A Digital Twin of the Social-Ecological System Urban Beekeeping

Carolin Johannsen, Diren Senger, and Thorsten Kluss

Abstract We describe the system design and setup of our digital twin of the social-ecological system urban beekeeping, with the aim to support agroecological methods in urban agriculture. The physical space consists of the bee populations, their beekeepers who are part of a beekeeping community, non-beekeepers who consume honey, organisational actors shaping rules and regulations and the environment. The virtual space is a multi-agent model, where autonomous agents can take actions and make decisions in partially observed Markov processes. To tie the physical and the virtual space, we embedded bee hives in an IoT environment and implemented an online documentation tool as a web application, where beekeepers take short notes about their work and observations. Bee hives are equipped with sensors, such as humidity, pressure and temperature sensors and a scale. Additionally, we pull data from the German weather service (Deutscher Wetter Dienst, DWD). In our system architecture, multiple levels on data fusion are performed, beginning with raw data quality estimation and sensor failure detection. On higher levels, states of entities are estimated, such as the health of a bee colony, and assessment made whether a state is normal or to be considered an anomaly. Finally on the highest level, we deal with the desires of our agents, how actions should be chosen in order to achieve or maintain desirable and rewarding world states. We hope to be able to refine our digital twin into a decision support tool for small-scale (bee) farmers and communal political actors that helps to reach desirable world states by predicting and simulating the effects of actions within the complex system of urban beekeeping.

Keywords Urban farming · Urban beekeeping · Agent-based modelling · Multi-agent models · Digital twin · Environmental modelling and simulation · Food supply system · Decision making · Decision support
1 Introduction

We need to change the way we do agriculture. This is formulated in the Sustainable Development Goals set by the UN, in the synthesis report of the International Assessment of Agricultural Knowledge, Science and Technology for Development (IAASTD) initiated by the World Bank and co-sponsored by WHO, FAO and UNESCO [1], or in the recent 2019 FAO Report on “The State of the Worlds Biodiversity for Food and Agriculture” summarized in [2], to only name a few sources. There is a broad consensus that implementing biodiversity-friendly production methods and encouraging sustainable low impact practices by providing incentives for the responsible management of natural resources are key elements in this change process. Also, there is agreement on the benefit to focus on small scale farmers, with agroecological methods having been identified as one of the most robust [3]. As the trend for urbanisation is unbroken and the majority of the world’s population will live in cities in the near future, it stands to reason to take a closer look at smart cities as the places where the suggested paradigm shifts and more sustainable practices could be implemented.

Mohammadi and Taylor suggest that digital twins can be a tool for (smart) cities to address the challenges of rapid urbanisation, as achieving urban sustainability and resilience objectives, including the planning and maintenance of water supply, recycling infrastructure, the allocation and consumption of resources [4]. Albeit they focus on human-infrastructure-technology interactions and do not explicitly include an edible green infrastructure [5] with non-human inhabitants and biodiversity in their perspective, the extension to include these aspects is straight forward. A city is a complex system of many and diverse actors, with a multitude of roles, tasks, functions, relationships and interactions. To capture this, the digital twin of a city can best be constructed as a composite, as an assembly of single twin entities as illustrated in [6]. This can be reflected in software architecture by a multi-agent system (MAS), which we will describe in more detail in Sect. 4.

Unlike unconscious digital twins in industrial contexts [7], our system includes digital twins of the inhabitants of the city, who are able to actively examine and deliberately alter their digital sibling through their behaviour. If used as a tool of empowerment for the citizens -small scale farmers and consumers alike—such a digital twin can be utilized for civic participation processes, not only reflecting the actions of each citizen in the citywide context, but visualizing them, giving the users an opportunity to virtually test the effects of their actions and behaviour changes. As Vickers and Grieves describe it in [8]: digital twins can help to decrease unpredicted undesirable outcomes by helping actors to better understand the system and their situation while the system is already in full operation and can not be stopped for testing.

Since the sustainability of urban agriculture largely depends on the institutional environment, including formal laws and regulations as well as social norms and rules [9], a planning and decision support system accessible to all stakeholders, urban farmers, political decision makers or institutions such as the veterinary office, seems
profitable. The allocation and consumption of resources viable for sustainable urban agriculture such as land, water or subsidies [10] can be made transparent, which can be seen as an implementation of IAASTD’s suggestion [1] to promote small-scale farmers by improving their access to political power, as transparency of political processes is improved.

A second dimension of the digital twin is leveraging sensor data of the agricultural processes and environmental conditions to improve the yield and resource usage, as well as observing the behaviour and health of plants and livestock, remotely and supported by AI [11].

2 Sensor-Supported Urban Beekeeping

In this paper, we focus on urban beekeeping as a subdivision of urban agriculture. Beekeeping has a long tradition in the area where we built our twin (Bremen, Germany) and is to date predominantly done by small scale farmers with less then 10 bee colonies under their care. None the less, a number of IoT systems for beehives exist [12–16] including our own development, described in more detail below and in [17].

A bee colony, with its division of work, different developing states of brood and well localised areas within the hive for different purposes, is in itself already a highly complex system. It can hardly be seen as an isolated entity, but as a component of an environment, dependent on climatic conditions, patch dynamics of the surrounding ecosystem, pollen availability, the density of bee populations in the area and current outbreaks of bee diseases. Its development strongly depends on the beekeepers, their experience and knowledge about bee biology and the decisions and beekeeping methods they perform.

Even though many beekeepers benefit from their experience of many years of beekeeping, decision making in beekeeping always remains a difficult task. The inside of a bee colony has a carefully managed micro-climate, where the bees make sure that their precious brood can grow in optimal conditions and undisturbed by sunlight. Every time a beekeeper decides to make a manual inspection to asses the development and health of the colony, he/she disturbs this climate, a deregulation inside the habitation which the bees have to thoroughly readjust after the inspection. Adult bees as well as the brood are vulnerable to the exposure to cold temperatures, so that during winter or cold periods even a single opening of the hive might harm the colony. The use of sensors within the habitation -the hive- promises to be a possibility to observe changes of the bee development and health without major disturbances and the risk of cold exposure.

The beekeeping task is therefore comparable with complex systems occurring in the industrial domain where the concept of digital twins helped to establish a better monitoring environment, e.g. for intelligent workshop environments [18], product life cycle management [19] and aerospace vehicle testing [20].
For the construction of the digital twin, we use a sensor monitoring system which was developed as Do-it-yourself (DIY) sensor kit as part of the citizen science project Bee Observer [21]. The advantage of a DIY sensor kit is that it is relatively cheap to purchase and can easily be transformed to match the needs rising from different bee hives and beekeeping techniques. As it was developed by citizen scientist with beekeeping background, it incorporates the experience of practical application of different earlier prototypes.

The sensor kit, schematically depicted in Fig. 1a, consists of six one wire digital temperature sensors (DS18B20), a single point load cell (Bosche H30A/ H40A) and a combined sensor for relative humidity, barometric pressure and ambient temperature (Bosch BME280). All sensors are connected to a circuit board and the measurements and the data transfer are performed by a ESP32 development board (FiPy, Pycom).

The scale (see Fig. 1a no. 2) is placed underneath the hive and the weight measurements can therefore reflect the nectar, pollen and water intake but also include the weight of the bees, the hive itself and snow or rainwater on the hive’s roof. Five of the six temperature sensors (see Fig. 1a, no. 1) are placed within the hive, either all in the same clearance between two frames or one in every second clearance between two frames. Bees can regulate the temperature within the hive by quickly vibrating with the muscles of their wings to produce heat or by using their wings as ventilators to cool down the temperature. The optimal temperature to raise brood is 35 °C. Since the brood can only be found in a smaller ball-shaped area called the brood nest, it can happen that the temperature sensors are not optimally placed amidst the brood, so the uncertainty about the distance to the brood nest and the resulting temperature gradient must be accounted for. Other events, such as for example the advance of swarming, when half of the bee population emerges from the hive with a new queen and heat is generated when they prepare their muscles for the take off, are also reflected by changes in temperature measurements.

The combined relative humidity, temperature and atmospheric pressure sensor is placed within a clearance (Fig. 1a, no. 3). The bees constantly work on removing the water content from nectar to make it storable or on adding water to stored honey to
make it available to feed the brood, and there is an optimal humidity value in the hive for raising brood. Therefore, this value also is an important indicator for the state of the bee population.

As all climatic variables are influenced by the outside climate and weather, data from the open weather database of the German Meteorological Service is used as an additional source of information, fused with the measurements of one temperature sensor placed outside the hive and the atmospheric pressure sensor inside the hive.

3 Recording Bee Keeper’s Actions and Expert Knowledge

To obtain direct information from the beekeepers, we built upon the online beekeeping application “BEEP”, which was originally developed by a team in the Netherlands under an open source license [22].

The application is build to be a documentation tool for (urban) beekeepers. The beekeepers use the application to document objective information about their beekeeping activities, for example if they supplied additional food, harvested honey, or tried to treat a disease. They can also use it to communicate their own assessment of the population’s health state, classified by items such as the activity of the bees, gentleness, pollen inflow, their brood status, abnormal observation such as bad wings, bad brood or diseases and parasites such as the varroa mite. The application provides the feature to file documentations for different locations of apiaries and different hive types. The users have the option to create groups for cooperative beekeeping, where group members share access to all hives of the group. The application also serves as a graphical user interface for the sensor measurements of the DIY sensor kit, which are visualised for each hive, which is shown in Fig. 2.

The entries made by the beekeepers about their actions and assessments are input for the digital twin, as they represent records of the behaviours of beekeepers and bee colonies.

4 The Digital Twin as a Multi-agent System

The domain we are modelling consists of multiple, diverse entities who are acting and interacting autonomously with each other and the environment. The model architecture has to be able to handle these properties. As formulated in the introduction, we see several use cases for the digital twin:

- Small scale beekeepers should be supported with the decisions in beekeeping. The decision support is based on inference on the bee colonies health and behaviour, on the grounds of sensor data as well as past observations and actions of the beekeeper.
• Political decision makers, such as members of the city council or officers in the veterinarian agency, should be able to simulate the effects of their actions, possible changes in rules and regulations, with the goal of avoiding undesirable system behaviour.

• Citizens should be able to see the allocation of resources such as funds or the rights granted to keep bees in certain locations.

Therefore, we defined the requirement specification that the system architecture is modular, so that a single beehive and each small scale farmer is individually represented, but at the same time the dynamic macro-scale behaviour of an entire city can be observed and simulated. A modelling approach which fulfils this requirement is agent based modelling. It comes with the bonus of being relatively intuitive for users, as the entities represented as individual, autonomous agents often have a limited repertoire of actions, from which the macro-scale system behaviour emerges. Multi-agent systems have been successfully applied in a wide range of application contexts [23], including sociology [24], socio-technical systems modelling [25] and human-nature or social-ecological systems [26, 27].

To be able to run simulations and explore system behaviour in regions of the state space that have previously not been reached, the digital twin does not only need a representation of the sensor data and system states, as for example shown in [11], but also an AI component for the decision making process of each agent, including the communication between agents.

In our architecture, each agent holds a probabilistic representation of the world and its state, called the agents’ beliefs. The agent constructs its internal representation of the world out of information it perceives coming from the environment or from other agents. This information is not complete, each agent only possesses a limited
number of senses and resources to perceive the world, or in other words, the world is only partially observable. We further make the Marcovian assumption, that an agent believes the current state of the world to be (solely) dependent on the state in the time step immediately before. The states of the world are denoted as a finite set $S$, with $s_t \in S$ as the the state at time $t$. All observations of those states are denoted by $O$. An agent does not blindly believe all information to be true, but does believe that its view of the world is true, which it updates after making an observation. The agent has a mental concept of how likely it is that a certain observation occurs under the given circumstances, its observation model, defining the probability of measuring $o \in O$ given the world is in state $s \in S$. It also has a mental model of how the world changes from one time step to the next, which is often referred to as the transition model, defining the probability of reaching the state $s_{t+1} \in S$ given a starting state $s \in S$ and performing an action $a \in A$. The transitions depend on the actions $A$ of the agent. Both of these mental concepts are individually learned and dynamically change through life experiences. However, in the current version of our architecture they are time-invariant. In short: we designed a partially observed Markov decision process (POMDP) model as a dynamic Bayes net as compact representation of the beliefs. See, for example [28] for a more detailed explanation.

At each time $t$, an agent has the opportunity to take some actions that will have an impact on the state of the world. Based on expert interviews, literature review and the data collected through the BeeObserverApp, we have constructed a library of actions $A$ which each agent can chose from. These actions can be sequentially combined into a policy $\pi$. Even though not all actions do require the same amount of time, they are atomic in a sense that an agent will not abort an action once it has started it. Even if some actions have a duration longer than one time step, the agent will finish the action and only afterwards update its beliefs and revise its current policy if necessary. The action library depends on the type of agent: bee colony, beekeeper, non-beekeeper (both of which are individual humans) and organisation. It is describes in more detail in [29], but can be summarized as follows:

**Bee colonies**

- **consume**: Eat from the honey/syrup stored in the cells;
- **forage**: If the weather permits, the foragers of the colony will fly out into the neighbourhood and gather nectar and thereby replenish the food stock;
- **storeSyrup**: If a beekeeper has provided syrup, this is taken from the container and stored within the hive as food stock;
- **raise queen**: If a colony is in need of a new queen, a queen cell is made;
- **swarm**: Half the colony leaves with the old queen to find a new home. The other half stays and keeps on living in the same hive;
- **die**: A colony can die;

**Beekeepers**

- **inspect**: In order to get information about the status of the colony, their health, food supplies and so on, the beekeeper performs an inspection;
- **feed**: Providing sugar syrup as a food supplement;
- **make new colony**: An equal fraction of all of the hives in care of the beekeeper is taken and combined to form a new colony;
- **register**: Each colony must be registered with the veterinary office once it comes into the possession of a beekeeper;
- **unregister**: A colony must be unregistered with the veterinary office
once a beekeeper no longer owns it (also in the case of death); **break queen cell:** If a queen cell is observed, the beekeeper can choose to destroy it (and frequently does so, to prevent swarming, for example); **combine colonies:** Two colonies are merged into one; **harvest:** Take honey out of the hive; **treat:** The parasite varroa destructor is a lethal threat to colonies. Treatment with medication such as organic acids, improves colony health; **sell a colony:** If a buyer for the colony can be found in the same or directly adjacent neighbourhoods; **buy supplies:** Medication, sugar syrup or empty hives, provided sufficient monetary funds; **sell honey:** We model a beekeeper to always consume what he/she needs for her own subsistence first and sell excess only; **remove hive:** If a colony dies, it’s remains need to be taken care of. Possibly, the honey it has left behind could be harvested. The now empty hive is stored for reuse;

**Beekeepers and non-beekeepers**

**catch the swarm:** If a colony swarms, the swarm flies away and any human who sees it can catch the swarm. This will turn the human into a beekeeper, if he/she has not been one; **eat:** An individual amount of honey every simulation time step; **buy a colony:** From a beekeeper in the same or immediately adjacent neighbourhood. This will turn the human into a beekeeper; **attend a beekeeping course:** To increase his/her knowledge on beekeeping. A course can be offered by an organization; **meet:** A random number m [0..50] of humans from its neighbourhood, sequentially. If a consumer and a beekeeper meet, a honey transaction occurs if the consumer is willing to buy and the beekeeper has honey in stock.

**Organisations**

There are currently only a limited number of organisations implemented. Each organisation has a very small and highly individual action library. The **veterinary office** can prohibit the placement of a new colony in the neighbourhood. This is a reaction to an registration attempt. It can also **set colony density**, setting a maximum number of colonies allowed in a neighbourhood. The **beekeepers association** can **offer a beekeeping course**.

The decisions of the real world twins are captured through sensor data, yet one application of the digital twin is to predict and simulate how changes will affect the systems’ states in the future. To be able to make such predictions, a decision making process needs to be implemented. Since we are dealing with a complex, dynamic environment, our decision making architecture should be able to predict dynamic, non-linear behaviour. A particle filter can be used to infer what the agent believes to be the future states of the world, under a chosen policy $\pi$. Out of all possible states of the present and the future, only a small subset is considered by the agent. For these states, particles are generated [30–33].

To be able to infer which decisions an agent should take, a reward (or utility) function $R$ is defined. In many works on POMDPs, the rewards are assigned to each action. But for a finite-horizon problem, this is equivalent to assigning a reward to states [34]. As we believe a majority of people would consider it desirable to work less and not more, we chose to reward states instead of actions. The agent plans its policies by maximising the total reward over a time horizon $\tau$ (or to infinity)
\[ \sum_{t}^{t+\tau} R(s_t, a, s_{t+1}) \] [35]. It is a common practice to calculate this iteratively, so that for each time step the maximum value is chosen, starting with the very last action of the sequence at time \( t = \tau \). There is very little research published on how humans or non-human animals determine a feasible length of a policy.

In our setting, as in many real world settings, there might be multiple objectives (or desires) an agents holds. For example, a beekeeper might want to keep his/her bees healthy, but also make a profit by selling honey. While it is often possible to restructure the reward function from a multi-objective into a into a single-objective case [36], the question remains how to set the rewards, how to determine what are desired states. Following a homeostasis approach [37], we have defined the initial goal states of staying alive and procreating for the bee colonies, keeping the colonies alive as goal for beekeepers as well as honey self sufficiency and maximising profit for beekeepers and non-beekeepers alike. These goals need to be refined as more data becomes available and to be extended to the motivations of organisational actors.

### 5 Connecting the Twin to Sensors

#### 5.1 Data Storage and System Architecture

The measurement readings of the sensors in a single bee hive are initiated by the ESP32 micro-controller every 5–10 s. Each sensor-kit is identified by a unique key. The vector of sensor measurements, consisting of seven temperature values, one humidity, one barometric pressure and one weight value, is transferred to an influxDB server using an HTTP API, associated with the unique sensor key and the current timestamp. The influx database is a no-sql database optimised for time-series with a high writing and querying throughput rate [38]. It is open source, implemented in Go and has a sql-like query language, featuring special filters and aggregations for the time-series data [39]. From the MAS, which is implemented in java and R, sensor measurements can be directly read using the influxdb http-api.

All information entered in the online BeeObserver application is stored in a MySQL database. For every hive equipped with a sensor-kit, a separate virtual hive is created in the app, identified by the unique sensor-kit key. A hive can be associated with an apiary, a group of hives at the same physical location. The user who created the virtual hive (the owner) can invite other beekeepers in the app for cooperative beekeeping, and all of these beekeepers are allowed to enter inspections. Therefore, there are different databases for different entities: hives, apiaries and inspections. The hive database includes information such as type and size of a hive and details about its bee population, such as the age of the queen bee. The apiary database can be accessed for the detailed location of an hive. The inspection database can be accessed regularly to get updates about performed actions of the bee keeper and their assessment of the bee populations health and development state. Current and forecast weather data is obtained from the German weather service (Deutscher Wetter Dienst, DWD) [40].
The integration of online Machine Learning and classical time series analysis are implemented in the statistical programming language R, while the simulation of the multi-agent system model with graphical user interface is implemented in java.

For illustration purposes, we show measurements for two exemplary events, a swarming event and the feeding by a beekeeper. During a swarming event in a hive with a new queen, half of the colony leaves the hive with the old queen, leading to a sudden decrease in weight. Also, the temperature continuously increases when bees prepare and gather to leave the hive, having its peak when most of the bees left already (Fig. 3).

To feed a bee colony, the beekeeper has to open the hive, leading to a decrease in weight. Next, an additional super is added together with syrup or fondant. After the feeding, the weight should decrease as the bees extract the sugar from the fed supplement and evaporate the water content (Fig. 4).

**Fig. 3** Sensory data from bee hive in Bremen, Germany, showing swarming event, with queen and half of the colony leaving the hive. Vertical line: Onset of weight decreases/estimated starting time of swarm. Left: Total weight in kg. Right: Measurements of central temperature sensor (degree Celsius).

**Fig. 4** Weight measurements before, during and after an inspection with feeding of a bee colony in Cologne, Germany. Vertical line indicates the reported point in time of the inspection by the beekeeper.
Referring to the levels of data fusion [41, 42], we have to perform multiple levels of data fusion. On level 0, we need to estimate the quality of our raw sensor readings, filter missing data, check for faulty or failed sensors and perform temporal data alignment. This includes inference about the time delay between an event observed by a beekeeper and the documentation thereof. Often, the time precision of a beekeeping intervention record is within the fraction of an hour, and not seconds, as digital sensor readings with timestamps (Fig. 4). Also, an observation of a longer lasting state, such as a queen bee missing, might occur some time after the onset of the event. On level 1, we estimate entity states and features, such as the health of a colony or the amount of food resources the colony has left, if a queen is present or if a beekeeper is currently issuing a treatment against an infection. On level 2, we perform the situation assessment, where we estimate whether the current state constitute an immediate or future problem. In this step, we regard the relationships between entities and their environment. On level 3, we deal with the desires of our agents, how action policies \( \pi \) should be chosen in order to achieve or maintain desirable and rewarding world states.

6 Decision Making Example

In this section, we describe one decision process relevant in urban beekeeping in more detail. After a harvest or when there is a shortage in forage, beekeepers feed their colonies with sugar syrup. This is a source for optimising profitability of a beekeeping operation [43] and an issue which came up during our expert interviews. The colony should only be fed the exact amount it needs, so no money is invested in syrup not needed by the hive, no time is wasted by needlessly refilling containers and no predators are attracted. But the colony also should be provided with the amount it needs, as a failure in feeding might result in suffering and long term negative health effects due to food deprivation. Arguments against a constant provision with sugar syrup are: (1) Syrup might turn bad if it is not brought into the hive in time. Even if it is still good, it can not be put back into storage when it is not collected by the bees. (2) Syrup might be left over after the winter, when the colony is already foraging. The syrup must be removed from the cells before harvesting, else it contaminates the honey. (3) Syrup fed in summer can attract robbery by parasites, such as wasps.

The knowledge on the proper amount to feed is implicit to many experienced beekeepers, but has been identified as an issue where decision support would be valuable. Two decision scenarios have been modelled: (1) How much syrup should be fed in autumn as preparation for overwintering. (2) When and in what quantity to provide syrup during spring or summer as an intervention to prevent malnutrition.

Figure 5 shows the Bayes net for the inference process. The decision support will run a query on the probability of \( g_{\text{SyrupNeeded}} \), a continuous node modelling the needed quantity of syrup in gram. This depends on the provisions the colony has left, the population size, and the time of year. Neither the provisions nor the population size can be measured directly, but indirectly. The population size is subject to the
How many provisions are left can also only be measured indirectly, as a fraction of the total weight of the hive. The weight constitutes of the empty housing, which depends on the hive type. This is documented in the BeeobserverApp. Since some hives are of the Langstroth type, made out of modular stackable boxes, the number of boxes is variable and changes during the different phases of the beekeeping year. The alteration of the hive (stacking, taking off of boxes) should be documented in the app. Precipitation is also an influencing factor, since all bees stay inside the hive when it is raining, and there could be - at rare times in Bremen- a layer of snow on the hive’s roof.

The net represents one distinct timepoint $t$. All dynamics of the state transition from $t$ to $t+1$, given an action $a_t$, is modelled in the transition model $T$, with probabilities $P_{as}(s_{t+1})$. For example, how much food will be gathered by the bees
is such an action \( a \), affecting a subset of world states, including \textit{provisions left}, but the amount depends on the value of several state variables, such as the outside temperature, precipitation, day of year, bee population and (not modelled here) the number of colonies in the vicinity and available forage.

The parameters of the probability distributions have been learned from the data we have already collected and supplemented with expert knowledge. However, these parametrizations can be regarded as preliminary only, as the data collection within the hives is subject to the yearly rhythm of the bees and surrounding nature.

7 Conclusion and Future Work

We have set up the digital twin, connecting the physical world of bee colonies and beekeepers to a multi-agent system and designed the system architecture which can handle the different levels of data fusion. The machine learning layer is capable of constantly learning and updating the conditional probabilities of the states and state transitions but needs much more data to meaningfully do so. The effects of different weather and climate conditions in combination with diverse beekeeping choices, for example, should be recorded for a number of individual colonies before a generalization can be made. This data will be collected with the sensors already in place, and the BeeObserver project is in the process of bringing more sensor-kits to the field, slightly slowed by Covid-19, but the time frames for data collection are subject to the rhythm of nature.

Many details of the implementation remain for future work, such as the modelling of all relevant decision processes and belief updates for all agent types, as well as identifying, predicting and assessing many more entity states, such as preparation for swarming or loss of a queen.

There is currently no data transfer from the real world to the agent-based model regarding the human interactions. Validation of the modelled system dynamics and simulation results will require additional data acquisition. We are currently investigating the feasibility of an interactive gamified approach, where the simulation of urban beekeeping can be played just like a farming simulator.

We wish to thank all citizen scientists of the Beeobserver project for their hard work and positive energy invested into the set-up and maintenance of the sensor network.

References

1. S. International Assessment of Agricultural Knowledge, T. for Development, Agriculture at a Crossroads - Synthesis Report. Tech. Rep. (2009)
2. Pilling, D., Bélanger, J., Hoffmann, I.: Nature food 1(3), 144 (2020). https://doi.org/10.1038/s43016-020-0040-y
3. Altieri, M.A., Funes-Monzote, F.R., Petersen, P.: Agronomy for sustainable development 32(1), 1 (2012). https://doi.org/10.1007/s13593-011-0065-6

4. Mohammadi, N., Taylor, J.E.: In: 2017 IEEE Symposium Series on Computational Intelligence, SSCI 2017 - Proceedings 2018-January, 1 (2018). https://doi.org/10.1109/SSCI.2017.8285439

5. Russo, A., Cirrella, G.T.: Palgrave Commun. 5(1), 1 (2019). https://doi.org/10.1057/s41599-019-0377-8

6. van Schalkwyk, P., Malakuti, D.S., Lin, S.W.: IIC J. Innov. (November) 2 (2003)

7. Tao, F., Qi, Q.: Nature 573(7775), 490 (2019). https://doi.org/10.1038/s41586-019-02849-1

8. in Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches, August 2017 (2016), pp. 1–327. https://doi.org/10.1007/978-3-319-38756-7

9. Pearson, L.J., Pearson, L., Pearson, C.J.: Urban agriculture: diverse activities and benefits for city society 5903, 7 (2011). https://doi.org/10.1057/ijas.2009.0468

10. Cohen, N., Reynolds, K.: Renewable Agric. Food Syst. 30(1), 103 (2015). https://doi.org/10.1017/S1742170514000210

11. Alves, R.G., Souza, G., Maia, R.F., Tran, A.L.H., Kaniensi, C., Soininen, J.P., Aquino, P.T., Lima, F.: In: 2019 IEEE Global Humanitarian Technology Conference, GHTC 2019 (October) (2019). https://doi.org/10.1109/GHTC46095.2019.9033075

12. Edwards-Murphy, F., Magno, M., Whelan, P.M., O’Halloran, J., Popovic, E.M.: Comput. Electron. Agric. 124, 211 (2016). https://doi.org/10.1016/j.compag.2016.04.008

13. Pešović, U., Marković, D., uраšević, S., Ranić, S.: Acta agriculturae Serbica 24(48), 157 (2019). https://doi.org/10.5937/aaser1948157p

14. Chen, Y.L., Chien, H.Y., Hsu, T.H., Jing, Y.J., Lin, C.Y.: Y. Lin. In: Yang, C.N., Peng, S.L., Jain, L.C. (eds.) Security with Intelligent Computing and Big-data Services, pp. 535–543. Springer International Publishing, Cham (2020)

15. Catania, P., Vallone, M.: Sensors (2020). https://doi.org/10.3390/s20072012

16. Hunter, G., Howard, D., Gauvreau, S., Duran, O., Busquets, R.: Proc. Inst. Acoust. 41(June), 339 (2019)

17. Johannsen, C., Senger, D., Kluß, T.: In: 2020 16th International Conference on Intelligent Environments (IE) (2020)

18. Zhang, Q., Zhang, X., Xu, W., Liu, A., Zhou, Z., Pham, D.T.: In: International Conference on Intelligent Robotics and Applications, pp. 3–14. Springer, Berlin (2017)

19. Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., Sui, F.: Int. J. Adv. Manuf. Technol 94(9–12), 3563 (2018)

20. Glaessgen, E., Stargel, D.: In: 53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/ASCE adaptive structures conference conference 14th AIAA (2012), p. 1818

21. Bee Observer BOB das ist unser citizen science projekt. https://hiverize.org/beo-observerbob-das-ist-unser-citizen-science-projekt/. Accessed 01 July 2020

22. (2020). https://beep.nl/home-english. Accessed on 15 Jan 2020

23. Dorri, A., Kanhere, S.S., Jurdak, R.: IEEE Access 6(April), 28573 (2018). https://doi.org/10.1109/ACCESS.2018.2831228

24. Bianchi, F., Squazzoni, F.: WIREs Comput Stat 7(August) (2015). https://doi.org/10.1002/wics.1356

25. Vespignani, A.: Nat. Phys. 8(1), 32 (2012). https://doi.org/10.1038/nphys2160

26. Schulze, J., Müller, B., Groeneveld, J., Grimm, V.: J. Artif. Societies Soc. Simul. 20(2), 8 (2017). https://doi.org/10.18564/jasss.3423. http://jasss.soc.surrey.ac.uk/20/2/8.html

27. An, L.: Ecol. Model. 229, 25 (2012). https://doi.org/10.1016/j.ecolmodel.2011.07.010

28. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. Massachusetts Institute of Technology (2006)

29. Johannsen, C.: In: Under Consideration for 24th European Conference on Artificial Intelligence, Qualitative Reasoning Workshop. Springer, Berlin (2020)

30. Doucet, A., Johansen, A.M.: Handbook of Nonlinear Filtering (December) 4 (2009)

31. Doshi, P., Gmytrasiewicz, P.J.: In: Proceedings of the International Conference on Autonomous Agents, pp. 463–470 (2005). https://doi.org/10.1145/1082473.1082522
32. Bard, N., Bowling, M.: Proceedings of the National Conference on Artificial Intelligence 1, 515 (2007)
33. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: Bayesian bounds for parameter estimation and nonlinear filtering/tracking 50(2), 723 (2007). https://doi.org/10.1109/9780470544198.ch73
34. Kaelbling, L.P., Littman, M.L., Cassandra, A.R.: Artif. Intell. 101, 99–134 (1998). https://doi.org/10.1007/s00726-010-0654-8
35. Ng, A., Harada, D., Russell, S.: ICML 99, 278 (1999)
36. Roijers, D.M., Vamplew, P., Whiteson, S., Dazeley, R.: J. Artif. Intell. Res. 48, 67 (2013). https://doi.org/10.1613/jair.3987
37. Drengstig, T., Jolma, I.W., Ni, X.Y., Thorsen, K., Xu, X.M., Ruoff, P.: Biophys. J. 103(9), 2000 (2012). https://doi.org/10.1016/j.bpj.2012.09.033
38. (2020). https://docs.influxdata.com/influxdb/v1.7/. Accessed on 29 Apr 2020
39. O.D.S. DWD, Data Source: Deutscher Wetterdienst (2020). http://shorturl.at/lpsV2. Accessed 28 Jan 2020
40. Llinas, J., Bowman, C., Rogova, G., Steinberg, A., Waltz, E., White, F.: In: Proceedings of the Seventh International Conference on Information Fusion, FUSION 2004 2, 1218 (2004)
41. Steinberg, A.N., Bowman, C.L: pp. 1–18 (2004). http://www.infofusion.buffalo.edu/tm/Dr.Llinas' stuff/RethinkingJDLDataFusionLevels_BowmanSteinberg.pdf
42. Somerville, D., Collins, D.: 63, 2007 (2015)