, ensuring transparency and traceability.

3.5 Integration with the blockchain platform

The AI-driven decisions generated by the system are subjected to decentralized validation and auditing through direct integration with a blockchain platform. This mechanism enforces transparency, ensures the immutability of actions, and supports computational trust within the distributed urban governance architecture. The integration encompasses the design and deployment of smart contracts, the interaction logic between IoT nodes and the Ethereum testnet, and the technical structure of validated transactions.

3.5.1 Smart contract design

A smart contract was developed in Solidity and deployed on the Ethereum Sepolia testnet. Its role is to validate whether the decisions inferred by the AI model meet the conditions defined in the governance policies. The contract encodes logical rules that govern the legitimacy of each action, based on contextual input and operational constraints.

Each transaction submitted to the smart contract includes the selected action, the corresponding sensory input vector $\mathbf{X}$, a unique node identifier, and a timestamp. The contract evaluates this input using a logical function that enforces predefined governance conditions. The validation process is defined as follows:

$$\text{Validation}(\mathbf{X}, a, t) = \begin{cases} \text{true} & \text{if } a \in \mathcal{P} \wedge \mathbf{X} \in \mathcal{R} \wedge t \in \Omega \\ \text{false} & \text{otherwise} \end{cases} \tag{4}$$

Where:

- a is the proposed action,
- $\mathcal{P}$ is the set of permitted actions under the governance logic,
- $\mathbf{X} \in \mathbb{R}^n$ represents the sensory input vector,
- $\mathcal{R}$ denotes the valid operational ranges for each input,
- t is the action timestamp,
- Ω defines the valid execution time window.

The deployed contract maintains a historical registry of all validated decisions, prevents duplication, and includes filtering mechanisms to ensure critical operational constraints such as power usage thresholds are respected. For instance, it blocks attempts to activate high-intensity lighting during daytime hours. This logic was tested during the evaluation scenarios and proved effective in enforcing the defined governance policies in real-time conditions.

3.5.2 Interaction between the IoT node and the blockchain

Once the AI model produces an inference, each coordinator node constructs a structured transaction containing the selected action, the corresponding sensory input vector, and the node's digital signature. This process is implemented using a Python interface built on the Web3.py library, which connects to an Ethereum RPC node and signs transactions using ECDSA cryptography (Puthiyidam et al., 2024).

The transaction payload includes the target smart contract address, node identifier, sensor data encoded in JSON format, and a cryptographic signature to ensure authenticity. The transaction is signed locally with the node's private key and sent via an RPC client configured with MetaMask in testnet mode.

Algorithm 1 - pseudocode summarizes the interaction implemented between the IoT node and the blockchain network.

The smart contract emits a confirmation event, which is captured by the node through a listener implemented in Web3.py. Based on the response, the node either proceeds with executing the validated action in the experimental system or records a denial if the action violates governance policies.

3.5.3 Transaction types and block structure

The system generates state-changing transactions that are submitted by IoT nodes and processed within the Ethereum testnet (Sepolia), where simulated gas consumption reflects real-world conditions. Each transaction is incorporated into a block on the chain, following the Ethereum standard format and containing the following fields:

- Hash of the previous block
- Block timestamp
- List of executed transactions
- Merkle root of the transactions
- Validation nonce
- Final storage state of the smart contract

The inclusion of each governance decision as a blockchain transaction ensures its immutability and prevents any unauthorized modification. Once written to the ledger, the action becomes a

```
Read current sensor data: X = [x1, x2, ..., xn]
Execute AI model: action ← model_inference(X)
Construct transaction TX:
   TX.payload ← {
     "action": action,
     "timestamp": current_time,
     "context": X
   }
Sign TX with node's private key: TX.signature
   ← Sign(TX.payload, private_key)
Connect to Ethereum network via Web3 RPC
Send TX to smart contract
Wait for confirmation of receipt and validation
Log transaction state: "Approved" or "Rejected"
```

Algorithm 1. Node decision submission to blockchain.

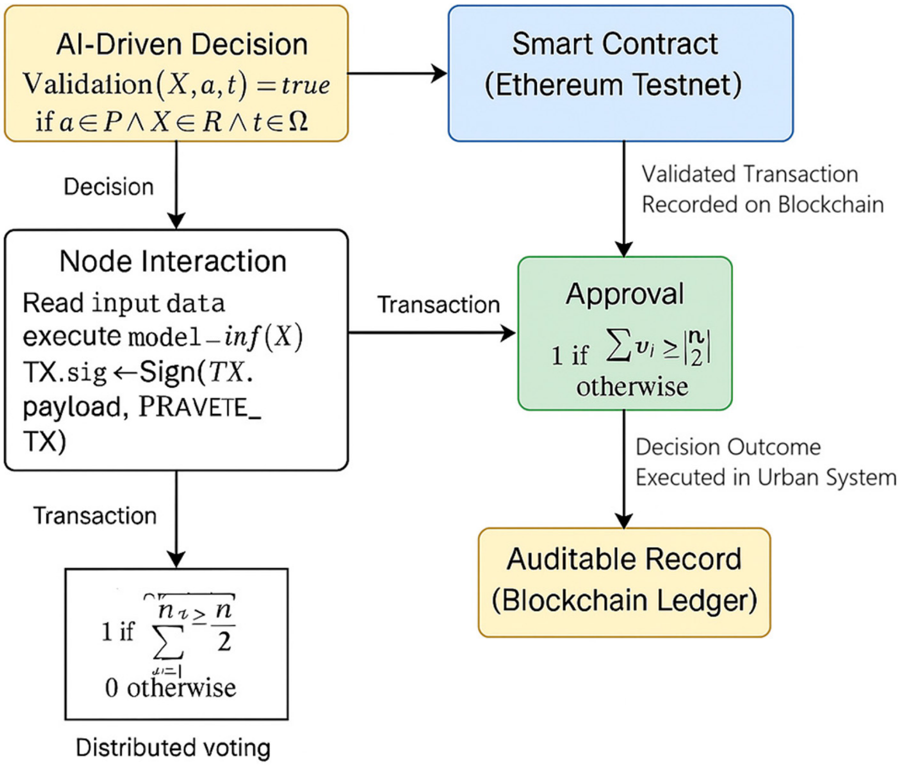

FIGURE 4
Distributed validation flow and decision execution in the urban governance system.

permanent, auditable record. This blockchain structure reinforces the integrity and traceability of every decision executed by the system.

3.5.4 Distributed governance

To evaluate decentralized decision-making within the governance model, a multi-node consensus mechanism was implemented. Multiple validator nodes, both physical and logical, participate in a distributed voting process that replaces unilateral decision enforcement with a majority-based rule. Each node independently evaluates the AI-inferred action using its local context or policy thresholds and casts a binary vote.

The system adopts a simple majority rule, formalized as:

$$\text{Approval} = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} v_i \geq \left\lceil \frac{n}{2} \right\rceil \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Here, $v_i \in \{0, 1\}$ denotes the vote from node i, and n is the total number of validators. The smart contract collects all votes via emitted events and registers the action only if the consensus threshold is reached. This logic reproduces realistic governance scenarios such as federated urban infrastructures or distributed municipal committees (Ghifari et al., 2021).

Figure 4 illustrates the operational flow: the AI module infers a decision, which is structured, digitally signed, and submitted to the blockchain. Validator nodes assess the decision, cast their votes, and, upon majority approval, the decision is immutably recorded and executed within the experimental urban control environment. This distributed validation scheme, combined with local AI inference and physical sensing, transforms the system into a fully functional experimental architecture for autonomous urban governance, founded on decentralization, algorithmic integrity, and transparent auditing.

3.6 Controlled urban simulation scenario

The system was validated in a controlled physical testbed designed to emulate the dynamics of a smart urban intersection. This environment enabled the evaluation of the system's performance under real-time operating conditions, including local AI inference, blockchain-based validation, and physical actuation. The constructed setup included scaled infrastructure with elements such as vehicle lanes, pedestrian crossings, street lighting, and traffic control systems, each connected to distributed sensors and actuators managed by coordinator nodes.

The sensing infrastructure included DHT22 modules for temperature and humidity, LDR sensors for ambient light measurement, and PIR sensors to detect pedestrian movement. These components were physically integrated with Arduino UNO microcontrollers, which communicated via a local Wi-Fi network with Raspberry Pi edge nodes. Each edge node performed real-time

TABLE 3 Controlled urban scenarios and evaluated conditions.

A	B	C	D	E	F
Scenario 1	Day	Absent	Absent	Constant ambient lighting	6
Scenario 2	Night	Moderate (10 s)	Moderate (10 s)	Adaptive lighting activation	6
Scenario 3	Day	Intense (4-6 s)	Intense (4-6 s)	Expected increase in energy consumption	6
Scenario 4	Night	Nil	Nil	Simulated failure of main luminaire	6
Scenario 5	Alternating	Variable (random)	Variable (random)	Combined events: motion + light + humidity	6

A: Scenario; **B**: Time Condition; **C**: Pedestrian Flow; **D**: Vehicle Flow; **E**: Simulated Events; **F**: Rounds.

inference and submitted decisions for validation on the Ethereum blockchain, following the protocol described in previous sections.

3.6.1 Test conditions and scenario design

The test environment incorporated dynamic conditions to replicate realistic urban contexts and evaluate the system's adaptability. The following physical variables were introduced and controlled:

- **Time of day simulation:** external light sources were adjusted to mimic day-night cycles, directly influencing the LDR sensor readings.
- **Pedestrian flow:** human movement patterns were induced near PIR sensors using controlled movement of objects and personnel at variable frequencies.
- **Microclimate variation:** localized temperature and humidity levels were altered in real time using thermal elements and vaporizers, directly impacting DHT22 sensor values.
- **Disturbance events:** predefined anomalies, such as lighting malfunctions and sensor signal saturation, were triggered to test the system's fault tolerance and recovery mechanisms.

These controlled variables reflect real-world operating scenarios and were combined to create complex, high-variability environments. The system responded to these stimuli with distributed inference, blockchain-based validation, and autonomous actuation, demonstrating its capacity for adaptive, decentralized urban governance.

To ensure the realism and applicability of the architecture, the simulation scenario was scaled to reflect a deployable structure consistent with the capacities of a typical university laboratory setting. The infrastructure included a total of 36 physical sensors, distributed across 12 terminal nodes, each composed of an Arduino UNO R3 connected to a DHT22, LDR, and PIR sensor. These nodes were coordinated by 6 Raspberry Pi 4 units functioning as edge processors, which performed real-time inference and local decision-making. Each processing unit managed two terminal nodes and controlled actuation components such as servomotors, solid-state relays, and LED arrays.

3.6.2 Experimental rounds and evaluation scenarios

The experimental validation was conducted across 30 physical test rounds, distributed over five defined scenarios. Each round lasted 20 minutes and included real-time data acquisition, AI-based inference, blockchain transaction submission, and physical execution of actions through actuators. All actions and contextual parameters were automatically logged for subsequent analysis of performance, latency, decision consistency, and system stability. Table 3 summarizes the configuration of each scenario evaluated during the testing phase:

Each scenario was explicitly designed to assess different operational dimensions of the system. Scenario 1 represents baseline behavior in static conditions. Scenarios 2 and 3 evaluate responsiveness and energy efficiency under increased activity. Scenarios 4 and 5 introduce anomalous and unpredictable events to measure the system's robustness and adaptability in volatile environments. Throughout the experimental rounds, metrics such as inference execution time, end-to-end latency of blockchain validation (on the Sepolia testnet), decision acceptance/rejection rates, and behavioral stability under fluctuating loads were continuously monitored.

3.7 Experimental procedure

The experimental procedure establishes the operational sequence used to validate the behavior of the decentralized urban governance system. This procedure is designed to ensure reproducibility, traceability, and complete control of the conditions under which the tests are executed. This allows for evaluating the performance of the artificial intelligence model, the efficiency of the blockchain validation process, and the consistency of the decisions generated.

The system is initialized, ensuring the integrity of each architecture node. This process includes turning on the Arduino microcontrollers, deploying the operating system on the Raspberry Pi, and activating the inference model in TensorFlow Lite format previously loaded on the edge nodes. A connection is established with the private local Wi-Fi network, specifically configured for the simulation, and the Web3 client is launched for direct communication with the RPC node on the Ethereum testnet (Ghifari et al., 2021). In parallel, the smart contract deployed on the blockchain is activated and configured to accept only transactions from nodes authenticated using ECDSA signatures. Once the system is initialized, the physical test environment is configured by activating controlled simulation mechanisms. Light levels are adjusted using programmable light sources that simulate day-night cycles; simulated pedestrian traffic is generated by moving past PIR sensors; and changes in climatic variables are induced by localized heaters and steam generators that alter the temperature and humidity in the sensory environment.

During each experimental round, active sensors capture environmental conditions in real time. Data acquired by the DHT22, LDR, and PIR sensors is transmitted via the MQTT protocol to the local processing nodes, where it is grouped and normalized using the parameters defined in the calibration phase. The generated input vector is encapsulated in a standard structure that includes the current state of the environment and is fed directly to the embedded inference model.

The AI model performs inference on the input vector and selects the most appropriate action from the possible defined classes. This action, sensory context, and timestamp are packaged as a structured transaction, digitally signed, and transmitted via Web3.py to the deployed smart contract. The contract verifies the transaction's validity according to the policies encoded in its internal logic and, if the decision meets the established criteria, records the transaction in a new block on the Ethereum network.

Subsequently, the physical subsystem associated with the decision is activated, which may include activating lighting, changing the state of a simulated traffic light, or modifying the energy operating mode. Each executed action is bidirectionally associated with the validated and recorded decision, allowing full traceability from the sensory context to the final urban action. At the end of each round, the system automatically records metrics that evaluate its performance. These metrics include total response time (from sensory capture to blockchain validation), cumulative energy consumption per node (measured using INA219 modules), the frequency of decisions rejected for violating contract policies, and contextual consistency, measured as the percentage of decisions that conform to the actual state of the environment versus a predefined reference.

These metrics are stored in structured files for later analysis and allow for direct comparisons between scenarios, system stability assessment under dynamic conditions, and measurement of each component's impact on the full cycle of distributed urban governance.

3.8 Performance evaluation metrics

The system is evaluated using a suite of quantitative metrics that assess its behavior in terms of responsiveness, decision reliability, energy efficiency, and resilience under adverse or uncertain conditions. These metrics are applied systematically during each simulation round, with data collected automatically via embedded scripts in each node.

3.8.1 System responsiveness and energy efficiency

The total system response time T_{resp} quantifies the elapsed interval between the sensory detection of an event and the execution of the corresponding action in the physical environment. It includes the inference delay, blockchain transaction validation time, and actuator activation latency. Synchronization is ensured using NTP across all nodes, and the response time is calculated as:

$$T_{\text{resp}} = T_{\text{ejec}} - T_{\text{event}} \tag{6}$$

In parallel, the energy consumption of each hardware component is measured using INA219 sensors. Current values are sampled per module category–sensor, microcontroller, processing unit, actuator, and accumulated in milliamp-hours (mAh). The total consumption is defined as:

$$E_{\text{total}} = \sum_{i=1}^{n} I_i \cdot \Delta t_i \tag{7}$$

These two indicators allow evaluating the latency-performance trade-off and the energetic footprint of blockchain-based urban automation under realistic operating loads.

3.8.2 Decision quality and blockchain validation success

Two complementary metrics are used to assess decision quality. The first is the validation success rate in blockchain (TVE), which captures the percentage of AI-inferred decisions that satisfy smart contract constraints and are correctly confirmed on-chain:

$$\text{TVE} = \frac{N_{\text{val}}}{N_{\text{tot}}} \cdot 100 \tag{8}$$

The second is the decision efficiency of the AI model (η_{IA}), which reflects the proportion of model outputs that match an ideal reference policy manually established for each context:

$$\eta_{\text{IA}} = \frac{N_{\text{correct}}}{N_{\text{evaluated}}} \tag{9}$$

These metrics quantify both the technical correctness of the inferences and the alignment of decisions with desirable urban behavior, providing a robust measure of functional performance.

3.8.3 Resilience and operational stability

The system's resilience is evaluated by injecting controlled disturbances, such as partial node disconnections, network congestion, and increased transaction latency. Two indicators are defined:

- **Functional recovery rate**: the proportion of disrupted cycles that the system recovers from autonomously without requiring manual intervention.
- **Tolerated time deviation**: the maximum temporal margin that can be added to T_{resp} while preserving functional coherence and system continuity.

These indicators characterize the system's fault tolerance and its ability to maintain autonomy and operational integrity in real-world, unpredictable environments typical of decentralized smart cities.

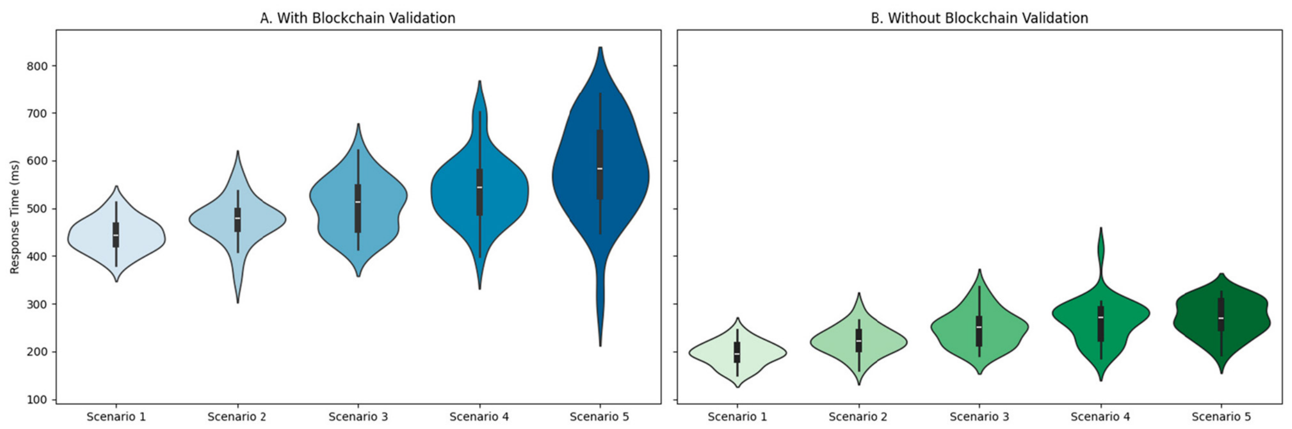

FIGURE 5
Response time distribution by scenario under different operating conditions; Graph **(A)**: response times with blockchain validation; Graph **(B)**: response times without blockchain validation.

TABLE 4 Average and maximum response times per scenario, with and without blockchain validation.

Scenario	Avg. time with blockchain (ms)	Max. time with blockchain (ms)	Avg. time without blockchain (ms)	Max. time without blockchain (ms)
Scenario 1	444.23	513.17	195.87	246.31
Scenario 2	474.70	574.31	222.96	293.90
Scenario 3	505.75	621.95	248.92	335.21
Scenario 4	539.46	702.03	264.85	414.11
Scenario 5	582.67	740.64	271.24	326.46

4 Results

4.1 System response time analysis

Response time is a critical metric in urban decision-making systems as it determines the system's ability to react promptly to contextual events. To evaluate this metric, 30 experimental rounds are conducted for each of the five scenarios defined in section 3.6, under two configurations: with and without blockchain validation. In each round, the time elapsed between the detection of the sensory event and the final execution of the corresponding action is measured, including all intermediate steps of the system.

Figure 5 presents the response time distribution for the five scenarios, differentiating between operating conditions with and without blockchain validation. Graph A shows that when smart contract validation is integrated, response time increases significantly, particularly in scenarios 3, 4, and 5, which involve adverse weather conditions or intense sensory activity. This behavior reflects the time penalty associated with sending, signing, verifying, and recording transactions on the decentralized network and the variable load of the testnet used.

In contrast, Graph B shows response times when the blockchain validation component is removed, allowing actions to be executed immediately after local inference. The curves show greater consistency between scenarios, with less dispersion and lower latency peaks. However, variability is still attributable to sensory complexity or the number of concurrent events. A direct comparison between the two graphs allows us to appreciate the structural impact of the distributed architecture on operational latency.

Table 4 presents the average and maximum response time values for each scenario. In Scenario 1, characterized by low sensory activity and stable conditions, the average time with blockchain is 444.23 ms, while without blockchain, it decreases to 195.87 ms. This difference progressively increases in more complex scenarios. For example, in Scenario 5, which simultaneously integrates weather changes, random pedestrian flow, and light alternation, the average with blockchain reaches 582.67 ms, with a maximum recorded time of 740.64 ms, compared to an average of 271.24 ms without distributed validation.

This behavior establishes that the system maintains acceptable latency performance, even under conditions of high sensor load and distributed logic. However, it also demonstrates that the time cost of decentralized governance is nonlinear and must be carefully considered in time-sensitive urban applications, such as traffic management or emergency response. Maximum tolerable latency limits should be defined according to the application domain and could be optimized through parallel validation mechanisms or more lightweight contracts.

4.2 Energy consumption assessment

The system's cumulative energy consumption represents a key criterion in designing decentralized urban solutions, especially

TABLE 5 Cumulative energy consumption by component and scenario (mAh).

Scenario	Sensors (mAh)	Processing nodes (mAh)	Actuators (mAh)
Scenario 1	11.94	30.96	16.92
Scenario 2	11.99	38.16	12.75
Scenario 3	12.10	41.57	18.08
Scenario 4	15.15	44.72	18.71
Scenario 5	14.86	47.74	13.94

when projected for implementation in distributed nodes with limited resources and restricted energy autonomy. The architecture includes low-power sensors, microcontrollers for data acquisition, embedded processing nodes (Raspberry Pi 4), and physical actuators that simulate traffic lights, lighting, and pedestrian control. Each component is individually evaluated using INA219 measurement modules, allowing continuous capture of current and voltage throughout the execution of each scenario.

Table 5 complements this analysis by providing the specific values by component and scenario. In Scenario 1, which simulates stable daytime conditions with low interaction, the total consumption of the processing nodes is 30.96 mAh. However, in Scenario 5, characterized by dynamic conditions and simultaneous sensory events, this value increases to 47.74 mAh, representing a 54.2% increase. This quantitative difference reflects the computational burden imposed by the need to make more frequent decisions, evaluate them, and validate them using smart contracts under distributed logic.

In the case of actuators, consumption variability is also significant and is directly related to the number of actions executed per scenario. For example, in Scenario 3, which simulates intense pedestrian flow and abrupt changes in weather conditions, the highest actuator consumption is recorded (18.08 mAh), attributable to the recurring activation of signaling and flow control mechanisms. In contrast, Scenario 2, with nighttime conditions and moderate flow, shows lower consumption (12.75 mAh), consistent with a lower frequency of state changes in the environment. The results identify the processing nodes as the dominant component in the system's energy profile. While these nodes are essential for ensuring real-time decisions and performing local validations without relying on external infrastructure, their energy costs suggest optimizing strategies. These include model compression, using lighter inference architectures such as MobileNet or TinyML, implementing lazy inference schemes, and reducing the blockchain validation rate for non-critical events.

Furthermore, this energy behavior raises key questions about the system's scalability in real-world environments, where hundreds or thousands of nodes must operate continuously. Decentralized energy management, adaptive contract design, and event prioritization based on their urban impact could constitute future improvements that enhance the proposed system's operational sustainability.

4.3 Successful validation rate in blockchain

The blockchain validation rate represents a critical metric for evaluating the trustworthiness of the proposed distributed governance system. This metric quantifies the percentage of decisions generated by the artificial intelligence module that are correctly accepted, signed, and recorded on the Ethereum testnet network using smart contracts. The validation process includes transaction generation, digital signature, submission through a local Web3 client, and evaluation of its conditions within the deployed contract.

During each experimental iteration, the system executes between 10 and 15 validated decisions in real time, totaling more than 150 transactions per scenario. Internal testnet event logs are used to calculate the proportion of successful transactions compared to the total submitted, allowing for the identification of failure or rejection patterns.

Figure 6 shows, in Graph A, the distribution of successful validation rates across the five simulated scenarios, using a violin plot to reflect the dispersion and density of the data obtained per round. Under stable conditions (Scenario 1), the validation rate remains high and stable, with average values close to 98%. However, as the complexity of the environment increases, Scenarios 4 and 5, where multiple parallel events and climate variability are introduced–a progressive reduction in the success rate is observed, reaching averages close to 85% with high dispersion. This decline can be attributed to the increased frequency of operational errors and the accumulated latency in the validation network.

Graph B complements this analysis with a swarm plot that categorizes validation errors according to their cause: signature errors, timeouts, violations of the contract's internal logic, and decisions that do not meet the conditions established in the smart contract. This representation visualizes how each type of error is distributed across scenarios. In less demanding scenarios, the predominant errors are invalid or poorly formatted signatures, possibly due to minor asynchronies or node connection losses. In contrast, in high-sensory-load scenarios (such as Scenario 5), timeout errors become the dominant cause, followed by logic violations, where the generated decisions do not meet the thresholds defined in the deployed smart contract.

Table 6 provides a quantitative breakdown of these causes. In Scenario 1, signature errors account for 34.8% of failures, while errors due to contract conditions barely reach 17.4%. This proportion is partially reversed in Scenario 5, where timeout errors exceed 33.3% of the total, and contract-related errors remain a recurring cause due to decisions made under conditions unforeseen by the contract logic. This behavior suggests contractual logic must adapt to dynamic contexts through automatic updates or more flexible governance schemes.

4.4 AI model accuracy in decision making

The accuracy assessment of the implemented AI model focuses on determining its ability to infer correct decisions based on the sensory conditions present in each scenario. The system operates

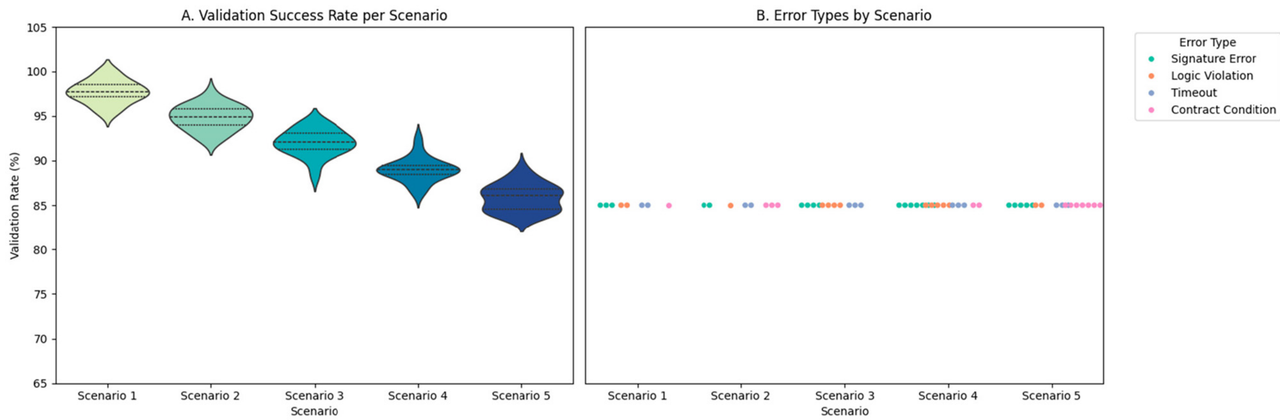

FIGURE 6
Validation process performance and error distribution by scenario; Graph **(A)**: distribution of the blockchain validation rate by scenario; Graph **(B)**: classification of validation errors by cause.

TABLE 6 Percentage distribution of validation errors by scenario and type of cause.

Scenario	Signature error (%)	Timeout (%)	Logic violation (%)	Contract condition (%)
Scenario 1	34.8	26.1	21.7	17.4
Scenario 2	30.0	25.0	30.0	15.0
Scenario 3	27.6	24.1	31.0	17.2
Scenario 4	25.0	29.2	27.1	18.7
Scenario 5	22.2	33.3	25.9	18.6

using a multi-category classifier, pre-trained with simulated data and tuned with contextual feedback to select one of several possible actions: turning the lights on or off, changing the traffic light, enabling pedestrian access, and activating energy-saving mode.

For this analysis, a sequence of ideal decisions defined by experts is considered a reference, representing the optimal response based on the environmental stimuli present. Each simulation round produces between 10 and 15 decisions, totaling more than 400 decisions evaluated for the entire system. Comparing inferred and ideal choices allows for calculating overall accuracy metrics and evaluating error trends by class.

Figure 7 presents two complementary visualizations. Graph A shows the evolution of the AI model's accuracy in the five scenarios, represented as a smoothed curve with variability bands. In Scenario 1, characterized by stable conditions and low sensory load, the model achieves an average accuracy of 95% with minimal deviation. However, as the complexity of the environment increases, for example, in Scenarios 4 and 5, which include multiple event sources and environmental variability, the model's accuracy progressively decreases, reaching an average of 81% with a dispersion more significant than 3.5%. This trend suggests a direct sensitivity of the model to the simultaneous occurrence of conflicting signals, such as the presence of pedestrians in bright light or atypical weather conditions.

Graph B represents a radar chart comparing accuracy by decision type, contrasting the distribution of ideal decisions versus those inferred by the model. This chart lets us identify which decision categories show the most significant divergence under real-world conditions. It is evident that decisions such as "Light On" and "Energy Save" maintain high consistency between inference and expectations, with deviations of less than 3%. In contrast, decisions such as "Change Signal" and "Pedestrian Pass" present a more significant gap, with differences of up to 7%. This indicates that these classes present more significant semantic ambiguity or greater dependence on the environmental context, affecting the inference quality.

Table 7 presents the consolidated confusion matrix for all scenarios. The most significant number of errors is concentrated in transitions between functionally similar classes, such as "Change Signal" and "Pedestrian Pass," which can be attributed to decisions contextualized in irregular pedestrian flows or shared crossing areas. The dominant diagonal indicates correct classification in most decisions, which validates the model as a basis for the proposed system. However, it also points out the need for fine-tuning in differentiating specific sensory patterns.

4.5 System resilience assessment

System resilience is assessed based on its ability to maintain essential operations and recover functionality following partial or total failures of some key components. To achieve this, five types of control failures are induced: sensor disconnection, network outage, inference node freeze, blockchain validation delay, and localized power fluctuation. Each type of failure is applied during multiple experimental rounds, replicating critical operating conditions that may occur in real-world distributed urban environments.

Figure 8 presents two complementary visualizations of system behavior. Graph A illustrates the functional recovery time by failure type. The system rapidly responds to minor failures, such as temporary sensor disconnection or minor network disruptions, with average recovery times of 4.5 and 7.2 s, respectively. However, more structural failures, such as the failure of the primary

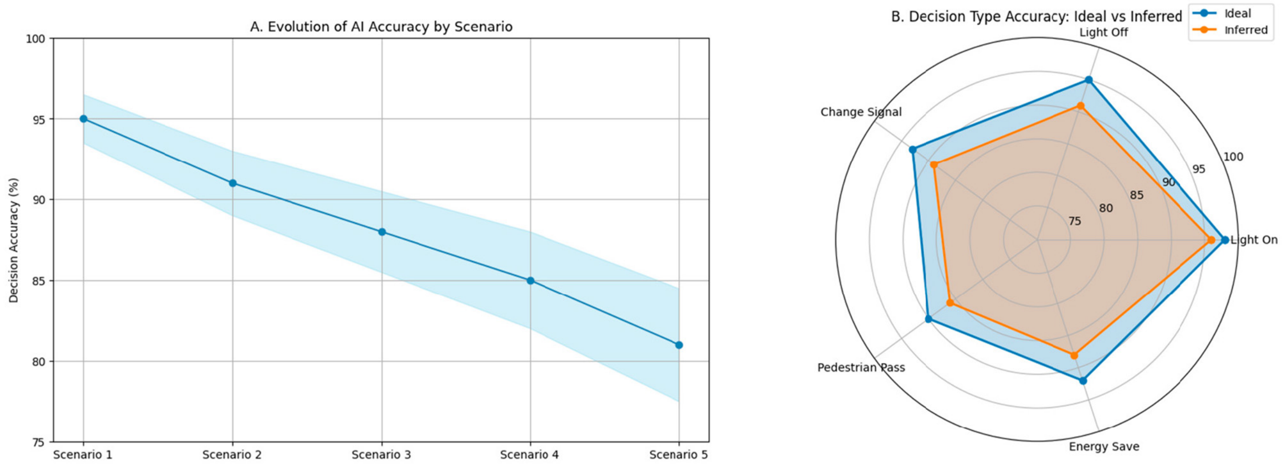

FIGURE 7
Assessment of the accuracy of the artificial intelligence model in simulated urban environments; Graph **(A)**: evolution of AI model accuracy by scenario; Graph **(B)**: comparison between ideal and inferred decisions by action type.

TABLE 7 Consolidated confusion matrix for AI model decisions.

A	B	C	D	E	F
Light on	42	3	1	0	1
Light off	2	40	3	1	2
Change signal	1	2	39	3	3
Pedestrian pass	0	1	4	37	3
Energy save	1	2	2	3	39

A: Ideal Decision \ Inferred Decision; **B**: Light On. **C**: Light Off. **D**: Change Signal. **E**: Pedestrian Pass. **F**: Energy Save.

processing node, require context transfer and reactivation of adaptive logic from a secondary node, which increases recovery time to an average of 9.1 s. This behavior demonstrates that distributed architecture, while fault-tolerant, presents a time penalty directly proportional to the criticality of the affected node.

Graph B shows the variability in model accuracy during the recovery period. The boxplot data shows that decision drift is lower during sensor or blockchain failures, where compensation mechanisms estimate missing values or apply lazy validation. In these cases, accuracy remains around 80% with controlled dispersion. However, when a failure occurs in the inference node, accuracy drops to values below 75% during the first few seconds of the recovery process, with increased dispersion indicating a temporary loss of coherence in decision-making. This behavior is consistent with the need for internal state reconstruction and data synchronization during inter-node transfer.

Table 8 details the compensatory actions automatically activated by the system for each type of failure. When sensors are disconnected, an estimation module based on moving averages and fuzzy logic is activated, allowing degraded functional decisions to be maintained without interruption. In situations of connectivity loss, the system implements a temporary local validation mechanism and a scheduled retry using adaptive timers. The failure of an inference node triggers a logical fallback protocol, where a secondary node assumes the functions of the primary node using a lightweight version of the model with local state persistence and an interrupted decision buffer. In blockchain delays, the validation logic is temporarily relaxed, allowing execution under optimistic logic, and validation is rescheduled with lazy signing. Furthermore, the system restarts safely using previously persisted information in scenarios with power fluctuations, ensuring continuity without loss of operational context.

The results show that the system maintains structural and operational stability under controlled adverse conditions, thanks to a distributed architecture, internal backup mechanisms, and functional persistence. Combining techniques such as secondary inference, optimistic logic, lazy validation, and context persistence allows the system to recover in acceptable timescales without compromising the intelligent governance of the simulated urban infrastructure. The recovery metrics obtained and the accuracy during stabilization confirm resilient behavior aligned with the autonomy, adaptability, and operational continuity principles required in decentralized urban scenarios.

4.6 Integrated multimetric analysis by scenario

The proposed system's integrated evaluation allows us to identify how its main performance dimensions interact under progressively more demanding simulated operating conditions. Five key metrics are considered: mean response time, AI model accuracy, cumulative energy consumption, successful blockchain validation rate, and functional failure recovery time. This consolidation allows us to establish structural behavior patterns, analyze technical trade-offs, and assess the system's robustness from a systemic perspective.

Table 9 summarizes the quantitative values obtained in each of the five scenarios. The mean response time increases from 230 ms

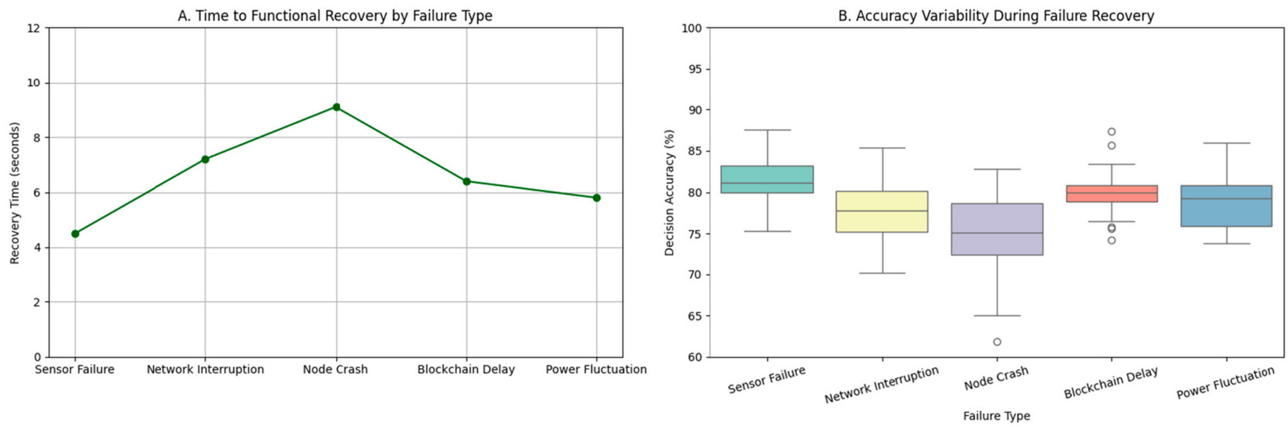

FIGURE 8
Assessment of the system's operational resilience to controlled failures; Graph **(A)**: functional recovery time after different types of failure; Graph **(B)**: variability in the precision of decisions during the recovery process.

TABLE 8 Compensatory actions of the system in the event of induced failures.

Induced failure type	Direct impact	Compensation mechanism enabled	System post-failure status
Sensor failure	Temporary loss of data input	Estimation based on moving averages and local fuzzy logic	Partial resume with degrade
Network interruption	Disruption in distributed validation	Temporal local validation + adaptive timer retry	Successful retry afterward
Node crash	Local inference node failure	Decision transfer to neighboring node with logical fallback	Recovery after 9 s
Blockchain delay	Delay in decision validation	Execution under optimistic logic + lazy validation	Retroactive acknowledgment logged
Power fluctuation	Partial restart of the control system	Safe startup with state persistence and decision buffer	Safe mode enabled briefly

TABLE 9 Multimetric summary of the system by simulation scenario.

Scenario	Average response time (ms)	AI model accuracy (%)	Total energy consumption (mAh)	Successful validation rate (%)	Average recovery time (s)
Scenario 1	230	95	61	97	4.5
Scenario 2	280	91	72	93	7.2
Scenario 3	310	88	77	89	9.1
Scenario 4	335	85	80	86	8.5
Scenario 5	360	81	86	83	10.0

in Scenario 1 to 360 ms in Scenario 5. This increase is attributable to the more significant number of simultaneous sensory events, the complexity of the inferences required, and the volume of validation operations executed on the blockchain network. This increase in latency directly impacts the system's ability to maintain consistent real-time decisions under an intensive load.

The accuracy of the AI model progressively decreases, dropping from 95% in Scenario 1 to 81% in Scenario 5. This deterioration in the model's inference capacity is associated with the increasing sensory ambiguity and operational overload faced by the processing nodes, which reduces their efficiency in discriminating complex patterns under noise, congestion, or simultaneous events.

Total energy consumption also shows an upward trend, with a 41% increase between Scenario 1 (61 mAh) and Scenario 5 (86 mAh). This increase is consistent with the increasing computational load, the more frequent activation of actuating components such as traffic lights and lighting systems, and the increased use of secondary nodes in functional recovery processes. A controlled decline is observed regarding the successful validation rate on the blockchain, ranging from 97% in Scenario 1 to 83% in Scenario 5. This reduction reflects the progressive saturation of the validation network, increased decisions generated under extreme conditions, and the appearance of errors due to timeouts or breaches of contractual conditions. However, the system maintains

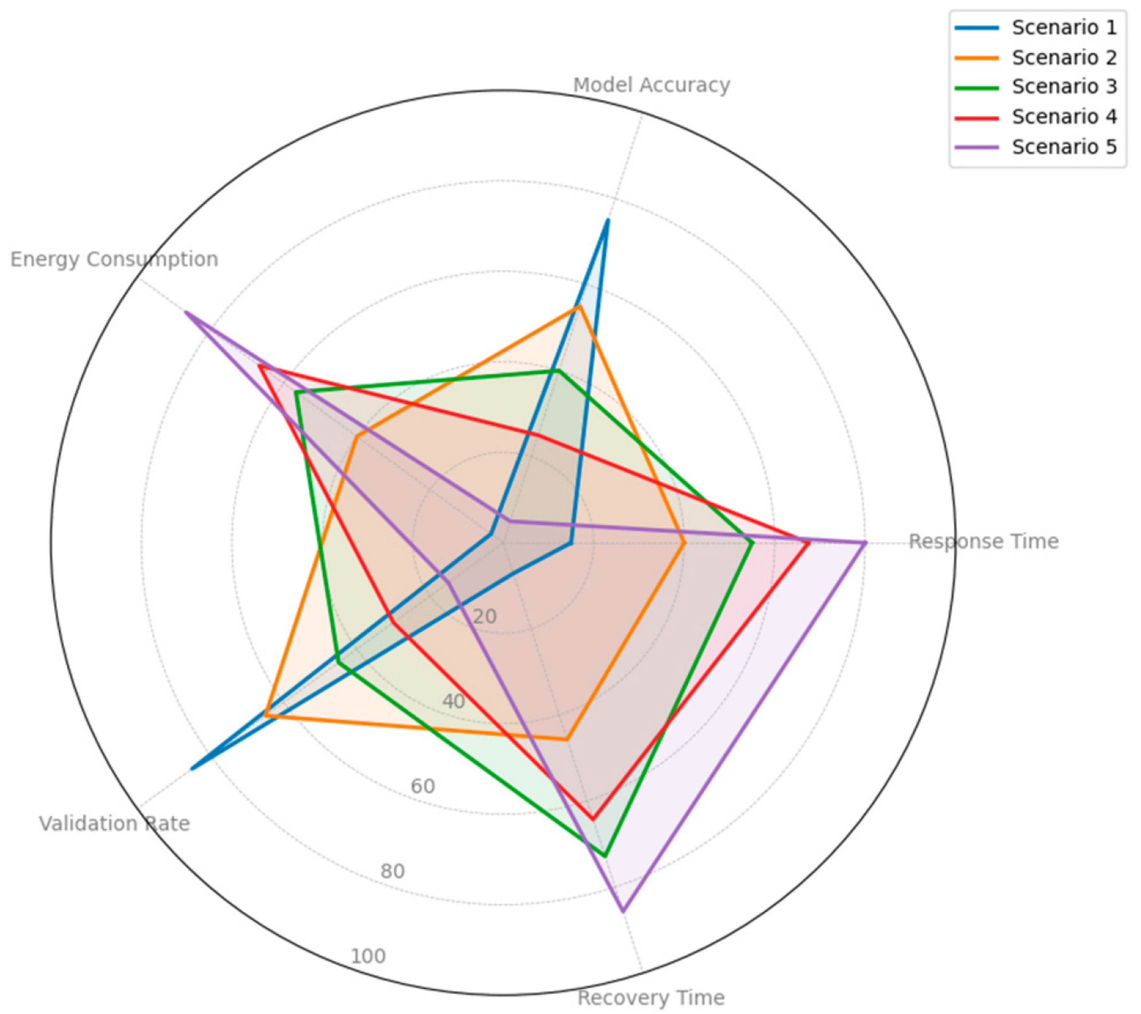

FIGURE 9
Integrated multimetric performance by scenario.

an operational rate above 80% in all cases, demonstrating adequate robustness in its distributed governance logic.

The last integrated metric is the mean time to functional recovery from induced failures. This metric increases nonlinearly from 4.5 s in Scenario 1 to 10.0 s in Scenario 5. The system's ability to recover the whole operation is affected not only by the complexity of the failure but also by the density of parallel events that interfere with compensation mechanisms, such as heuristic estimation or the activation of redundant nodes.

Figure 9 presents these metrics in an integrated form on a radar chart, where each axis represents a normalized metric. The graphical analysis reveals a progressive pattern of overall degradation as the scenarios approach complex urban operating conditions. The areas delimited by each scenario allow us to visualize how individual strengths (e.g., validation or recovery) are offset by increasing latency, energy consumption, and accuracy deficits. The radar shows that the system maintains functional equilibrium in the first three scenarios, while in scenarios 4 and 5, the tensions between efficiency and resilience become more pronounced.

Multimetric analysis confirms that the system responds appropriately to dynamic conditions and can maintain stable performance under various forms of technical stress. The interaction between distributed components, adaptive energy management, and blockchain-based governance provides a robust functional framework for decentralized urban scenarios, even when faced with high-load events and system failures.

4.7 Comparison with existing proposals

Table 10 compares the proposal developed in this study with four representative works identified in the literature review. This comparison is based on five key axes: the level of technological integration between AI, IoT, and blockchain; the validation environment; the presence and type of resilience mechanisms; and the governance structure implemented.

TABLE 10 Technical comparison between the developed proposal and identified studies.

Reference	Evaluation approach	Key technologies	Technical description of the model
Tan and Taeihagh (2020)	Multi-criteria evaluation with an ethical focus	AI, ethical indicators	The conceptual framework integrates transparency, inclusion, and algorithmic fairness indicators.
Ghahremani-Nahr et al. (2022)	Supply chain simulation	IoT, blockchain	Conceptual framework for sustainable management based on IoT and blockchain without physical assessment.
Cai and Hong (2024)	Hybrid architecture evaluation	AI, blockchain, IoT	Functional analysis of AI-BC architecture in smart cities using NS3 simulation.
This Work	Physical-simulated validation in an urban environment	Distributed AI, IoT, blockchain	Operational architecture deployed in a controlled environment with traceability and comprehensive assessment.

Regarding technological integration, most of the reviewed works address only partial components of the architecture required for fully distributed urban governance, the study by Ghahremani-Nahr et al. (2022), for example, proposes a conceptual framework that integrates IoT and blockchain in sustainable supply chain management but does not consider distributed inference architecture or active decision validation through smart contracts. Meanwhile, the work by Tan and Taeihagh (2020) addresses governance from a social and ethical perspective but without technological implementation. In contrast, the proposal developed here integrates the three components operationally: distributed physical sensors, artificial intelligence models embedded in the nodes, and computational validation through smart contracts deployed on a functioning blockchain network. This technical integration has been tested under physical and simulated conditions, allowing for its real-world quantitative evaluation.

Regarding the validation environment, this proposal is the only one that combines real physical infrastructure (Raspberry Pi, environmental sensors, actuators) with executable blockchain validation (Ethereum testnet). At the same time, other works, such as that of Ghahremani-Nahr et al. (2022), are limited to conceptual or simulated environments. This hybrid environment, which combines physical, logical, and distributed environments, allows for evaluating real-world technical behaviors such as network latencies, energy variability, node synchronization, and induced failures, which cannot be analyzed in theoretical frameworks or simulations without operational infrastructure.

Regarding resilience, none of the studies compared details of autonomous functional recovery mechanisms. Existing proposals assume ideal conditions or tolerance through structural redundancy without describing how the system responds to node loss, sensor failures, or network link degradation. In contrast, the system developed in this study implements an active resilience model, which includes logical fallback, heuristic estimation in the absence of data, state persistence, and lazy validation in smart contracts. This approach is tested under adverse conditions, allowing for measurement of recovery times, decision stability, and the functional continuity of the system.

From a governance perspective, the reviewed works present significant limitations. Tan and Taeihagh (2020) identify the need for institutional frameworks that prioritize transparency, algorithmic fairness, and citizen participation but do not implement a technical architecture that materializes these principles. Similarly, Ghahremani-Nahr et al. (2022) introduce the notion of traceability and reliability in distributed environments but without programmable validation mechanisms or real-time auditing. In contrast, the present proposal implements a functionally and computationally verifiable distributed governance, where each decision generated by the AI system must meet logical conditions embedded in smart contracts. These decisions are validated, recorded, and audited, allowing for traceability of simulated urban events such as lighting changes or traffic light adaptations. This level of control is not present in any of the compared proposals.

5 Discussion

The results demonstrate the technical and functional feasibility of implementing a decentralized urban governance system through the coordinated integration of physical IoT devices, distributed embedded artificial intelligence, and automatic validation via smart contracts on blockchain platforms. This approach, experimentally validated in a controlled environment with incrementally complex urban simulations, offers a functional advancement over existing proposals in the literature. For instance, Ghahremani-Nahr et al. (2022) propose conceptual frameworks involving IoT and blockchain but lack operational validation or mechanisms for autonomous inference. Similarly, Tan and Taeihagh (2020) emphasize ethical and transparent governance but do not advance toward a technically auditable implementation. Unlike these theoretical or partially realized models, the system developed herein integrates a physically validated, fully distributed architecture with embedded inference, blockchain-based decision validation, and operational resilience (Ambrose and Siddiki, 2024), constituting a replicable framework for real-world deployment.

Comparatively, while other national-scale implementations such as City Brain in China (Xu et al., 2024) demonstrate anticipatory governance capabilities through centralized AI systems, they rely heavily on large-scale integration and corporate-state coordination. In contrast, this work proposes a decentralized alternative grounded in edge intelligence and blockchain verification, enabling similar capabilities such as rapid response and transparency under a bottom-up, replicable approach. This highlights the potential of modular systems to complement large urban AI platforms, thereby democratizing access to intelligent governance mechanisms at smaller scales.

From a methodological standpoint, the hybrid environment used, combining physical sensors, edge inference nodes, and a blockchain testnet, enables the simultaneous evaluation of latency, energy consumption, and system fault tolerance (Tavakoli et al., 2024; Razzaq, 2024). The experimental design supports real-time data acquisition, inference, smart contract execution, and physical actuation under controlled yet diverse conditions. This setup has made it possible to compute critical metrics such as response time, transaction validation success rate, and resilience to node or network failures. The inclusion of radar charts, boxplots, and confusion matrices enhances the interpretability of these metrics and highlights the interplay between inference efficiency, contract execution success, and system overhead.

Across all evaluated scenarios, the system exhibits robust and consistent performance. The AI model achieves a decision accuracy exceeding 80% even under adverse or dynamic conditions, while blockchain validation rates remain above 83%, confirming the feasibility of executing computational governance in distributed environments. The system also demonstrates acceptable recovery behavior, maintaining operational continuity after induced faults such as sensor failure or communication delay, with recovery times ranging from 4.5 to 10 s. These findings validate the proposed architecture's resilience and its suitability for non-deterministic urban environments.

One of the most distinctive contributions of this system is its ability to enforce verifiable and auditable governance via blockchain. Each decision inferred by the AI component is submitted as a digitally signed transaction, validated by a smart contract, and immutably recorded on-chain. This ensures operational transparency and creates a tamper-proof audit trail–critical in urban settings where automated decisions can directly influence public safety, mobility, or energy distribution. Moreover, the contract logic serves as a safeguard by blocking invalid or inconsistent decisions and triggering alerts for post-event analysis. This automated validation mechanism, absent in most current proposals, aligns with contemporary demands for explainable and trustworthy AI systems.

Despite these strengths, several limitations must be acknowledged. The experimental setup, while physically implemented, remains confined to a controlled environment with a limited number of nodes. Consequently, absolute latency or energy values may not scale linearly in larger urban deployments. In addition, although the system uses a functional Ethereum testnet, it does not simulate mainnet congestion or gas fee dynamics, which could impact its feasibility in cost-sensitive real-world applications. Another constraint lies in the simplified AI model used; processing limitations on edge nodes (e.g., Raspberry Pi) required a compact architecture, potentially reducing generalization capacity under highly variable conditions. The system also assumes a minimum level of network stability and clock synchronization, which, although partially mitigated, may not always be guaranteed in urban scenarios. These constraints underline the importance of careful calibration and incremental deployment in future real-world implementations.

From the perspective of urban management, the system facilitates distributed decision-making by decentralizing control logic and guaranteeing traceability of autonomous actions. The architecture enables local, adaptive responses, such as dynamic signaling or infrastructure actuation, without relying on centralized infrastructure or manual intervention. This design offers tangible benefits for municipalities aiming to enhance transparency, reduce latency, and mitigate the risk of single points of failure in governance systems. It is especially applicable to small and medium-sized cities or digitally evolving urban districts, where lightweight, modular, and fault-tolerant systems can deliver scalable impact with minimal deployment cost. However, successful adoption may require minimal sensing infrastructure and institutional readiness to adopt decentralized governance paradigms.

Beyond the experimental testbed and small- to medium-scale applicability, the question of scalability requires explicit consideration. The proposed framework could be extended to large metropolitan areas by incrementally increasing the number of edge nodes and implementing hierarchical coordination among clusters of IoT devices. However, such scaling introduces non-linear complexities, including transaction throughput limitations in blockchain networks, increased latency in consensus protocols, and heterogeneous infrastructure integration. From a socio-economic perspective, adoption pathways will also differ. Highly digitized urban contexts may leverage existing broadband connectivity and institutional readiness, whereas cities in developing regions may face barriers related to cost, interoperability, and policy frameworks. These contrasts highlight that while the architecture is technically replicable, its deployment at larger scales will depend on both infrastructural maturity and governance capacity, making scalability multidimensional rather than a purely technical challenge.

These four analytical axes can also be mapped directly onto the multi-level conceptual framework introduced in Materials and Methods. The technological integration of IoT, edge AI, and blockchain corresponds to the technical layer of the framework, ensuring interoperability and operational cohesion. The validation environment, based on datasets and controlled testbeds, aligns with the evaluative layer, where performance and governance properties are systematically assessed. Resilience, expressed through redundancy, recovery times, and fault tolerance, acts as a cross-cutting property spanning both the technical and evaluative levels. Finally, traceable governance, implemented through blockchain validation and immutable audit trails, corresponds to the managerial layer of the framework, where autonomous decisions are verified and institutional accountability is reinforced. Establishing this alignment clarifies how the proposed architecture operationalizes conceptual principles across distinct but interconnected levels of analysis.

6 Conclusions and future work

This work presents a fully functional and experimentally validated architecture for decentralized urban governance that integrates IoT sensing, embedded artificial intelligence, and blockchain-based smart contracts. The system demonstrates the feasibility of executing autonomous decisions in real time, under dynamic conditions, and with verifiable validation mechanisms, all within a resource-constrained and distributed physical environment.

Unlike prior conceptual or simulation-only approaches, this implementation achieves tangible interoperability between contextual inference and computational governance. Each decision generated by the AI layer is evaluated and validated on-chain, ensuring traceability, consistency, and auditability, key features for future urban infrastructures.

The results confirm that the proposed architecture maintains acceptable levels of validation even under complex simulated urban conditions. However, critical issues regarding smart contract robustness and network fault tolerance have been identified and must be addressed to ensure the system's scalability and resilience.

The architecture's modular and energy-efficient design enables local processing without relying on central servers, making it suitable for small and medium-scale urban deployments. Furthermore, the system exhibits functional resilience to faults and latency, maintaining continuity of operation even in adverse conditions.

This work also advances the current body of research by building on prior proposals and extending their scope. Whereas earlier studies primarily offered conceptual models or simulation-only evaluations, our implementation provides experimental validation under realistic operating conditions. Moreover, it brings together IoT sensing, embedded AI inference, and blockchain-based governance in a unified framework, thereby complementing earlier fragmented approaches where these technologies had been examined separately. Finally, the system translates governance principles, transparency, auditability, and resilience into technically verifiable mechanisms. Taking together, these contributions support the transition from theoretical models to deployable systems and establish a replicable foundation for decentralized urban governance.

As a replicable framework, this work lays the groundwork for auditable, adaptive, and autonomous city management systems. Future efforts will extend its deployment to real urban contexts, incorporate mobile and mesh-based topologies, and enable runtime contract updates and policy learning, enhancing adaptability and long-term scalability.

Data availability statement

The data analyzed in this study is subject to the following licenses/restrictions: the dataset used in this study was generated within a controlled simulation environment that replicates urban governance processes and IoT-based interactions. Due to security considerations and the potential sensitivity of system configuration details, particularly those involving blockchain-based authentication mechanisms, the dataset is not publicly available. However, a sanitized and representative subset can be made available to the corresponding author upon reasonable request, for non-commercial research purposes only. Requests to access these datasets should be directed to william.villegas@udla.edu.ec.

Author contributions

WV-C: Validation, Writing – review & editing, Formal analysis, Conceptualization, Methodology, Supervision, Writing – original draft, Investigation, Visualization. RG: Data curation, Writing – original draft, Visualization, Validation, Software, Formal analysis. JG: Investigation, Software, Writing – original draft, Formal analysis, Methodology, Data curation. PP: Validation, Writing – review & editing, Supervision, Formal analysis.

Funding

The author(s) declare that no financial support was received for the research and/or publication of this article.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Gen AI was used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

Adreani, L., Bellini, P., Colombo, C., Fanfani, M., Nesi, P., Pantaleo, G., et al. (2024). Implementing integrated digital twin modelling and representation into the Snap4City platform for smart city solutions. *Multimed. Tools Appl.* 83, 37121–37146. doi: 10.1007/s11042-023-16838-0

Alasmari, R., and Alhogail, A. A. (2024). Protecting smart-home IoT devices from MQTT attacks: an empirical study of ML-based IDS. *IEEE Access* 12, 25993–26004. doi: 10.1109/ACCESS.2024.3367113

Ambrose, G., and Siddiki, S. (2024). Assessing drivers of sustained engagement in collaborative governance arrangements. *J. Public Adm. Res. Theory* 34, 498–514. doi: 10.1093/jopart/muae005

Cai, T., and Hong, Z. (2024). Exploring the structure of the digital economy through blockchain technology and mitigating adverse environmental effects with the aid of artificial neural networks. *Front. Environ. Sci.* 12:1315812. doi: 10.3389/fenvs.2024.1315812

Daneshvar, H., Niroomand, S., Boyer, O., and Hadi-Vencheh, A. (2023). Designing a hybrid intelligent transportation system for optimization of goods distribution network routing problem. *Decis. Mak. Appl. Manag. Eng.* 6, 907–932. doi: 10.31181/dma622023899

Ghahremani-Nahr, J., Aliahmadi, A., and Nozari, H. (2022). An IoT-based sustainable supply chain framework and blockchain. *Int. J. Innov. Eng.* 2, 12–21. doi: 10.59615/ijie.2.1.12

Ghifari, H. G., Darlis, D., and Hartaman, A. (2021). Pendeteksi golongan darah manusia berbasis tensorflow menggunakan ESP32-CAM. *ELKOMIKA* 9, 359–373. doi: 10.26760/elkomika.v9i2.359

Gracias, J. S., Parnell, G. S., Specking, E., Pohl, E. A., and Buchanan, R. (2023). Smart cities—a structured literature review. *Smart Cities* 6, 1719–1743. doi: 10.3390/smartcities6040080

Grossi, G., and Welinder, O. (2024). Smart cities at the intersection of public governance paradigms for sustainability. *Urban Stud.* 61, 2011–2023. doi: 10.1177/00420980241227807

Kaur, J., Rani, R., and Kalra, N. (2024). Attribute-based access control scheme for secure storage and sharing of EHRs using blockchain and IPFS. *Cluster Comput.* 27, 1–15. doi: 10.1007/s10586-023-04038-2

Kumar, D., Chaudhri, S. N., and Rajput, N. S. (2023). "Air quality prediction and monitoring using machine learning-based forecasting approach," in *Proceedings of the 2023 International Conference on IoT, Communication and Automation Technology (ICICAT 2023)* (Gorakhpur). doi: 10.1109/ICICAT57735.2023.10263594

Lee, W.-M. (2023). "Developing web3 dapps using python," in *Beginning Ethereum Smart Contracts Programming* (Berkeley, CA: Apress). doi: 10.1007/978-1-4842-9271-6_9

Liu, M., Yu, F. R., Teng, Y., Leung, V. C. M., and Song, M. (2019). Performance optimization for blockchain-enabled industrial internet of things (IIoT) systems: a deep reinforcement learning approach. *IEEE Trans. Ind. Inform.* 15, 3559–3570. doi: 10.1109/TII.2019.2897805

Lubis, S., Nurmandi, A., Ahmad, J., Purnomo, E. P., Purwaningsih, T., and Jovita-Olvez, H. D. (2025). Synergizing AI and blockchain: a bibliometric analysis of their potential for transforming e-governance in smart cities. *Front. Sustain. Cities* 7:1553816. doi: 10.3389/frsc.2025.1553816

Lustenberger, M., Wollenschläger, S., and Küng, L. (2025). DAO as digital governance tool for collaborative housing. *Front. Blockchain* 8:1523951. doi: 10.3389/fbloc.2025.1523951

Mahbub, M. (2021). "Blockchain technologies for securing IoT infrastructure: IoT-blockchain architectonics," in *EAI/Springer Innovations in Communication and Computing*. doi: 10.1007/978-3-030-65691-1_13

Miglani, A., and Kumar, N. (2021). Blockchain management and machine learning adaptation for IoT environment in 5G and beyond networks: a systematic review. *Comput. Commun.* 178, 37–63. doi: 10.1016/j.comcom.2021.07.009

Montakhabi, M., Madhusudan, A., Mustafa, M. A., Vanhaverbeke, W., Almirall, E., and Shenja van der, G. (2023). Leveraging blockchain for energy transition in urban contexts. *Big Data Soc.* 10, 1–14. doi: 10.1177/20539517231205503

Montori, F., Zyrianoff, I., Gigli, L., Calvio, A., Venanzi, R., Sindaco, S., et al. (2023). An IoT toolchain architecture for planning, running and managing a complete condition monitoring scenario. *IEEE Access* 11, 6837–6856. doi: 10.1109/ACCESS.2023.3237971

Mrabet, M., and Sliti, M. (2024). Integrating machine learning for the sustainable development of smart cities. *Front. Sustain. Cities* 6:1449404. doi: 10.3389/frsc.2024.1449404

Pawar, S., Panigrahi, N., Jyothi, A. P., Lokhande, M., Godse, D., and Jadhav, D. B. (2023). Evaluation of delay parameter of MQTT protocol. *Int. J. Eng. Trends Technol.* 71, 227–235. doi: 10.14445/22315381/IJETT-V71I3P223

Puthiyidam, J. J., Joseph, S., and Bhushan, B. (2024). Temporal ECDSA: a timestamp and signature mask enabled ECDSA algorithm for IoT client node authentication. *Comput. Commun.* 216, 307–323. doi: 10.1016/j.comcom.2024.01.016

Rasoulzadeh Aghdam, S., Bababei Morad, B., Ghasemzadeh, B., Irani, M., and Huovila, A. (2024). Social smart city research: interconnections between participatory governance, data privacy, artificial intelligence and ethical sustainable development. *Front. Sustain. Cities*. 6:1514040. doi: 10.3389/frsc.2024.1514040

Ravikumar, G., Venkatachalam, K., Masud, M., and Abouhawwash, M. (2022). Cost Efficient scheduling using smart contract cognizant ethereum for IoMT. *Intell. Autom. Soft Comput.* 33, 865–877. doi: 10.32604/iasc.2022.024278

Razzaq, A. (2024). A Web3 secure platform for assessments and educational resources based on blockchain. *Comput. Appl. Eng. Educ.* 32:e22677. doi: 10.1002/cae.22677

Sharma, A., Podoplelova, E., Shapovalov, G., Tselykh, A., and Tselykh, A. (2021). Sustainable smart cities: convergence of artificial intelligence and blockchain. *Sustainability* 13:13076. doi: 10.3390/su132313076

Singh, S. K., Rathore, S., and Park, J. H. (2020). BlockIoTIntelligence: a blockchain-enabled intelligent IoT architecture with artificial intelligence. *Future Gener. Comput. Syst.* 110, 1–14. doi: 10.1016/j.future.2019.09.002

Tan, S. Y., and Taeihagh, A. (2020). Smart city governance in developing countries: a systematic literature review. *Sustainability* 12:899. doi: 10.3390/su12030899

Tavakoli, P., Yitmen, I., Sadri, H., and Taheri, A. (2024). Blockchain-based digital twin data provenance for predictive asset management in building facilities. *Smart Sustain. Built Environ.* 13, 4–21. doi: 10.1108/SASBE-07-2023-0169

Tsuyuguchi, T., and Wang, H. (2024). Quasi-experimental research and spillover effects on Ethereum Merge. *Appl. Econ. Lett.* 32, 1613–1620. doi: 10.1080/13504851.2024.2308587

Xu, Y., Cugurullo, F., Zhang, H., Gaio, A., and Zhang, W. (2024). The emergence of artificial intelligence in anticipatory urban governance: multi-scalar evidence of China's transition to city brains. *J. Urban Technol.* 32, 9–33. doi: 10.1080/10630732.2023.2292823

Zakaie Far, A., Far, M. Z., Gharibzadeh, S., Naeini, H. K., Amini, L., Zangeneh, S., et al. (2025). Artificial intelligence for secured information systems in smart cities: collaborative IoT computing with deep reinforcement learning and blockchain. *arXiv preprint arXiv:2409.16444*. 1–48. doi: 10.48550/arXiv.2409.16444

Zhu, X., Samadh, S. A., and Yu, T. Y. (2018). "Large scale active vehicular crowdsensing," in *IEEE Vehicular Technology Conference* (Chicago, IL). doi: 10.1109/VTCFall.2018.8690631