

## Review

# Embedding autonomy in large-scale IoT ecosystems using CAO and L4G-CAO

Iakovos T. Michailidis<sup>1</sup> · Athanasios Ch. Kapoutsis<sup>1</sup> · Christos D. Korkas<sup>1</sup> · Panagiotis T. Michailidis<sup>1,2</sup> · Kyriaki A. Alexandridou<sup>1</sup> · Christos Ravanis<sup>1</sup> · Elias B. Kosmatopoulos<sup>1,2</sup>

Received: 18 September 2020 / Accepted: 4 January 2021

© The Author(s) 2021 **OPEN**

## Abstract

Recently, special attention has been paid in developing methodologies and systems for embedding autonomy within smart devices (Things). Moreover, as Things typically operate in an interconnected IoT ecosystem, autonomous operation must be performed in a cooperative fashion so the different Things coordinate their autonomous actions towards meeting high-level objectives and policies. Embedding Things with cooperative autonomy typically requires a tedious and costly effort not only during the original ecosystem deployment but throughout its lifetime. The current study describes CAO (Cognitive Adaptive Optimization)—and its distributed counterpart L4G-CAO (Local for Global Cognitive Adaptive Optimization)—which can overcome this shortcoming. CAO and L4G-CAO—which have recently been introduced and tested in a variety of IoT applications—can embed Things with *cooperative autonomy in a plug-n-play fashion*, i.e., without requiring the aforementioned tedious and costly effort. Results of the application of the aforementioned approaches in three different application domains (smart homes and districts, intelligent traffic systems and coordinated swarms of robots) are also presented. The presented results demonstrate the potential, of both approaches, to exploit the IoT automation functionalities in order to significantly improve the overall IoT performance without tedious effort.

**Keywords** Autonomous IoT · Distributed Systems · Intelligent Control

## Abbreviations

AC	Air-Conditioning
ACH	Air Chillers
ADMM	Autonomy Decision-Making Mechanism
AHU	Air Handling Unit
AUV	Autonomous Underwater Vehicles
BCS	Base Case Scenario
BMS	Building Management System
CAO	Cognitive Adaptive Optimization
HVAC	Heating, Ventilation, and Air Conditioning
IoT	Internet of Things.
L4G CAO	Local4Global CAO
LS	Least-Squares

✉ Iakovos T. Michailidis, michaild@iti.gr; Athanasios Ch. Kapoutsis, athakapo@iti.gr; Christos D. Korkas, chriskorkas@iti.gr; Panagiotis T. Michailidis, panosmih@iti.gr; Kyriaki A. Alexandridou, kikialexan@iti.gr; Christos Ravanis, cravanis@iti.gr; Elias B. Kosmatopoulos, kosmatop@iti.gr | <sup>1</sup>Information Technologies Institute, Centre for Research and Technology Hellas (ITI-CERTH), Building B, 6th km Thessaloniki-Thermi Road, Thessaloniki 57001, Greece. <sup>2</sup>Department of Electrical and Computer Engineering, Democritus University of Thrace, Kimmeria Campus, Building B, Xanthi 67100, Greece.



Discover Internet of Things

(2021) 1:8

| <https://doi.org/10.1007/s43926-021-00003-w>

Published online: 24 February 2021

 Springer

NREC	Non-Renewable Energy Consumption
PID	Proportional Integral Derivative
PPD	Predicted Percentage Dissatisfied
PS	occupants Presence contact System
PV	Photovoltaic
RBC	Rule-Based-Controllers
RWTH	Rheinisch-Westfälische Technische Hochschule
SHE	Smart Energy Homes
SoS	System of Systems
STC	Smart Traffic Control
TD	manual Temperature Dial
VFC	Volume Flow Control
WS	Window-opening Sensor

## 1 Introduction

Recent research efforts have been focusing on developing methodologies that embed autonomy within smart devices (Things). Next generation IoT must be autonomous as well as cooperative so as to be able to autonomously coordinate Things actions towards meeting common high-level goals. Moreover, Things should also be able to compute and implement their intelligent actions in a highly *distributed/ self-sustained* manner as it is not possible to employ traditional centralized approaches in massive scale systems. Consider, for instance, smart home systems which are becoming more and more affordable for the home user. Embedding smart home systems with cooperative autonomy, where smart thermostats, electric appliances, electric chargers, etc., autonomously act and coordinate their actions based on indoor and weather conditions, varying energy prices, renewables' generation and user preferences can result in tremendous energy bill savings [5, 7, 12, 31, 44, 45, 54]. Most importantly, a massive implementation of cooperative intelligence capable of optimizing energy consumption to the benefit of an entire community (smart neighborhoods or smart cities) can have even more significant social and business impacts.

Distributed intelligent control methodologies are probably the best candidate for embedding Things with cooperative autonomy. The vast majority of such methodologies needs a model (mathematical or simulation-based) of the IoT ecosystem [8, 11, 14, 18, 24, 25, 33, 41, 49, 52]. Developing, however, a model for an IoT ecosystem is usually a quite complex and cumbersome - or sometimes not feasible at all - task; especially when large and heterogeneous (multi-domain) IoT implementations are considered. Most importantly, since the IoT ecosystems are constantly subject to changes (e.g., failures of some nodes, geographical expansion of the IoT ecosystem, addition/removal of Things, changes in external factors such as users' behavior), a repetitive revising/re-engineering process and verification of the model is usually needed. On the other hand, intelligent control methodologies that do not require an accurate model or are model-free [10, 16, 17, 29, 35, 50, 55], may exhibit an unacceptable performance due to poor adaptation while their application is typically limited to small- or medium-scale applications.

The authors have recently developed CAO (Cognitive Adaptive Optimization) [29, 35] and its distributed counterpart—the L4G-CAO (Local4Global Cognitive Adaptive Optimization) [30]. These two toolsets have extensively been demonstrated in a variety of large-scale real-life IoT applications, exhibiting a remarkably efficient behavior in embedding Things with cooperative autonomy that can overcome the above-mentioned shortcomings of state-of-the-art systems and approaches [1, 3, 4, 9, 20, 21, 27, 34, 36–39, 47]. CAO and L4G-CAO are model-free but contrary to the existing tools they do not present any poor performance problems. Thanks to their self-learning/self-tuning mechanisms, they are able to optimize the IoT performance in a rapid, safe and smooth-transient manner. Moreover, they are highly scalable as they can handle IoT applications of a very large-scale and complexity as well as applications that involve highly heterogeneous elements/entities. Finally, due to their self-adapting and self-learning capabilities, their operational and maintenance costs are minimal i.e., there is no need for tedious programming, verification and calibration prior or during the application due to IoT topology and ecosystem diversions.

The main purpose of this paper is to provide an overview of the use of CAO and L4G-CAO for embedding autonomy within IoT ecosystems. This overview covers theoretical results (reported in [29, 30, 35]) as well as practical implementations in different IoT-related applications (reported in [1, 3, 4, 9, 20, 21, 27, 34, 36–39, 47]) and concerns:

- A unified mathematical formulation of the problem of embedding cooperative autonomy within IoT ecosystems and the demonstration of how CAO and L4G-CAO can be employed for addressing such a problem;
- An overview of the main functionalities and mathematical attributes of CAO and L4G-CAO when applied for embedding autonomy in IoT ecosystems ;
- A brief overview of the results and main conclusions of implementing CAO and L4G-CAO in challenging real-life large-scale IoT ecosystems.

## 2 The Problem set-up

Let us consider an IoT ecosystem consisting of  $N$  Things (smart devices) with each of the Things being embedded with an *Autonomy Decision-Making Mechanism* - (ADMM) as follows:

$$u_i(t) = \varpi(\theta_i, z_i(t), d_i(t)) \quad (1)$$

where  $t$  denotes the time index;  $\varpi(\cdot)$  is a non-linear vector function;  $z_i(t)$ ,  $d_i(t)$  denote the vectors of *local data* (e.g, sensor measurements) and *external data* (e.g., information available through the web/cloud), respectively, available to the  $i$ th Thing at time  $t$ ; and  $\theta_i$  is a vector of *tunable parameters* configuring the ADMM of the  $i$ th Thing, i.e., for different choices of  $\theta_i$  we obtain different autonomous behaviours for the  $i$ th Thing. Let  $z$ ,  $d$ ,  $u$ ,  $\theta$  denote the *augmented vectors* of local and external data, actions and tunable parameters, of the overall IoT ecosystem:

$$z = \begin{bmatrix} z_1^r \\ \vdots \\ z_N^r \end{bmatrix}, d = \begin{bmatrix} d_1^r \\ \vdots \\ d_N^r \end{bmatrix}, u = \begin{bmatrix} u_1^r \\ \vdots \\ u_N^r \end{bmatrix}, \theta = \begin{bmatrix} \theta_1^r \\ \vdots \\ \theta_N^r \end{bmatrix}$$

The performance of the overall IoT ecosystem, is evaluated through an objective function (performance index) over a time-horizon  $T$

$$J = \sum_{t=0}^{T-1} \pi_t(z(t), d(t), u(t)) \quad (2)$$

where  $\pi_t$  are known non-negative functions.

**Example 1** To better understand the above definitions consider the example of a smart home that is comprised of  $N$  rooms: a smart device in each room is used to autonomously control the room's A/C (Air-Conditioning) set-points so as to (a) minimize energy bills and (b) keep the rooms' climate conditions (e.g., temperature, humidity, etc.) within some pre-specified limits. Then,  $z_i$  contains the  $i$ th room's indoor sensor conditions (e.g, temperature, humidity, etc.),  $u_i$  denotes the  $i$ th room's A/C set-point and  $d_i$  contains external information such as the current and forecasted external weather data, energy prices, etc. The functions  $\pi_i$  are typically calculated as follows:

$$\begin{aligned} \pi_t(z(t), d(t), u(t)) = & a \text{ Energy Consumed } (t) \\ & + b \text{ Penalty for indoor conditions } (t) \end{aligned}$$

where the function "Penalty for indoor conditions" penalizes the cost whenever some room's indoor conditions exceed the pre-specified limits and  $a$ ,  $b$  are appropriately defined weighting/normalizing factors. Finally, the time-horizon  $T$  is typically selected to be one day. See e.g., [4, 9, 27, 36] for more details on the above definitions.  $\square$

**Remark 1** Typically, the ADMM is designed using parametrized rule-based logics or it is based on standard control system theory tools. Apparently, the choice of the ADMM is crucial for the efficiency of the IoT ecosystem: it must be designed in such a way that different choices of its tunable parameter AC set-point meters  $\theta_i$  should cover all possible and feasible autonomous behaviours. The reader is referred to the practical applications described in the next two sections where examples of choices for the ADMM are provided.  $\square$

**Remark 2** The above formulation is valid not only in the case of IoT ecosystems consisting of homogeneous Things (like the smart home example provided above) but also for cases where heterogeneous Things live and interact in the same

ecosystem. Moreover, the formulation is still valid—under some minor modifications—in a System-of-Systems (SoS) set-up, where if instead of  $N$  Things, the overall ecosystem is “split” into  $N$  constituent “smaller” ecosystems with each of them consisting of a group of Things.  $\square$

Using standard results from systems theory (see e.g., [32]), it can be seen that the local data states are evolving according to an equation of the form

$$\begin{aligned} z_i(t+1) = & g_i(z(t), z(t-1), \dots, z(t-T_z), \\ & u(t), u(t-1), \dots, u(t-T_u), \\ & d_i(t), d_i(t-1), \dots, d_i(t-T_d)) \end{aligned} \quad (3)$$

where  $g_i(\cdot)$  is a non-linear vector function of its elements and  $T_z, T_u, T_d$  denote the local data state *memories*. The above equation describes the effect of the Thing's actions to the IoT ecosystem environment. For instance, in the case of Example 1, the above equation corresponds to effect the A/C set-points (controlled through the Thing's ADMM) to the rooms' climate conditions.

Replacing (3) into (2) and using (1), it can be seen after some algebraic manipulations [32] that *the performance index  $J$  is a function of the tunable parameters  $\theta$  and the history of the external data over the time-horizon  $T$ , i.e.,*

$$J \equiv J(\theta, D_T) \quad (4)$$

where  $D_T = [d^r(1), \dots, d^r(T)]$ . Therefore, *the problem of optimizing the overall IoT ecosystem performance can be mathematically formulated as the problem of finding the values for the tunable parameters  $\theta$  that optimize the cost criterion (4)*. Please note that the dynamics (3) are “hidden” in the equation (4): in other words, the computation of (4) requires knowledge of both the cost function elements  $\pi_i(\cdot)$  as well as the functions  $g_i(\cdot)$ . As a result, there are two main limitations when attempting to solve such an optimization problem:

- (Limitation 1). It is difficult, if feasible at all, to apply standard optimization approaches (such as e.g., gradient descent). Standard optimization approaches require an analytic form of the cost function (4) and since this function depends on the dynamics (3), knowledge of the analytic form of the overall IoT ecosystem dynamics is required. However, extracting the analytic form of the IoT ecosystem dynamics is an extremely difficult task, if not impossible at all, even for small-scale implementations. To make things even worse, as the IoT ecosystem is usually subject to minor or major changes (e.g., addition/removal of devices, changes in the end-users behaviour, etc.), a constant adaptation of the model for the IoT ecosystem is required.
- (Limitation 2). Intelligent, adaptive and/or learning approaches which do not require knowledge of the analytic form of the IoT ecosystem dynamics may exhibit [29, 35] a very poor performance due to adaptation which, in turn, may put safety of operations at stake. Moreover, typically such approaches are applicable to small- or medium-scale applications.

### 3 Centralized version: the cognitive-based adaptive optimization tool

CAO [29, 35] can overcome both Limitations 1 and 2, described in the previous section. Below, we provide a brief description of CAO along with its main properties. To start with, let us briefly explain how CAO is implemented. CAO starts with an initial set of tunable parameters  $\theta(0)$  and lets the ADMM mechanisms operate the Things over a time-horizon  $T$  by keeping the tunable parameters constant and equal to  $\theta(0)$ .<sup>1</sup>

After the system operates over  $T$  time-units, CAO evaluates its performance through the cost function  $J(0)$  and calculates  $\theta(1)$  using the algorithm of Table 1. This procedure is repeated for the next  $T$  time-units so as for CAO to calculate

<sup>1</sup> Typically, the initial set of tunable parameters  $\theta(0)$  is chosen based on past experience so as to provide a performance for the IoT which is safe and acceptable but, of course, far from being efficient. For instance, in applications such as smart homes/districts or intelligent traffic systems, such initial parameters  $\theta(0)$  can be easily extracted from the usual practice, see [3, 34, 38] for more details. More advanced choices of the initial parameters  $\theta(0)$  can also be used, see e.g. [4, 9, 27, 36, 37] for instance, where the initial parameters correspond to control-based principles. On the other hand, the choice of the time-horizon  $T$  depends on the particular application. For instance, in smart homes/districts and traffic systems, the horizon  $T$  is chosen to be a whole day (24h) as the system preserves a 24h periodicity.

**Table 1** The CAO Algorithm

At every  $k$ th iteration (where each iteration involves the IoT ecosystem operating for  $T$  time-units with  $\theta$  being constant and equal to  $\theta(k)$ ) measure the IoT ecosystem performance  $J(k)$  and update  $\theta$  using the following steps:

1. Construct an estimator for  $J(k+1)$  as follows:

$$\hat{J}(k+1) = \vartheta^T(k) \phi\left(\theta(k), D_T(k)\right) \quad (5)$$

where  $\hat{J}(k+1)$  denotes the estimate (prediction) of  $J(k+1)$ ,  $\phi$  is the regression vector and  $\vartheta$  is the estimator vector. Standard function approximation schemes (e.g. polynomials) can be used to construct estimator (5). The reader is referred to [29, 34] for more details on how to construct such an estimator (it must be emphasized that it suffices to use estimators of very “simple” structure and not very elaborate ones). The estimation vector  $\vartheta$  is constructed using standard Least-Squares (LS) estimation, i.e.,

$$\vartheta(k) = \underset{\chi}{\operatorname{argmin}} \sum_{\ell=k-W(k)}^{k-1} \left( \chi^T \phi\left(\theta(\ell), D_T(\ell)\right) - J(\ell+1) \right)^2$$

where  $W(k)$  denotes the time-window over which the LS estimation is taking place.

2. Choose a positive function  $\alpha(k)$  to be either a constant positive function or a time descending function satisfying  $\alpha(k) > 0$ ,  $\sum_{k=0}^{\infty} \alpha(k) = \infty$ ,  $\sum_{k=0}^{\infty} \alpha^2(k) < \infty$ .
3. Generate—randomly or pseudo-randomly—a set of  $L$  candidate perturbations  $\delta\theta^{(1)}(k), \delta\theta^{(2)}(k), \dots, \delta\theta^{(L)}(k)$  where  $\delta\theta^{(j)}(k)$  are vectors of the same dimension as  $\theta(k)$  and  $L$  is an integer satisfying  $L \geq 2\dim(\theta)$ .
4. Estimate the effect of each of the candidate perturbations to the current vector  $\theta(k)$  by employing the estimator (5) and pick the candidate perturbation with the “best” effect, i.e., choose the vector  $\delta\theta^{(j^*)}(k)$  that satisfies

$$\delta\theta^{(j^*)}(k) = \underset{j=1, \dots, L}{\operatorname{argmin}} \left\{ \vartheta^T(k) \phi\left(\theta(k) + \alpha(k) \delta\theta^{(j)}(k), \hat{D}_T(k+1)\right) \right\}$$

where  $\hat{D}_T(k+1)$  denotes an estimate (prediction) of  $D_T(k+1)$ .

5. Set  $\theta(k+1) = \theta(k) + \alpha(k) \delta\theta^{(j^*)}(k)$
  6. Go to step 1 until performance convergence has been achieved.
- The reader is referred to [29, 34] for more details on the CAO algorithm as well as for guidelines for the selection of the different design parameters of the algorithm (regression vector  $\phi$ ,  $\alpha(k)$ ,  $W(k)$ , etc.).

$\theta(2)$  using  $J(1)$ , then for the next  $T$  time-units in order to calculate  $\theta(3)$  using  $J(2)$  and so on. The details on how  $\theta(1), \theta(2), \dots$  are calculated are provided in Table 1. The next Theorem summarizes the main properties of CAO. Its proof can be found in [29].

**Theorem 1** Let  $D_T(k) - \hat{D}_T$  be zero-mean and bounded. Then, under some mild conditions on the continuity of  $J$ , the following hold:

- (a)  $\theta(k) \mapsto \theta^*$   
where  $\theta^*$  denotes a local optimum of  $J$ , i.e.,  $\nabla J(\theta^*, D_T(k)) = 0$ .
- (b)  $J(k+1) \leq J(k) + \mathcal{O}(|D_T(k) - \hat{D}_T|) + \epsilon(k)$   
where  $\epsilon(k)$  is a term that decays to zero exponentially fast.

In simple words, the above Theorem states that:

- CAO guarantees that the tunable parameters of the ADMM mechanisms will converge to their locally optimal values, provided that the prediction  $\hat{D}_T$  satisfies some typical assumptions (see the 3rd item in this list for more details). Apparently, the performance improvement depends on the nature of the particular local optimum  $\theta(k)$  where it will converge. If the ADMM mechanism is suitably chosen then the improvements that such local optimum may provide could be significant: for instance, in the practical applications, which are described later in this paper, the improvements can reach 30% or even higher. On the other hand, if the ADMM is chosen according to the procedure suggested in [2],  $\theta(k)$  converges to the point whose distance from the *globally optimal performance* is proportional

to the complexity of the ADMM mechanism: the more complex is the ADMM mechanism the closer to the globally optimum performance is obtained (at the expense, of course, of a convergence speed which is inversely proportional to the ADMM complexity).

- Part (b) of Theorem 1 establishes that CAO does not face the risk of the poor performance (which is one of the main shortcomings of other adaptive/learning approaches): the cost  $J(k + 1)$  is less than its value of the previous iteration plus two terms: (a) an—unavoidable—term that depends on the accuracy of the prediction of the external data and (b) a term that converges to zero exponentially fast. The exponentially fast convergence to zero is the best that any adaptive/learning algorithm can achieve [19]. As a matter of fact, in the vast majority of adaptive/learning schemes, a term similar to  $\epsilon(k)$  is always present - with the difference that such a term do not converge exponentially fast: as a result such a term may take significantly large values during adaptation, leading to situations of very poor or, even, unsafe performance.
- The properties of CAO are established based on some typical assumptions on the prediction  $\hat{D}_T$ . Apparently, any type of algorithm depends on the accuracy of the prediction  $\hat{D}_T$  which corresponds e.g., to weather predictions in the case of smart home/districts, traffic predictions in the case of traffic systems, etc.
- Last, but not least, it is emphasized that due to the model-free nature of CAO, it possesses self-reconfiguration capabilities: if the IoT infrastructure changes (e.g., nodes added/removed), then CAO will automatically re-learn and re-adjust the tunable parameters towards optimizing the altered system. The robotic application mentioned in this paper exhibits such an attribute: whenever the IoT system changes (because a node joins/leaves the system) or whenever the user requirements change (which corresponds to a change in the cost function  $J$  structure), CAO rapidly reconfigures itself towards efficiently optimizing the altered system.

### 3.1 Smart Traffic Control (STC): real-life application in the city of Chania, Greece

One of the benefits of the impressive recent advances of the field of IoT, is that it becomes more and more affordable and “easier” to deploy Smart Traffic Control (STC) systems to intelligently and more efficiently control and manage traffic operations [3, 34, 38]. Unfortunately, embedding STC systems with intelligence requires a tremendous amount of human effort and time for programming and tuning the IoT involved in these operations. The programming and tuning procedure involves the calibration, adjustment and programming of hundreds of parameters, rules, operational schedules, decision-making mechanisms, etc. and are typically performed by experienced personnel. Thus, because of the complexity of the problem, there is no guarantee that the overall programming and tuning procedure will end-up successfully.

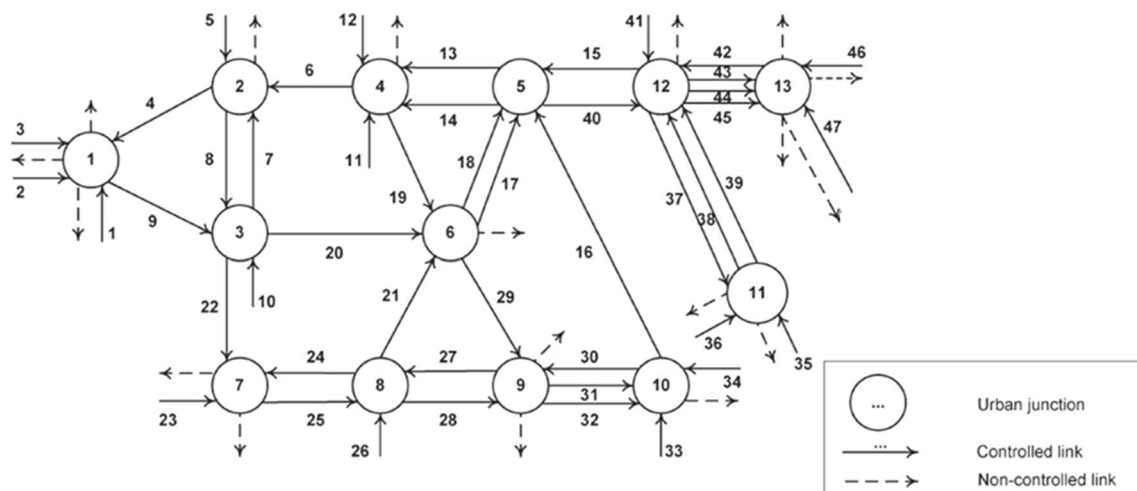
The CAO system has been implemented in a real-life STC system towards demonstrating its potential for providing an automated and systematic approach that will neglect the need for the tedious and costly human involvement. The particular STC system where CAO has been implemented is the STC system for the urban road network of the city of Chania, Greece (see Fig. 1) which is a highly challenging traffic network: it involves a very complex signalling structure; frequent illegal or double-parking which change the network characteristics and junction capacities in an unpredictable way; and a traffic demand that changes significantly throughout the year (Chania is a touristic city with its population increasing by almost 100% during summer time). It is also emphasized that the ADMM employed and its original parameters (i.e., the tunable parameters  $\theta$  before their tuning by CAO) correspond to a very well-designed STC system, achieving the best the state of the art can offer [28]. Table 2 provides the details of the CAO implementation for the STC system of the city of Chania.

The real-life results after implementing CAO for about 60 days (see Fig. 2), indicate that CAO was able to provide ~ 50% improvements over a well-designed STC system. The improvements have been calculated based on the productivity index (the mean speed achieved inside the network multiplied by the traffic demand). The calculations for estimating the cost savings in Table 2 assume a fuel consumption of 10L per 100Km [53] in urban areas and a price of 1.2 €/L. Table 3 summarizes the result of CAO application in the STC system of the city of Chania.

### 3.2 Smart energy homes (SEH): real-life applications in two large-scale buildings

Calculating the optimal decisions that balance energy and user needs is by no means an easy task. Extensive research and real-life experiments performed over the last decades exhibited that demand-optimized actions require modifying the operating set-points many times during the day, in an intelligent and delicate manner. Such decisions should also consider the complex interplays between diverse factors such as equipment and envelope dynamics, user comfort and needs, occupancy schedules, weather conditions, etc. [9, 13, 27, 40]. Things become way more complicated when local energy generation (renewable sources, spinning reserves etc.) and storage are involved: in this case, the problem of generating

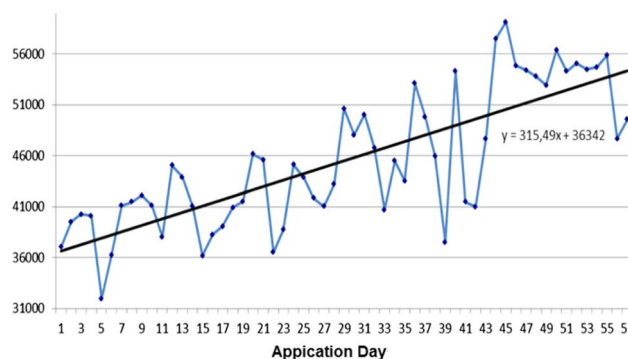




**Fig. 1** City of Chania Traffic Network

**Table 2** Details of CAO application in the STC system of the city of Chania

$N = 20$	Smart traffic junctions controlling in real-time green times
$u_i(t)$	Green times of traffic lights of the $i$ th junction
$z_i(t)$	Number of vehicles in each of the links (incoming roads) to the $i$ th junction
$d_i(t)$	No of vehicles (for each incoming link) entering the traffic network
ADDM	The ADDM consists of the strategy $TUC(\theta, z(t))$ , a well-established traffic control strategy which is based on control systems principles [28]. The initial set of parameters $\theta(0)$ were the ones obtained after a quite lengthy and tedious manual tuning in the past.
$T = 24h$	Time-horizon
$J(t)$	(average mean speed of the whole traffic network) $\times$ (total no of vehicles entering the traffic network)= System Productivity



**Fig. 2** Real-life Application of the CAO system to the urban road STC system of Chania: traffic network performance improvements (blue=real data; black=linear fit of the real-data). The x-axis corresponds to the number of days the CAO system is operating. The y-axis reflects the daily system performance in terms of (speed $\times$ demand), which is known as the system productivity

**Table 3** Results of CAO application in the STC system of the city of Chania

Annual fuel savings (due to reduction of travel times) as compared to the “best state-of-the-art”	1-2 Million €/year for an urban area of 100,000 people
Improvement of Traffic Network Performance as compared to the “best state-of-the-art”	~50%



**Fig. 3** **a** AFCON Ltd. building, Tel-Aviv, Israel (left); **b** Technical University of Crete building, Chania, Greece (right)

optimal decisions that guarantee the aforementioned attributes becomes way more complicated. Unfortunately, existing methods for calculating such optimal decisions usually rely on the analytic knowledge of the building dynamics [26, 42]. Apparently, such assumption is not realistic since developing an analytic model is an extremely “expensive” and cumbersome procedure. Moreover, such models would require continuous recalibration since SEH ecosystems are not static in time, a fact which renders the model “maintenance” extremely expensive when it comes to large-scale deployments. To make things worse, even in the cases where an elaborate model is available, existing methods for calculating the optimal decisions are computationally quite expensive [51].

The CAO system has been implemented in two real-life, large-scale SEH systems towards employing an automated and systematic control approach that is able to overcome the aforementioned drawbacks involved in existing solutions:

### 3.2.1 Application to the office building of AFCON Ltd. (Tel Aviv, Israel)

The first SEH system concerns the main office building of AFCON Ltd., which is located in a suburban area of Tel-Aviv, Israel. It was built in 2004 to host over 600 employees. The building is comprised of 5 floors: Floors 1 and 2 are used as storage spaces without any air conditioning units, while floors 3, 4 and 5 are consisted of offices (about 70 offices and rooms per floor) (see Fig. 3a). The net heated floor area (3rd, 4th and 5th floor in total) is around  $2350m^2$ . The daily energy demand is approximately  $11879kWh$  during spring period. Two couples of chillers are installed for indoor climating purposes; each of the chiller can deliver up to 150 refrigeration tons (total of 600 tons) which corresponds to  $527kW$  per chiller (total  $2.108MW$ ). The indoor air-conditioning system includes AHUs (Air Handling Units) for offices located on the same floor—on average 10 offices share the same AHU.

It should be noted that the comparison performance benchmark (base case scenario - BCS) is the common control practice adopted in the real-life building employing a constant chiller set-point of  $11^\circ C$  during working days. The AHU thermostats were constantly set to  $21^\circ C$ . Experiments were conducted focusing on the 3rd, 4th and 5th floors consisting of offices. The test period refers to the period from Monday 30/3/2015 to Friday 10/4/2015, when, due to mediocre outdoor conditions, the energy-efficiency of the BCS was poor. Table 4 provides the details of the CAO implementation for this application.

The evaluation results demonstrated that CAO led to substantial power savings of  $\sim 35\%$  translated into  $6711kWh$  average daily consumption, without violating the acceptable comfort bounds. An estimation of the potential savings—summarized in Table 5—in terms of energy cost, can be extracted considering that the benchmark control application requires  $11879kWh/day$  in average and CAO requires only  $6711kWh/day$ . Using the EU-28 average price of  $0.125€/kWh$  for industrial consumers [15], such difference can be translated in a daily amount of  $646€$  savings during summer period.

### 3.2.2 Application to an office building of Technical University of Crete, (Chania, Greece)

The second SEH application involves a 2-floor office building, located inside the campus of the Technical University of Crete, Greece (see Fig. 3b). The building area of  $450m^2$  is divided into 10 offices, each equipped with a  $12000btu$  conventional air conditioning unit as well as indoor temperature and humidity sensors. The building is also equipped with a photovoltaic (PV) panel, which provides solar energy to the building. The building is considered as a conventional building with poor insulation characteristics which render the problem of optimization and efficient control design to an extremely challenging one, due to the strong dependence of the indoor conditions to the outdoor ones. The energy



**Table 4** Details of CAO application in the SEH system of AFCON Ltd

$N = 2 \times 21$	AHU thermostat set-points regulating in real-time the water temperature
$u_i(t)$	Set-points of the $i - th$ chiller
$z_i(t)$	Indoor temperature for all 210 offices located on the 3rd, 4th and 5th floor
$d_i(t)$	Current and forecasted ambient temperature, total solar radiation and occupancy
ADDM	Combination of a linear controller and a rule-based controller
$T = 24h$	Time-horizon
$J(t)$	Weighted summation of the active chiller energy consumption and indoor comfort

**Table 5** Results of CAO application in the SEH system of AFCON Ltd.

Daily energy savings during spring period as compared to the “usual practice”	5168 kWh/day
Daily economic savings during spring period as compared to the “usual practice”	646 €/day

**Table 6** Details of CAO application in the SEH system of Chania Building

$N = 10$	Air-conditioner set-points
$u_i(t)$	Set-points of the $i - th$ air-conditioner
$z_i(t)$	Indoor temperature and humidity for all 10 offices
$d_i(t)$	Current and forecasted ambient temperature, outdoor humidity, total solar radiation and occupancy
ADDM	Combination of a linear controller and a rule-based controller
$T = 24h$	Time-horizon
$J(t)$	Weighted summation of the total energy consumption and indoor comfort

**Table 7** Results of CAO application in the SEH system of Chania Building

Daily energy savings during summer period during spring period as compared to the “usual practice”	35 kWh/day
Daily economic savings during summer period during spring period as compared to the “usual practice”	3.25 €/day

consumption is highest during the summer period when large cooling loads are required to achieve an acceptable indoor thermal conditioning. Large glass surfaces, combined with the Greek summer and the poor insulation factor of the building, usually lead to overheating. Therefore the respective tests focused on reducing the air-conditioners energy consumption during the summer period. The simple rule-based control strategy, which is used in the building control practice, was adopted as the base case for comparison purposes. The rule-based control employs a very simple strategy, which consists of keeping the air-conditioner set points constantly equal to 25 °C during the office hours, and turn them off outside office hours (8 : 00 – 17 : 00). Table 6 provides the details of the CAO implementation for this application.

CAO was able to reduce energy consumption by 19%, while indoor comfort conditions remained within acceptable comfort bounds. An estimation of the potential savings—summarized in Table 7—in terms of energy cost, can be extracted by considering that the benchmark control application requires 126kWh/day in average while CAO requires only 100kWh/day. Using the EU-28 average price of 0.125€/kWh for industrial consumers [15], such energy consumption difference can be translated in 3.25€ daily savings during the summer period.

### 3.3 Autonomous trajectory design system for AUVs: real-life application in the Port of Porto, Portugal

Another instance of IoT application is the deployment of underwater robots (AUVs) to accomplish underwater mapping, see e.g., [6, 22, 43, 46, 48]). Despite these advances, however, almost all underwater map-building methods are characterized by low autonomy, since they typically rely on a set of pre-defined trajectories and often on human intervention. In other words, AUVs usually follow trajectories designed off-line, before the actual deployment, which is a limiting factor



**Fig. 4** The three AUVs used in the Multi-AUV underwater experiments

**Table 8** Details of CAO application in the multi-AUV mapping test-case

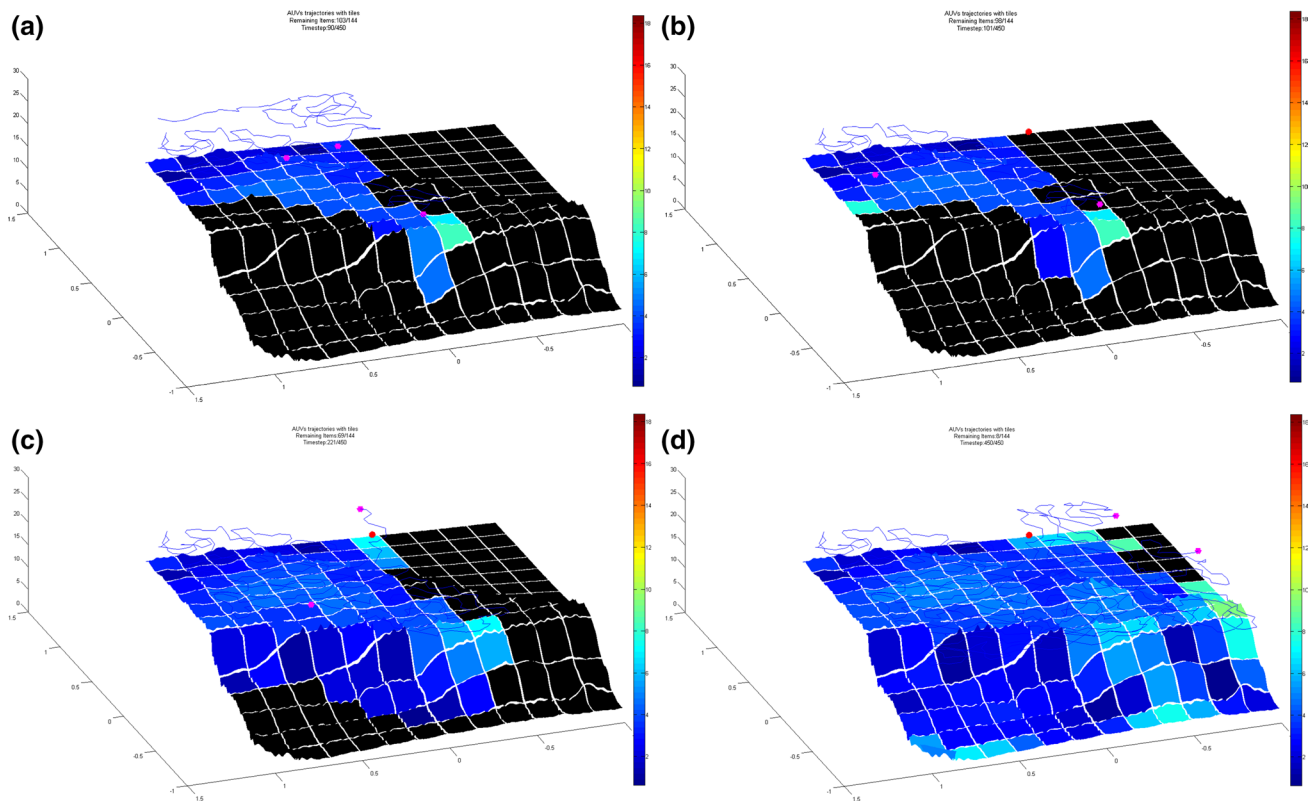
$N = 3$	Number of AUVs (Noptilus-1, Noptilus-2 and Noptilus-3)
Operation terrain $Q$	Square area with dimensions equal to $240 \times 240$ meters
$u_i(t)$	Robots' movements
$z_i(t)$	Terrain measurements (may be corrupted by noise)
$T = 450$ timesteps	Time-horizon (where by a new time-step is defined whenever new waypoints are sent to the AUVs)
$d = 1m$	Safety distance from the ground
$d_r = 0.5m$	Safety distance between any two robots
$J(t)$	Summation of the mapping performance on each tile

when a totally unknown area is to be mapped underwater: pre-defined trajectories are quite likely to “miss” areas rich in information or AUVs may waste valuable time focusing on low informative regions. A common approach for tackling these problems in practice, is to perform the following repetitive procedure. Initially, AUVs map the sea-floor following blindly defined trajectories (usually in a lawnmower pattern). Once this first step is accomplished, new trajectories are generated, always off-line, but now using the existing seabed knowledge from the constructed maps and this procedure is repeated many times.

To alleviate the previously described shortcomings we apply the centralized CAO algorithm. The aim of this research is to generate on-line trajectories for a team of AUVs in order to construct fast and accurate sea-floor maps [21] while also enabling the possibility to simultaneously track a dynamic event. Two different experiments were conducted in the Leixões Port, located in the city of Oporto, Portugal. Both experiments involved a fleet of 3 AUVs (called *Noptilus-1*, *Noptilus-2* and *Noptilus-3*) shown in Fig. 4. Table 8 provides the details of the CAO implementation for this application. Next we summarize the details of the 2 real-life experiments.

### 3.3.1 1st experiment: one AUV faces hardware malfunction during the mapping mission

In this experiment, we deployed the fleet of the 3 AUVs having as an objective to perform cooperative mapping of the seafloor using their bathymetric measurements. Figure 5a illustrates the progress of the 3 AUVs (blue lines) until time-step 90. The AUVs' positions, at this time-step, are depicted with the magenta spheres. The black tiles correspond to areas where the AUVs have not yet acquired any measurement, while the colorful ones correspond to the areas where the AUVs have started (and may have completed) their estimation process. The color in each one of them is an error index that varies from dark-blue, in case where the AUVs have acquired a perfect match from the ground truth, to dark-red in case where the measurements do not have any correspondence with the actual surface (ground truth map) that underlines the specific tile. It should be highlighted that the CAO algorithm does not use any information regarding the ground truth map (or error index): during the exploration process, the AUVs adjust their movements taking as input only their bathymeters' measurements and their locations (as estimated by the localization module). Figure 5b depicts the



**Fig. 5** Multi-AUV 1st experiment: **a** Exploration time-step 90 (top-left); **b** Noptilus-1 has stopped its exploration process (red thick sphere), Exploration time-step 100 (top-right); **c** Noptilus-2 undertakes the tiles of Noptilus-1, Exploration time-step 221 (bottom-left); **d** Completion of the experiment, Exploration time-step 450 (bottom-right)

time-step where a (simulated) hardware malfunction took place. The malfunctioned vehicle had to return immediately to the base-station to avoid jeopardizing such an extremely expensive infrastructure. Figure 5c exhibits the adaptation in the navigation schemes of the two remaining AUVs. The important feature here is that, one AUV autonomously chose to cover the tiles that would have been assigned, under normal conditions, to the damaged AUV. The mapping process was terminated after 450 time-steps when the AUVs covered the majority of the operation area having estimated 136 from 144 tiles (Fig. 5d). It is worth mentioning that in the majority of the estimated tiles, the AUVs acquired a satisfactory number of bathymeter's measurements, different in each case, since it is highly dependent on the actual morphology that underlines the tile. A comparison was also performed versus the usual practice of mapping using pre-defined trajectories [21]. The results of the comparison are summarized in Table 9.

### 3.3.2 2nd experiment: performing target tracking simultaneously with the mapping task

In this scenario, the task was to construct a map of the seafloor area while, concurrently, tracking the trajectory of a moving target. In this scenario we utilized a fleet of only 2 vehicles, due to the fact that the third available vehicle was utilized as the moving target. The information regarding the moving target was available through AUV-to-moving-target distance. In other words, the two AUVs do not know the position of the moving target, but they are using their AUV-to-moving target distance measurements in order to estimate the—dynamic—position of the target. Even from the initial time-steps, the difference from the previous experiment is evidential. Figure 6a depicts such an initial state, where one AUV seems to approach almost directly the position of moving target in order to minimize their in-between distance.

In a subsequent timestep (Fig. 6c) another feature of the utilized navigation algorithm can be observed. At this very moment, the distance between the target and any of the two AUVs was more or less the same. However, the bathymetric information below the AUV which was responsible for tracking the target, was far more important than the other one. The CAO algorithm without any build-in mechanism to detect and appropriately act on such cases, chose to “switch”

**Table 9** Results of CAO application in the multi-AUV mapping test-case

Accuracy as compared to the “usual practice”	1.5 more accurate maps
Mission Time as compared to the “usual practice”	50% reduction (at least)
Preparation/Pre-deployment Effort	90% reduction

the tasks between the two AUVs. By doing so, the AUVs (as a whole) were able to keep track of the movements of the moving target without undesired spikes on the estimated trajectory and, at the same time, to dedicate one vehicle to gather sensor data from regions where the mapping accuracy was low (Fig. 6d). The aforementioned switching process was performed several times during the experiment, in cases where the AUVs had more or less the same distance from the target and there was a clear advantage for the specific switching. It is worth highlighting that, the algorithm chose to make the transitions only when the AUVs' distances from the target were the same, in order to avoid sudden increases in the estimation error of the target's motion.

The experiment was terminated after 450 time-steps where the AUVs had accurately estimated, mainly but not limited, the area where the target was moving, while at the same time had almost perfectly estimated the target's trajectory.

#### 4 Distributed version: the Local4Global cognitive adaptive optimization tool

The CAO algorithm described in the previous sections, assumes a *centralized* form. However, in large-scale IoT implementations, such a centralized formation is not practically implementable: instead, the local parameters  $\theta_i$  of the  $i$ th Thing must be updated using only locally available information (plus information about the global criterion time-history). L4G-CAO [30] suitably revises CAO so as to meet such a requirement. Table 10 describes the details of the L4G-CAO algorithm.

The following Theorem provides the basic attributes of L4G-CAO which—despite the distributed nature of L4G-CAO—are similar to those of CAO.

**Theorem 2** Let  $D_{i,T}T(k) - \hat{D}_{i,T}$  be zero-mean and bounded. Then, under some mild conditions on the continuity of  $J$ , the following hold:

- (a)  $\theta(k) \mapsto \theta^*$   
where  $\theta^*$  denotes a local optimum of  $J$ , i.e.,  $\nabla J(\theta^*, D_T(k)) = 0$ .

$$J(k+1) \leq J(k) + \mathcal{O}(\sup_i |D_{i,T}(k) - \hat{D}_{i,T}|) + \epsilon(k)$$

- (b)  
where  $\epsilon(k)$  is a term that decays to zero exponentially fast.

**Proof** The proof—see also [35]—can be established by using standard results from representing state-space systems with input/output models. Using these results it can be seen that Theorem 2 is a direct application of Theorem 1. More precisely:

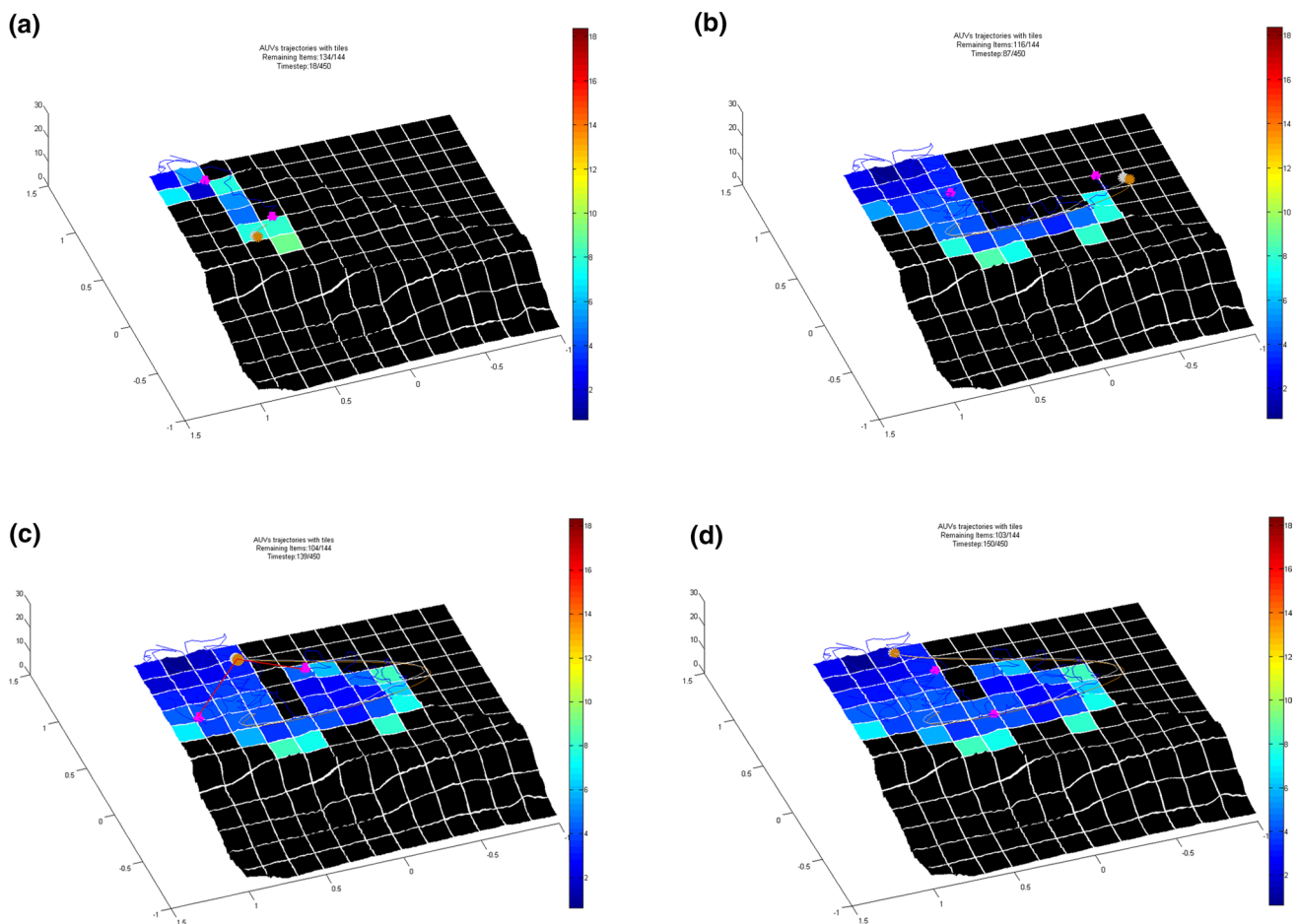
As a first step, it is not difficult for someone to see that the L4G-CAO algorithm assumes a mathematical form as follows:

$$\theta_i(k+1) = P_i(\theta_i(k), \hat{D}_{i,T}(k), J(k-1), \dots, J(k-d)) \quad (7)$$

for some nonlinear vector function  $P_i(\cdot)$ . Therefore, the overall L4G-CAO dynamics can be written in state-space form as follows:

$$\begin{aligned} \bar{\theta}(k+1) &= F(\bar{\theta}(k), \hat{D}_T(k), J(k-1), \dots, J(k-d)) \\ y(k) &= h(\bar{\theta}(k), \theta_i(k), D_T(k)) \end{aligned}$$

where  $\bar{\theta} = [\theta_1^r, \theta_2^r, \dots, \theta_{i-1}^r, \theta_{i+1}^r, \dots, \theta_N^r]^r$ ,  $F = [P_1^r, P_2^r, \dots, P_{i-1}^r, P_{i+1}^r, \dots, P_N^r]^r$  and  $y = J$ . Please note that  $\theta_i$  is considered as an exogenous input in the above equations. Using standard results from transforming state-space into input/output systems (see e.g., Theorem 2 in [32]) we can see that



**Fig. 6** Multi-AUV 2nd experiment: **a** Noptilus-3 approaches the target in order to improve its estimation, Exploration time-step 18 (top-left); **b** Noptilus-3 keeps tracking of the target, while the Noptilus-1 take measurements in order to produced a detailed map, Exploration time-step 87 (top-right); **c** The target tracking task is assigned to Noptilus-1, Exploration time-step 139 (bottom-left); **d** Noptilus-3 is re-sensing the underestimated tiles, while Noptilus-1 keeps tracking of the target. Exploration time-step 150 (bottom-right)

$$y(k+1) \equiv J(k+1) = \mathfrak{F}_i \left( J(k), J(k-1), \dots, J(k-d), \theta_i(k), D_T(k), \hat{D}_T(k) \right)$$

where  $\mathfrak{F}_i(\cdot)$  denotes an unknown nonlinear function. Therefore, the global performance index  $J(k)$  can be calculated—at the  $i$ th Thing level—through a nonlinear function  $\mathfrak{F}_i(\cdot)$  by using the previously measured values of  $J$ . By defining  $\mathcal{D}(k) = [J(k), J(k-1), \dots, J(k-d), D_T^r(k), \hat{D}_T^r(k)]^T$ , we have that the problem of optimizing  $J$  can be transformed into the problem of optimizing the cost  $\bar{J}_i$  at the  $i$ th Thing level, where  $\bar{J}_i$  is as follows:

$$\bar{J}_i \left( \theta_i(k), \mathcal{D}(k) \right) \equiv \mathfrak{F}_i \left( J(k), J(k-1), \dots, J(k-d), \theta_i(k), D_T(k), \hat{D}_T(k) \right)$$

and thus the CAO algorithm—and its attributes—are directly applicable by replacing  $J, D_T$  in CAO by  $\bar{J}_i, \mathcal{D}$ , respectively.  $\square$

#### 4.1 Distributed smart energy systems (DSES): real-Life application in a large-scale building

The first of L4G-CAO experiments concerns the case where there is a number of independent SEH (Smart Energy Home) systems in a large building, with each SEH system operating over a distinct part of a building (e.g., each apartment or office of the building is equipped with a distinct SEH system that operates independently of the others). The different



**Table 10** The L4G-CAO Algorithm

At every  $k$ th iteration (where each iteration involves the IoT ecosystem operating for  $T$  time-units with  $\theta_i$  being constant and equal to  $\theta_i(k)$ ) measure the IoT ecosystem performance  $J(k)$  and assume that the value of  $J(k)$  is available to each of the Things. Then,  $\theta_i$  is updated using the following steps:

1. Construct an estimator for the global performance  $J(k+1)$  at the  $i$ th Thing level as follows:

$$\hat{J}_i(k+1) = \vartheta_i^T(k) \phi \left( \theta_i(k), D_{i,T}(k), J(k-1), \dots, J(k-d) \right) \quad (6)$$

where  $\hat{J}_i(k+1)$  denotes the estimate (prediction) of  $J(k+1)$ ,  $\phi$  is chosen as in the CAO case,  $d$  is a positive integer chosen typically in the range 5 – 10 and  $D_{i,T}(k) = [d_i^T(1), \dots, d_i^T(N)]^T$ . Please note that each Thing has its own estimator.

The estimation vector  $\vartheta_i$  is constructed using standard Least-Squares (LS) estimation, i.e.,

$$\vartheta_i(k) = \underset{\chi}{\operatorname{argmin}} \sum_{\ell=k-W(k)}^{k-1} \left( \chi^T \phi \left( \theta_i(\ell), D_{i,T}(\ell), J(\ell-1), \dots, J(\ell-d) \right) - J(\ell+1) \right)^2$$

where  $W(k)$  denotes the time-window over which the LS estimation is taking place.

2. Choose  $\alpha(k)$  as in the case of CAO.
3. Generate—randomly or pseudo-randomly—a set of  $L_i$  candidate perturbations  $\delta\theta_i^{(1)}(k), \delta\theta_i^{(2)}(k), \dots, \delta\theta_i^{(L_i)}(k)$  where  $\delta\theta_i^{(j)}(k)$  are vectors of the same dimension as  $\theta_i(k)$  and  $L_i$  is an integer satisfying  $L_i \geq 2\dim(\theta_i)$ .
4. Estimate the effect of each of the candidate perturbations to the current vector  $\theta_i(k)$  by employing the estimator (6) and pick the candidate perturbation with the “best” effect, i.e., choose the vector  $\delta\theta_i^{(j^*)}(k)$  that satisfies

$$\delta\theta_i^{(j^*)}(k) = \underset{j=1, \dots, L_i}{\operatorname{argmin}} \left\{ \vartheta_i^T(k) \phi \left( \theta_i(k) + \alpha(k) \delta\theta_i^{(j)}(k), \hat{D}_{i,T}(k), J(k-1), \dots, J(k-d) \right) \right\}$$

where  $\hat{D}_{i,T}(k+1)$  denotes an estimate (prediction) of  $D_{i,T}(k+1)$ .

5. Set

$$\theta_i(k+1) = \theta_i(k) + \alpha(k) \delta\theta_i^{(j^*)}(k)$$

6. Go to step 1 until performance convergence has been achieved.

SEH systems are not allowed to communicate to each other due to e.g., privacy preserving reasons. The only information that is common to all different SEH systems is the total daily energy performance of the whole building along with a daily comfort index indicating the degree of satisfaction in all the different apartments/offices (for instance, this index may correspond to the worst of comfort conditions among all different apartments/offices).

The particular building where the L4G-CAO experiments were performed is an office-building that belongs to E.ON. Energy Research Centre of RWTH University and is located in Aachen, Germany. Figure 7 below, illustrates the building's south façade and its ground-floor plan. The available control and sensing infrastructure consisted of:

- sensors: room temperature (T), room CO<sub>2</sub> level, occupants' presence contact (PS), window-opening sensor (WS), manual temperature dial (TD) and energy measuring devices in each room, and;
- actuators: (i) Air Chiller (ACH) systems for cooling the supply air from the central air handling unit individually for each room; and (ii) Volume Flow Control (VFC) systems, for adjusting the air flow rate individually for each room, separately in supply and exhaust air duct.

It must be emphasized that the energy supplied was a mixture of renewable and non-renewable (i.e. from the power distribution grid) energy provided by the central supply system.

The usual case for buildings located in northern climates suggests that the largest amount of the total energy demand is consumed during winter and autumn periods, mainly for heating purposes. For this reason, the L4G-CAO real-life experiments were conducted during 21<sup>st</sup> – 26<sup>th</sup> of November 2016. The goal of L4G-CAO was to reduce the Non-Renewable Energy Consumption (NREC) while keeping user comfort at satisfactory levels. Table 11 provides the details of the L4G-CAO implementation for this application.

For comparison purposes, the L4G-CAO strategy is compared with the base case control strategy. The base case control strategy has been designed and implemented in the respective Building Management System (BMS) by the planners and the commercial system provider in a conventional manner. Such a strategy employs a closed PID-based control-loop, designed to react on room temperature and CO<sub>2</sub> deviations on ACs and VFCs. It should be noted that three rooms of about 30m<sup>2</sup> each were utilized for the L4G-CAO application (see Fig. 7b blue area). Moreover, two neighboring rooms





**Fig. 7** **a** RWTH E.ON. Building south facade (left); **b** RWTH E.ON. Building ground-floor plan overview (right)

with similar thermal characteristics, where the benchmark control was applied during the experimental period, served as the base case control scenario test-bed (see Fig. 7b red area). The real-life application of the L4G-CAO optimization tool employed a distributed topology to ensure seamless scalability and confirmed all of the aforementioned properties in real-life operating conditions. It is worth mentioning that NREC improvements could be observed even from the very first experimental day. The total improvement of the defined NREC index was 34% during the considered test period. In particular, during the experiments the average daily NREC consumption was about  $0.067 \text{ kWh/m}^2/\text{day}$  in the benchmark control case (see red circled area in Fig. 7b) while in the L4G-CAO case it was reduced to  $0.043 \text{ kWh/m}^2/\text{day}$  (see blue circled area in Fig. 7b). Note that internal solar heat gains were also negligible during the experimental period therefore indoor solar heat gains did not affect the evaluation process. In addition, the indoor comfort levels achieved were similar in both L4G-CAO and the base case control strategy.

An estimation of the potential savings in terms of non-renewable energy cost can be extracted considering that the benchmark control application requires  $0.067 \text{ kWh/m}^2/\text{day}$  in average and L4G-CAO  $0.043 \text{ kWh/m}^2/\text{day}$ . Using the EU-28 average price of  $0.125 \text{ €/kWh}$  for industrial consumers [15], daily savings of  $0.003 \text{ €/m}^2/\text{day}$  during the cold period of the year can be obtained (Table 12).

## 4.2 Distributed Smart Energy Systems (DSES): simulated application in a microgrid of 100 buildings

The second experiment of L4G-CAO concerns a simulated experiment of a connected microgrid of 100 buildings with each of the buildings equipped with each own independent SEH system (see Fig. 8). Moreover, the buildings of the microgrid share different energy sources: first, renewable energy sources (photovoltaic panels) are shared as a 'must-take' source, i.e. photovoltaic energy is always used when it is available; as a second source, the microgrid is also connected to the main electricity grid, i.e. if the output of the renewable energy sources is not enough, the extra electricity is absorbed from the main grid. In the following, more details about the different components of the microgrid are given.

It is important to underline that each one of the 100 buildings has a different size, different orientation, and different occupancy schedule (cf. Table 13): this implies that each building has different energy needs. For example, because of the orientation, each building receives a different portion of solar radiation, which might influence drastically the selection of the Heating, Ventilation, and Air Conditioning (HVAC) set point in each room (and thus the energy need). The size of the building and the fact that the building is occupied or not are additional factors influencing the selection of the HVAC set point. In particular, Table 13 shows that buildings may have 10, 6 or 4 rooms: the size of the buildings goes from  $300$  to  $900 \text{ m}^2$ , and the rooms in a single building have the same size. Buildings may host office activities, commercial activities, or residential activities. Each activity has its own occupancy schedule. It is assumed that all the rooms of a building exhibit the same occupancy pattern. Table 14 provides the details of the L4G-CAO implementation for this application.

**Table 11** Details of L4G-CAO application in the SEH system of RWTH E.ON

$N = 6$	ACH and VFC subsystem per test room (total of 3 test rooms)
$u_i(t)$	ACH set-points and VFC exhaust and supply set-points in each room
$z_i(t)$	Indoor temperature and $CO_2$ for all 3 rooms
$d_i(t)$	Current and forecasted ambient temperature, total solar radiation and occupancy
ADDM	Combination of a linear controller and a rule-based controller
$T = 24h$	Time-horizon
$J(t)$	Weighted summation of the NREC and indoor comfort index

**Table 12** Results of L4G-CAO application in the SEH system of RWTH E.ON

Daily NREC savings during test period compared to the “usual practice”	0.024 kWh per $m^2$ per day
Daily economic savings during test period compared to the “usual practice”	0.003€ per $m^2$ per day

The L4G-CAO results are compared to two the base case control strategies  $RBC_{24^\circ C}$  and  $RBC_{24^\circ C}$ , which are two Rule-Based-Controllers setting the building set-points to  $24^\circ C$  or  $24^\circ C$  when occupants are present. The two histograms presented in Fig. 9, have been obtained from a one-week simulation. The first histogram presents the energy absorbed from the grid in € for each type of buildings and the whole microgrid. The second histogram presents the mean percentage of people who are dissatisfied. Similarly to the single building test case, L4G-CAO achieves better scores in both histograms. In particular, with respect to  $RBC_{24^\circ C}$ , L4G-CAO manages to save more than 400€ for the whole system, while maintaining the comfort at better levels. On the other hand, L4G-CAO achieves a slightly better energy cost than  $RBC_{24^\circ C}$ : the energy cost is slightly better despite the pre-cooling effect implemented by L4G-CAO that demands more energy consumption. Table 15 summarizes the results of the application of L4G-CAO to the microgrid case.

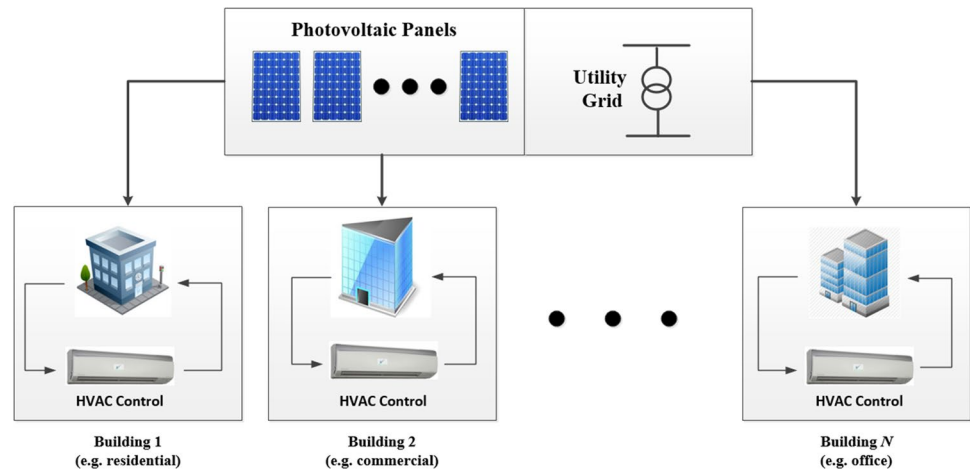
### 4.3 Continuous monitoring/inspection of critical infrastructures utilizing a team of robots (simulated experiment)

The final L4G-CAO application concerns a multi-robot mission where the objective is to continuously monitor an area of interest using the team of robots. Such tasks can be found in several real-life applications including: surveillance in hostile environments (i.e. areas contaminated with biological, chemical or even nuclear wastes), environmental monitoring (i.e. air quality monitoring, forest monitoring), and law enforcement missions (i.e. border patrol), etc. The task of continuous monitoring can be shortened to the task of designing the robots trajectories, in real-time, so that:

- (1) the part of the terrain that is monitored (i.e. visible) by the robots is maximized;
- (2) for every point in the terrain, the closest robot is as close as possible to that point.

The second objective is significant for two practical reasons: (a) at first, the closer is the robot to a point in the terrain, the better its ability to monitor this point becomes and (b) secondly, in many multi-robot monitoring applications, fast and accurate robot intervention (when needed) is highly essential. More information about this problem set-up along with the specialized version of the distributed-CAO algorithm for it can be found in [23].

To validate our approach in a realistic environment, we used data which were collected from the Birmensdorf area in Zürich. The main constraints imposed on the robots are that they must remain within the terrain's limits, i.e. within  $[x_{min}, x_{max}]$  and  $[y_{min}, y_{max}]$  in the  $x$ - and  $y$ - axes, respectively. At the same time they have to satisfy a maximum height requirement whilst not hitting the terrain, i.e. they must remain within  $[z + d, z_{max}]$  along  $z$ -axis. Moreover, the operational robots had a maximum threshold regarding their sensors' capabilities, i.e.  $\|x_i - q\| < thres$  where  $x_i$  denotes the

**Fig. 8** Microgrid test case**Table 13** Building composition and type of activity/occupancy schedule for the microgrid test case

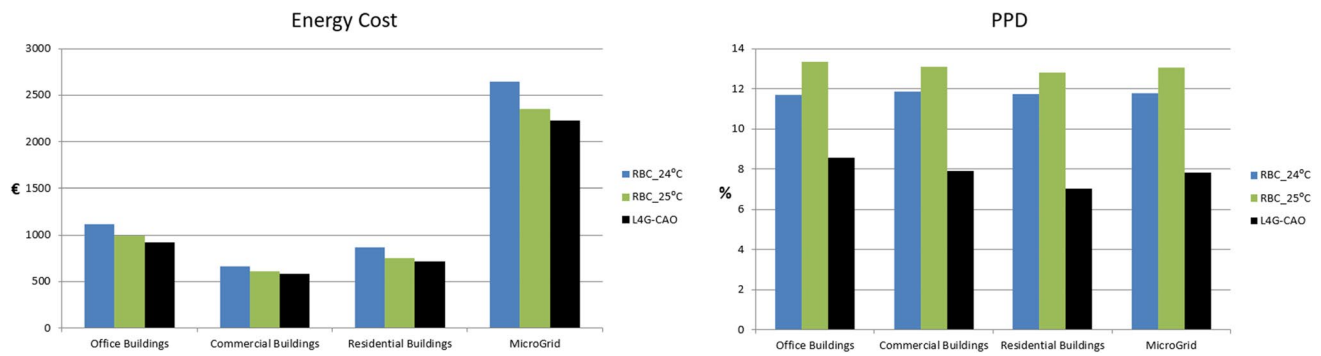
No. of buildings	Rooms for building	Size of building
40	10	From 300 to 900 m <sup>2</sup>
30	6	From 300 to 900 m <sup>2</sup>
30	4	From 300 to 900 m <sup>2</sup>
Type of activity	Occupancy schedule	
Office	6–18	
Commercial	7–14 and 17–21	
Residential	0–7 and 14–18 and 21–24	

**Table 14** Details of L4G-CAO application in the microgrid of 100 Buildings

$N = 700$	AC subsystem per test room
$u_i(t)$	AC set-points
$z_i(t)$	Indoor temperature, humidity and occupancy for all rooms
$d_i(t)$	Current and forecasted ambient temperature, total solar radiation and occupancy
ADDM	Constant AC set-point = 24 °C or Constant AC set-point = 24 °C
$T = 168h$	Time-horizon
$J(t)$	Weighted summation of the NREC and indoor comfort index

3D position of the  $i$ th robot and  $q$  any point of the surface. Finally, any two robots should have always a safety distance of  $d_r$ , i.e.  $\|x_i - x_j\| < d_r, \forall i, j \in \{1, \dots, N\}$ . The details of the performed experiments are summarized in Table 16.

Several initial configurations for the robot team were tested. In Fig. 10 the cost function of an illustrative scenario is presented, while the initial and the final configuration of the team (for the same scenario) is displayed in Figs. 11 and 12 respectively. Please note that, in both figures, the color in each cell of the surface, denotes the closest robot that actively monitors that cell. If the cell is marked with black color, it means that no robot is able to monitor that cell, either due to the maximum visibility range or the geometry of the environment. In Table 17 the final achieved coverage percentage for different initial configurations and different clustering in the Birmensdorf area, is presented.



**Fig. 9** District: energy cost in € and Percentage of Dissatisfied People during a 1 week experiment

**Table 15** Results of L4G-CAO application in the microgrid of 100 Buildings

NREC savings during test period compared to the “usual practice”	40€ per week per building
Indoor comfort conditions compared to the “usual practice”	33% improvement

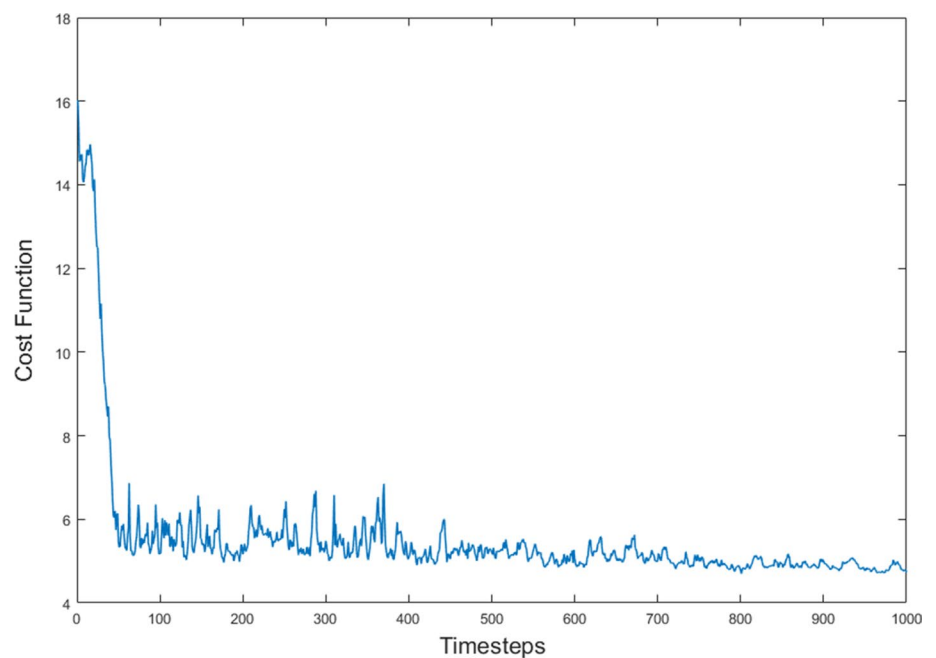
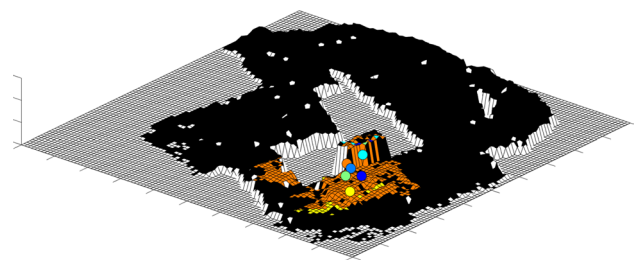
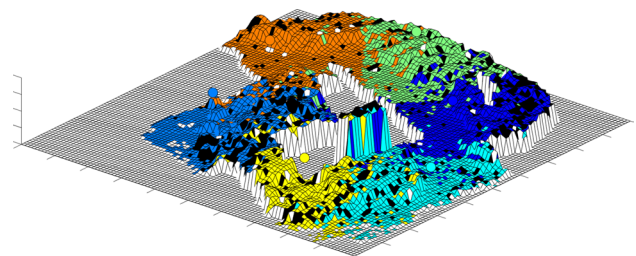
## 5 Conclusions

Despite the complexity and heterogeneity aspects involved in IoT, the CAO and L4G-CAO methodologies have presented a quite robust and inter-operable behavior in all application domains considered herein. The absence of elaborate simulation models and analytic knowledge of the specified use case scenario did not hinder the applicability of both methodologies due to their model-free operation feature.

CAO and L4G-CAO applications proved the high potential of model-free intelligent control in orchestrating a cooperative web of autonomously acting entities in order to improve the overall IoT performance in a real-time cognitive manner. Both have been evaluated in three different application domains under diverse conditions and scenarios presenting a quite promising behavior. CAO and L4G-CAO were able to improve significantly the overall IoT performance as compared to well-established base case strategies.

**Table 16** Details of L4G-CAO application in the multi-robot monitoring test-case

$N = 6$	Number of robots
$u_i(t)$	Robots' movements
$z_i(t)$	Terrain measurements (may be corrupted by noise)
$T = 1000$ timesteps	Time-horizon
$d = 0.5m$	Safety distance from the ground
$d_r = 0.5m$	Safety distance between any two robots
$thres = 16m$	Maximum visibility of the robots
$J(t)$	Summation of the distance between any point of the terrain with the closest robot and the number of invisible points
Advantages over usual practice	No need of tele-operation and explicit coordination

**Fig. 10** Cost function evolution in the scenario of monitoring an unknown terrain**Fig. 11** Initial robots' configuration. Black area corresponds to the area that has to be monitored**Fig. 12** Final robots' positions along with their sub-areas of responsibility

**Table 17** Coverage percentage for different initial configurations and different clustering in the Birmensdorf area

	(% coverage)				
Test case	1	2	3	4	5
Initial configuration	8.62	44.35	19.78	15.46	29.89
Final configuration	86.61	86.27	86.23	85.53	85.95

**Acknowledgements** The research leading to these results has been partially funded by the European Commission:—FP7-ICT-2013.3.4, Advanced computing, Embedded Control Systems [Contract Number 611538] Local4Global - <http://www.local4global-fp7.com/>,—H2020-EU.2.1.1.3. - Future Internet: Software, hardware, Infrastructures, technologies and services [Contract Number 645220] RAWFIE <http://www.rawfie.eu/>,—H2020-SEC-20-BES-2016: Border Security: autonomous systems and control systems, Sub-topic 1: Autonomous surveillance [Contract Number 740593] ROBORDER <http://roborder.eu/>,—H2020-EU.2.1.5.2. - Technologies enabling energy-efficient systems and energy-efficient buildings with a low environmental impact [Contract Number 768735] PLUG-N-HARVEST <http://www.plugin-harvest.eu/>,—H2020-EU.2.1.5.2., Technologies enabling energy-efficient systems and energy-efficient buildings with a low environmental impact [Contract Number 958284] PRECEPT <https://www.precept-project.eu/>.

**Authors' contributions** All authors approved the version to be published; and agree to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved. In specific: Iakovos TM, ACK, CDK, EBK; made substantial contributions to the conceptual idea of the work; the experiments execution, data acquisition, analysis, and interpretation; drafted as well as thoroughly revised the work for important intellectual content. PM, KA, CR; made substantial contributions to the analysis and interpretation of data; drafted the work accordingly. All authors read and approved the final manuscript.

**Funding** The research leading to these results has been partially funded by the European Commission:—FP7-ICT-2013.3.4, Advanced computing, Embedded Control Systems [Contract Number 611538] Local4Global - <http://www.local4global-fp7.com/>,—H2020-EU.2.1.1.3. - Future Internet: Software, hardware, Infrastructures, technologies and services [Contract Number 645220] RAWFIE <http://www.rawfie.eu/>,—H2020-SEC-20-BES-2016: Border Security: autonomous systems and control systems, Sub-topic 1: Autonomous surveillance [Contract Number 740593] ROBORDER <http://roborder.eu/>,—H2020-EU.2.1.5.2. - Technologies enabling energy-efficient systems and energy-efficient buildings with a low environmental impact [Contract Number 768735] PLUG-N-HARVEST <http://www.plugin-harvest.eu/>,—H2020-EU.2.1.5.2., Technologies enabling energy-efficient systems and energy-efficient buildings with a low environmental impact [Contract Number 958284] PRECEPT <https://www.precept-project.eu/>.

**Availability of data and materials** All authors ensure that all data and materials analyzed herein; support their published claims and comply with field standards.

**Code availability** Not applicable.

**Competing interests** The authors have no conflicts of interest to declare.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Aliubavicius U, Obermaier J, Fourati W, Manolis D, Michailidis IT, Diakaki C, Kosmatopoulos EB, Krause M. Use of system of systems and decentralized optimization concepts for integrated traffic control via dynamic signalization and embedded speed recommendation. *Transp Res Proc.* 2016;14(Supplement C):3416–25. <https://doi.org/10.1016/j.trpro.2016.05.300>.
2. Baldi S, Michailidis I, Kosmatopoulos EB, Ioannou PA. A “plug and play” computationally efficient approach for control design of large-scale nonlinear systems using cosimulation: a combination of two “ingredients”. *IEEE Control Syst.* 2014;34(5):56–71. <https://doi.org/10.1109/MCS.2014.2333272>.
3. Baldi S, Michailidis I, Ntampasi V, Kosmatopoulos EB, Papamichail I, Papageorgiou M Simulation-based synthesis for approximately optimal urban traffic light management. In: 2015 American Control Conference (ACC), 2015;868–873. <https://doi.org/10.1109/ACC.2015.7170843>
4. Baldi S, Michailidis I, Ravanis C, Kosmatopoulos EB. Model-based and model-free “plug-and-play” building energy efficient control. *App Energy.* 2015;154(Supplement C):829–41. <https://doi.org/10.1016/j.apenergy.2015.05.081>.
5. Bhati A, Hansen M. Energy conservation through smart homes in a smart city: a lesson for singapore households. *Energy Policy.* 2017;104(Supplement C):230–9. <https://doi.org/10.1016/j.enpol.2017.01.032>.



6. Carrasco PLN, Bonin-Font F, Campos MM, Codina GO. Stereo-Vision Graph-SLAM for robust navigation of the AUV SPARUS II. *IFAC-Papers-onLine*. 2015;48(2):200–5.
7. Carroll J, Lyons S. Reducing household electricity demand through smart metering: the role of improved information about energy saving. *Energy Econ*. 2014;45(Supplement C):234–43. <https://doi.org/10.1016/j.eneco.2014.07.007>.
8. Cetinkunt S, Yu WL. Closed-loop behavior of a feedback-controlled flexible arm: a comparative study. *Int J Rob Res*. 1991;10(3):263–75. <https://doi.org/10.1177/027836499101000307>.
9. Korkas CD, Baldi S, Michailidis I, Kosmatopoulos EB. Intelligent energy and thermal comfort management in grid-connected microgrids with heterogeneous occupancy schedule. *Appl Energy*. 2015;149(Supplement C):194–203. <https://doi.org/10.1016/j.apenergy.2015.01.145>.
10. Claessens BJ, Vanhoudt D, Desmedt J. Model-free control of thermostatically controlled loads connected to a district heating network. *Energy Build*. 2018;159(Supplement C):1–10. <https://doi.org/10.1016/j.enbuild.2017.08.052>.
11. D'Angelo G, Ferretti S, Ghini V. Simulation of the internet of things. In: 2016 International Conference on High Performance Computing Simulation (HPCS), 2016;1–8. <https://doi.org/10.1109/HPCSim.2016.7568309>.
12. Ding D, Cooper RA, Pasquina PF, Fici-Pasquina L. Sensor technology for smart homes. *Maturitas*. 2011;69(2):131–6. <https://doi.org/10.1016/j.maturitas.2011.03.016>.
13. Dounis AI, Caraiscos C. Advanced control systems engineering for energy and comfort management in a building environment—a review. *Renew Sustain Energy Rev*. 2009;13(6):1246–61.
14. D'Angelo G, Ferretti S. Multi-level simulation of internet of things on smart territories. *Simulat Model Pract Theor*. 2017;73(Supplement C):3–21. <https://doi.org/10.1016/j.simpat.2016.10.008>.
15. Eurostat: Europe in figures - Eurostat yearbook (2017). [http://ec.europa.eu/eurostat/statistics-explained/index.php/Energy\\_price\\_statistics](http://ec.europa.eu/eurostat/statistics-explained/index.php/Energy_price_statistics). Accessed 7 Sep 2017.
16. Gao W, Jiang ZP, Lewis FL, Wang Y Cooperative optimal output regulation of multi-agent systems using adaptive dynamic programming. In: 2017 American Control Conference (ACC), 2017;2674–2679. <https://doi.org/10.23919/ACC.2017.7963356>.
17. Hashim HA, El-Ferik S, Lewis FL. Neuro-adaptive cooperative tracking control with prescribed performance of unknown higher-order nonlinear multi-agent systems. *Int J Control*. 2017. <https://doi.org/10.1080/00207179.2017.1359422>.
18. Iivari A. Building a simulation-in-the-loop sensor data testbed for cloud-enabled pervasive applications. *Proc Comp Sci*. 2015;56(Supplement C):357–62. <https://doi.org/10.1016/j.procs.2015.07.219>.
19. Ioannou PA, Sun J. *Robust Adaptive Control*. New Jersey: Prentice Hall; 1996.
20. Kapoutsis AC, Chatzichristofis S, Doitsidis L, de Sousa JB, Kosmatopoulos EB. Autonomous navigation of teams of unmanned aerial or underwater vehicles for exploration of unknown static & dynamic environments. In: 21st Mediterranean Conference on control and automation, Chania; 2013. p. 1181–1188. <https://doi.org/10.1109/MED.2013.6608870>.
21. Kapoutsis AC, Chatzichristofis SA, Doitsidis L, de Sousa JB, Pinto J, Braga J, Kosmatopoulos EB. Real-time adaptive multi-robot exploration with application to underwater map construction. *Auton Robots*. 2016;40(6):987–1015.
22. Kapoutsis AC, Chatzichristofis SA, Kosmatopoulos EB. Darp: divide areas algorithm for optimal multi-robot coverage path planning. *JINT*. 2017;86(3–4):663–80.
23. Kapoutsis AC, Chatzichristofis SA, Kosmatopoulos EB. A distributed, plug-n-play algorithm for multi-robot applications with a priori non-computable objective functions. *Int J Robot Res*. 2019;38(7):813–32.
24. Khan M, Din S, Jabbar S, Gohar M, Ghayvat H. Context-aware low power intelligent smarthome based on the internet of things. *Comp Electr Eng*. 2016;52(Supplement C):208–22. <https://doi.org/10.1016/j.compeleceng.2016.04.014>.
25. Kollar T, Roy N. Trajectory optimization using reinforcement learning for map exploration. *Int J Robot Res*. 2008;27(2):175–96.
26. Kopanos GM, Georgiadis MC, Pistikopoulos EN. Energy production planning of a network of micro combined heat and power generators. *Appl Energy*. 2013;102:1522–34.
27. Korkas CD, Baldi S, Michailidis I. Occupancy-based demand response and thermal comfort optimization in microgrids with renewable energy sources and energy storage. *Appl Energy*. 2016;163(Supplement C):93–104. <https://doi.org/10.1016/j.apenergy.2015.10.140>.
28. Kosmatopoulos E, Papageorgiou M, Bielefeldt C, Dinopoulou V, Morris R, Mueck J, Richards A. International comparative field evaluation of a traffic-responsive signal control strategy in three cities. *Transport Res Part A*. 2006;40(5):399–413.
29. Kosmatopoulos EB. An adaptive optimization scheme with satisfactory transient performance. *Automatica*. 2009;45(3):716–23. <https://doi.org/10.1016/j.automatica.2008.09.014>.
30. Kosmatopoulos EB, Michailidis I, Korkas CD, Ravanis C. Local4global adaptive optimization and control for system-of-systems. In: 2015 European Control Conference (ECC), 2015; 3536–3541. <https://doi.org/10.1109/ECC.2015.7331081>.
31. Lee D. Energy savings by energy management systems: a review. *Renew Sustain Energy Rev*. 2016;56(Supplement C):760–77. <https://doi.org/10.1016/j.rser.2015.11.067>.
32. Leontaritis IJ, Billings SA. Input-output parametric models for non-linear systems part i: deterministic non-linear systems. *Int J Control*. 1985;41(2):303–28. <https://doi.org/10.1080/0020718508961129>.
33. Ma Z. Model-based optimization of ground source heat pump systems. *Energy Proc*. 2017;111(Supplement C):12–20. <https://doi.org/10.1016/j.egypro.2017.03.003>.
34. Manolis D, Papamichail I, Kosmatopoulos EB. Automated tuning of its management and control systems: results from real-life experiments. *Transport Res Part C*. 2016;66(Supplement C):119–35. <https://doi.org/10.1016/j.trc.2015.08.020>.
35. Michailidis I, Baldi S, Kosmatopoulos EB, Ioannou PA. Adaptive optimal control for large-scale nonlinear systems. *IEEE Trans Autom Control*. 2017;62(11):5567–77. <https://doi.org/10.1109/TAC.2017.2684458>.
36. Michailidis IT, Baldi S, Pichler MF, Kosmatopoulos EB. Proactive control for solar energy exploitation: a german high-inertia building case study. *Appl Energy*. 2015;155(Supplement C):409–20. <https://doi.org/10.1016/j.apenergy.2015.06.033>.
37. Michailidis IT, Korkas C, Kosmatopoulos EB, Nassie E. Automated control calibration exploiting exogenous environment energy: an israeli commercial building case study. *Energy Build*. 2016;128(Supplement C):473–83. <https://doi.org/10.1016/j.enbuild.2016.06.035>.
38. Michailidis IT, Michailidis P, Rizo A, Korkas C, Kosmatopoulos EB. Automatically fine-tuned speed control system for fuel and travel-time efficiency: A microscopic simulation case study. In: 2017 25th Mediterranean Conference on Control and Automation (MED), pp. 915–920 (2017). <https://doi.org/10.1109/MED.2017.7984236>.

39. Michailidis IT, Schild T, Sangi R, Michailidis P, Korkas C, Fütterer J, Müller D, Kosmatopoulos EB. Energy-efficient hvac management using cooperative, self-trained, control agents: a real-life German building case study. *Appl Energy*. 2018;211(Supplement C):113–25. <https://doi.org/10.1016/j.apenergy.2017.11.046>.
40. Oldewurtel F, Parisio A, Jones CN, Gyalistras D, Gwerder M, Stauch V, Lehmann B, Morari M. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy Build*. 2012;45:15–27.
41. Palacios-Gasós JM, Montijano E, Sagüés C, Llorente S. Distributed coverage estimation and control for multirobot persistent tasks. *IEEE Trans Robot*. 2016;32(6):1444–600.
42. Palensky P, Dietrich D. Demand side management: demand response, intelligent energy systems, and smart loads. *IEEE Trans Industr Inform*. 2011;7(3):381–8.
43. Paull L, Huang G, Seto M, Leonard JJ. Communication-constrained multi-AUV cooperative SLAM. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA; 2015. pp. 509–516. <https://doi.org/10.1109/ICRA.2015.7139227>.
44. Pritoni M, Meier AK, Aragon C, Perry D. Energy efficiency and the misuse of programmable thermostats: the effectiveness of crowdsourcing for understanding household behavior. *Energy Res Soc Sci*. 2015;8(Supplement C):190–7. <https://doi.org/10.1016/j.erss.2015.06.002>.
45. Rocha P, Siddiqui A. Improving energy efficiency via smart building energy management systems: a comparison with policy measures. *Energy Build*. 2015;88(Supplement C):203–13. <https://doi.org/10.1016/j.enbuild.2014.11.077>.
46. Salavasidis G, Kapoutsis AC, Chatzichristofis SA, Michailidis P, Kosmatopoulos EB. Autonomous trajectory design system for mapping of unknown sea-floors using a team of AVUs. In: 2018 European Control Conference (ECC), Limassol; 2018. p. 1080–1087. <https://doi.org/10.23919/ECC.2018.8550174>.
47. Sangi R, Schild T, Daum M, Fütterer J, Streblow R, Müller D, Michailidis I, Kosmatopoulos E. Simulation-based implementation and evaluation of a system of systems optimization algorithm in a building control system. In: 2016 24th Mediterranean Conference on Control and Automation (MED); 2016. p. 1296–1301. <https://doi.org/10.1109/MED.2016.7535890>.
48. Solbach M, Bonin-Font F, Burguera A, Oliver G, Paulus D. Robust world-centric stereo EKF localization with active loop closing for AUV s. *Pattern Recognit Image Anal*. 2016;26(1):205–15.
49. Song Y, Han B, Zhang X, Yang D. Modeling and simulation of smart home scenarios based on internet of things. In: 2012 3rd International Conference on Network Infrastructure and Digital Content, Beijing; 2012. p. 596–600. <https://doi.org/10.1109/ICNIDC.2012.6418824>.
50. Su Z, Khorasani K. A neural-network-based controller for a single-link flexible manipulator using the inverse dynamics approach. *IEEE Trans Ind Electron*. 2001;48(6):1074–86. <https://doi.org/10.1109/41.969386>.
51. Trčka M, Hensen JL, Wetter M. Co-simulation of innovative integrated hvac systems in buildings. *J Build Perform Simul*. 2009;2(3):209–30.
52. Wei J, Cao D, Liu L. Global mode method for dynamic modeling of a flexible-link flexible-joint manipulator with tip mass. *Appl Math Model*. 2017;48(Supplement C):787–805. <https://doi.org/10.1016/j.apm.2017.02.025>.
53. Wikipedia: Fuel economy in automobiles (2017). [http://en.wikipedia.org/wiki/Fuel\\_economy\\_in\\_automobiles](http://en.wikipedia.org/wiki/Fuel_economy_in_automobiles). Accessed 20 Nov 2017.
54. Wilson C, Hargreaves T. Benefits and risks of smart home technologies. *Energy Policy*. 2017;103(Supplement C):72–83. <https://doi.org/10.1016/j.enpol.2016.12.047>.
55. Yazdizadeh A, Khorasani K. Identification of a two-link flexible manipulator using adaptive time delay neural networks. *IEEE Trans Syst Man Cybern B Cybern*. 2000;30(Supplement C):165–72. <https://doi.org/10.1109/3477.826956>.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.