OPEN ACCESS
International Journal of Software Engineering
and Knowledge Engineering
(2024) 1–17

DOI: 10.1142/S0218194024500025

World Scientific
www.worldscientific.com

Modeling and Control of Drinking Water Supply Infrastructures Through Multi-Agent Systems for Sustainability

Carlos Calatayud Asensi

Aguas de Valencia S.A
Avda. Marqués del Turia 46005, Valencia, Spain
ccalatayud@globalomnium.com

José Vicente Berná Martínez*, Lucía Arnau Muñoz†
and Francisco Maciá Pérez‡

Department of Informatics Technology and Computation, University of Alicante
Carretera San Vicente del Raspeig s/n, 03690 San Vicente del Raspeig, Spain
**jvberna@ua.es*
†lucia.arnau@ua.es
‡pmacia@dtic.ua.es

Received 6 August 2023
Revised 27 October 2023
Accepted 19 January 2024
Published 19 March 2024

Traditionally, drinking water supply infrastructures have been designed to store as much water as possible and to do so during the energy cheap hours. This approach is unsustainable today. The use of digital systems capable of modeling the behavior of infrastructures and the creation of intelligent control systems can help to make drinking water supply systems more efficient and effective, while still meeting minimum service requirements. This work proposes the development of a control system, based on multi-agent systems (MAS), capable of generating an intelligent control over a drinking water infrastructure, based on the use of local interests of the agents and with an emergent behavior coherent with the needs. To validate the proposal, a simulator based on the infrastructures of a medium-sized Spanish city of 5000 inhabitants has been built and the control has been simulated using the MAS. The results show how the system can maintain the objectives set, handling unknown situations, and facilitating the development of future physical systems based on a just-in-time paradigm that guarantees sustainability, as it allows the generation of virtualizations of the infrastructures and their behavior, thus being able to study the best option for an infrastructure to resolve a supply situation.

Keywords: Intelligent control; infrastructure modeling; multi-agent system; intentions; behaviors.

* Corresponding author.

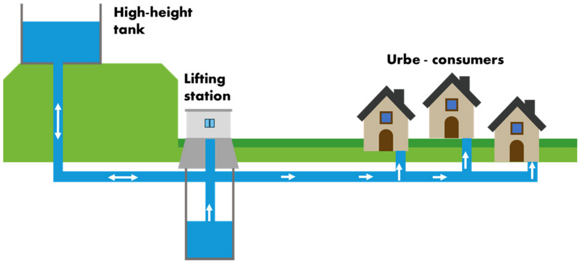
1. Introduction

The drinking water supply systems use the storage of large-quantity storage of water treated in water tanks at height, as a mechanism for the cost cover and service assurance. Producing water during the hours of the day where economically it is cheaper because the cost of energy is lower or because you get photovoltaic plants [1, 2] has been and remains the mainly used strategy. On the other hand, solving the possible shortage of consumers at times when it is possible that, due to high demand, drinking water production systems may not be sufficient, also uses the storage of as much available water ready to be served [3].

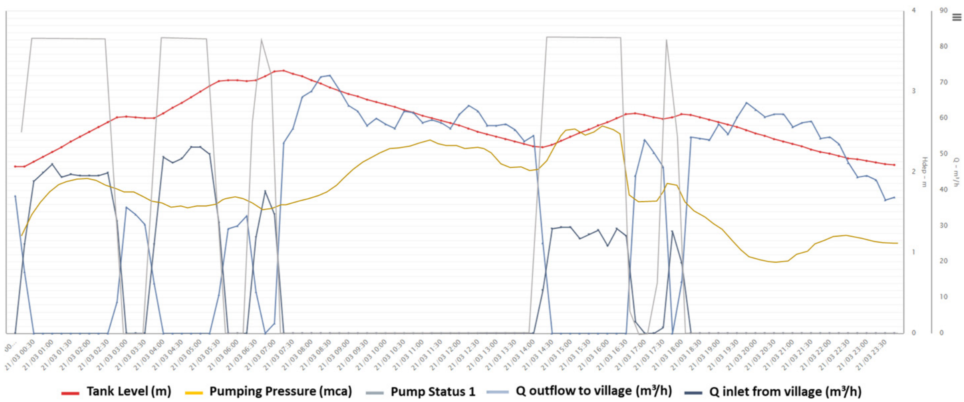
However, this approach generates certain problems that are so far ignored in favor of lowering the cost of production or preventing water shortages. In the first place, storing water in deposits has a cost in conjunction with sustainability, as it involves extracting water from the underground water to be stored in large deposits, with an environmental impact [4] and with the need to treat for its potability by chemicals. Secondly, keeping water tanks filled to the brim using cheap energy causes the existing volume of water to put pressure on water infrastructures and pumping stations. In tanks with a bottom inlet, this means that the higher the level in the tank, the more energy the pumps must use to lift the water [5]. Thirdly, when water consumption is occurring at a given moment, the pumping of water into the tanks has to overcome the inertia produced by the movement of the mass of water, which means that the pumps not only have to lift the water into the tanks, but also overcome the flow of the existing flow. This extra effort means higher energy consumption, and the flow of water into the tank is less, and therefore the time to fill it also increases, which has a negative impact on energy consumption. In addition, this generates an undesired overpressure on the installations, which is also detrimental to the integrity of the infrastructures. Figure 1(a) shows a basic diagram of the supply infrastructure together with an empirical study. Figure 1(b) shows the effects described above for a typical installation in a medium-sized Spanish city of 5000 inhabitants.

In Fig. 1(b), the pressure (yellow line) decreases when water is consumed from the city, and that this pressure increases when the pumping station injects water into the network to refill the water tank. In addition, the sheet of water stored in the tank does not fall below a height of 2 m (red line), which forces the pumps to consume more energy to bring the water up to the tank due to the pressure they must exert.

The problem we face is therefore the optimization of the use of the infrastructures, minimizing the sheet of water stored, starting pumping at the most appropriate times depending on consumption and existing storage, but at the same time ensuring the city's supply. Furthermore, in the case of much more complex and distributed scenarios, where there may be different pumping stations, segmented consumer networks that adopt different consumption patterns (e.g. city vs. beach, seasonal, etc.), different water storage systems, and even unpredictable unknown situations (heat waves, mass events, breaks and breakdowns), the complexity of optimized management can generate an intractable problem [6].



(a)



(b)

Fig. 1. (a) (Color online) Diagram of the typical supply network of a city, with a pumping group that supplies a high water tank and a network that connects all the elements and consumers. (b) Graph showing the evolution of the pressure, the state of the pumps, the filling level of the water tank and the consumption from the city or the production of water to the water tank. Data based on the empirical study of the company Aguas de Valencia.

The problems related to the management of the water supply network refer, among others, to the optimization of its parameters and the energy saving use, ensuring the correct distribution of water with the appropriate quality. Given the complexity of this task, multiple IA-related techniques have been used [7]. The different alternatives depend on the optimization objective and specific characteristics of the infrastructures. For example, the use of machine learning is presented as a suitable way for nonlinear problems in the optimization treatment [8], the use of neural networks based on multilayer perceptrons is suitable for multi-objective systems but in closed and isolated environments as a sample [9] and deep learning techniques may be suitable in systems in open context but with a single objective [10]. However, the characterization of the problem addressed in this work is that of an open system that can evolve over time, with multiple objectives and where these objectives can change. The problem with traditional AI methods based on networks and algorithms is that they are trained is that change involves retraining the systems [11]. This involves validation and testing of the models once they are trained and above all it requires analysis and design of the algorithms in the face of changes. This is why AI techniques based on systems precisely designed for change are ideal in this type of context, i.e. multi-agent systems (MASs) [12]. MASs allow adding, removing or modifying objectives, modifying only the logic that is isolated in the agent that manages it, it is even possible to modify the relevance of the influence that an agent generates, thus making an objective more primordial or not in the system. This property makes MASs ideal in environments where there are changes in the infrastructure, in the behavior of the system, in the users of the infrastructure and even in the objectives, since the control system is flexible and modular [13], being able to modify a part while maintaining the effectiveness of the rest of the system.

The aim of this work, therefore, will be the creation of a control system capable of satisfying all the optimization objectives set regarding the minimum water level, maintenance of pressures and minimizing the work of the pumps during the moments of greatest inertia. For this purpose, the use of a deliberative MAS is proposed, since these types of agents model very well multi-objective systems, are open to change, and where it is possible they are able to easily modify the control to adapt it to new situations. For this purpose, a real case study based on a Spanish city of 5000 inhabitants will be used as an example infrastructure, together with its consumption patterns and data on pressures and state of the infrastructures. The rest of the paper is structured as follows: Sec. 2 describes the formal framework that characterizes the MAS model; Sec. 3 specifies the functions of the model for a deliberative system with internal history; Sec. 4 develops the objectives by modeling agents for the control of each one of them and establishing the specific parameters of the model; Sec. 5 shows the tests and validation of the model, where it is verified that the model meets the objectives; Sec. 6 discusses the results; finally, Sec. 7 shows the conclusions of our work and future lines.

2. Proposed Solution

Our proposal uses a MASs approach as its core. MASs were developed to solve large problems where data may be distributed and have different natures. Furthermore, they are particularly suited to problems with multiple resolution methods, multiple perspectives and/or multiple elements that can provide a solution to the problem. The purpose is to achieve objectives through a distributed system of sensing, communication, processing and control [14]. The use of MAS in complex sensor-related systems has also been shown to enable the generation of complex behaviors and is even able to cope with unknown situations [15], to ultimately generate highly efficient specialized control systems [16].

A MAS consists of a set of autonomous agents, each of which tries to execute actions on the system. Each agent observes a part of the system, and makes decisions based on this fragmented knowledge, to modify the world to achieve its individual goal. This is what allows the introduction of multi-objective pursuit within the control system. It is the summa of all these attempts to influence the world that generates an overall behavior close to the overall goal of the system.

According to our proposal, formally each agent can be described as an entity α capable of obtaining information from its environment, which we call the $\mathit{Percept}_\alpha$, obtaining a perceived state Φ_α from the global environment. It can store this new perceived state, what we call Mem_α, in an internal state Σ_α, which will be the result of combining the perceived data and its own knowledge up to that point. Using this perceived information and its own internal state it can make decisions, called $\mathit{Decision}_\alpha$, and finally because of the decided action, execute it in the Exec_α system. Using formal definitions an agent α is defined as

$$\alpha = \langle \Phi_\alpha, \Sigma_\alpha, P_\alpha, \Gamma_\alpha, \mathit{Percept}_\alpha, \mathit{Mem}_\alpha, \mathit{Decision}_\alpha, \mathit{Exec}_\alpha \rangle, \tag{1}$$

where $\Phi_\alpha = \langle \varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_n, \rangle$ and φ_i is a list of signal-value pairs of perceptions of the world, the information that an agent will extract from its context and that comes either from the managed system or from other agents producing new information.

$\Sigma_\alpha = \langle \varsigma_1, \varsigma_2, \varsigma_3, \ldots, \varsigma_n, \rangle$ and ς_i is a list of the system's internal signal-value pairs, the information it will store internally.

$P_\alpha = \langle \rho_1, \rho_2, \rho_3, \ldots, \rho_n, \rangle$ and ρ_i is a list of signal-value pairs that define an action, an intention to change.

$\Gamma_\alpha = \langle \gamma_1, \gamma_2, \gamma_3, \ldots, \gamma_n, \rangle$ and γ_i is a list of output signal-value pairs, called influences, which constitute the centre's attempt to change the state of the world by outputting new values that it wishes to change in some element.

$\mathit{Percept}_\alpha\colon W \to \Phi_\alpha$ function that generates a perception from the state of the world W.

$\mathit{Mem}_\alpha\colon \Phi_\alpha \to \Sigma_\alpha$function that generates a new internal state from the perceived state.

$\mathit{Decision}_\alpha\colon \Phi_\alpha\ x\ \Sigma_\alpha \to P$ function that generates an action from the perceived and internal state.

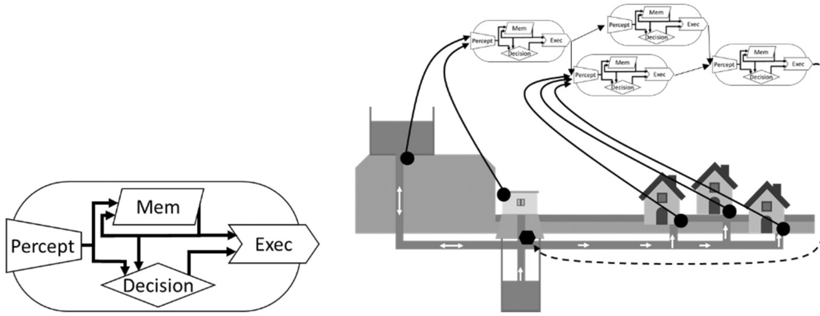
$\mathit{Exec}_\alpha\colon P \to \Gamma$ function that generates an influence from the action taken.

The execution of a given action by an agent at a given time does not directly imply the alteration of the state of the system but is to be taken as an attempt to change its state, to exert an influence γ of change. Figure 2(a) shows the internal scheme of an agent. It is therefore the execution of all decided actions, the sum of all influences of each agent taking part in the system that generates a change from one state to another. Formally the future state of the system $\sigma(t+1)$ can be defined as the reaction, *React*, of the system in its current state $\sigma(t)$ together with the union of all the influences of the agents in the system:

$$\sigma(t+1) = \mathit{React}\Big(\sigma(t), \bigcup\nolimits_1^n (\gamma_i)\Big), \tag{2}$$

where each γ_i is defined as

$$\gamma_i = \mathit{Exec}_i(\mathit{Decision}_i(\mathit{Percept}_i(\sigma(t)), \varsigma_i(t))).$$



(a) (b)

Fig. 2. (a) Visual diagram of an agent and its functions. (b) Diagram of a water supply system and the control exercised by the MAS.

We can therefore see a MAS as a control system made up of multiple elements, each of which is focused on a local interest and tries to influence the whole system. The MAS-based management system would be a set of agents, which would receive information from the world (the water supply system), from all sources of information (sensors), each agent would obtain the list of those data about its interest, and based on this information would try to influence the system. It should also be considered that the influence of an agent can be on another agent and not directly on the physical system. Figure 2(b) summarizes this idea.

The diagram in Fig. 2(b) shows how the control system can receive information about the sensors of the physical system or the results of other agents. Finally, signals are sent to the actuators that are the result of the execution of each agent locally.

3. Definition of Local Functions

Once the general scheme has been defined, it is necessary to define the agents that will form part of the control system, for which the internal *Percept*, *Mem*, *Decision* and *Exec* functions of each agent must be defined.

3.1. *Percept*

This function simply defines the list of signals that will be observed by an agent. That is, it will be a list of signal names to which the agent will react by perceiving the state of the world, when one of them changes. Therefore, each agent will define its watch list: *watchList*=[*singnal*1, *signal*2, ..., *signaln*].

3.2. *Mem*

The memorization function is responsible for storing in the agent's internal memory a new state of the perceived world, if there has been a substantial change. We can therefore say a threshold μ which, when exceeded, generates a new internal state to be stored. To compare the current world state and whether it has changed sufficiently we use a distance function. Given an agent, and the list of perceived signals (φ) and the stored state (ς), the distance function is defined as shown in the following equation:

$$distance = \frac{\sqrt{\sum_1^n \frac{\varphi(t)_i^2}{\max(\varphi)_i^2}} - \sqrt{\sum_1^n \frac{\varsigma(t-1)_i^2}{\max(\varsigma)_i^2}}}{card(\varphi)}. \tag{3}$$

To avoid that there are signals that can affect the distance more or less (for example, if a signal moves between the values 1 and 1000 while the rest of the signals move in the range 0 to 1), each signal will be divided by its maximum value. This scales all signals to the same range of values between 0 and 1. The value of the perceived state is subtracted from the value of the stored state. Finally, the value is divided by the cardinality of φ, i.e. the number of signals that are stored. Distance will therefore be a value between 0 and 1.

Finally, *Mem* will store the new state of the world whenever $distance > \mu$. If we want the system to be very sensitive to changes, it will be enough to make $\mu = 0$, and then the system will react to any change. If we do not want the agent to be very reactive, the value should be close to $\mu = 1$. Each agent can be configured with a particular μ.

3.3. *Decision*

The decision function generates an action in case the center detects a condition of interest. Generating an action implies generating output signals that will try to influence the world, trying to bring about a change. We can define this function as

$$SetSignalValue(FunD(\varphi, \varsigma)) \quad \text{if } PreD(\varphi, \varsigma) \text{ is verified,} \tag{4}$$

where

- *PreD* (φ, ς): Decision precondition function that relates False or True to a percept and a given internal state: *PreD*: $\Phi_\alpha \times \Sigma_\alpha \rightarrow Boolean$. This defines the trigger conditions.
- *FunD* (φ, ς): Function that associates the perception and internal state of the agent with a list of efferent signals that define an action *FunD*: $\Phi_\alpha \times \Sigma_\alpha \rightarrow P_\alpha$.

Therefore, each center will have to determine the signals that generate actions.

3.4. *Exec*

Finally, we will define the execution function as a function that emits the signals generated by Decision, provided that a certain precondition of execution is met:

$$PostE(\rho) \quad \text{if } PreE(\varphi) \text{ is verified.} \tag{5}$$

The *PostE* function generates the output signals or influences. They can be the same signals defined by the action ρ. The function $PreE(\varphi)$ allows to condition the influence attempt to the perceived state, although it can be set to *true* and thus influence would always be generated.

3.5. *Configuration table*

After defining the internal functions, Table 1 shows the elements to be specified for each agent that is part of the control system.

Table 1. Parameters and functions to be configured.

Element	Description
watchList	List of signs to watch out for
μ	Attention threshold
PreD	Decision precondition, triggers an action, can be TRUE
FunD	Action taken by the centre (signal modification)
PostE	Influence generation from action
PreE	Condition to trigger influence, can always be TRUE

We will use as *PreD* and *PreE* the function TRUE, they will always be triggered.

4. Agent-based Control System

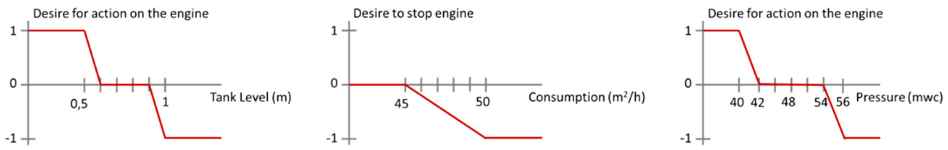
Based on the general definitions, we will define the agents that make up the control system. Each agent concentrates intentions according to its intentions and wishes so that we will assign an agent to pursue an intention in the system. We must decompose the desired global behavior into intentions that can be implemented in the agents and the priority relations between these intentions following a logical functional decomposition [17]. The definition of system intentions will be as follows:

- **Intention 1:** Minimize the sheet of water stored, determining as lower limit 0.5 m and maximum limit 1. It is necessary to maintain this minimum of 0.5 to avoid the dragging of sediments from the water tank.
- **Intention 2:** To avoid pumping water during periods of maximum consumption in the city, thus avoiding the work of the pumping stations against inertia.
- **Intention 3:** Maintain the pressure of the infrastructure around a reasonable value, avoiding overpressure or low pressure.

For each of these intentions, we are going to define an agent with its corresponding configuration table.

4.1. *Intention 1*

To model the behavior of this agent, we are going to establish its output as the desire to turn the pumping groups on or off in reaction to the water level, D_l, i.e. the further



(a) (b) (c)

Fig. 3. (a) Graph modeling the behavior of agent 1. (b) Graph modeling the behavior of agent 2. (c) Graph modeling the behavior of agent 3.

away the value of the water level in the tank is from the ideal, the more the agent will try to act on the motors, when the value is close to 0.5 or lower it will want to start the motor (value 1), and when the water level is close to 1 it will want to stop them (value –1). When the water level is between 0.5 and 1, the agent will not want to take any action (value 0). Figure 3(a) shows the graph with the behavioral model. Based on the desire to stop or start, we define the agent's table of parameters and functions as shown in Table 2.

Table 2. Agent 1 — Intention 1.

Element	Values	
watchList	sheetWater	
μ	0.01 m	
FunD	$D_l = 1$	if sheetWater < 0.5
	$D_l =$ (0.6 – sheetWater)*10	if $0.5 \leq$ sheetWater ≤ 0.6
	$D = 0$	if $0.6 <$ sheetWater < 0.9
	$D_l =$ (0.9 – sheetWater)*10	if $0.9 \leq$ sheetWater ≤ 1
	$D_l = -1$	if sheetWater > 1
PostE	D_l	

4.2. *Intention 2*

This agent monitors the consumption that is being produced in the city, so when the pumping groups are not running, the outflow from the tank will be used as a measure (*waterFromTank*), and when the groups are running, the water production flow of the pumps will be used (*waterFromPump*) plus the water flow towards the tank (*waterToTank*). The sensing function would be responsible for filtering the sensing. The empirical study of the installations indicates that with a consumption of more than 50 m^2/h, the inertia of the water can generate pressure and it would be better not to activate the motors, the agent will try to stop the motors when this value is exceeded, or what we call the desire to stop due to inertia D_i. Figure 3(b) shows the graph with the behavioral model. The configuration table of this agent is configured as shown in Table 3.

Table 3. Agent 2 — Intention 2.

Element	Values	
watchList	waterConsumption → waterToTank, waterToPump, waterToTank	
μ	$0.5\,\mathrm{m}^2/\mathrm{h}$	
FunD	$D_i = 0$	if waterConsumption < 45
	$D_i = -$ (waterConsumption – 45)/5	if 45 ≤ waterConsumption ≤ 50
	$D_i = -1$	if waterConsumption > 50
PostE	D_i	

4.3. *Intention 3*

This agent will be responsible for determining the desire to stop or start if the pressure is not adequate. A pressure of around 48 mwc (meter of water column) is considered adequate; when the pressure is higher, the desire to compensate for the pressure, D_p, is to stop (negative values) and if the pressure is lower, the desire is to start (positive value) the pumping. Figure 3(c) shows the graph with the behavioral model, and its configuration table is given in Table 4.

Table 4. Agent 3 — Intention 3.

Element	Values	
watchList	waterPressure	
μ	0.1 mwc	
FunD	$D_p = 1$	if waterPressure < 40
	$D_p = 1 -$ (waterPressure – 40)/2	if 40 ≤ waterPressure < 42
	$D_p = 0$	if 42 ≤ waterPressure ≤ 54
	$D_p = -$ (waterPressure – 54)/2	if 54 < waterPressure ≤ 56
	$D_p = -1$	if waterPressure > 6
PostE	D_p	

4.4. *Modeling the relationship between actors*

The result of managing the intentions is that the system now produces three new signals that materialize the desire to change the state of the system:

- D_l: desire for the level of the sheet of water;
- D_i: desire for inertia;
- D_p: desire for pressure.

The values of these signals oscillate between −1 and 1, with −1 indicating a strong desire to stop engines, 1 indicating a desire to start the engine at maximum power, and 0 indicating no desire to alter the state of the system. It should be understood that 0 in any of the signals does not indicate stopping the engine, but rather that the agent has no intention in changing the state of the system. Faced with these desire signals, a final agent will be responsible for converting this desire for change into a signal that really influences the state of the engines. To do so, it will

Table 5. Agent 4 — Influence on engine.

Element	Values
watchList	D_l, D_p, D_i
μ	0.1
FunD	$I_m = 0.5 * D_l + 0.3 * D_p + 0.2 * D_i$
PostE	I_m

weigh the desires against each other, giving a greater weight to D_l, since a possible lack of supply depends on it, secondly to D_p, since the security of the infrastructure partly depends on it, and lastly to D_i, since it favors a correct use of the infrastructure. This last agent will be characterized with the configuration table as shown in Table 5.

As can be seen, I_m will generate an influence towards the engine, a positive or negative value, between -1 and 1, which will generate a response in the engine that it wants to stop or that it wants to increase its pumping power. Figure 4 shows an image of the control system with all the agents and signals.

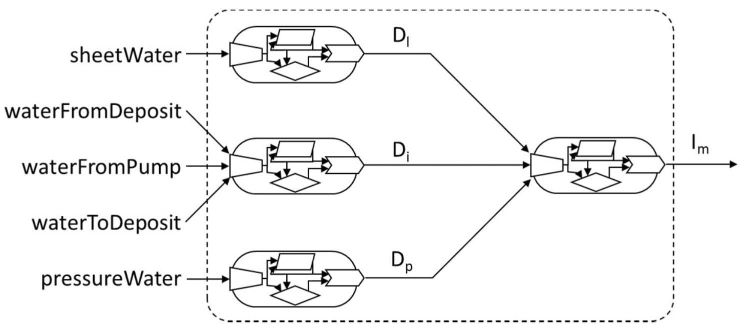


Fig. 4. Multi-agent control system and signals handled and produced.

5. Testing and Validation

To test and validate the proposal, a MAS has been built to show the modeled behavior on a virtual simulation of the infrastructure of a medium-sized Spanish city used during the study. This system has been developed using TypeScript, in an Angular 14 web interface, and allows the observation of the system variables and their evolution, both the input signals to the MAS and the internal ones (D_l, D_i and D_p) and the output signal I_m. The application can be found in the public repository [18]. The system is configured with a tank at a height of 48 m, which added to an initial water level of 0.9 m, means an initial pressure in the infrastructure of 48.9 mwc. In this configuration, different consumption situations are generated. The system simulates the evolution of the signals every 10 min, so that each instant of time is equivalent to that time having elapsed in the infrastructures.

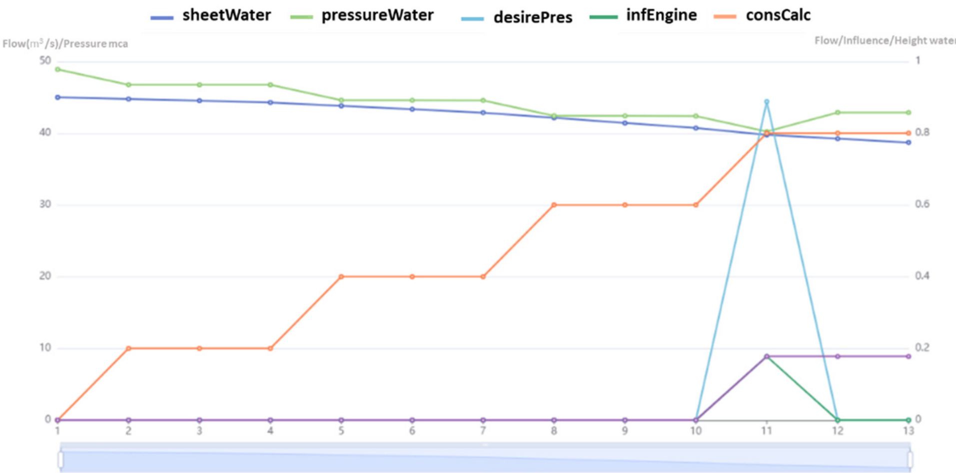


Fig. 5. Behavior of the system when water consumption starts.

Figure 5 shows the evolution of the system's signals when water consumption in the city begins.

Consumption increases by $10\,\mathrm{m}^3/\mathrm{h}$ every 30 min up to $40\,\mathrm{m}^3/\mathrm{h}$. The water level drops and when the water consumption increases to $40\,\mathrm{m}^3/\mathrm{h}$, the pressure drops below the desired pressure of 42 mwc, and this causes the pumps to be activated to compensate the pressure. It can be seen how the desire generated by the drop in pressure (D_p), "desirePres" in the graph, increases at instant 11, and when the pumps are activated and the pressure is compensated above 42 mwc, the desire falls back to 0. As the pumps are running at a rate below 20%, water is being generated from the pumps but at about $22\,\mathrm{m}^3/\mathrm{h}$, which causes the water level to continue to fall. Figure 6 shows the simulation of consumption over several hours and it can be

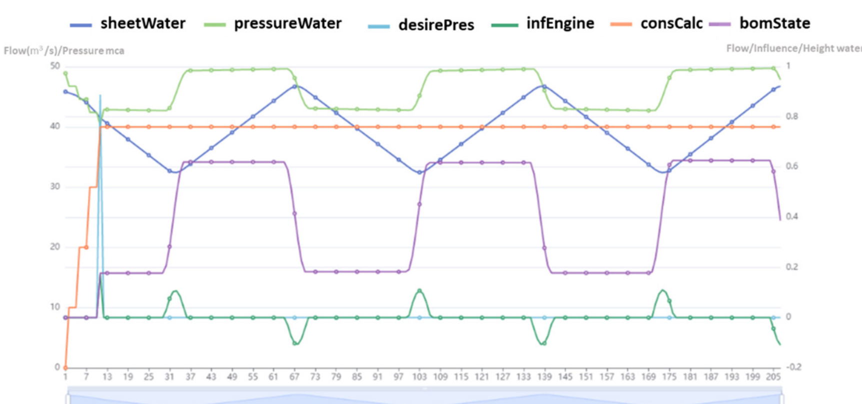


Fig. 6. The system after several hours of operation.

seen how the water level drops to below 0.6 m, which activates the motors to a higher working range, around 60%, which is enough to fill the tank to the desired level of 0.9. And when this optimum level is reached again, the motors lower their working range to compensate the pressure again, but without the need to supply water to the tank.

When we subject the system to sudden changes in consumption, going from 40 m^3/h to 80 m^3/h, the agents respond by increasing the activity of the pumps. Figure 7 shows how at instant 97 the increase in consumption occurs (*consCalc* line) and at that instant the system compensates for the drop in pressure by activating the pumps. In addition, the level of pump activation is now higher, since to fill the tank, it is necessary for the pumps (line *bomState*) to work at a higher speed. It can be seen that from instant 97, the pumps have a 100% working speed, whereas before they were pumping at around 60%.

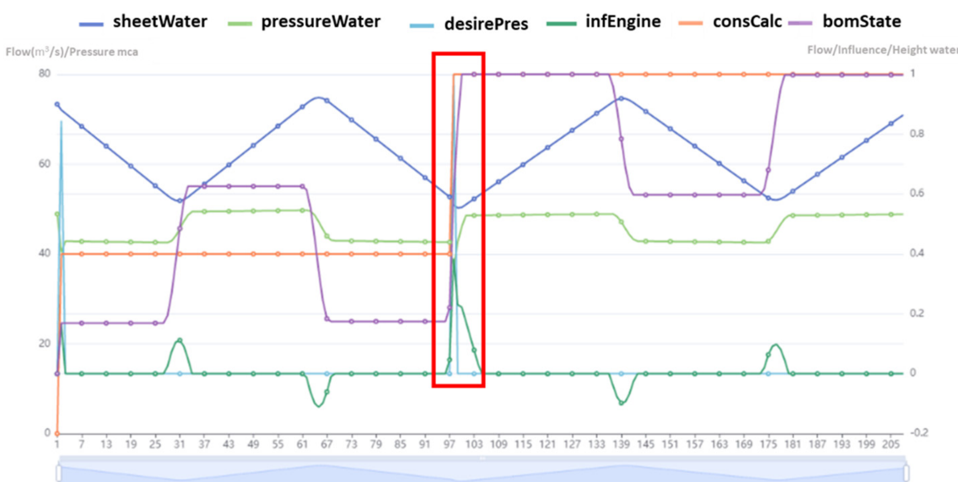


Fig. 7. Behavior of the system when consumption is increased to 80 m^3/h.

In the simulator, the model has been subjected to all kinds of system evolutions, always showing coherent behavior, trying to fill the tank at the desired levels, maintaining an adequate pressure and avoiding pumping when inertia is high. The interesting thing about the system is that it offers correct behavior in any situation, even those not contemplated during the modeling and that have not been observed in the real system. This means that the behavior of the system is continuous and that it always offers a response that tries to maintain the local interests of each agent, favors the stability of the infrastructure, and brings out a control aligned with what is sought in this control system.

As this is a simulator, the working ranges of the pumps have been idealized, so that the system aims for the minimum possible water storage, keeping the infrastructure conditions stable, but in exchange for having pumps capable of working at various power levels.

6. Discussion of the Results

The proposed control system presents several strengths, but also some problems. The main advantages are those concerning the management of multiple objectives. As mentioned above, each agent manages a local intelligence that takes care of part of the control. They only need to perceive a part of the world, and this makes modeling an objective easy, as it is not necessary to consider the complete state of the context. To modulate the influence that each objective has on the control, it is sufficient to adjust the agent that combines the influences, or the strength with which the influences are emitted on each agent. Adding or removing objectives, i.e. to change the nature of the control system, is done by adding or removing agents. It is even easy to temporarily isolate an objective and remove it from the control system by simply turning that agent off, i.e. setting its influence to neutral.

Another advantage is the robustness of the system in the face of incidents. As the logic is isolated in several agents, we are dealing with a purely distributed system, where each agent can even be separated into a different execution environment. Furthermore, they can be monitored so that if an agent fails, it is easy to replicate it in another environment. As for the scalability of the system, since it is a distributed system, it is only necessary to ensure that one agent can be executed, and as we have seen, each agent actually contains very limited logic.

However, there are also several disadvantages. One of the first that we can detect is that the system becomes indeterministic. This means that we cannot be sure that, faced with the same state of the world, the response will be the same, because among other things it will depend on the memory of the agents, that is, on their past experience, and on how the influence they generate affects the world. For as we have said, agents do not generate an exact and known change, but an attempt to change the world. For example, at a certain moment, an agent may want a water pump to turn on because it requires it to adjust a value, but nevertheless, the world may either change, or it may change for the worse. The point is that the system tries to adjust the world to its needs but cannot ensure it.

On the other hand, the control system is limited to the speed of communication with the real world, i.e. control depends on the reception of signals coming from the infrastructures, which trigger deliberation and generate influences. The triggers of control are therefore the signals from the world. In this case, we are therefore limited to being able to receive and send these signals. In addition to this communication, it will also be conditioned by the degree of distribution of the system, i.e. if each agent resides in a different execution context, the communication time between each execution context will have to be added to the time for sending and receiving data from the infrastructures. This can be restrictive in case we need real-time control responses.

7. Conclusions

The main contribution of our work is the creation of a sustainability-oriented control system, with different objectives from cost-oriented control systems, capable of controlling infrastructures in contexts of uncertainty, contradictory objectives, generic infrastructures, and reactive real-time control. Through the simulation of the system and using real data, it has been demonstrated that the new system generates the desired behavior, ensuring the continuous supply of the city.

This work allows the desired behavior to be divided into entities that use the signals of the physical system to achieve the local objectives of each agent. The union of all the interests makes the desired global behavior emerge, being a coherent behavior even in situations that were not initially modeled, such as the sudden change of consumption in ranges that would be abnormal (such as doubling consumption in a single instant). The main advantage offered by this system compared to other artificial intelligence alternatives is that it allows interests to be easily aggregated and control to be made more complex, starting from a divide-and-conquer strategy, and without knowing the whole system.

Another aspect is that it allows modeling the behavior of the system in any situation, which is especially useful in the design and development of decision support systems or even virtual modeling systems or digital twins. This way of constructing the system makes it possible to reflect any infrastructure, however complex it may be, and it is also possible to observe it closely as it evolves. This makes it easier not only to achieve the desired behavior but also to understand what is happening at any given moment and to observe how intentions influence the execution of the system. Giving more or less weight to the intentions makes the system more or less reactive, for example, to pressure variations, to the height of the water sheet or to the fact that there are inertias in the system. In large infrastructures, with several water tanks, distributed pumping systems and multiple consumption areas, being able to adopt strategies even at the local level can be of great interest. Other artificial intelligence techniques are not observable in their evolution, as they require additional techniques for the explainability of the algorithms, such as machine learning and deep learning.

Another very interesting aspect is that this allows you to raise the construction of in-fracture construction under another new paradigm, with a “just-in-time” approach, where the minimization of water storage is prioritized, in front of the traditional approaches to maximize it. Being able to model the system and check how it would behave in any situation makes it possible to study the design of new systems from the digital world, or the modification of existing ones to supply new situations, as cities grow or shrink, change their stationary nature or even industrial change generates new consumption models. So far, when developing systems to optimize infrastructure, only the optimization of energy consumption or infrastructure wear and tear is used, but not an approach based on environmental sustainability.

At present, prediction agents are being incorporated as a future line of work. Existing control agents are based on the perceived state of the world, but thanks to machine learning techniques, it is possible to generate predictions based on historical information. Our goal at this point is to bring this predictive character to the system so that it is able to make decisions not only on the basis of the current state, but in anticipation of the predicted future state. These synthetic signals, since they would not belong to the sensorization of the infrastructure, would provide new sources of information that would allow the sophistication of the system's behavior. In the long term, and once this feature has been incorporated, we are studying the development of a pilot experience on the water infrastructures of a small medium-sized Spanish city, thus being able to check *in situ* how the digital model manages the real world.

Acknowledgments

This work was supported by the UAIND22-01B Project "Adaptive Control of Urban Supply Systems" by the Office of the Vice President of Research of the University of Alicante, and the INNTA3/2022/3 project "Adaptive Control of Urban Supply Systems", by Agencia Valenciana de la Innovación.

ORCID

Carlos Calatayud Asensi https://orcid.org/0009-0007-5744-631X
José Vicente Berná Martínez https://orcid.org/0000-0002-9007-6054
Lucía Arnau Muñoz https://orcid.org/0009-0008-4555-4409
Francisco Maciá Pérez https://orcid.org/0000-0002-2516-4728

References

1. G. Merei, J. Moshövel, D. Magnor and D. U. Sauer, Optimization of self-consumption and techno-economic analysis of PV-battery systems in commercial applications, *Appl. Energy* **168** (2016) 171–178.
2. N. Good, E. Karangelos, A. Navarro-Espinosa and P. Mancarella, Optimization under uncertainty of thermal storage-based flexible demand response with quantification of residential users' discomfort, *IEEE Trans. Smart Grid* **6**(5) (2015) 2333–2342.
3. B. Ming, P. Liu, J. Chang, Y. Wang and Q. Huang, Deriving operating rules of pumped water storage using multiobjective optimization: Case study of the Han to Wei interbasin water transfer project, China, *J. Water Res. Plann. Manage.* **143**(10) (2017) 05017012.
4. J. Hoogesteger and P. Wester, Regulating groundwater use: The challenges of policy implementation in Guanajuato, Central Mexico, *Environ. Sci. Policy* **77** (2017) 107–113.
5. D. Kaya, F. Çanka Kılıç and H. H. Öztürk, Energy efficiency in pumps, in *Energy Management and Energy Efficiency in Industry: Practical Examples* (Springer International Publishing, Cham, 2021), pp. 329–374.
6. S. S. Skiena, *The Algorithm Design Manual*, Vol. 2 (Springer, New York, 1998).

7. X. Xiang, Q. Li, S. Khan and O. I. Khalaf, Urban water resource management for sustainable environment planning using artificial intelligence techniques, *Environ. Impact Assess. Rev.* **86** (2021) 106515.
8. L. Li, S. Rong, R. Wang and S. Yu, Recent advances in artificial intelligence and machine learning for nonlinear relationship analysis and process control in drinking water treatment: A review, *Chem. Eng. J.* **405** (2021) 126673.
9. P. Satrio, T. M. I. Mahlia, N. Giannetti and K. Saito, Optimization of HVAC system energy consumption in a building using artificial neural network and multi-objective genetic algorithm, *Sustain. Energy Technol. Assess.* **35** (2019) 48–57.
10. A. Candelieri, R. Perego and F. Archetti, Bayesian optimization of pump operations in water distribution systems, *J. Global Optimiz.* **71** (2018) 213–235.
11. M. Xiao and H. Yi, Building an efficient artificial intelligence model for personalized training in colleges and universities, *Comput. Appl. Eng. Educ.* **29**(2) (2021) 350–358.
12. A. Dorri, S. S. Kanhere and R. Jurdak, Multi-agent systems: A survey, *IEEE Access* **6** (2018) 28573–28593.
13. W. Shen, *Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing* (CRC Press, Ed. Taylor & Francis, Oxon, United Kingdom, 2019).
14. S. Knorn, Z. Chen and R. H. Middleton, Overview: Collective control of multiagent systems, *IEEE Trans. Control Network Syst.* **3**(4) (2015) 334–347.
15. J. V. Berna-Martinez and F. Marcia-Perez, Robotic control systems based on bioinspired multi-agent systems, *Int. J. Adv. Eng. Sci. Technol.* **8**(1) (2011) 32–38.
16. F. M. Pérez, L. Z. Mendez, J. V. B. Martínez, R. S. Lima and I. L. Fonseca, Architectural model of the human neuroregulator system based on multi-agent systems and implementation of system-on-chip using FPGA, *Microprocess. Microsyst.* **89** (2022) 104431.
17. J. V. Berna-Martinez and F. Macia-Perez, Model of competencies for decomposition of human behavior: Application to control system of robots, *Res. J. Appl. Sci., Eng. Technol. (06)* (2013) 2180–2191.
18. MASWaterControlSimulator, Public repository on GitHub with project source code, 2023, https://github.com/jvberna/MASWaterControlSimulator.