Smart Agricultural Technology 11 (2025) 100885

Contents lists available at ScienceDirect

Smart Agricultural Technology

journal homepage: www.journals.elsevier.com/smart-agricultural-technology

Optimization of irrigation and fertigation in smart agriculture: An IoT-based micro-services framework

Tommaso Adamo [a], Danilo Caivano [b], Lucio Colizzi [b,*], Giovanni Dimauro [b], Emanuela Guerriero [a]

[a] Dipartimento di Ingegneria dell'Innovazione, University of Salento, Via Monteroni, Lecce, 73100, Italy
[b] Dipartimento di Informatica, University of Bari Aldo Moro, Via E. Orabona, 4, Bari, 70125, Italy

ARTICLE INFO

Keywords:
Smart farming
IoT sensors
Fertigation optimization
Irrigation optimization
Micro-services
Sustainable agriculture

ABSTRACT

Efficient management of water and fertilizer resources is crucial for achieving sustainability and productivity in agriculture. This paper presents an AI-powered microservices solution that optimizes irrigation and fertigation practices. The proposed system integrates IoT nodes for real-time data collection on environmental conditions, soil moisture levels, and nutrient crop needs. Fertigation and irrigation decision-making are modeled as a data-driven sequential decision problem. At each decision stage, real-time data serve as input to an AI planning model aimed at satisfying nutrient and water demands while minimizing water and fertilizer waste. The system allows supervision by the farmer through a mobile app and a Digital Twin, enabling the design of crop planting layouts and providing detailed information on real-time decisions implemented in the field, as well as water and fertilizer consumption. The proposed solution manages diverse crop species with distinct water and nutrient requirements. Efficient data exchange is facilitated through a push-pull communication paradigm between the IoT nodes and cloud services. This approach offers several benefits, including greater control over data flow, energy savings, and increased flexibility in resource management.

1. Introduction

Recent years have seen a surge in interest in developing digital farming solutions. According to Kumar Kasera et al. [24], the number of scientific papers started to grow considerably in 2016, making it one of the most discussed research topics in the last two years. Rapid advancements in Internet of Things (IoT), distributed systems architectures, and Cloud Computing are driving a fourth farming revolution. The push for sustainability in agricultural production, along with the ongoing reduction in the cost of IoT devices, has led to a substantial increase in ICT (Information and Communication Technologies) solutions for the farming industry, which is predicted to grow from \$6.5 billion in 2019 to \$21 billion by 2030 [32]. The agronomic processes receiving the most attention from academia and the world of industrial innovation are undoubtedly irrigation and fertigation. Providing water and fertilizers only where and when needed represents an important challenge to pursue today. Indeed, optimizing and rationalizing water and input nutrients contribute to both the economic sustainability of production and the impact that irrigation and fertigation processes have on the environment [35]. To this aim, integration of ICT into conventional agricultural practices can be carried out according to two main approaches known as *Precision Agriculture* and *Smart Agriculture*. Precision Agriculture (PA) refers to the use of advanced technologies such as global positioning systems, sensors, drones, and data management software to gather and analyze real-time and geostatistical information on soil, climate, and crop conditions, in order to make more informed agronomic decisions and optimize the use of agricultural resources. While precision agriculture is about using precise data to optimize specific farming practices, Smart Agriculture (SA) aims to create an overall intelligent farming ecosystem through the integration of various advanced technologies [17]. Indeed, SA refers to the integrated use of different precision agriculture technologies and IoT, allowing for real-time data collection and remote control of activities in the field. The focus is rather on access to data and the application of these data and how the collected information can be used in a smart way. In a recent contribution, Zhang [51] assesses that researchers are not clear about whether there are distinguishable differences among three categories: *digital agriculture*, *precision agriculture*, and *smart agriculture*. To this aim, the author randomly se-

* Corresponding author.
E-mail address: lucio.colizzi@uniba.it (L. Colizzi).

https://doi.org/10.1016/j.atech.2025.100885
Received 27 September 2024; Received in revised form 25 February 2025; Accepted 6 March 2025

Available online 13 March 2025

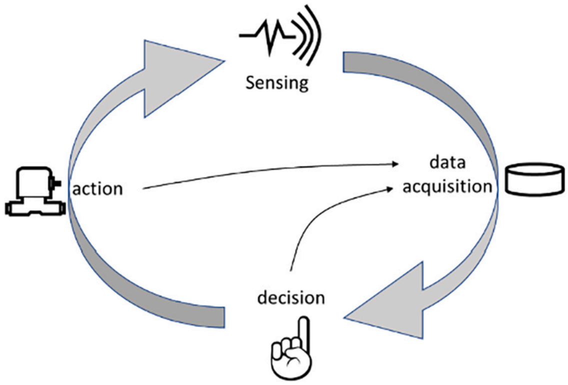


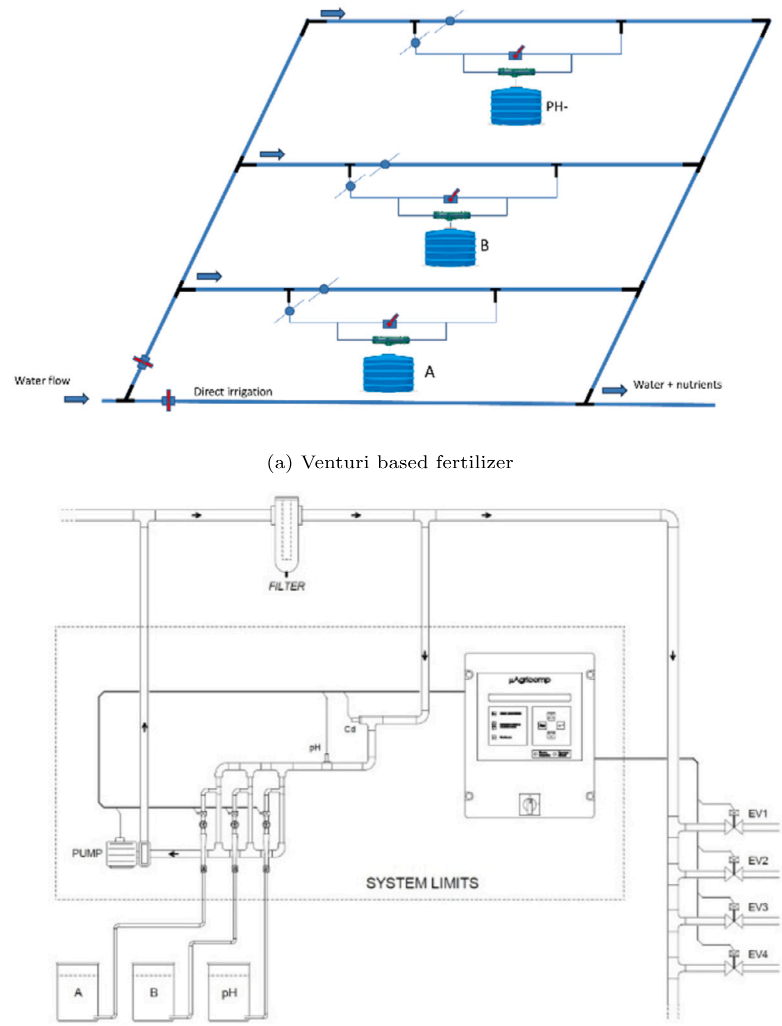
Fig. 1. 1-chain of events in a smart farming system.

lects 100 contributions from 1800 papers with one of these three terms in the title or keyword lists and finds that most papers could be listed under all three categories. Zhang [51] also suggests using the term *precision agriculture* to describe the current development stage of agriculture and *smart precision agriculture* (SPA) to mark advancements of precision agriculture from very simple approaches in its early days to more autonomous and intelligent agricultural systems.

As illustrated in Fig. 1, a SPA system aims to integrate sensors, actuators, and decision algorithms to control the status of growing crops, optimizing the use of agronomic inputs such as fertilizers, water, energy, organic compounds, etc. [33]. In particular, agronomic inputs are optimized through the combination of Site-Specific Crop Management (SSCM) and real-time field measurements and interventions. The integration of SSCM and real-time intervention can be seen as the application of information technologies, along with agronomic experience, to optimize production efficiency, quality, and environmental impact, as well as to minimize production risk, all at the site-specific level.

In this paper we focus on the use of SPA to manage *fertigation*, a portmanteau of "fertilization" and "irrigation". Fertigation is a sophisticated agricultural practice that combines the delivery of fertilizers with irrigation water directly to the root zone of plants. This method offers several advantages over traditional fertilization techniques, primarily by enhancing the efficiency of nutrient delivery and water usage. In a modern fertigation system, various components work in a synergistic manner to achieve nutrient management, tailored to the specific needs of crops while minimizing environmental impact [19]. At the heart of a modern fertigation setup lies a fertigation unit, a sophisticated device designed to accurately blend fertilizers with irrigation water according to predefined formulas and real-time monitoring data. This unit typically consists of a series of tanks or reservoirs containing concentrated fertilizers, which are automatically mixed with water in precise proportions through pumps or injectors. The mixing process is carefully controlled to ensure uniform distribution of nutrients throughout the irrigation system (Bassett et al. [8], Lemonnier & Hervieu [25]). Farmers can program the system to deliver nutrients at specific times or in response to changing environmental conditions, ensuring optimal plant growth. In modern fertigation systems, a relevant aspect is related to the integration with advanced monitoring and control technologies. The fertigation unit typically measures parameters such as pH and Electrical Conductivity (EC) of the nutrient solution being dispensed. Then the system acts to keep nutrient solution balanced according to a recipe provided by the agronomic advisor. Consequently, while modern fertigation systems provide crops with precise amounts of water and fertilizers, they may lack sensed data from the soil, potentially limiting their ability to satisfy the actual needs of the crops. Another limitation of fertigation systems lies in the variety of crop species that can be served. A typical setup involves three tanks, usually labeled A, B, and acid. The nutrient mixture is tailored for a specific crop system, depending on factors such as the cultivated species, soil characteristics, and water quality. Once the mixture is prepared, it needs to be split into the A and B tanks with a specific concentration factor. This split is necessary to avoid potential chemical incompatibilities among nutrients, which could lead to aggregation and precipitation at the bottom of the tanks rather than proper solubilization. Additionally, the third tank is utilized to adjust the pH of the nutrient solution. Fig. 2 illustrates two typical examples of commercial fertigation systems. The *simplified* schema in Fig. 2a shows an example of the former approach, considered by farmers a low-cost solution, where the fertigation unit consists of a set of parallel Venturi injector lines [13]. Fig. 2b depicts an example of the latter approach, considered by farmers a *high-tech* fertigation unit, which adjusts EC and pH to target values of the water-fertilizer mix before delivering it to the field [4]. These currently available solutions fall short of helping farmers realize the underlying promise of technological progress in precision smart farming. As stated by Zhang [51] numerous sensors are being developed and marketed for SPA, yet effectively utilizing the sensed data to support decision-making in production challenge. We aim to address this challenge by focusing on the optimization of water and nutrient management. As mentioned earlier, decision-making in current fertigation systems is primarily based on measuring EC and pH of the nutrient mixture. Over the past two years, some low-cost electrochemical sensors capable of measuring nitrogen, phosphorus, and potassium content in soil and/or substrates have emerged on the market [9]. These sensors allow for the collection of data necessary to make fine-grained fertigation decisions. However, the short-term planning of irrigation and fertigation must properly account for the non-negligible recovery times of these sensors, which are ion-selective but not ion-specific. In this research, we devise an IoT-based framework to handle these issues through an information-to-action decision-making process, modeled as a data-driven sequential decision problem. Specifically, we propose an optimal policy based on the solution of an integer linear programming model (ILP). Our primary goal is to prevent water excess or over-fertilization, both of which can adversely affect crop yield and environmental health. Indeed, the risk of water excess arises when irrigation operations are not adequately monitored or controlled, leading to soil moisture levels exceeding plant requirements. This can result in waterlogging, soil compaction, and root oxygen deprivation, all of which compromise plant growth and increase the risk of root diseases. Conversely, over-fertigation occurs when the quantity of fertilizer applied exceeds plant needs or when nutrients are unevenly distributed in the soil. This can lead to the contamination of groundwater and surface water with excess nutrients such as nitrogen and phosphorus, negatively impacting the environment and water quality. After a fertigation phase, soil moisture may already be adequate for the plants' needs, reducing the necessity for additional irrigation with non-fertilized water. However, if salts accumulate in the soil due to fertigation, it is crucial to activate the irrigation process to leach the salts and make them available for plant absorption through osmotic processes. Therefore, managing fertigation and irrigation processes separately allows for the precise adjustment of water and nutrient supply to meet the specific needs of the plants. This approach promotes healthy plant growth and helps reduce the risk of salt accumulation in the soil [44].

It is important to highlight that the issues discussed are relevant not only for agronomic reasons but also for sustainability, both economic and environmental. Consider, for example, the impact of geopolitical situations on fertilizer prices. Additionally, there is the environmental impact of over-irrigation and over-fertilization, i.e., the delivery of water and nutrients that are not directly related to the actual needs of the plants. This necessitates investing in new solutions that facilitate the transition from traditional fertigation methods, which distribute water and fertilizers evenly at fixed intervals, to a more intelligent system that delivers water and fertilizers precisely where and when they are needed, ensuring optimal resource use. To this end, this paper proposes an innovative system for smart irrigation and fertigation that integrates moisture, pH, nitrogen, phosphorus, and potassium sensors connected to IoT nodes and utilizes Artificial Intelligence (AI) planning algorithms



(a) Venturi based fertilizer

(b) An example of automatic fertigation unit proposed by (Agricontrol, 2024)

Fig. 2. A schematic overview of an inline fertigation system.

within a micro-services architecture. In particular, the system's decision-making framework is modeled as an intelligent (problem-solving) agent powered by an ILP model [41]. Following such an AI-based modeling approach, the proposed system evaluates both the current system and environmental states, dynamically choosing actions that maximize a utility function reflecting sustainable resource management. By balancing competing objectives -such as minimizing water and fertilizer use while satisfying crop-specific nutrient needs- the agent ensures optimal, autonomous, and data-driven decision-making.

The proposed approach distinguishes our system from traditional PID (Proportional Integral Derivative) controller-based solutions in several key aspects:

- **Complexity Management**: The ILP model effectively handles a larger number of variables and constraints compared to PID controllers. This allows for the consideration of over-fertilization, fertilizer compatibility, the prevention of mixing incompatible nutrients, and a more comprehensive approach to decision-making.
- **Multiobjective Optimization**: While PID controllers excel at regulating individual variables, they fall short in optimizing the overall use of resources. The ILP model, on the other hand, is designed to optimize multiple key performance indicators. The proposed ILP model minimizes water and fertilizer consumption while meeting the specific needs of different crops, aligning with the goals of sustainable agriculture.

Our framework surpasses the capabilities of conventional PID controllers, offering a more sophisticated and sustainable approach to managing irrigation and fertigation processes.

2. Our contribution

The main goals of this paper can be summarized as follows.

1. The paper proposes a framework for an IoT-based smart fertigation system that incorporates short-term planning for water and fertilizer management. To the best of our knowledge, this is the first contribution to devise an IoT-based framework that relies on information-to-action decisions aimed at fine-tuning the supply of individual macronutrients and water to each distribution channel.
2. We model fertigation and irrigation decision-making as a data-driven sequential decision problem, implemented through a problem-solving autonomous agent [41]. At each decision stage, real-time data is used as input to an Integer Linear Programming model designed to meet nutrient and water demands while minimizing resource waste.

3. A comprehensive evaluation of the proposed framework was conducted in greenhouses with various horticultural species and growing methods, including both soil and soilless techniques. This assessment highlights the framework's capabilities and advantages, demonstrating its potential to advance agricultural practices.

The remainder of this paper is organized as follows. Section 3 reviews previous related contributions. Section 4 presents the IoT-based micro-services framework. Section 5 describes how the proposed system models fertigation and irrigation as a data-driven sequential decision problem. Section 6 introduces an optimal policy for these processes based on the solution of an ILP model. Section 7 details the system implementation. Section 8 provides a critical evaluation of the proposed approach. Finally, Section 9 concludes the paper and discusses potential avenues for future research.

3. Background and related work

As aforementioned, SPA is a promising field in agriculture that leverages IoT technology to enable real-time monitoring and control of various farming processes. Several contributions deal with IoT systems designed with the aim of improving crop yields, reducing water usage, and enhancing overall sustainability. Research on this topic is fairly recent and scattered. In this paper, we are interested in methods and technologies to achieve information-to-action decision-making for the management of water and fertilizer resources. To this aim, we review the most recent contributions that specifically address the integration of field data acquisition systems, actuation systems, and front-end monitoring or data analytics components within the domain of irrigation and fertigation. The key technological components in our review are IoT nodes and AI-based decision-making. The interested reader may refer to Kumar Kasera et al. [24] for a thorough literature review of IoT and AI-based applications in farming activities, including, among others, disease analysis, pathogen detection, seed selection, land preparation, and crop selection.

3.1. Irrigation

IoT-enabled irrigation systems offer several advantages, including reduced overall water consumption, increased cost-efficiency, improved performance efficiency, lower energy consumption, and minimized crop wastage [34]. The drastic reduction in IoT device costs and their easy availability online has given rise to a multitude of low-cost solutions that were unimaginable just a few years ago (Benyezza et al. [10]).

Dahane et al. [16] propose a smart agriculture system based on IoT and cloud computing for crop growth monitoring and intelligent irrigation management. The proposed system employs wireless sensors to gather real-time environmental data, which is then transmitted to both an edge-layer for deep learning-based decision-making and a cloud-layer for data storage and analysis. Subsequently, the cloud server sends the optimized irrigation strategy to the irrigation controller, effectively reducing water usage while enhancing crop yields.

Trilles et al. [46] propose a system that provides recommendations to farmers based on the analysis of sensed data. The proposed architecture is modular, scalable, and relies on micro-services and serverless paradigms. It consists of web services and includes sensor nodes, gateways, cloud services, and a web application. The system aims to offer real-time monitoring and decision-making support to farmers, incorporating several layers such as the data acquisition layer, the data processing layer, and the user interface layer.

Another approach to IoT architecture for precision smart farming is fog computing. Fog computing extends cloud computing to the edge of the network, bringing computation and data storage closer to the source of data generation, thus enabling faster processing and real-time analytics. The fog computing-based IoT framework proposed by Guardo et al. [20] is designed specifically for precision agriculture applications, enabling information-to-action decision-making. The architecture includes two fog layers where nodes act as gateways for data collection and processing, and a cloud-based platform that provides data storage and analysis capabilities. The framework is designed to support a large number of IoT nodes and heterogeneous data sources.

Kaur et al. [22] propose an IoT-based precision irrigation system that utilizes machine learning algorithms. The system employs a wireless sensor network to collect environmental data, which is then analyzed by machine learning algorithms to optimize the irrigation strategy. This approach aims to increase crop yield while reducing water usage.

Benyezza et al. [12] present an IoT-based zoning irrigation system that uses fuzzy logic to optimize irrigation timing and reduce water and energy consumption. In the proposed system, a wireless sensor network collects soil moisture and temperature data, then processed by a fuzzy logic controller to determine the necessary irrigation time for each zone. The authors highlight the importance of considering sensor range, water flow, and plant size when making irrigation decisions. The experimental results show significant water and energy savings compared to traditional irrigation methods. However, the authors acknowledge that the benefits, particularly the financial aspect, may not be immediately apparent due to the initial investment required for system installation. In 2023, the same authors extended their research by presenting an IoT platform for comprehensive greenhouse climate management (Benyezza et al. [11]). The authors propose a fuzzy logic approach for decision-making components in an IoT-based intelligent platform for monitoring and controlling greenhouse irrigation and internal climate. For an in-depth review of smart irrigation systems and monitoring and advanced control strategies for precision irrigation, interested readers may refer to Obaideen et al. [34] and Abioye et al. [1]. The study Abioye et al. [2] explores the application of machine learning models and the integration of algorithms with digital interfaces to enhance water efficiency and reduce environmental impact compared to conventional methods. The effectiveness of such learning systems, however, depends significantly on the availability of site-specific datasets, which can be challenging and costly for many farms to collect and maintain.

We finally observe that optimization models based upon Linear Programming (LP) technique has been used extensively for irrigation scheduling and management. Smout & Gorantiwar [45] propose a water allocation model, which takes into account deficit irrigation for optimizing the use of irrigation water. A mathematical model is developed to optimize multi-crop irrigation areas in a reservoir-irrigation system, integrating reservoir operation policies. Moradi-Jalal et al. [31] propose a LP model, that aims to maximize annual benefits by allocating irrigation water while considering reservoir releases, monthly water distribution, and spills. Constraints include reservoir balance, crop water demand, evaporation loss, and operational equations, formulated as a linear programming model. Sensitivity analysis on inflows and irrigation policies is performed. The approach is validated through a real case study in Iran, demonstrating its applicability. Singh [42] proposes a LP model to optimize land and water resource allocation in the Jhajjar district of Haryana, India, characterized by poor-quality groundwater. The model maximizes net annual returns and incorporates water production functions to assess crop yields under varying irrigation water qualities and includes a groundwater balance constraint to prevent waterlogging. Cid-Garcia et al. [15] propose an ILP-based approach that prescribes when and how much to irrigate or fertilize for the Real-Time Irrigation Problem (RTIP). It is worth noting that the real-time component is related only to the IoT sensors. The proposed framework has no direct connection between the mathematical model and an automated actuator system.

Some contributions explore the use of stochastic linear programming to incorporate uncertainty by modeling certain parameters as random variables with known probability distributions. This approach is particularly valuable in irrigation management, where factors such as rainfall, water availability, and crop water demand are inherently uncertain. By optimizing decisions while considering these uncertainties, stochastic linear programming helps allocate water resources efficiently, reducing the risks associated with droughts or fluctuating supply. A common framework is two-stage stochastic programming, where an initial irrigation plan is devised based on expected conditions, and adjustments are made once actual water availability and climatic conditions are observed (Li et al. [27], Wang et al. [49]). However, irrigation planning often requires a more dynamic and adaptive approach, making multistage stochastic programming a more suitable alternative. In a multistage setting, decisions are updated at multiple time steps as new information becomes available, allowing for a more flexible and responsive allocation of water resources (Li & Hu [26], Mahdavimanshadi & Fan [29]). For a review of optimizing water resource allocation using various programming techniques, the interested readers may refer to Marquez et al. [30] and Singh [43].

3.2. Fertigation

Fertigation is the technique of applying fertilizers and irrigation water simultaneously through an irrigation system. This approach can enhance crop yields, reduce water usage and labor costs, and provide precise control over nutrient application. The previous section discussed typical operating schemes of fertigation systems and the limitations of current commercial solutions. Several scientific studies have aimed to introduce innovative aspects to address these limitations.

Rosli et al. [40] illustrate the design and implementation of an IoT system that facilitates monitoring, control, and database storage to enhance crop yield. The authors examine key parameters such as the electrical conductivity of fertilizers, pH levels, air humidity, and temperature in the environment. The system is specifically tailored to support positive pressurized irrigation methods. However, one limitation of this approach is its reliance on cumulative measurements of soluble salts in the nutrient solution, as indicated by pH and EC sensors. While these measurements offer valuable insights into the overall nutrient levels, they do not provide specific details about individual macronutrients that may be deficient in the soil. This can be a disadvantage when aiming to address specific nutrient deficiencies in a targeted manner for optimal crop growth and productivity.

Ahmed et al. [5] utilize Xbee modules to integrate IoT nodes, facilitating the injection of fertilizers during irrigation phases. The soil moisture sensor employed is resistive, which is cost-effective but susceptible to degradation over time. Additionally, the decision-making model is designed to achieve target values for electrical conductivity and pH by injecting pre-calculated nutrient blends tailored to the specific crop needs.

Rode et al. [39] propose a small-scale system that uses a pH sensor to determine the amount of nitrogen to inject. For phosphorus and potassium, the system relies on reference values specific to the plant species being experimented with. The system manages fertilizer dispensing by activating solenoid valves until a threshold value is reached. However, this approach is not suitable for large-scale farming systems, where determining fertilizer amounts solely based on pH values is impractical. It is crucial to account for the biochemical interactions between the substrate and plants in relation to the applied fertilizers. Overcoming these limitations requires integrating ion-selective or ion-specific sensors into the IoT nodes.

Parimala et al. [36] integrate a modern fertigation bench with a monitoring and control system operated through a mobile app. The injection formula is standard, utilizing three containers: A, B, and acid. The system allows for the injection of fertilizers according to precise concentration ratios, all remotely controlled. While the remote monitoring component is innovative compared to the state of the art, the decision-making model (resulting from the system architecture) is also based on tracking pH and EC values.

Anuar et al. [6] also propose to make decisions about distribution of water with nutrients based on the measurement of EC in the mixture. The system utilizes a Total Dissolved Solids (TDS) sensor to quantify the total dissolved salts and metals. The value is then compared with thresholds that depend on the specific cultivated species. It should be noted that the delivery of agronomic inputs (both water and fertilizer mixtures) is carried out in constant and planned time windows.

Zailani & Jumaat [50] propose a fertigation system based on the Blynk client-server architecture. Fertigation decisions are made based on soil moisture measurements, and the use case of separating the irrigation process from fertigation is not considered. This implies that the nutrient mixture must be predetermined based on the species to be cultivated.

Valecce et al. [47] propose a *solar-fertigation* IoT architecture designed for smart agriculture. This system integrates solar power generation with IoT technologies to enable real-time monitoring and control of water and nutrient delivery to crops. The proposed framework utilizes Message Queuing Telemetry Transport (MQTT) and is capable of supporting a large number of IoT nodes. It includes multi-hop routing and traffic modeling to manage heterogeneous data sources effectively.

Recently, Cheruvu et al. [14] propose a new sensor technology that measures the quantity of macronutrients in the soil in parts per million (ppm) or milligrams per kilogram (mg/kg). This represents a significant advancement in the field, as it introduces a decision support system for precise macronutrient distribution. The proposed decision-making model employs the machine learning technique known as Random Forest, trained on a dataset of 2,200 entries from 22 different plant species. The dataset encompasses parameters such as nitrogen content, potassium content, phosphorus content, temperature, humidity, soil pH, and rainfall.

Lin et al. [28] propose an IoT-based framework for fertigation. The authors model fertigation management as an ILP aimed at maximizing farmers' profits. The corresponding optimization problem is a short-term planning problem, which is heuristically solved using a hybrid genetic algorithm. The result of this optimization serves as input for long-term planning. The short-term planning solution determines the injection of the nutrient mixture. It is worth noting that IoT sensors measure the electrical conductivity (EC) of the mixture, which serves as an indicator of the total amount of nutrients in the soil. However, they do not provide specific information on macronutrient needs. As a result, fertigation is implemented through a standard fertigation bank, where the EC and pH of the nutrient mixture are maintained within predefined values.

4. The proposed IoT-based framework

In this section we illustrate the proposed IoT-based framework. Fig. 3 depicts the main components of the proposed system. We describe each layer in the following section.

4.1. Physical layer

The physical layer includes fertigation channels and distribution lines, featuring, respectively, an inject pump and a solenoid valve controlled by IoT actuator node. When one or more fertigation channels are open, each open distribution line delivers fertilized water to crops associated with it. On the other hand, if no fertigation channel is open, the set of open distribution lines, if any, delivers irrigation water. Each distribution line is equipped with a set of IoT sensors nodes. Fig. 4 reports an example consisting of 4 fertigation channels and 6 distribution lines, each equipped with 3 IoT sensor nodes. It is worth noting that each single irrigation/fertigation decision targets a distribution channel, which may consist of one or more distribution lines. The number of lines associated with a single distribution channel defines the resolution level of irrigation/fertigation decisions, which can be customized according to the specific application setting.

4.2. IoT node layer

The IoT layer comprises hardware and software components that enable each physical device to function as an IoT node. This layer includes

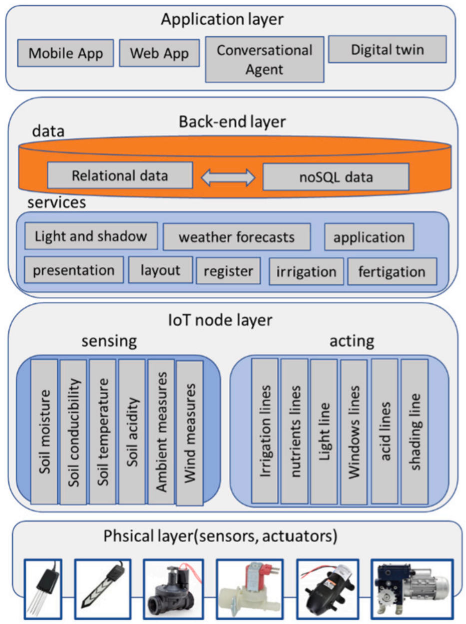


Fig. 3. The proposed IoT-based smart irrigation and fertigation system.

two types of IoT nodes: sensors and actuators. Each node in the IoT layer can request or provide services to the back-end component. IoT actuators are associated with fertigation channels and distribution channels in a one-to-one relationship. Conversely, each distribution channel has a one-to-many relationship with IoT sensors, while each IoT sensor is assigned to exactly one distribution channel. To ensure the modularity, extensibility, and resilience of the system, no hybrid nodes are designed to implement decisions (i.e., acting) and take measurements (i.e., sensing) simultaneously. It is worth noting that decision-making in current fertigation systems is mainly based on measuring EC and pH of the nutrient mixture. As aforementioned, the decision-making of the proposed framework is enabled by low-cost electrochemical sensors capable of measuring nitrogen, phosphorus, and potassium content in soil and/or substrates [9]. These sensors are ion-selective but not ion-specific, which implies non-negligible recovery times that must be properly accounted for. In the following section, we illustrate how these issues are handled by an information-to-action decision-making process, completely managed by the back-end component. This means that the ‘intelligence’ of the system is entirely confined to the back-end component, where services cooperate to make decisions. The only decision-making process retained at the IoT layer is local fault management.

4.3. Back-end layer

The back-end component of the proposed architecture is structured into two key areas: (1) a suite of microservices, and (2) a data management layer designed for systematic data organization and efficient querying to derive meaningful insights. A defining characteristic of the proposed framework lies in the structured organization of cloud-exposed services. Each service operates with a distinct identity and well-defined autonomy, enabling it to function independently while achieving its specific objectives. Furthermore, the shared back-end data repository fosters seamless integration among services, resulting in a hybrid microservices architecture that enhances resilience against service failures. The microservices can be invoked by both **IoT nodes** and **applications**, and they perform highly specialized tasks. For example:

- *Light and Shadow* responds to actuators that dim lights or control shading systems in a greenhouse environment.
- *Layout* optimizes transplant positions to maximize a **biodiversity index** (Adamo et al. [3]).
- *Register* and *Presentation* handle the **automatic registration** of an IoT node upon startup and the **sensor-to-actuator mapping**, respectively.
- *Irrigation* and *Fertigation* are services described in **Section 5** of this work.

This modular approach ensures that each microservice focuses on a specific functionality, promoting **scalability**, **flexibility**, and **ease of maintenance** across the architecture. In the back-end layer, data management is strategically divided into two distinct components to address the exponential growth of data generated by real-time data streams from IoT devices. As fresh data continuously flows into the system, maintaining query efficiency and scalability becomes a critical challenge. To overcome this, the architecture separates real-time operational data from historical datasets, optimizing both responsiveness and analytical capabilities:

1. *Relational data component.* Optimized for real-time decision-making, this component handles fresh, time-sensitive data essential for immediate DSS operations. It builds actionable knowledge ‘on the fly’ and efficiently manages data related to the plant layout and associated IoT nodes, ensuring minimal latency and maximum responsiveness.
2. *NoSQL component.* Designed for scalability and analytical workloads, this component stores historical data, enabling advanced descriptive and predictive analytics without burdening real-time operations. By offloading older datasets from the relational component, it preserves the efficiency of transactional queries while supporting long-term data retention and retrospective insights.

This separation ensures a clear division of responsibilities: the relational database focuses on operational agility, while the NoSQL database supports extensive historical analysis, creating a robust and scalable architecture tailored for both immediate responsiveness and long-term data-driven insights.

4.4. The data model

As aforementioned, all micro-services share a data repository managed by the relational component of the back-end layer (see Fig. 3). This repository encompasses two main areas: (1) the configuration of the planting area, which includes the arrangement of transplants, the positions of individual plants, and the parameters of agronomic species; and (2) real-time data from IoT nodes, consisting of measurements from sensors and the current status of actuators. In the relational model, the main entity is the IOT_NODE, which can be instantiated as a SENSOR or an ACTUATOR. Relationships between sensors and actuators are modeled as an oriented graph (Fig. 5a). This logic mapping is encoded through a many-to-many relationship between SENSORS and ACTUATORS, supported by a linking table (Fig. 5b). This enables the assignment of each sensor node to a set of selected actuators and vice versa. This modeling approach provides the foundation for defining (1) flexible network configurations of IoT nodes, (2) complex control rules during the system’s configuration stage, and (3) push-pull interaction policies among IoT nodes as described above. Latitude and longitude coordinates are attributes of the IOT_NODE entity, enabling the matching between physical nodes and their digital representation. Additionally, the IOT_NODE has attributes that represent its location within the planting area, specifically a pair of values referencing the distribution channel (i.e., entity LINE in Fig. 6) and the corresponding position (i.e., entity IRRIGATION_POINT in Fig. 6) where the physical node is located. This allows for the smart and precise delivery of nutrients and/or water exactly where and when it is needed. Moreover, the IoT nodes can be relocated or removed to prevent damage during harvesting operations.

The push pull communication paradigm is implemented by acquiring data from IoT nodes. Fig. 7 illustrates how the relational model repre-

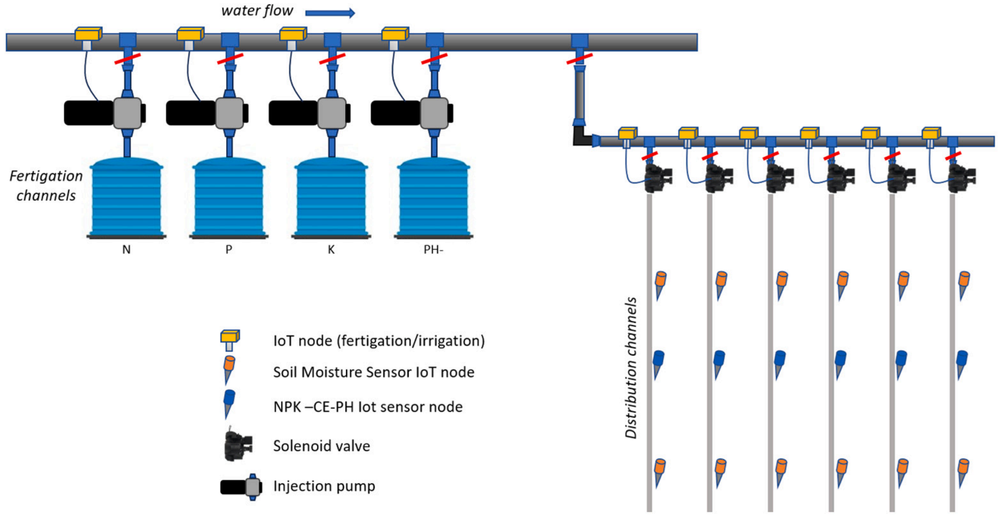
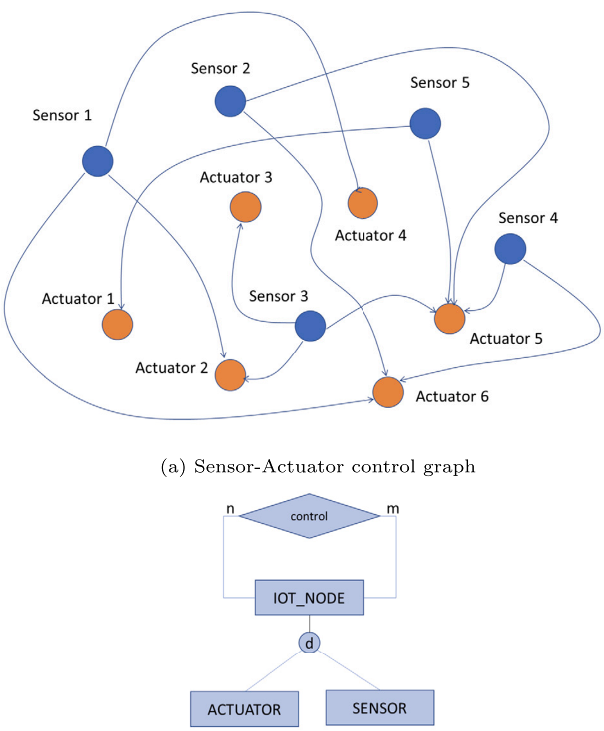


Fig. 4. IoT-based fertigation and distribution system with sensors.



(a) Sensor-Actuator control graph

(b) Actuator-Sensor Enhanced Entity Relationship model

Fig. 5.

sents these real-time data. Sensor IoT nodes *push* measurements, while actuator IoT nodes *pull* its next state. Both data type are stored as a response to a request sent by IoT node to a specific micro-service. Data are stored with their timestamps and in fully asynchronous mode.

4.5. Application layer

The **Application layer** serves as the **front-end interface** for end-users, providing access to a variety of tools and functionalities. Applications such as the *mobile App* and the *Digital Twin* have been developed to **validate decision-making processes**, **monitor system performance**, offer advanced **analytics capabilities**, and enable **intervention in case of malfunctions**. The Application layer leverages a set of cooperating **microservices** from the back-end layer to deliver functionalities such as **real-time reporting**, **data visualization**, and **instant alerts** for efficient crop and field management. Future developments include the integration of a **conversational agent**, designed to implement **text-to-action policies**, allowing users to interact with the system more intuitively (see Section 8).

The digital twin consists of a front-end component developed in Angular and a set of back-end REST API services managed through the Java Spring framework. These services enable real-time data transmission, collecting information from IoT nodes (sensor measurements and actuator states) and storing it in the NoSQL database component. The user interface is structured into two main sections: a 3D visualization module and advanced analytics dashboards. The 3D model of the greenhouse, designed in .FBX format, follows a modular structure of 8 by 10 meters. These modules are downloaded client-side and automatically assembled based on layout information stored in the relational database. Rendering is performed client-side using Three.js, integrated into the front end. All IoT nodes, including their 3D representations, are georeferenced according to a predefined coordinate system for each greenhouse, with the zero point consistently positioned at the front-left corner relative to the entrance. User interaction with the model allows node selection, triggering back-end events that retrieve and display relevant data. The same mechanism enables real-time interaction with the physical greenhouse, allowing users to control actuators in case of unexpected events, such as remotely opening or closing an electromechanical valve or manually triggering fertilizer injection. The analytics section provides advanced data visualization through interactive dashboards, offering real-time insights into key agronomic parameters such as nutrient levels (NPK), soil and environmental pH, moisture, temperature, and humidity. It also includes detailed monitoring of water and fertilizer consumption, allowing users to track resource usage over time, compare historical trends, and optimize irrigation and fertigation efficiency. By integrating dynamic charts and heatmaps, the system provides a clear overview of consumption patterns, helping to reduce waste and improve sustainability. Finally, the digital twin enables real-time 3D visualization of irrigation and fertigation decisions made by the AI service, dynamically animating specific sections of the irrigation system to enhance situational awareness and decision-making support.

4.6. The push-pull communication model

IoT-enabled smart farming systems face several critical requirements to operate effectively and efficiently. The communication paradigm, in particular, should properly address issues related to: (1) managing a large-scale network of IoT nodes; (2) asynchronous polling by sensor

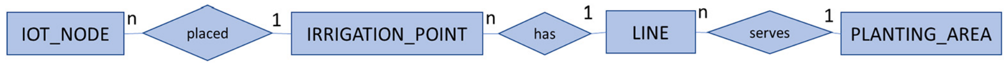


Fig. 6. Entity relationship for IoT sensor-actuation granularity.

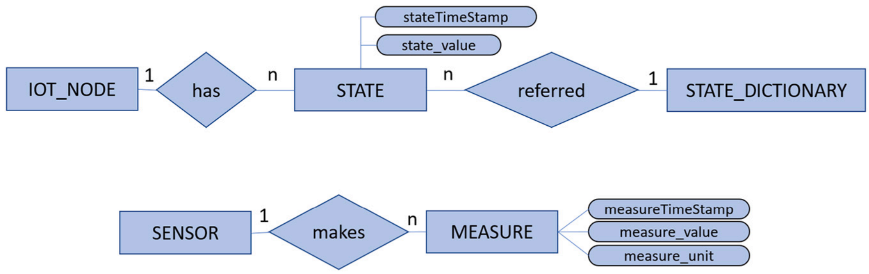


Fig. 7. Entity Relationship model for actuator-state and sensor data management.

and actuator IoT nodes; and (3) optimizing IoT node activity for energy saving.

In the proposed architecture, the interaction mechanism between IoT nodes and back-end components is based on a push-pull policy. This communication paradigm operates on the idea that IoT sensors can periodically send the collected data (e.g., soil moisture) to a back-end micro-service (*push*), making such data available for processing. If the data transmission rate is high (e.g., on the order of minutes), the system can react in real-time. Conversely, with a low transmission rate, the system cannot respond promptly, making it more suitable for monitoring applications.

Asynchronously, IoT-actuator nodes send requests to inquire (*pull*) about the next state to a gateway service. The gateway potentially invokes other decision-maker micro-services and then confirms the current state or communicates a new one (e.g., *start/stop irrigation of distribution channel 1*).

Such a push-pull mechanism offers several advantages. Firstly, it is suitable for large-scale applications where assigning a public IP address to every IoT node is impractical. The pull logic allows IoT nodes to operate with non-public addresses assigned by a DHCP server, as the responsibility for triggering events lies solely with the nodes. They only need to know the address of the back-end component exposing a gateway service.

Another benefit is that sensor and actuator nodes can asynchronously poll the system. Sensor nodes periodically send collected data (e.g., soil moisture levels, temperature) to the back-end system, while actuator nodes send requests to the back-end system to determine the next state or action (e.g., *start or stop irrigation*). This asynchronous polling allows for flexible and efficient data collection and decision-making processes without requiring direct interactions between sensor and actuator nodes.

In Fig. 8, a Unified Modeling Language (UML) activity diagram clarifies the push-pull logic of the proposed system Bahga & Madisetti [7].

It is important to note that IoT nodes often rely on battery power or renewable energy sources. Excessive activity, especially frequent polling or data transmission, can rapidly deplete their power reserves. Implementing efficient communication protocols and reducing unnecessary data transmissions are essential to prolonging the operational lifespan of IoT nodes in the field. Finally, it is worth noting that the back-end system adopts a serverless architecture, intentionally avoiding reliance on an orchestrator service. Each IoT node possesses direct knowledge of the specific REST API's reachability required to request a service. This design choice not only reduces system complexity and minimizes latency by removing intermediary layers, but also improves scalability and fault tolerance by enabling each node to operate independently (Rajan [38]).

5. Modelling fertigation/irrigation as a data-driven sequential decision problem

This study proposes modeling fertigation and irrigation decision-making as a data-driven sequential decision problem, implemented through a problem-solving autonomous agent [41]. The agent's decisions are guided by two vectors of state variables. The first vector represents the system state, encoding the current configuration of IoT actuators, each associated with either a distribution or a fertigation channel. Each component of the system state indicates whether the corresponding channel is currently open or closed. The second vector represents the environmental state, providing, for each distribution channel, a set of sensed values obtained from IoT sensors. The agent evaluates these state variables to optimize its decisions, aiming to satisfy nutrient and water demands while minimizing resource waste. Specifically, the agent processes real-time data from five types of IoT sensors, which quantify the nutrient and water needs of each distribution channel by measuring soil conditions, including moisture (H2O), nitrogen (N), phosphorus (P), potassium (K), and pH. This approach ensures that the agent selects actions that maximize overall system performance, balancing resource use efficiency and environmental sustainability.

Notation Decisions are encoded by a set of actions resulting in a new *system state*. In particular, the set $N = \{\mathrm{H2O}, \mathrm{N}, \mathrm{P}, \mathrm{K}, \mathrm{pH+}, \mathrm{pH\text{-}}\}$ encodes the set of IoT actuators used to satisfy nutrient/water demand and adjust soil pH values. The symbol H2O models a dummy IoT actuator needed to encode a state where the physical layer is delivering only irrigation water, i.e. at least one distribution channel open and all (physical) fertigation channels close. Moreover, symbols pH- and pH+ represent, respectively, two distinct channels devoted to lowering pH values (i.e., alkaline soil) and increasing pH values (i.e., acid soil). To ease the discussion and without loss of generality, we refer to $N = \{\mathrm{H2O}, \mathrm{N}, \mathrm{P}, \mathrm{K}, \mathrm{pH+}, \mathrm{pH\text{-}}\}$ as the set of fertigation channels or the set of nutrients.

Let $L = \{1, \dots, \ell\}$ denote the set of ℓ distribution channels. At each decision stage, for each distribution channel $j \in L$ and fertigation channel $i \in N$, the system compares the corresponding measurements (i.e., environmental state) to a minimum value LB_{ij} and a maximum value UB_{ij}. These threshold values are set by the agronomic advisor based on the crop species and the corresponding growth stage. Then, for each distribution channel $j \in L$, a set of categorical values a_{ij} is determined according to the following policy for $i \in N \setminus \{\mathrm{pH\text{-}}\}$:

- $a_{ij} = 0$ if the measured level of nutrient i belongs to the interval $[LB_{ij}, UB_{ij}]$;
- $a_{ij} = -1$ if the measured level of nutrient i is lower than the minimum value LB_{ij};
- $a_{ij} = 1$ if the measured level of nutrient i is greater than the maximum value UB_{ij}.

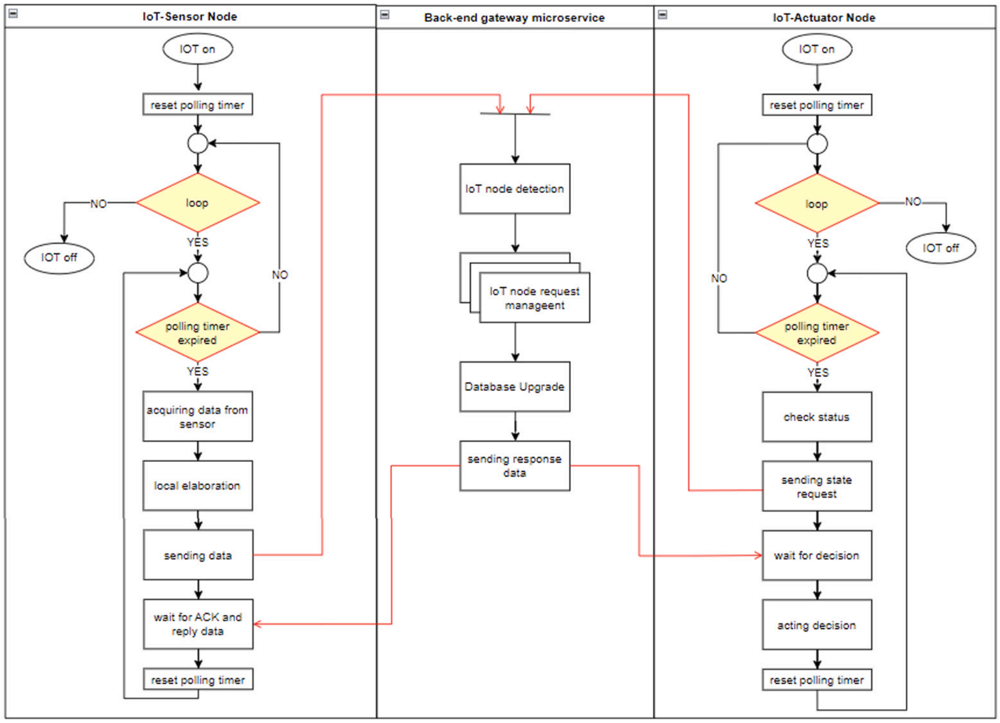


Fig. 8. UML activity diagram of the push-pull model of the proposed system.

A special case is represented by demands for lowering pH values for which $a_{\text{pH-},j} = -1 \times a_{\text{pH+},j}$ for any $j \in L$.

The symbol ζ denotes the current total nutrient demand, computed by counting the -1 entries in the matrix $a_{ij} (i \in N, j \in L)$, i.e.,

$$\zeta = \sum_{i=1}^{N} \sum_{j=1}^{L} -1 \times \min\{0, a_{ij}\}.$$

At each decision stage, the algorithm determines a set of actions that open/close channels so that the environment evolves to a new state corresponding to a possibly lower total nutrient demand, i.e., a lower value of ζ.

The ultimate objective is to satisfy every nutrient demand using a finite sequence of actions $\pi = (\alpha_1, \alpha_2, \dots, \alpha_T)$ so that the environment evolves from an initial state σ to the goal state τ, i.e. $\alpha_T \circ \cdots \circ \alpha_1(\sigma) = \tau$. The ultimate goal state τ is defined by $a_{ij} = 0$ for any $i \in L, j \in N$. When this environment state is reached, the prescribed action is to close every distribution channel $i \in L$ and fertigation channel $j \in N$.

A constructive heuristic policy $\mathcal{P}_1$ may satisfy each nutrient/water demand one by one. $\mathcal{P}_1$ defines an upper bound to the number of stages (up to ℓ) required to satisfy all nutrient/water requests. To model rule stating that IoT actuators $j, j' \in N$ cannot be open simultaneously, we use parameter $p_{jj'}$ for $j, j' \in N$: $p_{jj'} = 1$ if j and j' are incompatible, $p_{jj'} = 0$ otherwise. We also define set $A_i = \{j \mid j \in N : a_{ij} = -1\}$ of nutrients required by channel $i \in L$, and set $B_j = \{i \mid i \in L : a_{ij} = 1\}$ of channels with an high level of nutrient $j \in N$. Table 1 summarizes all used symbols.

6. An optimal policy for fertigation/irrigation decisions

We propose a policy that maps the system state to a fertigation/irrigation action. In particular the proposed policy is based on the solution of a linear program. The goal is to maximize at each stage the total nutrient/water demand. The binary variable x_i models the open/close decision of a distribution channel $i \in L$. The binary variable y_j models the open/close decision of the channel j to inject or not the nutrient $j \in N$. Finally, the binary variables z_{ij} are used to count the satisfied demands.

$$\max \quad \Big(\sum_{i\in L}\sum_{j\in A_i} z_{ij}, \sum_{i\in L} -x_i, \sum_{j\in N} -y_j\Big) \tag{1}$$

$$s.t.$$

$$x_i \le \sum_{j\in A_i} y_j \qquad i \in L \tag{2}$$

$$y_j \le \sum_{\substack{i\in L:\\ j\in A_i}} x_i \qquad j \in N \tag{3}$$

$$x_i + y_j \le 1 \qquad j \in N, i \in B_j \tag{4}$$

$$z_{ij} \le x_i \qquad i \in L, j \in N \tag{5}$$

$$z_{ij} \le y_j \qquad i \in L, j \in N \tag{6}$$

$$y_i + y_j \le 1 \qquad i, j \in N : p_{ij} = 1 \tag{7}$$

$$x_i \in \{0,1\} \qquad i \in L \tag{8}$$

$$y_j \in \{0,1\} \qquad j \in N \tag{9}$$

$$z_{ij} \in \{0,1\} \qquad i \in L, j \in N \tag{10}$$

The objective function maximizes (in lexicografic order) the number of satisfied nutrient demands, minimize waste of both water and fertilizer. Constraint (2) closes a distribution channel if none of the requested nutrients is open. In a dual way, the constraint (3) closes a fertigation channel if there is no open distribution channel among those that require it. Constraint (4) avoids over-fertigation. Constraints (5)-(6) connect the variables z to x and y. Constraints (7) prevent that two incompatible fertigation channels are open. Domains of the variables are encoded by (8)-(10).

Variable fixing At each decision stage further sets $L_1 \subseteq L$, $L_2 \subseteq L$ and $N_1 \subseteq N$ are defined and the following variable fixing are included in the constraints set of the linear integer program:

$$\max \quad \Big(\sum_{i\in L}\sum_{j\in A_i} z_{ij}, \sum_{i\in L} -x_i, \sum_{j\in N} -y_j\Big)$$

$$s.t. \quad (2) - (10)$$

Table 1
Index sets, parameters and decision variables.

Index sets	
L	Set of distribution channels
N	Set of fertigation channels
L_1	Subset of distribution channels open in fertigation mode
L_2	Subset of distribution channels closed in recovery mode
N_1	Subset of fertigation channels j with $a_{ij} = -1$ and $i \in L_1$
A_i	$= \{j \mid j \in N : a_{ij} = -1\}$ with $i \in L$
B_j	$= \{i \mid i \in L : a_{ij} = 1\}$ with $j \in N$
Parameters	
LB_{ij}	lower threshold value for IoT sensor associated to $j \in N$ and distribution channel $i \in L$
UB_{ij}	upper threshold value for IoT sensor associated to $j \in N$ and distribution channel $i \in L$
a_{ij}	categorical value associated to sensed data with $i \in L, j \in N$
$p_{j_1 j_2}$	binary coefficient stating if nutrient $j_1 \in N$ and $j_2 \in N$ are compatible
Δ	time interval between two consecutive decision stage
Δ_i^1	duration of fertigation/irrigation time window for the distribution channel $i \in L$
Δ_i^2	duration of recovery time window for the distribution channel $i \in L$
Binary variables	
z_{ij}	equal to 1 if demand $j \in N$ from distribution channel $i \in L$ is satisfied, 0 otherwise.
x_i	equal to 1 if distribution channel $i \in L$ is open, 0 otherwise
y_j	equal to 1 if fertigation channel $j \in N$ is open, 0 otherwise

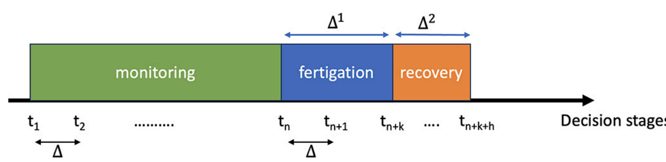


Fig. 9. Basic sequence of stage changes for a distribution channel.

$$x_i = 1 \qquad i \in L_1 \tag{11}$$

$$x_i = 0 \qquad i \in L_2 \tag{12}$$

$$y_j = 1 \qquad j \in N_1 \tag{13}$$

Variable fixing constraints aim to model sensor recovery times for ion-specific and ion-selective sensors. Recovery times refer to the period required for the sensors to return to their baseline readings or to a stable state after being exposed to a particular ion concentration. This is crucial for ensuring accurate and reliable measurements. For these reasons, the status of distribution channel i can be of three types:

- *monitoring mode*, i.e. close status with $i \notin L_1 \cup L_2$;
- *fertigation mode*, i.e. open status with $i \in L_1$;
- *recovery mode*, i.e. close status with $i \in L_2$.

A distribution channel is managed by repeating a basic sequence of state changes, *close-in-monitoring-mode→open-in-fertigation-mode→close-in-recovery-mode* (see Fig. 9). Categorical values a_{ij} are updated only for distribution channels that are *close-in-monitoring-mode*, i.e. $i \notin L_1 \cup L_2$. In the following we describe the three main steps associated to each basic sequence of Fig. 9.

Transition from monitoring mode to fertigation mode If at the decision stage t, the optimal solution $(\mathbf{x}^*, \mathbf{y}^*)$ prescribes to open a distribution channel $\bar{i} \notin L_1 \cup L_2$, then the portion of system status related to distribution channel $\bar{i}$ is updated according to the following four-step procedure.

STEP 1. The irrigation/fertigation time window $TW^1_{t\bar{i}} = [t, t + \Delta^1_{\bar{i}}]$ is started;
STEP 2. The overall system status is updated according to $(\mathbf{x}^*, \mathbf{y}^*)$,
STEP 3. The distribution channel $\bar{i}$ is included in L_1,
STEP 4. For each fertigation channel $j \notin N_1$ if $a_{\bar{i}j} = -1$ and $y^*_j = 1$, then the fertigation channel j is added to N_1.

At the next decision stage $t' > t$ if the irrigation/fertigation time window is expired, i.e. $t' > t + \Delta^1_{\bar{i}}$, then the system closes the distribution channel $\bar{i}$ currently *open-in-fertigation-mode*. This is done by updating the portion of system status related to distribution channel $\bar{i}$ as follows.

Transition from fertigation mode to recovery mode

STEP 1. The recovery time window $TW^2_{t\bar{i}} = [t', t' + \Delta^2_{\bar{i}}]$ is started;
STEP 2. The status of distribution channel $\bar{i}$ is set to *close*,
STEP 3. The distribution channel $\bar{i}$ is removed from L_1, and added to L_2.

It is worth noting that to remove a fertigation channel j from N_1 and close it, we need to check that no distribution channel $i \in L_1$ is such that $a_{ij} = -1$. At the next decision stage $t'' > t'$, if the recovery time window is expired, i.e. $t'' > t' + \Delta^2_{\bar{i}}$, then the final step of the basic sequence takes place as follows.

Transition from recovery mode to monitoring mode This transition requires a single step

STEP 1. The distribution channel $\bar{i}$ is removed from L_2.

We finally observe that time windows durations Δ^1_i and Δ^2_i are input parameters set by the agronomic advisor. In particular their values are multiple of the time interval Δ occurring betweentwo consecutive decision stages (see Fig. 9).

Fertigation/irrigation micro-service Summing up, at each decision stage our algorithm executes the following steps:

STEP 1. Check expired fertigation time windows and apply transitions from fertigation mode to recovery mode
STEP 2. Check expired recovery time windows and apply transitions from recovery mode to monitoring mode
STEP 3. Update categorical values a_{ij} for $i \notin L_1 \cup L_2$, $j \notin N_1$.
STEP 4. Counts the total number of $a_{ij} = -1$, if 0 go to final state (close all channels), STOP;

STEP 5. Determine $(\mathbf{x}^*, \mathbf{y}^*)$ by solving the linear integer program (1)-(13);
STEP 6. Compute the objective function of current system state.
STEP 7. If $(\mathbf{x}^*, \mathbf{y}^*)$ dominates current system state, then apply transitions from monitoring mode to fertigation mode

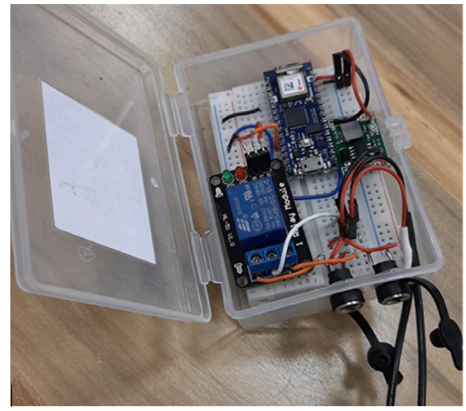
7. System implementation details

This section focuses on the implementation details of the system layers reported in Fig. 3.

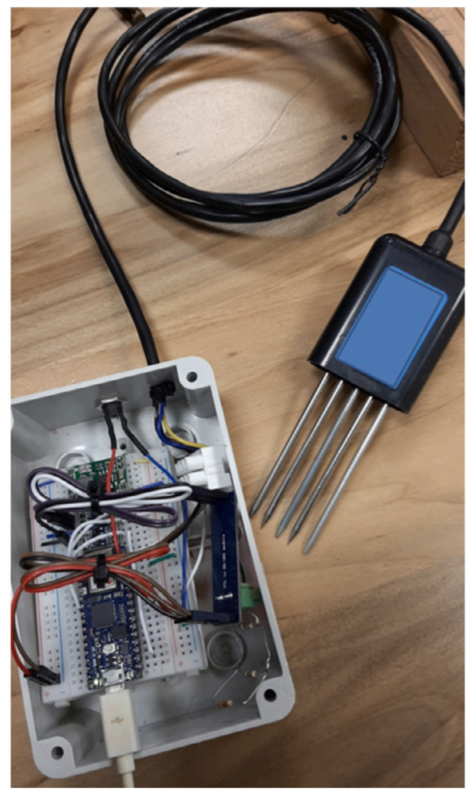
Physical layer We start by observing that each single distribution channel is equipped with one IoT node actuator, which determines its state, i.e., open or closed. This implies that the system's field flow rate is dynamically determined by the number of distribution channels simultaneously open. To properly address this time-dependent flow rate, the IoT actuator associated with each fertigation channel has been designed to vary its injection rate in real-time. The main goal is to deliver fertilized water to each individual crop with a *drop* concentration equal to 1. It operates on a 5 V DC power supply, drawing a maximum current of 100 mA. Communication with the IoT Micro Controller Unit (MCU) is facilitated via RS485 interface. Each distribution channel features a solenoid valve controlled by an IoT node, with the microcontroller being the Arduino NANO 33 IoT module. Additionally, another IoT node (also managed by the Arduino NANO 33 IoT) is installed on each irrigation line, equipped with an electrochemical sensor that measures values such as Nitrogen (N), Phosphorus (P), Potassium (K) in mg/kg, electrical conductivity (μs/cm), temperature (C), pH, and moisture. The NPK sensor features a measurement range of 0-1000 mg/kg for nitrogen and 0-500 mg/kg for both phosphorus and potassium, with an accuracy of ±5% for each parameter. The sensor offers a response time of 60 seconds for all three elements.

IoT node layer The Arduino Nano 33 IoT MCU has been utilized for both sensor and actuator nodes. In Fig. 10, the proof of concept of the IoT actuator node (Fig. 10b) and the IoT sensor node (Fig. 10a) are presented. The Arduino Nano 33 IoT board features an Arm Cortex-M0+ microcontroller running at 48 MHz, with 256 KB of Flash memory and 32 KB of SRAM. It includes a built-in WiFi and Bluetooth connectivity, allowing for seamless IoT integration. Additionally, it has a variety of digital and analog pins for versatile sensor and actuator interfacing. The board supports a wide range of programming languages and development environments. For the system proposed the IoT node has been programmed in C++ under Arduino integrated development environment.

Regarding IoT node power supply and energy consumption, in greenhouse applications, IoT sensor nodes are directly powered by the electrical grid, operating at a continuous voltage of 12 V for sensors and 24 V for actuator nodes. Greenhouses are generally equipped with a stable power supply, making direct electrical connection the most efficient solution. However, in open-field applications, energy consumption becomes a critical factor. IoT nodes are designed to operate autonomously using a combination of solar power and battery storage. Each node is equipped with a 12,000 mAh LiFePO4 battery and a 10 W photovoltaic panel, ensuring energy self-sufficiency. The solar panel recharges the battery during the day, providing enough power to sustain operations even at night or during overcast conditions. Additionally, sensor data is aggregated into a single JSON packet before transmission, optimizing power efficiency by reducing communication overhead. A key factor in energy management is that irrigation and fertigation are not performed at night, as plant metabolism slows down. As a result, during nighttime hours, IoT nodes operate in their lowest power mode (0.2 W idle state), further reducing energy consumption. The estimated battery autonomy for different types of nodes, without solar recharging, is as follows:



(a) IoT actuator node



(b) IoT sensor node

Fig. 10. The proof of concept of the IoT nodes.

- Sensor nodes, which operate continuously at 0.8 W, can sustain operations for approximately 180 hours (7.5 days) on a full battery charge.
- Irrigation actuator nodes, which remain in idle mode (0.2 W) at night and activate solenoid valves only during the day (7 W for 10 minutes per cycle, 3 times per day), can operate for approximately 421 hours (17.5 days) before requiring a recharge.

With the integration of the solar panel, the system is effectively energy self-sufficient, as the daily energy generated (50 Wh) exceeds the consumption needs of both sensor and actuator nodes. This ensures continuous operation even in variable weather conditions and enables long-term deployment with minimal maintenance. To further enhance efficiency and scalability, future developments will incorporate a lightweight publish-subscribe protocol, particularly for solenoid valve management, to optimize power consumption and network communication in large-scale deployments. Additionally, an NB-IoT module will be integrated to improve connectivity in remote agricultural areas.

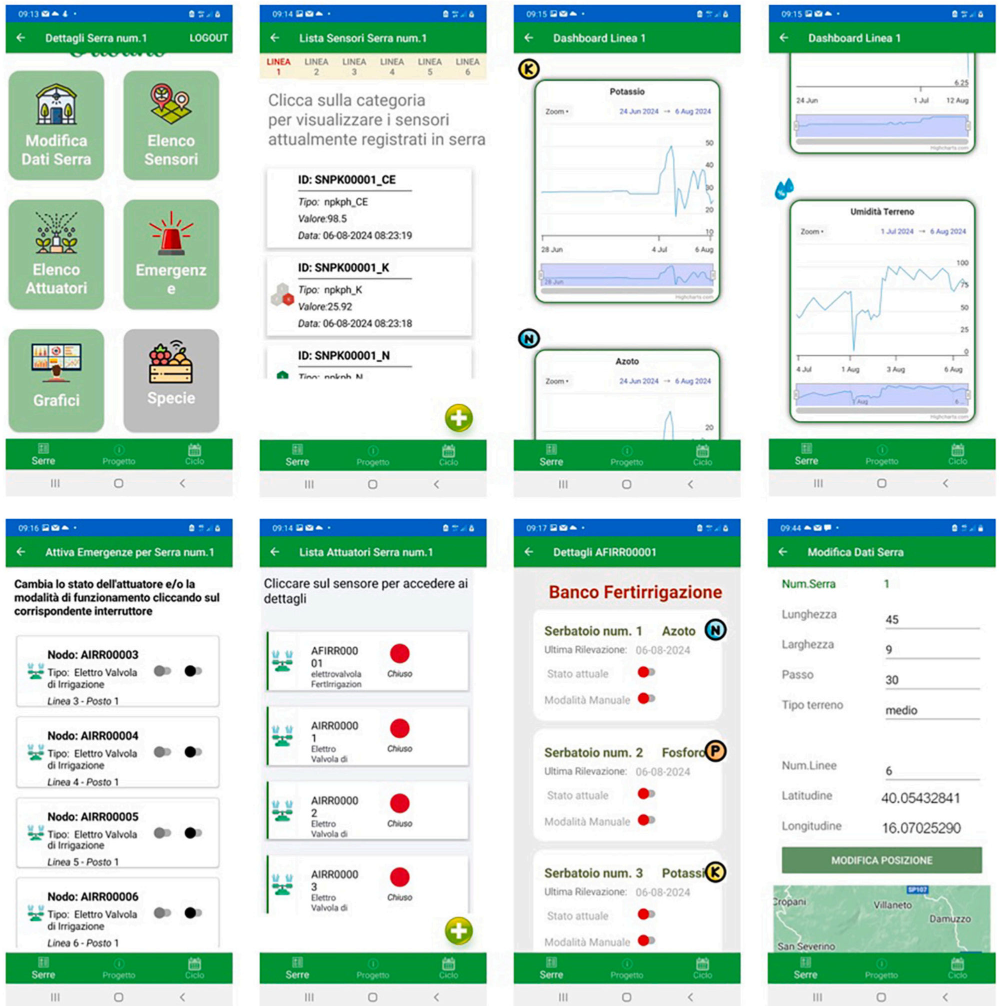


Fig. 11. Details of the management mobile app.

If the current downward trend in NB-IoT traffic costs continues, this solution could become a cost-effective and scalable approach for large-scale IoT-based agricultural systems, ensuring reliable and widespread connectivity. This design guarantees high resilience of the IoT nodes, enabling autonomous operation in diverse agricultural environments, from greenhouses with stable grid power to open-field deployments relying on solar energy.

Back-end layer The relational database stores the knowledge base of our system and has been implemented using the open-source MySQL database management system. It records every end user request and response. Its main purpose is to manage all information related to irrigation layouts, threshold values for each species, agricultural cycles (opening and closing), sensor measurements, and actuator states. The No-SQL data component is used to offload data related to measurements and actuator states. This data, when not involved in decision-making, can be transferred from the relational database to the non-relational database for analytical purposes. This component, anticipated in the architecture, will be the subject of future developments.

The irrigation/fertigation micro-service has been coded in Java using Spring Boot framework responsible for back-end validation, ILP problem instance encoding and resolution through the invocation of ILP solver micro-service.

It is worth noting that the ILP model (1)-(13) is characterized by few hundred variables and constraints, which made it suitable to be solved by a commercial solver. For these reasons the *ILP solver micro-service* runs an off-the-shelf black-box ILP solver in order to provide a solution for a given ILP problem instance. We implemented the ILP model (1)-(13) using the Optimization Programming Language [48], a proprietary modeling language equipped with a repertoire of functions tailored for CPLEX [21], the commercial ILP solver developed by IBM ILOG. The computational results demonstrated that the proposed approach is not energy-intensive to run. It only takes a few minutes to generate optimal solutions using computational resources typically available on a laptop or smartphone. We finally observe that the framework also includes the *The CPLP micro-service* proposed by Adamo et al. [3] and invoked by the mobile application to define the crop planting layout.

Application layer The mobile management application is a user-friendly native application developed for iOS and Android using Microsoft Xamarin. In addition to handling end-user input errors, it transparently invokes all services exposed on the cloud in the back-end component. There is no direct connection between the application and the irrigation/fertilization micro-service. Instead, the mobile application communicates with the irrigation/fertilization gateway service to provide essential information: the configuration of the irrigation system, the configuration of the fertigation channels, and the selected plant species. The gateway then invokes the decision-making service and records all data consistently in the relational database. Thanks to this architectural model, most improvements and corrective actions made on the gateway do not require reinstallation of the mobile app on the farmer's smartphone. Fig. 11 reports some representative screens captured by the mobile application. To adequately support agronomic experiments, a Digital Twin was developed (Fig. 12), which allows the user to: (a) vi-

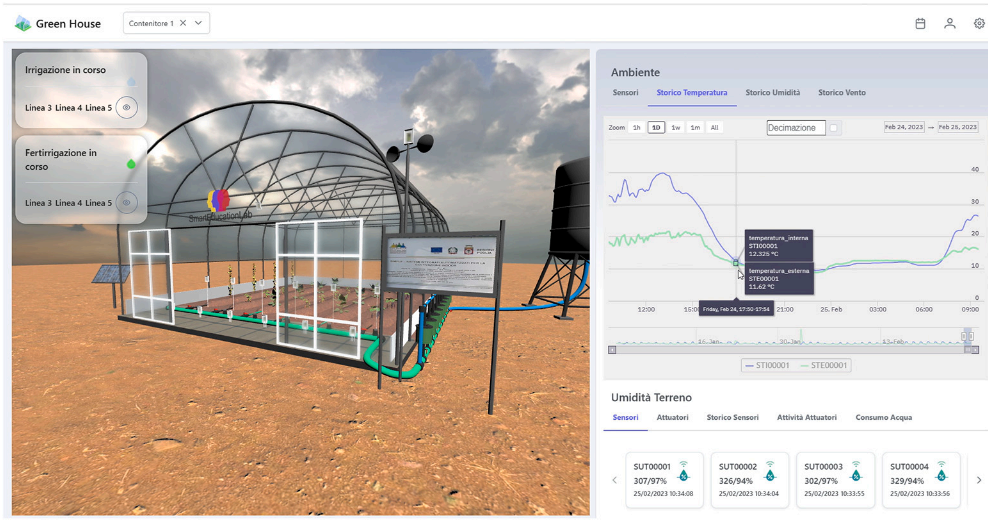


Fig. 12. Digital Twin Graphical User Interface.

sualize the state of the fertigation system in real time, including sensor measurements, actuator statuses, and decisions made by the decision-making model; (b) analyze historical data to identify trends and optimize irrigation and fertigation strategies through parameter tuning, such as Δ^1, Δ^2, LB_{ij}, and UB_{ij}.

8. Tests, results and discussion

This work is part of a research project that aims to test the potential of decision-making systems based on IoT and artificial intelligence in various agricultural processes. The proposed framework was tested in two different greenhouses. The former with a mixed layout, both in terms of cultivated species and agronomic techniques (soil and soilless); the latter with only one species (strawberries) cultivated with soilless technique. This choice provided a test field to verify the system adaptability in setting requiring different demand patterns in terms of water and nutrients. The main objective was to evaluate the robustness of the proposed system. In this direction, the soilless open-loop cultivation was particularly significant. The location of the greenhouses (South-east Italy) and the chosen soilless cultivation (strawberries) implied an extremely vulnerable configuration with respect to irrigation and fertigation stress.

In contrast, the soil-based experiment aimed to evaluate the robustness of the proposed system in managing distribution channels with heterogeneity in both soil and the chosen cultivation. The configuration parameters for the planting areas were provided by the farmers involved in the research project as stakeholders.

8.1. First experiment

The first experiment was carried out from January 11, 2024, to May 31, 2024, in a greenhouse located in Apulia, Italy. The planting layout was designed according to principles known as “strip intercropping”, guided by organic and sustainable agricultural practices [3]. These agronomic techniques are part of what is known as ‘synergistic agriculture.’ Specifically, synergistic agriculture involves increasing the number of species per square meter to trigger positive interactions between them, thus improving soil biodiversity and strengthening the resilience of the cropping system.

The objective of this first experiment was twofold: firstly, to verify the validity of the approach by ensuring the healthiness of the plants at each of their phenological stages and to verify that each plant would bear fruit; and secondly, to ensure an adequate level of yields.

Six distribution channels, numbered from 1 to 6, were installed in the greenhouse. Each distribution channel accommodated a specific number of plants of the same species, spaced according to given transplanting distances. The planting layout was:

- CHANNEL 1: tomato,
- CHANNEL 2: tomato,
- CHANNEL 3: soilless tomato,
- CHANNEL 4: zucchini,
- CHANNEL 5: cucumber,
- CHANNEL 6: soilless strawberry.

The system was configured with three fertigationchannels, a pH reducer channel (pH-).

Table 2 details for each fertigationchannel the corresponding nutrient solution.

The system carries out an initialization step for moisture and pH. In particular, an initial irrigation phase is started to bring the substrate/soil to a moisture level of 20%, defined in accordance with the technical specifications of the NPK sensor, which stipulates a minimum moisture level for reliable measurements. Then, the system injects the acid channel (pH-) to adjust the pH level where necessary. Once the initialization step is completed, the proposed framework autonomously irrigated and fertigated the greenhouse for four months. Fig. 13 shows the evolution over time of the species grown along the distribution channels. As can be seen, all plants developed and fruited consistently with their phenological phases. Table 3 reports average plant production detailed for each species.

At the end of the growing cycle, all plants were healthy, except for the cucumbers, which were attacked by powdery mildew after two harvest cycles. Since no pest treatment was applied, the cucumber plants eventually died. The observed production values are consistent with the expected average yields. Additionally, the soilless cherry tomato cultivation exhibited a reduced leaf area and a slight decrease in the average fruit size. This observation led to the implementation of corrective adjustments to the nutrient threshold values for tomatoes grown in soilless systems.

8.2. The second experiment

The second experiment was carried out from June 1, 2024, to July 31, 2024, in an industrial setting devoted to the production of strawberries grown using a soilless method in Abruzzo, Italy. The proposed system was installed in an industrial greenhouse measuring 8 m x 40 m. Table 4 reports comparative analysis of fertilizer consumption between the proposed system and the conventional fertigation system installed in an adjacent greenhouse. This conventional system was equipped with a

Table 2
Fertigation channels and the corresponding nutrient solutions.

Channel	Components
Nitrogen (N)	Calcium nitrate for agricultural use, Ammonium nitrate, Chelated iron EDDHA, Microelements
Phosphorus (P)	Monopotassium phosphate, Chelated iron EDDHA, Microelements
Potassium (K)	Potassium nitrate, Potassium sulfate, Chelated iron EDDHA, Microelements
pH reducer	Nitric acid

Fig. 13. Plant growth from transplanting to the fourth month.

Table 3
Average production per plant for crops.

Species	Average Production per Plant
Cherry Tomato (soil cultivation)	2.5 kg
Cherry Tomato (soilless cultivation)	3.2 kg
Zucchini (soil cultivation)	5.8 kg
Cucumber (soil cultivation)	3.4 kg
Strawberry (soilless cultivation)	0.93 kg

Dosatron device (differential piston injection system) [18], configured according to the classic three-tank setup (nutrient solutions A, B, and PH-), as illustrated in Fig. 2a. Column headings are self-explanatory. The proposed system always provided a saving, with an average value above 50%. In particular, the conventional system was programmed by the farmer with a daily schedule, consisting of 9 fertigation time windows, with a constant 5-minute duration.

As far as the farmer irrigation policy is concerned, the conventional system was never utilized to delivery only irrigation water. The proposed IoT based system saved 30% of water consumption. This implies that while the conventional system has consistently over-fertigated, the proposed system remarkably reduced the consumption of water and fertilizers by exploiting more efficiently nutrients naturally present in the irrigation water. As in the first experiment plants developed and fruited consistently with their phenological phases. We finally observe that the user-friendliness of the mobile app played a central role in enhancing the way of thinking of the farmer involved in the experiment. In particular, the remarkable fertilizer and water savings have marked the initial stride towards integrating the research project's outcomes into a large-scale industrial setting.

8.3. Discussion

A substantial body of research has explored the use of AI algorithms and IoT technologies for the precision management of water and fertilizer resources, laying a critical groundwork for our study. Table 5 provides a general comparison between the proposed solution and recent contributions. Several notable gaps remain in the existing literature:

- Most current studies focus on applying control technology to replenish water and fertilizer for specific crops at certain growth stages. Decision-making is often based on rule-based approaches, which lack explicit optimization goals. Few studies have tackled the issue of fertilizer optimization from the perspective of individual macronutrients, such as Nitrogen, Phosphorus, and Potassium.
- While existing IoT-based smart fertigation systems enable real-time irrigation and fertilization based on environmental and crop growth data, they often overlook the fine-grained, information-to-action decisions necessary to address the precise water and nutrient needs of crops along individual distribution lines.

Table 4
Fertilizer consumption per day per linear meter.

Fertilizer	Conventional System (A)	Proposed System (B)	Saving% $= \frac{A-B}{A}$
Nitric Acid	0.000415 liters	0.000218 liters	47.5%
Calcium Nitrate	0.000400 Kg	0.000086 Kg	78.50%
Ammonium Nitrate	0.000067 Kg	0.000032 Kg	52.20%
Potassium Nitrate	0.000195 Kg	0.000081 Kg	58.50%
Magnesium Nitrate	0.000108 Kg	0.000014 Kg	87.00%
Monopotassium Phosphate	0.000168 Kg	0.000067 Kg	60.12%
Potassium Sulfate	0.000325 Kg	0.000122 Kg	62.50%
Chelated Iron	0.000020 Kg	0.000011 Kg	45.00%
Microelements	0.000009 Kg	0.000004 Kg	55.55%

Table 5
Comparison of the proposed solution with some new previous proposed contributions.

Study	Purpose/application Area	System architecture	Communication strategy	Algorithm
Dahane et al. [16]	Irrigation monitoring and acting	Edge-Fog-Cloud	Polling to Edge level	Deep learning (LSTM, GRU)
Trilles et al. [46]	Irrigation monitoring and recommendation	Serverless micro-services	MQTT M2M connectivity on 3G connectivity	based on micro-services implemented
Guardo et al. [20]	Irrigation monitoring and acting	two-tier Fog	Lora Protocol and MQTT over mobile network	clustering algorithm
Kaur et al. [22]	Irrigation monitoring	ThingSpeak services	http polling to ThingSpeak services on WiFi network	K-nearest neighbors
Benyezza et al. [11]	Irrigation monitoring and acting	Fog level in local server and cloud services	wireless sensors network (WSN) based on radio frequency (RF)	fuzzy logic controller (FLC)
Rosli et al. [40]	Fertigation acting and monitoring	Firebase serverless cloud services	http on WiFi network	IF-THEN rules on EC and pH values
Ahmed et al. [5]	Fertigation acting and monitoring	Client-server on ethernet	xBee	IF-THEN rules on EC and pH values
Rode et al. [39]	Fertigation acting	Stand-alone system	Wired connection	IF-THEN rules on EC and pH values
Parimala et al. [36]	Fertigation acting	ThingSpeak services	http polling to ThingSpeak services on WiFi network	IF-THEN rules on EC and pH values
Anuar et al. [6]	Fertigation acting	Stand-alone proof of concept	Wired connection	IF-THEN rules on EC and pH values
Zailani & Jumaat [50]	moisture monitoring	Blynk client-server	http on WiFi network	No decision making
Cheruvu et al. [14]	Fertigation acting	ThingSpeak services	http polling to ThingSpeak services on WiFi network	ML Random Forest
Lin et al. [28]	Fertigation monitoring and acting	client-server	Wireless Sensor Network (WSN) with RF communication, NRF24L01+ ShockBurst protocol	Hybrid genetic algorithm optimizing fertigation decisions.
Proposed Method	Fertigation/irrigation monitoring and acting	serverless micro-services	Push-Pull	Mathematical Programming approach optimizing fertigation/irrigation decisions.

We contribute to bridging these gaps by modeling fertigation/irrigation decision-making as a data-driven sequential decision problem. Specifically, we propose a policy that maps the state (i.e., macronutrient needs at the distribution line level) to an action (i.e., open/close distribution/fertigation channels). The proposed policy leverages the solution of a linear integer programming model. To the best of our knowledge, this is the first contribution to propose an IoT-based framework that relies on information-to-action decisions aimed at fine-tuning the supply of individual macronutrients and water to each distribution channel. Nevertheless, there are two main limitations underscoring the importance of further research and development to make the proposed system more resilient and adaptable to various agricultural contexts.

The first limitation lies in the assumption of unlimited water availability for irrigation and fertigation. This hypothesis contrasts with the reality of many agricultural regions, where water scarcity is a growing challenge. Keesstra et al. [23] emphasizes that optimizing water use and its equitable allocation are fundamental to the sustainability of agriculture. To make the proposed system more effective and adaptable, it is necessary to incorporate water availability constraints into the decision-making process. For instance, the parameters Δ^1 and Δ^2 (representing the duration of the irrigation/fertigation time window and the sensor recovery time, respectively) could be integrated into the integer linear programming model as decision variables. This enhancement would enable the system to adapt to periods of limited water availability. A look-ahead policy could also be developed to proactively optimize resource allocation and irrigation strategies, considering the specific needs of different crops under different scenarios of water restriction.

The second limitation stems from the use of predefined thresholds for agronomic inputs (i.e., parameters LB_{ij} and UB_{ij} of the decision model), determined based on general crop uptake models. These thresholds do not account for site-specific variability or real-time crop conditions. Future developments could leverage generative AI, allowing decision makers (i.e. agronomic advisors) to input observations of crop conditions in a more natural, intuitive, and adaptable manner. For instance, a farmer might input the following text to a Large Language Model:
Soilless strawberry leaves exhibit dark green margins and apices, with central areas fading to light green near the primary and secondary veins. The leaf tips show a slight brown discoloration, which darkens towards the edge.
The system would then combine this description with a structured database to identify species-specific thresholds for optimizing nutrient levels (Qin et al. [37]). The generative-AI service could respond with recommendations such as:
Step 1: Adjusting nitrogen (N), phosphorus (P), and potassium (K) levels based on symptoms.
Step 2: Generating SQL queries to update thresholds in the database:

UPDATE SPECIE

$$\begin{aligned}\text{SET } & N_{\text{min}} = 19,\ N_{\text{max}} = 24,\\ & P_{\text{min}} = 15.5,\ P_{\text{max}} = 20.5,\\ & K_{\text{min}} = 46,\ K_{\text{max}} = 51,\\ & \text{ph}_{\text{min}} = 5.5,\ \text{ph}_{\text{max}} = 6.2\\ \text{WHERE } & \text{pk\_specie} = 141;\end{aligned}$$

The system then directly forwards these queries to the database, enabling automatic tuning of thresholds. To ensure system stability and prevent harmful actions, such services would need to be paired with quality evaluation modules to validate the AI-generated responses before execution.

9. Conclusions

This paper presents a micro-services IoT framework for managing irrigation and fertigation, leveraging an asynchronous push-pull paradigm to enhance flexibility, extensibility, and resilience. The core innovation lies in fine-grained decision-making for supplying macronutrients and water, modeled as a data-driven sequential decision problem solved through integer linear programming. This approach surpasses traditional PID controllers by holistically managing interconnected variables and constraints.
The system was tested in soil and soilless greenhouses, achieving significant resource savings: 47.5% less nitric acid, 67.7% less macronutrients, and 30% less water compared to conventional systems. Despite its efficacy, two key limitations remain. First, the assumption of unlimited water availability is unrealistic in many agricultural contexts. Future work should integrate water scarcity constraints into the model, treating parameters like irrigation/fertigation time windows as decision variables.
Second, the use of predefined thresholds for agronomic inputs does not account for site-specific variability or real-time crop conditions. Generative AI could address this by enabling intuitive farmer inputs (e.g., natural language descriptions of crop conditions) to dynamically adjust thresholds and optimize nutrient levels. Pairing AI with quality evaluation modules would ensure system reliability.
Finally, reinforcement learning could further enhance decision-making by integrating learned heuristics or adaptive policies into the optimization framework. Addressing these limitations will make the system more resilient and adaptable, offering a robust tool for sustainable and precision agriculture.

CRediT authorship contribution statement

Tommaso Adamo: Writing – review & editing, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Danilo Caivano:** Writing – review & editing, Validation, Methodology, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Lucio Colizzi:** Writing – original draft, Validation, Supervision, Software, Methodology, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Giovanni Dimauro:** Writing – review & editing, Validation, Methodology, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Emanuela Guerriero:** Writing – original draft, Validation, Supervision, Software, Methodology, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors are grateful to Dr. Nicola Ciminielli (Io-Basilicata), the farmer involved in the experiment, for his collaboration in providing data and information. The authors also thank Eng. Loredana Verardi (SmartEducationLab) for the deployment of the overall system in an industrial setting. This work was supported by the Ministero dell'Università e della Ricerca (Progetti DM 1062 del 10/08/2021 - Azione IV.4 - Azione IV.6 - cod. 02-G-14860-1) and the Agritech National Research Center, funded by the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4 – D.D. 1032 17/06/2022, CN00000022). This manuscript reflects only the authors' views and opinions; neither the European Union nor the European Commission can be held responsible for them.

Data availability

Data will be made available on request.

References

[1] E.A. Abioye, M.S.Z. Abidin, M.S.A. Mahmud, S. Buyamin, M.H.I. Ishak, M.K.I.A. Rahman, A.O. Otuoze, P. Onotu, M.S.A. Ramli, A review on monitoring and advanced control strategies for precision irrigation, Comput. Electron. Agric. 173 (2020) 105441, https://doi.org/10.1016/j.compag.2020.105441, https://www.sciencedirect.com/science/article/pii/S0168169919314826.
[2] E.A. Abioye, O. Hensel, T.J. Esau, O. Elijah, M.S.Z. Abidin, A.S. Ayobami, O. Yerima, A. Nasirahmadi, Precision irrigation management using machine learning and digital farming solutions, AgriEngineering 4 (2022) 70–103.
[3] T. Adamo, L. Colizzi, G. Dimauro, E. Guerriero, D. Pareo, Crop planting layout optimization in sustainable agriculture: a constraint programming approach, Comput. Electron. Agric. 224 (2024) 109162.
[4] Agricontrol, Unitá di Fertirrigazione Computerizzate AGRIMIX-L, http://www.agricontrol.it/wp-content/uploads/2013/09/schema-agrimix-LK-1024x718.jpg, 2024. (Accessed 6 November 2024).
[5] O.M.E. Ahmed, A.A. Osman, S.D. Awadalkarim, A design of an automated fertigation system using IoT, in: 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), IEEE, 2018, pp. 1–5.
[6] N.S. Anuar, Z.B. Zamani, A.N. Mohamad, A.A. Safiqa, N.A. Sabirrullah, Microcontroller-based fertigation farming automation system, EDUCATUM J. Sci. Math. Technol. 9 (2022) 49–55.
[7] A. Bahga, V. Madisetti, Internet of Things: A Hands-on Approach, Vpt, 2014.
[8] M. Bassett, D. Winterbone, R. Pearson, Calculation of steady flow pressure loss coefficients for pipe junctions, Proc. Inst. Mech. Eng., Part C, J. Mech. Eng. Sci. 215 (2001) 861–881.
[9] A. Bellosta-Diest, M.Á. Campo-Bescós, J. Zapatería-Miranda, J. Casalí, L.M. Arregui, Evaluation of nitrate soil probes for a more sustainable agriculture, Sensors 22 (2022) 9288.
[10] H. Benyezza, M. Bouhedda, K. Djellout, A. Saidi, Smart irrigation system based thingspeak and Arduino, in: 2018 International Conference on Applied Smart Systems (ICASS), IEEE, 2018, pp. 1–4.

[11] H. Benyezza, M. Bouhedda, R. Kara, S. Rebouh, Smart platform based on IoT and wsn for monitoring and control of a greenhouse in the context of precision agriculture, Internet of Things 23 (2023) 100830.

[12] H. Benyezza, M. Bouhedda, S. Rebouh, Zoning irrigation smart system based on fuzzy control technology and iot for water and energy saving, J. Clean. Prod. 302 (2021) 127001, https://doi.org/10.1016/j.jclepro.2021.127001, https://www.sciencedirect.com/science/article/pii/S0959652621012208.

[13] P. Chandramohan, P. Balamurugan, S. Dinesh, Effect of fluid flow diversion and venturi size on drip irrigation fertilizer feeding system, Flow Meas. Instrum. 65 (2019) 250–256.

[14] B. Cheruvu, S.B. Latha, M. Nikhil, H. Mahajan, K. Prashanth, Smart farming system using npk sensor, in: 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1, IEEE, 2023, pp. 957–963.

[15] N.M. Cid-Garcia, A.G. Bravo-Lozano, Y.A. Rios-Solis, A crop planning and real-time irrigation method based on site-specific management zones and linear programming, Comput. Electron. Agric. 107 (2014) 20–28.

[16] A. Dahane, R. Benameur, B. Kechar, An IoT low-cost smart farming for enhancing irrigation efficiency of smallholders farmers, Wirel. Pers. Commun. 127 (2022) 3173–3210.

[17] D. De Benedetto, A. Castrignanò, M. Rinaldi, S. Ruggieri, F. Santoro, B. Figorito, S. Gualano, M. Diacono, R. Tamborrino, An approach for delineating homogeneous zones by using multi-sensor data, Geoderma 199 (2013) 117–127.

[18] Dosatron, Dosatron official website, https://www.dosatron.com/it-it/dosing-pump-manufacturer, 2025. (Accessed 21 January 2025).

[19] M.R. Goyal, Sustainable Micro Irrigation: Principles and Practices, CRC Press, 2014.

[20] E. Guardo, A. Di Stefano, A. La Corte, M. Sapienza, M. Scatà, A fog computing-based IoT framework for precision agriculture, J. Internet Technol. 19 (2018) 1401–1411.

[21] IBM ILOG, User's manual for CPLEX solver: v22.1.1, https://www.ibm.com/docs/en/icos/22.1.1, 2023.

[22] A. Kaur, D.P. Bhatt, L. Raja, et al., Developing a hybrid irrigation system for smart agriculture using IoT sensors and machine learning in Sri Ganganagar Rajasthan, J. Sens. 2024 (2024).

[23] S. Keesstra, S. Sannigrahi, M. López-Vicente, M. Pulido, A. Novara, S. Visser, Z. Kalantari, The role of soils in regulation and provision of blue and green water, Philos. Trans. R. Soc. B, Biol. Sci. 376 (2021), https://doi.org/10.1098/rstb.2020.0175.

[24] R. Kumar Kasera, S. Gour, T. Acharjee, A comprehensive survey on IoT and AI based applications in different pre-harvest, during-harvest and post-harvest activities of smart agriculture, Comput. Electron. Agric. 216 (2024) 108522.

[25] H. Lemonnier, E. Hervieu, Theoretical modelling and experimental investigation of single-phase and two-phase flow division at a tee-junction, Nucl. Eng. Des. 125 (1991) 201–213.

[26] Q. Li, G. Hu, Multistage stochastic programming modeling for farmland irrigation management under uncertainty, PLoS ONE 15 (2020) e0233723.

[27] W. Li, Y. Li, C. Li, G. Huang, An inexact two-stage water management model for planning agricultural irrigation under uncertainty, Agric. Water Manag. 97 (2010) 1905–1914.

[28] N. Lin, X. Wang, Y. Zhang, X. Hu, J. Ruan, Fertigation management for sustainable precision agriculture based on Internet of things, J. Clean. Prod. 277 (2020) 124119.

[29] M. Mahdavimanshadi, N. Fan, Multistage stochastic optimization for semi-arid farm crop rotation and water irrigation scheduling under drought scenarios, J. Agric. Biol. Environ. Stat. (2024) 1–24.

[30] J. Marquez, L.H. Talero-Sarmiento, H. Lamos, Multistage stochastic programming to support water allocation decision-making process in agriculture: a literature review, Chem. Proc. 10 (2022) 26.

[31] M. Moradi-Jalal, O.B. Haddad, B.W. Karney, M.A. Marino, Reservoir operation in assigning optimal multi-crop irrigation areas, Agric. Water Manag. 90 (2007) 149–159.

[32] A. Naseer, M. Shmoon, T. Shakeel, S. Ur Rehman, A. Ahmad, V. Gruhn, A systematic literature review of the IoT in agriculture—global adoption, innovations, security, and privacy challenges, IEEE Access 12 (2024) 60986–61021.

[33] O. Naud, J. Taylor, L. Colizzi, R. Giroudeau, S. Guillaume, E. Bourreau, T. Crestey, B. Tisseyre, Support to decision-making, in: Agricultural Internet of Things and Decision Support for Precision Smart Farming, Elsevier, 2020, pp. 183–224.

[34] K. Obaideen, B.A. Yousef, M.N. AlMallahi, Y.C. Tan, M. Mahmoud, H. Jaber, M. Ramadan, An overview of smart irrigation systems using IoT, Energy Nexus (2022) 100124.

[35] G. Papadopoulos, S. Arduini, H. Uyar, V. Psiroukis, A. Kasimati, S. Fountas, Economic and environmental benefits of digital agricultural technologies in crop production: a review, Smart Agric. Technol. (2024) 100441.

[36] M. Parimala, M.J. Rani, K.B. Sai, M. Sunitha, IoT based intelligent fertigation through drip irrigation, in: 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), IEEE, 2021, pp. 410–414.

[37] B. Qin, B. Hui, L. Wang, M. Yang, J. Li, B. Li, R. Geng, R. Cao, J. Sun, L. Si, et al., A survey on text-to-sql parsing: concepts, methods, and future directions, arXiv preprint, arXiv:2208.13629, 2022.

[38] R.A.P. Rajan, Serverless architecture-a revolution in cloud computing, in: 2018 Tenth International Conference on Advanced Computing (ICoAC), IEEE, 2018, pp. 88–93.

[39] S. Rode, R. Saraf, J. Veigas, N. Shetty, S. Shardul, Design and fabrication of IoT based agricultural automation system, in: 2023 5th Biennial International Conference on Nascent Technologies in Engineering (ICNTE), IEEE, 2023, pp. 1–6.

[40] M.S.H. Rosli, S.M. Salleh, A. Sapit, M.A. Razali, B.A.M. Zain, Implementation of IoT system for crop fertigation plantation, in: AIP Conference Proceedings, vol. 2530, AIP Publishing, 2023.

[41] S.J. Russell, P. Norvig, Artificial Intelligence: a Modern Approach, Pearson, 2020.

[42] A. Singh, Optimal allocation of resources for the maximization of net agricultural return, J. Irrig. Drain. Eng. 138 (2012) 830–836.

[43] A. Singh, Irrigation planning and management through optimization modelling, Water Resour. Manag. 28 (2014) 1–14.

[44] B.B. Sinha, R. Dhanalakshmi, Recent advancements and challenges of Internet of things in smart agriculture: a survey, Future Gener. Comput. Syst. 126 (2022) 169–184.

[45] I.K. Smout, S. Gorantiwar, Multilevel approach for optimizing land and water resources and irrigation deliveries for tertiary units in large irrigation schemes. I: method, J. Irrig. Drain. Eng. 131 (2005) 254–263.

[46] S. Trilles, A. González-Pérez, J. Huerta, An IoT platform based on microservices and serverless paradigms for smart farming purposes, Sensors 20 (2020) 2418.

[47] G. Valecce, S. Strazzella, A. Radesca, L.A. Grieco, Solarfertigation: Internet of things architecture for smart agriculture, in: 2019 IEEE International Conference on Communications Workshops (ICC Workshops), IEEE, 2019, pp. 1–6.

[48] P. Van Hentenryck, The OPL Optimization Programming Language, MIT Press, 1999.

[49] Y. Wang, Z. Li, S. Guo, F. Zhang, P. Guo, A risk-based fuzzy boundary interval two-stage stochastic water resources management programming approach under uncertainty, J. Hydrol. 582 (2020) 124553.

[50] M.Z.M. Zailani, S.A. Jumaat, A monitoring system of soil moisture for fertigation system using IoT application, Evol. Electr. Electron. Eng. 2 (2021) 856–866.

[51] Q. Zhang, Opinion paper: precision agriculture, smart agriculture, or digital agriculture, Comput. Electron. Agric. 211 (2023) 107982.