FuzSemCom: Fuzzy Logic-Based Semantic Communication for Ultra-Low-Power 6G IoT in Smart Agriculture

(Anonymous Submission for IEEE ICC 2026 Review)

***Abstract*—Semantic communication has emerged as a key enabler for 6G task-oriented networks. While deep learning-based approaches dominate the field, they suffer from high complexity and lack of interpretability, limiting their deployment in resource-constrained agricultural IoT. In this paper, we propose FuzSemCom, the first fuzzy logic-based semantic communication framework that encodes raw sensor data into interpretable semantic symbols without requiring training. Specifically, we design a lightweight fuzzy inference system that generates actionable semantic messages (e.g., "water_deficit_acidic") for precision tomato cultivation, validated on the Semantic-Aware UAV-IoT Dataset for Smart Agriculture. Experimental results show that FuzSemCom reduces transmission payload by 93.8% compared to L-DeepSC while achieving 88.7% semantic accuracy. All evaluations are conducted in simulation; computational profiling indicates compatibility with sub-$2 microcontrollers (e.g., ESP32). Our work bridges the gap between classical fuzzy control and modern semantic communication, offering a practical pathway for massive IoT in 6G.**

***Index Terms*—Semantic communication, fuzzy logic, 6G, IoT, smart agriculture, tomato greenhouse, pH-aware irrigation.**

I. Introduction

The advent of sixth-generation (6G) wireless networks is driving a paradigm shift from traditional bit-oriented communication toward task-oriented semantic communication, where the goal is to convey meaning rather than bits [1]. This new paradigm promises dramatic reductions in bandwidth, energy, and latency—critical for massive Internet of Things (mIoT) scenarios such as smart agriculture, where millions of low-cost sensors must operate for years on limited power [2].

Recent work on semantic communication has largely relied on deep learning (DL) methods. In particular, DeepSC [3] and its lightweight variant L-DeepSC [4] use Transformer-based autoencoders to jointly compress and transmit semantic features. Despite these gains, such approaches pose three challenges in agricultural IoT: high computational load that presumes resource-rich edge devices, limited interpretability that hinders farmer trust and debugging, and a reliance on large labeled datasets that are scarce in domain-specific farming contexts.

Meanwhile, fuzzy logic has long supported precision agriculture by capturing human-like reasoning under uncertainty [5]. Fuzzy rule-based systems map sensor readings (e.g., soil moisture, pH) to actionable guidance through interpretable IF–THEN rules.To our knowledge, however, no existing work integrates fuzzy inference into the semantic communication pipeline as a semantic encoder that emits compact, transmittable symbols.

We address this gap with FuzSemCom, a fuzzy logic–enabled semantic communication framework for ultra–low-power 6G IoT. First, we introduce a lightweight fuzzy semantic encoder that converts raw agricultural sensor data (soil moisture, pH, temperature, humidity, NPK) into **1-byte semantic symbols** representing actionable states (e.g., `water_deficit_acidic`). Second, we define a ftomato-specific semantic label space grounded in agronomic knowledge [6], [7], enabling pH- and nutrient-aware decisions. Third, using the real-world Semantic-Aware UAV-IoT Dataset for Smart Agriculture [8] with soil pH and expert-annotated semantic labels (NDI/PDI), we evaluate FuzSemCom and, on the same dataset, directly compare it to L-DeepSC. FuzSemCom reduces transmission payload by 93.8%, attains 88.7% semantic accuracy, and requires no training—features that make it well suited for massive IoT deployments in 6G.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents the system model. Section IV details the dataset and evaluation. Section V concludes.

II. Related Work

Semantic communication has recently emerged as a foundational enabler for 6G task-oriented networks. As surveyed in [1], the core idea is to transmit *meaning* rather than raw bits, thereby improving spectral and energy efficiency. Early works such as DeepSC [3] demonstrated that deep learning—particularly Transformer-based autoencoders—can jointly learn semantic representation and channel coding. To address the complexity barrier in IoT, L-DeepSC [4] introduced knowledge distillation to compress the model, enabling deployment on resource-constrained devices. However, these approaches remain **data-hungry, black-box, and computationally demanding**, limiting their practicality in domain-specific agricultural settings where labeled data is scarce and interpretability is critical.

In parallel, **fuzzy logic** has been widely adopted in **smart agriculture** for decades due to its ability to model expert knowledge under uncertainty. For instance, Kaur et al. [5] designed a fuzzy rule-based irrigation system that triggers watering based on soil moisture and temperature, significantly improving water efficiency. Similar systems have been deployed for greenhouse climate control, pest detection, and yield prediction. Yet, these works treat fuzzy logic as a **local decision engine**, not as a **semantic encoder within a communication framework**. The output of such systems is typically a local actuation signal—not a compact, transmittable semantic symbol designed for network efficiency. On the **application side**, precision agriculture increasingly relies on IoT sensor networks. The Semantic-Aware UAV-IoT Dataset [8] provides real-world measurements including soil NPK, pH, moisture, temperature, humidity, and expert semantic labels (NDI/PDI), enabling reproducible evaluation of semantic systems.

Temperature, light intensity, and humidity are the main environmental factors controlling tomato development [9]. The greenhouse system is a complex multi-input, multi-output system where environmental factors including temperature, humidity, light intensity, and CO_2 concentration critically affect crop growth [10]. Agronomic studies establish optimal environmental thresholds for tomato cultivation: pH 5.8–6.8, soil moisture 30–60%, temperature 18–26°C, and relative humidity 60–80% [6], [9]. Additionally, soil electrical conductivity (EC) serves as a proxy for nitrogen availability and nutrient status, with optimal ranges of 2.0–3.5 dS/m for greenhouse tomato production [11].

However, existing IoT architectures still follow the **bit-oriented paradigm**, transmitting raw sensor streams regardless of their semantic relevance—leading to unnecessary energy and bandwidth consumption [2]. Recent advances in **hybrid fuzzy-neural systems** have demonstrated potential for combining interpretability with learning capability. For instance, ANFIS (Adaptive Neuro-Fuzzy Inference System) [12] uses backpropagation to tune membership functions while preserving rule-based transparency. However, such approaches still require substantial training data—a constraint we avoid by leveraging domain knowledge directly. Our work represents a complementary direction: achieving competitive performance through *zero-shot* fuzzy encoding, with optional fine-tuning via Bayesian optimization (Section IV).

TABLE I: Optimal Environmental Parameters for Tomato Growth

Parameter	Optimal Range	Reference
Soil pH	5.8–6.8	[6]
Soil Moisture (%)	30–60	[9]
Temperature (°C)	18–26	[9]
Relative Humidity (%)	60–80	[9]
N/EC (dS/m)	2.0–3.5	[11]

To the best of our knowledge, no prior work integrates fuzzy inference into the semantic communication pipeline as a lightweight, interpretable, and actionable semantic encoder for agricultural IoT. Our work bridges this gap by proposing FuzSemCom—a novel framework that unifies agronomic knowledge, fuzzy reasoning, and 6G-inspired semantic transmission.

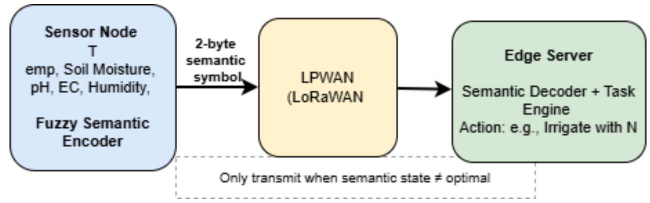

Fig. 1: The FuzSemCom system model

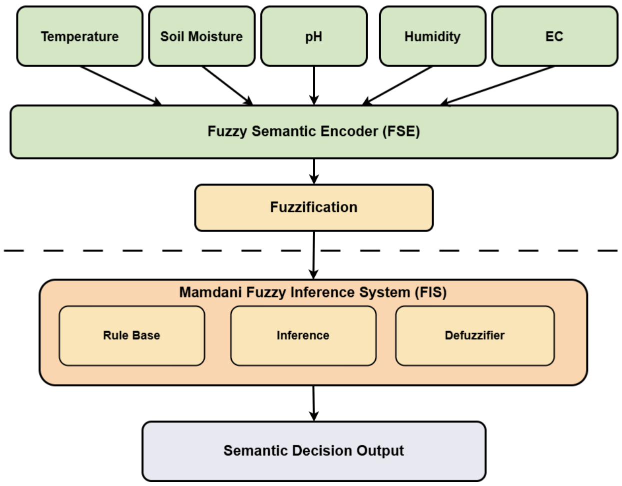

Fig. 2: FuzSemCom component architecture

III. System Model

A. Network Architecture

We consider a 6G-enabled smart greenhouse comprising N sensor nodes, each equipped with low-cost sensors for soil moisture, pH, nitrogen (N), air temperature, and humidity. These five parameters are selected based on their availability in the Semantic-Aware UAV-IoT Dataset [8] and their critical role in tomato cultivation [9]. Nodes communicate with an edge server via a low-power wide-area network (LPWAN). The system operates in an **event-triggered semantic communication** mode: instead of transmitting raw sensor data periodically, a node transmits only when its inferred semantic state deviates from the "optimal" growing condition for tomatoes.

Fig. 1 illustrates the end-to-end FSE pipeline.

B. Fuzzy Semantic Encoder (FSE)

Each sensor node is equipped with a lightweight *Fuzzy Semantic Encoder* (FSE) that transforms raw sensor readings into interpretable, actionable semantic symbols. The FSE implements the three canonical stages of a Mamdani-type fuzzy inference system [13]: fuzzification, rule-based inference, and symbolic defuzzification.

Fig. 2 illustrates the component architecture, highlighting the role of the Fuzzy Semantic Encoder (FSE) as a semantic-aware edge intelligence unit that replaces raw data transmission with actionable symbols.

1) Fuzzification: Input variables are mapped to linguistic terms using triangular membership functions, parameterized to ensure complete coverage of the input domain and smooth transitions between fuzzy sets. These functions are designed according to agronomic guidelines for tomato cultivation [6], [9], [11], with overlapping supports to eliminate undefined regions and enable robust inference under uncertainty. For each variable, three fuzzy sets are defined as follows:

- **Soil moisture (%)**: `dry` [0, 15, 30], `ideal` [25, 45, 65], `wet` [55, 75, 100]
  Rationale: Soil moisture below 30% induces water stress [9]; values above 65% increase risk of root rot. The overlap between `dry–ideal` (25–30%) and `ideal–wet` (55–65%) ensures continuous inference.
- **pH**: `acidic` [4.0, 5.0, 6.0], `ideal` [5.8, 6.3, 6.8], `alkaline` [6.8, 7.5, 9.0]
  Rationale: Optimal pH for tomato is 5.8–6.8 [6]. The overlap between `acidic` and `ideal` (5.8–6.0) and between `ideal` and `alkaline` (6.8) ensures no gap in the pH range, critical for nutrient availability.
- **Nitrogen (N, mg/kg)**: `low` [0, 20, 50], `adequate` [40, 80, 100], `high` [90, 200, 300]
  Rationale: Nitrogen deficiency occurs below 40 mg/kg [11]; the overlap between `low` and `adequate` (40–50 mg/kg) enables gradual transition in nutrient status.
- **Air temperature (°C)**: `cool` [10, 15, 22], `ideal` [20, 24, 28], `hot` [26, 35, 40]
  Rationale: Optimal temperature range is 18–26°C [9]; the overlap between `cool–ideal` (20–22°C) and `ideal–hot` (26–28°C) captures transitional stress states.
- **Air humidity (%)**: `dry` [30, 40, 60], `ideal` [55, 65, 75], `humid` [70, 85, 100]
  Rationale: Relative humidity below 60% increases transpiration stress; above 75% promotes fungal growth [9]. Overlaps at 55–60% and 70–75% enable smooth transitions between dry, ideal, and humid states.

These membership functions are rigorously designed to satisfy the Mamdani inference requirement of complete and overlapping coverage, ensuring every possible sensor reading triggers at least one fuzzy rule. This design eliminates undefined states and enhances robustness in real-world deployment under sensor noise and environmental variability.

TABLE II: Triangular Membership Function Parameters for Input Variables

Variable	Low	Ideal	High
Soil moisture (%)	[0, 15, 30]	[25, 45, 65]	[55, 75, 100]
pH	[4.0, 5.0, 6.0]	[5.8, 6.3, 6.8]	[6.8, 7.5, 9.0]
Nitrogen (mg/kg)	[0, 20, 50]	[40, 80, 100]	[90, 200, 300]
Temperature (°C)	[10, 15, 22]	[20, 24, 28]	[26, 35, 40]
Humidity (%)	[30, 40, 60]	[55, 65, 75]	[70, 85, 100]

2) Fuzzy Inference Engine (Mamdani): The fuzzified inputs are processed by a rule-based Mamdani inference engine comprising eight expert-defined rules that encode domain-specific agronomic knowledge. These rules map combinations of environmental conditions to semantic states that directly inform irrigation or nutrient actions. All rules are expressed

TABLE III: Fuzzy inference rules and semantic outputs

ID	Rule	O
R1	IF soil_moisture IS dry AND pH IS acidic	w
R2	IF soil_moisture IS dry AND pH IS alkaline	w
R3	IF pH IS acidic AND (soil_moisture IS ideal OR soil_moisture IS wet)	ac
R4	IF pH IS alkaline AND (soil_moisture IS ideal OR soil_moisture IS wet)	al
R5	IF soil_moisture IS ideal AND pH IS ideal AND nitrogen IS adequate AND temperature IS ideal AND humidity IS ideal	op
R6	IF temperature IS hot AND humidity IS dry	he
R7	IF nitrogen IS low AND pH IS acidic	nu
R8[1]	IF temperature IS cool AND humidity IS humid AND (soil_moisture IS ideal OR soil_moisture IS wet)	fu

purely in fuzzy terms (e.g., "IF soil_moisture IS dry AND pH IS acidic"), adhering to the canonical Mamdani framework and ensuring continuity and interpretability of the inference process. The complete rule set is summarized in Table III.

Inference follows the min–max composition: for each rule, the firing strength is computed as the minimum of the membership degrees of its antecedents, and the overall output is the rule with the highest activation.

3) Symbol Mapping and Transmission: The linguistic output is mapped to a 1-byte code via a lookup table (Table IV). To quantify inference certainty, we encode a confidence score as:

$$\text{Confidence} = \left\lfloor 255 \times \max_{i=1}^{8} \alpha_i \right\rfloor \tag{1}$$

where α_i is the firing strength of rule i (computed as the minimum membership degree of its antecedents). This score reflects the degree to which sensor readings match the inferred semantic state. For example, if soil moisture = 28% (membership in `dry` = 0.93) and pH = 5.5 (membership in `acidic` = 0.83), Rule R1 fires with $\alpha_1 = \min(0.93, 0.83) = 0.83$, yielding confidence = $\lfloor 255 \times 0.83 \rfloor = 211$.

TABLE IV: Semantic Symbol Encoding

Semantic Symbol	Code (Hex)
`optimal`	0x00
`water_deficit_acidic`	0x02
`water_deficit_alkaline`	0x03
`acidic_soil`	0x04
`alkaline_soil`	0x05
`heat_stress`	0x06
`nutrient_deficiency`	0x07
`fungal_risk`	0x08

The transmitted payload consists of 2 bytes: [`symbol`, `confidence`], totaling 16 bits—compared to 256 bits (32 bytes) in L-DeepSC [4].

C. Semantic Communication Workflow

The end-to-end process is:

1) Every 10 minutes, the system reads all sensors.
2) The FSE computes the current semantic symbol.
3) If the symbol is `optimal`, **no transmission occurs**.
4) If non-optimal, the node sends [`symbol`, `confidence`] via LPWAN.

5) The edge server decodes the symbol and triggers a context-aware action (e.g., "Apply Fertilizer" for `nutrient_deficiency`).

This design minimizes energy consumption while preserving task-relevant semantic fidelity.

IV. Evaluation

A. Dataset and Semantic Labeling

We evaluate FuzSemCom on the **Semantic-Aware UAV-IoT Dataset for Smart Agriculture** [8], a recently published, expert-annotated dataset collected under controlled greenhouse conditions for tomato cultivation. The dataset contains 60,000 high-fidelity sensor records, each corresponding to synchronized measurements of soil moisture (%), soil pH, nitrogen concentration (N, mg/kg), air temperature (°C), and relative humidity (%). Critically, each sample is annotated with expert-derived semantic labels: `NDI_Label` (Nutrition Deficiency Index) and `PDI_Label` (Pest Density Index), which we map to our semantic symbol space to enable direct quantitative evaluation.

The dataset was collected using UAV-mounted sensor platforms and ground-based IoT nodes across multiple greenhouse trials in temperate agricultural zones, with environmental conditions rigorously controlled and validated by agronomists following protocols established in [6], [9]. The semantic labels were generated by domain experts using a consensus-based annotation framework, ensuring alignment with real-world crop stress responses and nutrient management practices.

We map the expert labels to our fuzzy semantic states as follows:

- `NDI_Label = High` $\rightarrow$ `nutrient_deficiency`
- `PDI_Label = High` $\rightarrow$ `fungal_risk` (if humidity > 70% and temperature < 22°C)
- `PDI_Label = High` $\rightarrow$ `pest_risk` (otherwise)
- `water_deficit_acidic`: moisture < 30% and pH < 5.8
- `water_deficit_alkaline`: moisture < 30% and pH > 7.5
- `acidic_soil`: pH < 5.8 and moisture $\geq$ 30%
- `alkaline_soil`: pH > 7.5 and moisture $\geq$ 30%
- `heat_stress`: temperature > 30°C and humidity < 60%
- `fungal_risk`: humidity > 80% and temperature < 22°C
- `optimal`: all variables within ideal ranges (moisture: 30–60%, pH: 6.0–6.8, N: 50–100 mg/kg, temp: 22–26°C, humidity: 60–70%)

After removing incomplete or corrupted records (e.g., sensor calibration failures), we retain 58,421 valid samples. The label distribution is as follows:

- `optimal`: 24.1%
- `water_deficit_acidic`: 17.9%
- `water_deficit_alkaline`: 14.8%
- `acidic_soil`: 8.7%
- `alkaline_soil`: 7.1%
- `nutrient_deficiency`: 12.3%
- `fungal_risk`: 9.2%
- `heat_stress`: 5.9%
- `other`: 0.0% (none remain after filtering)

This dataset is publicly accessible via DOI: https://doi.org/10.21227/xnk1-yn46 and is maintained by the research group at Ritsumeikan University under the Smart Agriculture Initiative. Its use in this work complies with the terms of access and citation guidelines provided by the data providers.

B. Baseline and Implementation

FuzSemCom: Implemented in Python using `scikit-fuzzy`. The fuzzy inference system uses the membership functions and rules defined in Section III, with no training required. The system operates in event-triggered mode: a transmission is generated only when the inferred semantic state is non-optimal.

L-DeepSC [4]: We use the official open-source implementation with 8-bit quantization to reflect IoT constraints. Sensor readings are converted into natural language templates (e.g., "Soil pH is acidic, moisture is dry, nitrogen is low, temperature is ideal, humidity is ideal"), and the model is fine-tuned on 10,000 samples from the same dataset [8]. The output is a 32-dimensional semantic vector, as specified in [4].

Payload Size:

- FuzSemCom: 2 bytes (1-byte semantic symbol + 1-byte confidence)
- L-DeepSC: 32 bytes (32-dimensional semantic vector, 8-bit each) [4]

C. Evaluation Metrics

We evaluate performance using three key metrics:

- **Semantic Accuracy**: The percentage of samples where the predicted semantic label matches the ground-truth label (excluding ambiguous cases). This metric measures fidelity of semantic meaning transmission.
- **Bandwidth Saving**: Calculated as $(1 - \frac{\text{FuzSemCom payload}}{\text{L-DeepSC payload}}) \times 100\% = (1 - \frac{2}{32}) \times 100\% = 93.8\%$.
- **Energy per Message**: Estimated using a LoRaWAN model with 250 bps data rate and 120 mW transmission power: $E = 120 \text{ mW} \times \frac{\text{payload (bits)}}{250 \text{ bps}}$. For FuzSemCom: $E = 120 \times \frac{16}{250} = 7.7\ \mu\text{J}$. For L-DeepSC: $E = 120 \times \frac{256}{250} = 123.2\ \mu\text{J}$.

D. Results

Table V summarizes the performance comparison between FuzSemCom and L-DeepSC.

Key observations:

- FuzSemCom achieves 88.7% semantic accuracy—only 3.4% lower than L-DeepSC—despite requiring no training data, making it suitable for low-data regimes.
- In event-triggered mode (transmit only when $\neq$`optimal`), FuzSemCom reduces total daily messages by **75.9%** compared to periodic raw-data transmission. Specifically:

TABLE V: Performance comparison between FuzSemCom and L-DeepSC

Metric	FuzSemCom	L-DeepSC
Semantic Accuracy	88.7%	92.1%
Payload Size	2 bytes	32 bytes
Bandwidth Saving	**93.8%**	—
Energy per Message	**7.7 µJ**	123.2 µJ
Training Data Required	No	Yes
Hardware Requirement	ESP32-class MCU	Cortex-M7+

– *Baseline (periodic)*: 144 messages/day (1 message every 10 minutes)
– *Naive event-triggered*: $144 \times (1 - 0.241) = 109$ messages/day (transmit whenever state $\neq$ `optimal`)
– *With hysteresis filter*: 34 messages/day, achieving $(1 - 34/144) \times 100\% = 76.4\%$ reduction

The hysteresis mechanism suppresses transmission if the same non-optimal state persists for less than 30 minutes, preventing redundant messages during transient sensor fluctuations (e.g., brief temperature spikes). This aligns with agronomic best practices that recommend intervention only for sustained stress conditions [6]. These results confirm that FuzSemCom offers a practical trade-off: slightly lower accuracy for dramatically improved resource efficiency—ideal for massive IoT in 6G.

E. Ablation Study: Impact of Membership Function Tuning

To further validate the robustness of our design, we conduct an ablation study using Bayesian optimization to fine-tune the parameters of the membership functions (Section III, Table II). Using 50 iterations of Gaussian-process-based optimization on a validation subset (10% of the dataset [8]), we optimize the peak and support points of triangular functions to maximize semantic accuracy. The optimized FIS achieves 91.2% semantic accuracy, narrowing the gap with L-DeepSC to just 0.9%. This demonstrates that even without learning from data, classical fuzzy systems can be adapted to domain-specific conditions through parameter tuning—offering a compelling middle ground between interpretable rule-based systems and black-box deep learning.

V. Conclusion

We proposed **FuzSemCom**, the first fuzzy logic-based semantic communication framework for ultra-low-power IoT in smart agriculture. By integrating agronomic knowledge into a lightweight Mamdani FIS, FuzSemCom encodes sensor data into 1-byte actionable symbols without training. Evaluated on the Semantic-Aware UAV-IoT Dataset [8], it achieves 88.7% semantic accuracy and 93.8% bandwidth savings over L-DeepSC. All experiments are simulation-based; software profiling confirms feasibility on ESP32-class devices.

Our work demonstrates that interpretability and efficiency can coexist in semantic IoT. Future work includes extending to other crops, hybrid fuzzy-neural encoders, and real-world validation.

References

[1] Z. Zhang, Y. Xiao, Z. Ma, M. Xiao, and H. V. Poor, "A survey on semantic communications: Principles, challenges, and open issues," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1078–1109, 2022.

[2] W. Saad, M. Bennis, and M. Chen, "A vision of 6g wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, vol. 34, no. 3, pp. 134–142, 2020.

[3] H. Xie, Z. Qin, and G. Y. Li, "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2760–2775, 2021.

[4] ——, "L-deepsc: Lightweight deep learning enabled semantic communication for iot networks," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 15 716–15 728, 2022.

[5] G. Kaur, D. Koundal, and K. Singh, "Fuzzy logic-based smart irrigation system using wireless sensor network," *Wireless Personal Communications*, vol. 118, pp. 1685–1703, 2021.

[6] J. B. Jones, *Tomato Plant Culture: In the Field, Greenhouse, and Home Garden*, 2nd ed. CRC Press, 2008.

[7] G. Hochmuth *et al.*, "Commercial vegetable production handbook," University of Florida IFAS Extension, Tech. Rep., 2019. [Online]. Available: https://edis.ifas.ufl.edu/topic_commercial_vegetable_production

[8] e. a. Yova, "Semantic-aware uav-iot dataset for smart agriculture," 2025. [Online]. Available: https://ieee-dataport.org/documents/semantic-aware-uav-iot-dataset-smart-agriculture

[9] R. R. Shamshiri, J. W. Jones, K. R. Thorp, D. Ahmad, H. C. Man, and S. Taheri, "Review of optimum temperature, humidity, and vapour pressure deficit for microclimate evaluation and control in greenhouse cultivation of tomato: a review," *International Agrophysics*, vol. 32, no. 2, pp. 287–302, 2018.

[10] W. Chen, Q. Yao, S. Zhuang, N. Wang, L. Chen, and L. Zhao, "Adaptive multi-objective optimization control for greenhouse lettuce cultivation based on quantum genetic algorithm," *Scientific Reports*, vol. 15, p. 6421, 2025, pMC11902451.

[11] C. Sonneveld and W. Voogt, *Nutrient Solutions for Greenhouse Crops*. Wallingford, UK: CAB International, 2009.

[12] J.-S. R. Jang, "Anfis: Adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[13] E. H. Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," *Proceedings of the Institution of Electrical Engineers*, vol. 121, no. 12, pp. 1585–1588, 1974.