Resource Allocation Using Virtual Objects in the Internet of Things: a QoI Oriented Consensus Algorithm

Andrea Carta, Virginia Pilloni, Luigi Atzori
DIEE, University of Cagliari, Italy
{*virginia.pilloni,l.atzori*}@*diee.unica.it*

***Abstract*—The pervasive spread of smart objects is encouraging the development of smart environments, such as Smart Cities and Smart Homes. In the Internet of Things (IoT) vision, even the most common and simple object is expected to acquire information from the surrounding ambient and to cooperate with other objects to achieve a common goal, fulfilling the expected quality requirements. In such a heterogeneous and complex scenario, optimal allocation of resources (e.g. available energy, computing speed, storage capacity) is paramount in order not to overload some objects.**

In this paper, a framework that makes use of Virtual Objects (VOs) to manage the objects of an IoT system is proposed. Using VOs, the resources, functionalities and capabilities available on the objects are virtualised and exposed to the other objects to cooperate for executing the deployed applications. A distributed algorithm for resources allocation based on consensus has been developed to: distribute the workload among the objects that can cooperate to the same task; ensure Quality of Information (QoI) requirements. Simulation results show that, compared to a static frequency allocation, the algorithm enhances the performance of the system with an average improvement of 62% in network lifetime, and confirm the compliance to QoI requirements.

***Index Terms*—Resource allocation; Internet of Things; Virtual Objects**

I. Introduction

The Internet of Things (IoT) [1] is a new technological paradigm that refers to an evolution of the Internet network, characterized by huge amounts of objects that dynamically cooperate and make their resources available, with the aim of achieving a common objective. The IoT is defined as a revolution for communications and people's lifestyle [2]. Although the IoT has not yet spread much despite its potentiality, several studies prove how the objects will assume a predominant role in the future Internet network. The IoT will offer amazing improvements in the collection, processing and distribution of knowledge and information, thanks to the pervasive spread of smart objects. The exploitation of information and data collected by the objects is going to improve users' knowledge, their relationship with nature and their lifestyle. Technological and social progresses have led to a transition from Web 2.0 [3] to the semantic, ubiquitous and pervasive Web 3.0 [4]. Now this transition is involving the IoT and it is going to encourage its evolution. Not only will the IoT technology enable users to communicate with objects: the objects themselves, including the most common and simple, will have the ability to communicate with each other and gain the intelligence to provide information on their status or acquire data from other objects.

This work has been partially funded by the project Artemis JU Demanes, Design, Monitoring and Operation of Adaptive Networked Embedded Systems, grant agreement no. 295372.

The first step of this work has been studying the state of the art and the architectural features of the IoT. In particular, we focused on the concept of *Virtual Object* (VO) [5], the virtual counterpart of one or more Real World Objects (RWO), which virtualises their resources, capabilities, functionalities and data collected. Starting from the iCore architecture [6], we have focused on the mechanisms for identifying and selecting objects, capable of performing a specific task, leveraging the capabilities of VOs. The logic for the creation of VOs and the parameters of interest required for the identification and selection of candidates capable of performing the required tasks have been analysed.

Taking advantage of the features offered by the VOs, we propose an optimisation process that allows to distribute adequately the workload generated by the execution of IoT application tasks, among the objects that can carry them out. The objective of the consensus-based task allocation algorithm presented in this paper is twofold: considering Quality of Information (QoI) constraints in the process of allocating tasks to the IoT objects, so that the fulfillment of application requirements is ensured; optimising the use of resources of the underlying IoT system.

In Section II some previous studies on the concept of virtualisation in IoT are presented. Section III provides a functional analysis of the reference architecture and the problems related to the allocation of services focused on QoI achievement. Section IV tackles the description of the resource allocation model developed. The implemented solutions have been tested through simulations on two application scenarios specifically modeled. Simulations and experimental results will be presented in Section V. Finally, conclusions and future works are presented in Section VI.

II. PRELIMIARIES

The IoT consists of intelligent objects connected to the Internet, which cooperate to support the execution of complex applications and services [1]. These objects are equipped with sensors and actuators, which provide context-awareness and enable them to gather, process and exchange data, in order to react to external stimuli. The need to represent, store, discover, search, exchange and manage the huge amount of information generated by the objects, motivated the development of semantic technologies [7].

Although the IoT lays its basis on the use of simple technologies such as RFID tags [8], the use of Cloud Computing has greatly enhanced its capabilities. With Cloud Computing [9], even devices characterised by limited computational capacities are able to execute intricate computations required for effective performance of assigned applications.

With a thorough comparative analysis between Cyber Physical Cloud (CPC), Cloud of Sensors (CoS) and IoT, the authors in [10] show how these three technologies exploit Cloud Computing potentialities, and how much they are related in the objective of linking digital and real worlds. They base on the concept of object *virtualisation*, according to which the physical components of an object can be abstracted and made available as virtual resources. Virtualisation allows the higher layers of the IoT architecture to: i) interface with devices; ii) provide device with the required commands, adapted to their native communication protocol; iii) monitor their activities and connection capabilities. A VO is the virtual counterpart of one or more real objects, and as such it inherits all their functionalities, characteristics and acquired information [11]. Since virtualisation is such a fundamental component of the IoT, many well-known middlewares, such as SENSEI [12], IoT-A [13] and iCore [5], are based on it.

Combining virtualisation with context-awareness, the IoT system is able to achieve a clear knowledge of the resources and functionalities made available by its objects. Since the IoT is characterised by scarse resources, they need to be managed and orchestrated in an efficient way. The process of detecting the most appropriate IoT objects' resources that are able to fulfill the applications' requirements, needs to be accomplished in a distributed and automatic way, in order to cope with the dynamic nature of the IoT.

Resource allocation has been extensively studied in Wireless Sensor Networks (WSNs), particularly with reference to network lifetime. In [14] a distributed task allocation that focuses on the reduction of the overall energy consumption and task execution time into a heterogeneous WSN is proposed, with attention to nodes' residual energy. A similar approach is studied in [15], where a distributed algorithm based on particle swarm optimization is proposed. In [16], the issue of energy saving in Wireless Cooperative Networks is addressed. The algorithm proposed in this paper aims to find a trade-off between efficiency and fairness, by using a game-theoretic approach. Since the main criticality of wireless networks is their lifetime, all these algorithms mainly focus on maximizing

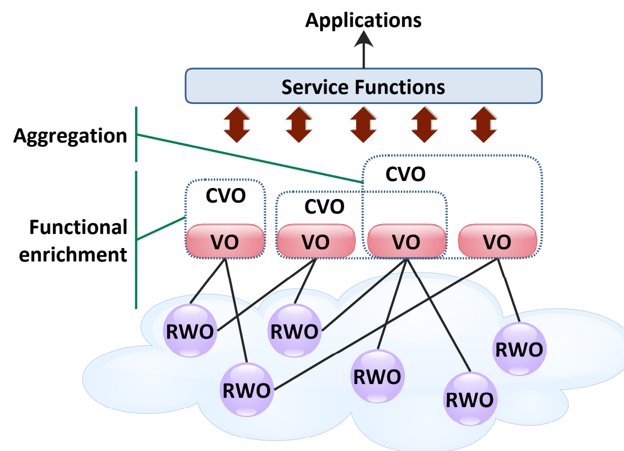


Fig. 1. iCore architecture [6]

this resource. Nevertheless, IoT nodes have more heterogeneous characteristics and capabilities, and therefore even other resources, such as residual memory and processing capacity, are considered scarce.

As far as IoT networks are concerned, resource allocation is an open issue. Most of the existing studies on resource allocation for IoT are focused on IoT service provisioning, such as in [17] and [18]. In these studies, the aim is to allocate the resources that enable service execution. However, they do not focus on finding the best configuration that corresponds to an optimal resource allocation. None of the works found in the literature tries to find the optimal resource allocation associated to the lowest impact of the application assigned to the network. A first attempt in resource optimisation was made in [19], where QoI was not taken into account.

III. REFERENCE ARCHITECTURE

The reference architecture proposed in this paper is based on the iCore framework [6]. As shown in Figure 1, it is subdivided into three functional levels: i) VO Level where VOs perform object virtualisation; ii) CVO Level that has the aim of fulfilling the application requirements by VO composition and functional enrichment; iii) Service Level that processes user requests to decompose services into applications.

The entire architecture is based on the concept of VO, which is a semantically and functionally enriched representation of one or more RWO [20]. Thanks to its self-management and self-awareness capabilities, it offers cognitive and intelligent services and enables objects' (services) composition.

The result of objects' composition is a Composite Virtual Object (CVO), which is a mash-up of VOs and other CVOs. CVOs are autonomic entities able to render services according to user perspectives and application requirements.

The highest layer of the architecture is the Service layer, which receives the user requests and dynamically maps them to the CVOs and VOs required to perform them.

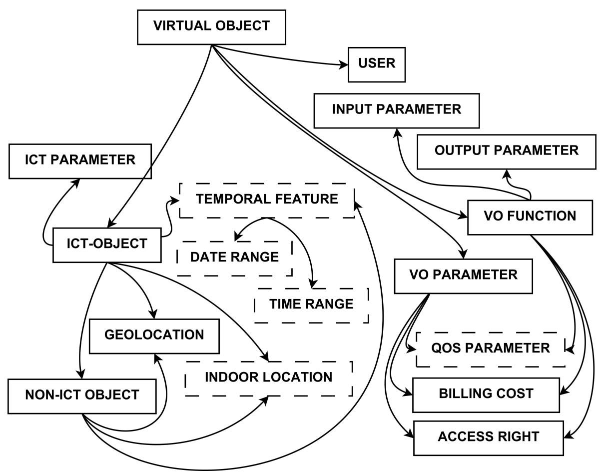


Fig. 2. VO Information Model used. Solid border boxes correspond to elements included in the iCore VO Information Model. Dashed border boxes are new elements introduced by the proposed architecture

A. The VO Creation Process

The VO creation process is divided into: VO template creation and VO instantiation [20]. The *VO template*, or VO description, enables the virtualisation of the object resources, characteristics and functionalities. It is created so that new VOs can be easily described by reusing the same template used by similar VOs. Furthermore, the VO template is envisioned to ease the automated VO discovery, search and selection functionalities. As soon as an object joins the IoT, a corresponding VO is instantiated using an appropriate VO template, and it is linked to it. As depicted in Figure 2, the iCore information model, according to which templates are created, gathers the informations related to ICT objects (e.g. sensors, smartphones, RFID tags), non-ICT objects (e.g. tables, rooms), and VO functional characteristics.

In order to account for object mobility and QoI specification, we propose a modified version of the iCore information model. This enhancement is meant to improve the VO search, discovery and selection processes that enable the assignment of tasks, in which services are divided, to the most appropriate VOs, with a QoI-oriented perspective. Figure 2 shows the new elements in dashed border boxes:

- Indoor Location: evaluated using local patterns, it is used to improve the object indoor research, where geo-localisation is not sufficient.
- Temporal Feature: it refers to the last activity of an object, or the last collection of data. It is needed to know whether a resource is available, or how old its updates are.
- QoI Parameter: it encompasses all the QoI parameters related to the object, such as data accuracy and timeliness.

B. The Reference Scenario

The distributed nature of the IoT leads to the conclusion that some functionalities can be distributed across the RWOs controlled by the VO layer. The objective of the architecture developed in this paper is to enable the distribution of functionalities that are required by the Service layer, to those VOs that have the required capabilities and/or characteristics. In particular, we focus on the allocation of application tasks to RWOs that can cooperate to perform them, in order to ensure an optimal exploitation of the available resources (i.e. lifetime, storage capacity and computational speed), and the compliance to QoI requirements.

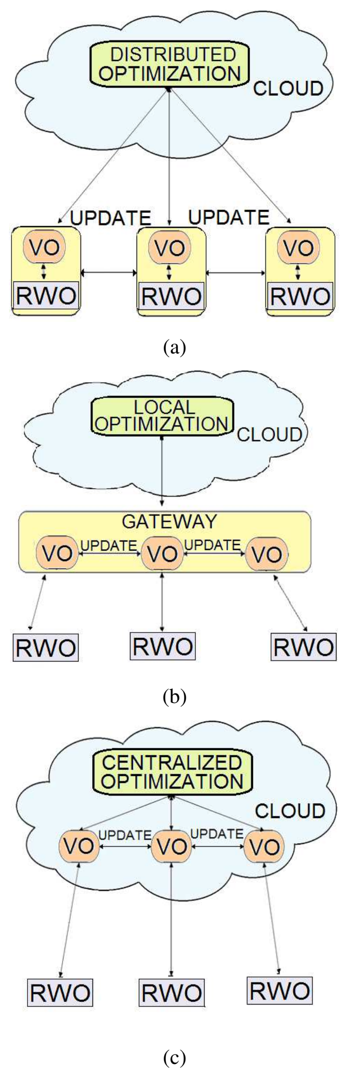


Fig. 3. Location of the proposed algorithm into three typical IoT scenarios with reference to objects' resource allocation

We identified three possible scenarios, depicted in Figure 3. In the first case, the assigned task requires resources that are located in the same area. RWOs are intelligent objects characterised by sufficient resources, which communicate using short-range technologies. In this case, the optimal task allocation process itself can be implemented on the appropriate RWOs, so that resource management is located as closest as possible to where they are used (see Figure 3(a)).

In the second case (Figure 3(b)), RWOs are still located in the same area, but resources provided by them are scarce. In this case, the resource allocation process cannot be performed on RWOs, but it can still take place on the gateway, which can act as a VO including all the involved RWOs.

The third case (Figure 3(c)) refers to RWOs located in different areas far from each other, which need to connect to the Cloud to communicate. It is obvious that a distributed solution would not be suitable to this case, and therefore a centralised one is preferable.

In the following Sections we focus on the first and second scenarios.

IV. The Resource Allocation Model

Herein, we present the proposed QoI-oriented resource allocation algorithm, which provides mechanisms that enable applications to exploit the best set of available objects resources, which are exposed through their related VOs. The objective of the proposed algorithm is twofold:

- *Resource optimisation.* It optimises the use of the available resources, assigning to the VOs the most appropriate frequency to execute the task, so that they cooperate to perform it according to their capabilities.
- *QoI accomplishment.* In order for QoI constraints to be applied, the algorithm uses QoI parameters measured on RWOs and selects the VOs that are able to achieve at least the minimum QoI required. Furthermore, in order to cope with the accuracy required by the application, the task execution frequency is assigned to VOs taking into account the minimum execution frequency required by the Service layer.

A. Virtual Object Selection

The optimisation mechanism proposed relies on the CVO layer, where the requests coming from the Service layer, and converted into specific parameters, are processed. The CVO layer first detects which VO Templates are needed to compose the CVO. Then, a query is sent to the VO layer, which starts a semantic search to detect all the available VOs that comply with functionality, location and reference time requirements. As a result, these VOs, that are able and available to take part in the execution of the application task, are activated by the VO layer. The resource allocation optimisation algorithm is then run on this group of VOs.

B. Consensus-Based Resource Allocation Optimisation

The resource optimisation strategy proposed in this paper relies on a consensus-based algorithm where VOs decide the amount of resources to allocate to a task, according to the constraints request by the higher layers. The consensus algorithm presented in the following focuses on lifetime optimisation, but it can be easily extended to focus on other objects' resources than residual energy, such as storage capacity or processing speed.

As defined in [19], the lifetime of a node is the time until it depletes its battery. The lifetime of the node associated to VO i at time t is expressed as

$$\tau_i(t) = \frac{E_i^{res}(t)}{\sum_k P_{ik}^c(t) + P_i^o(t)} = \frac{E_i^{res}(t)}{\sum_k E_{ik}^c \cdot f_{ik}(t) + P_i^o(t)} \quad (1)$$

where $E_i^{res}(t)$ is its residual energy, $P_{ik}^c(t)$ and E_{ik}^c are the power and energy consumed by the RWO associated to VO i to perform task k, $f_{ik}(t)$ is the frequency at which VO i performs task k, and $P_i^o(t)$ is the offset power consumed by the other activities of the node (e.g. tasks that are assigned directly by the user).

The network lifetime is defined as the time in which at least one node has exhausted its energy reserve from the battery: in fact, when this condition is reached, the network topology is disrupted [21]. We base on the assumption that optimising the network lifetime is equivalent to adjusting the VOs' power consumption so that their associated nodes reach the same lifetime [19]. This means that, taken two VOs i and j that received an activation request for task k, at time t_c when the algorithm converges $\tau_i(t_c) = \tau_j(t_c)$. Therefore

$$\sum_k \alpha_{ik}(t_c) \cdot f_{ik}(t_c) + P_i^o(t_c) = \sum_k \alpha_{jk}(t_c) \cdot f_{jk}(t_c) + P_j^o(t_c) \quad (2)$$

where $\alpha_{ik}(t) = E_{ik}^c / E_i^{res}(t)$. Defining the total amount of power consumption contributions with the exception of task k as $\delta_{ik}(t) = \sum_{l \neq k} \alpha_{il}(t) \cdot f_{il}(t) + P_i^o(t)$, from Equation 2 follows that

$$f_{jk}(t_c) = \frac{\alpha_{ik}(t_c)}{\alpha_{jk}(t_c)} \cdot f_{ik}(t_c) + \frac{\delta_{ik}(t_c) - \delta_{jk}(t_c)}{\alpha_{jk}(t_c)} \quad (3)$$

According to accuracy constraints provided by the higher layers, the collaborative completion of a task is required to be performed at a reference frequency $F_k^{ref} = \sum_j f_{jk}(t_c)$. Using Equation 3 in this identity, after some simple computations and multiplying and dividing by the number N_k of VOs involved in task k, it is straightforward to obtain

$$\alpha_{ik}(t_c) \cdot f_{ik}(t_c) = \frac{\bar{\varphi}_k}{\bar{\beta}_k(t_c)} + \frac{\bar{\gamma}_k(t_c)}{\bar{\beta}_k(t_c)} - \delta_{ik}(t_c) \quad (4)$$

with $\bar{\varphi}_k = \frac{F_k^{ref}}{N_k}$, $\bar{\beta}_k(t_c) = \frac{1}{N_k} \cdot \sum_j \frac{1}{\alpha_{jk}(t_c)}$ and $\bar{\gamma}_k(t_c) = \frac{1}{N_k} \cdot \sum_j \frac{\delta_{jk}(t_c)}{\alpha_{jk}(t_c)}$. The $\bar{\varphi}_k$ value can be forwarded directly by the VO layer. It is easy to notice that $\bar{\beta}_k(t_c)$ and $\bar{\gamma}_k(t_c)$ represent mean values evaluated over all the VOs that are able to perform task k. Therefore, their value can be estimated in a distributed way using an average consensus algorithm [22].

We suppose to have a system that is not subject to perturbations and where nodes stay connected until the algorithm is converged. Nevertheless, the update functions can be easily adjusted according to [22], in order to be robust against perturbations and topology changes. Since variations of α and δ are negligible over the time needed by the algorithm to converge, in the following we consider them constant and omit their dependence from time. Nevertheless, if substantial variations of them are experienced, the algorithm needs to start again.

C. Lifetime Optimisation Algorithm

As soon as VO i receives an activation request for task k from the VO layer, it verifies if it is able to satisfy the minimum level of QoI required by the higher levels. If it is not, it sets f_{ik} to 0 and informs the VO layer about it, so that it can update the $\bar{\varphi}_k$ value. Otherwise, it initialises its local values $\beta_{ik} = 1/\alpha_{ik}$ and $\gamma_{ik} = \delta_{ik}/\alpha_{ik}$ and starts the consensus with its neighbours. Whenever VO i receives an update from one of its neighbours j, it computes the following updates:

$$\beta_{ik}^{+} = \beta_{ik} - \lambda_1 \sum_{j} (\beta_{ik} - \beta_{jk}) \tag{5a}$$

$$\gamma_{ik}^{+} = \gamma_{ik} - \lambda_2 \sum_{j} (\gamma_{ik} - \gamma_{jk}) \tag{5b}$$

$$\tau_i^{+} = \frac{\beta_{ik}^{+}}{\varphi_{ik}^{+} + \gamma_{ik}^{+}} \qquad f_{ik}^{+} = \frac{1}{\alpha_{ik}} \cdot \left(\frac{1}{\tau_i^{+}} - \delta_{ik} \right) \tag{5c}$$

where $\lambda_1 > 0$ and $\lambda_2 > 0$ are tuning parameters that affect the convergence time and steady-state accuracy [22]. If $f_{ik}^{+} > 0$ and if its value has changed after the update, the VO sends the updated value of β_{ik}^{+} and γ_{ik}^{+} to its neighbours. It may happen that $f_{ik}^{+} \leq 0$. In this case, the VO cannot participate into executing task k. Therefore, it sets f_{ik} to 0 and informs the VO layer, which updates its $\bar{\varphi}_k$ value. The algorithm can be considered converged when f_{ik} does not change consistently after the updates.

V. Performance Analysis

In order to demonstrate how the presented VO management framework improves the performance of the entire underlying infrastructure, a healthcare application scenario was identified and modeled. This scenario comprises three smart environments: a Smart Home, a doctor's office and a drugstore. In the modeled scenario several objects collaborate in the execution of tasks for the deployed applications, e.g. vital signs and ambient sensors, smart cameras, and smartphones. To demonstrate the performance of the proposed algorithm, a wide series of tests was run using the Simulink simulation tool. Each test takes into consideration the activation of various tasks at the same time, and different configurations of devices that can simultaneously perform the required application. The following parameters were set for each task: the reference frequency for the execution, the values of the calibration constants of the algorithm, and the minimum QoI parameters to achieve. For all the modeled configurations different reference frequencies have been used and, from time to time, VOs corresponding to different resources are enabled to participate in the optimisation process.

Figure 4 shows the algorithm's behaviour in an illustrative example. The plot refers to a test in which four devices participated to the optimisation process. The frequency allocation is performed by the algorithm when each task is activated. The peaks correspond to the points where a new task is activated. At each task's activation the execution frequency is equally divided among the devices that can perform the task, initializing the algorithm that starts the process as shown by peaks corresponding to this point. At each step the algorithm decreases the devices' lifetime difference to reach converge. In this case only three devices are able to perform all the three tasks (solid, dashed, and dash-dot lines). Therefore, their

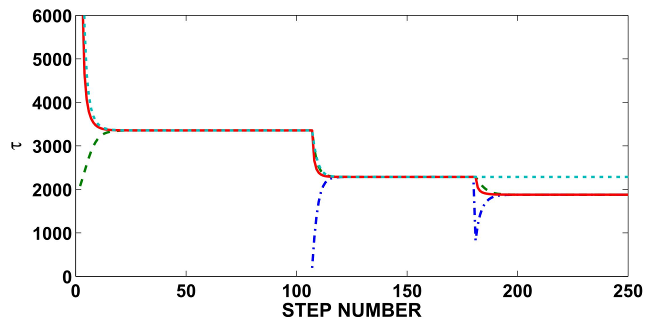


Fig. 4. Example plot of lifetime's convergence: each line represents the lifetime for a different member of a group of four objects implementing the proposed algorithm

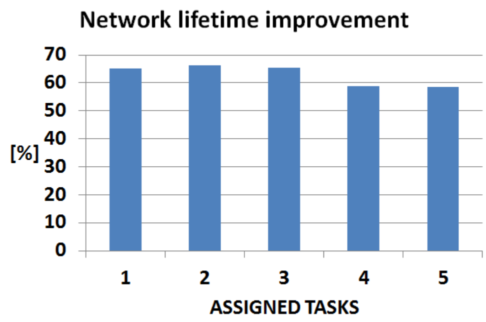
lifetime will converge to a value that's lower than that of the fourth device (dotted line), which is not able to perform the third task. Its lifetime will be higher than that of the other devices because they will take charge of a higher workload.

The data analysis shows that the optimisation process brings in all cases to an improvement of the lifetime of the network. The tasks are assigned in an optimised manner, so that the execution is heavier on devices with higher energy resources, preserving the energy expenditure of the others. To demonstrate the algorithm's validity and the proper functioning of the optimisation process, we compared the network lifetime results to the network lifetime calculated using the task's reference frequency, divided by the number of devices available to run it. In this way, without any optimisation, the frequency is distributed equitably among devices regardless of their energy resources. This value of non-optimised lifetime was then compared with that resulting from the optimisation process, in order to evaluate the obtained average percentage of lifetime improvement. Figure 5 shows the average values of percentage improvement in network lifetime, depending on the number of assigned tasks (Figure 5(a)) and the number of devices involved in the optimisation process (Figure 5(b)). In all the considered cases the obtained values from the tests have led to a significant improvement of the lifetime, thanks to the optimisation process. The graph shows how the increase in the number of assigned tasks does not impact on the improvement of the average lifetime. However the increase in the number of involved devices, leads to an increase in the average improvement of lifetime.

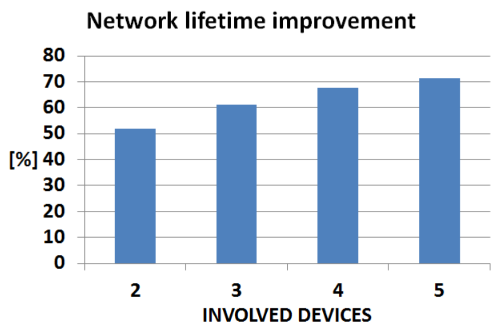
TABLE I
Average steps of algorithm's convergence

	1	**2**	**3**	**4**	**5**
Assigned tasks	43	92	156	159	181
Involved devices	–	88	112	166	171

We also analysed the time performance of the process. Table I shows the algorithm's convergence time in function of the average number of assigned tasks assigned and the number of devices involved in the optimisation process. As it can be seen from results, the algorithm's convergence time



(a)



(b)

Fig. 5. Average values of the percentage improvements in network lifetime

is influenced both by the number of assigned tasks and the number of devices that participate in the optimisation process. From them depends the number of steps that the algorithm has to perform to achieve the lifetime convergence. The tests have shown that for a higher number of assigned tasks or involved devices the convergence time of the algorithm increases.

VI. Conclusions

The analysis of the issues related to the identification and selection of resources through the use of VOs, has allowed to implement a process of optimisation of the allocation of tasks, that improves the QoI offered by the object resources in an IoT scenario. The consensus-based algorithm on which the process is based uses the parameters measured on the physical resources and subdivides the frequency of tasks' execution, required by the application, among the VOs, so as to provide the best possible QoI. The modeled scenario ensures the validation of the proposed framework and the improvement of its performance. In all the tests performed the simulation results have demonstrated an average improvement of 62% in network lifetime.

The optimisation process implemented has the goal to select VO instances that would guarantee the minimum QoI level and improve the lifetime of objects. Future developments will focus primarily on the implementation of the proposed framework on real devices, in order to assess directly its behavior in the case of real transmissions between objects. Another aspect upon which future development will be focused will be the study of a multi-objective algorithm that will also take into account other resources, such as storage capacity and processing speed.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
[2] D. Evans, "The internet of things," *How the Next Evolution of the Internet is Changing Everything, Whitepaper, Cisco Internet Business Solutions Group (IBSG)*, 2011.
[3] T. O'reilly, "What is web 2.0: Design patterns and business models for the next generation of software," *Communications & strategies*, no. 1, p. 17, 2007.
[4] J. Hendler, "Web 3.0 emerging," *Computer*, vol. 42, no. 1, pp. 111–113, 2009.
[5] iCore Project. (2015) iCore: Empowering iot through cognitive tecnologies.
[6] ——. (2015) iCore: D2.3 architecture reference model.
[7] P. Barnaghi, W. Wang, C. Henson, and K. Taylor, "Semantics for the internet of things: early progress and back to the future," vol. 8, no. 1. IGI Global, 2012, pp. 1–21.
[8] C. M. Roberts, "Radio frequency identification (rfid)," vol. 25, no. 1. Elsevier, 2006, pp. 18–26.
[9] P. Parwekar, "From internet of things towards cloud of things," in *Computer and Communication Technology (ICCCT), 2011 2nd International Conference on*. IEEE, 2011, pp. 329–333.
[10] V. K. Sehgal, A. Patrick, and L. Rajpoot, "A comparative study of cyber physical cloud, cloud of sensors and internet of things: Their ideology, similarities and differences," in *Advance Computing Conference (IACC), 2014 IEEE International*. IEEE, 2014, pp. 708–716.
[11] J. Pacual-Espada, O. Sanjuan-Martinez, B. C. P. G-Bustelo, and J. M. C. Lovelle, "Virtual objects on the internet of things," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 1, no. 4, 2011.
[12] SENSEI-ICT. (2008) Integrating the physical with the digital world of the network of the future.
[13] IOT-A. (2015) Internet of things architecture.
[14] V. Pilloni, P. Navaratnam, S. Vural, L. Atzori, and R. Tafazolli, "Tan: a distributed algorithm for dynamic task assignment in wsns," *Sensors Journal, IEEE*, vol. 14, no. 4, pp. 1266–1279, 2014.
[15] Y. Shen and H. Ju, "Energy-efficient task assignment based on entropy theory and particle swarm optimization algorithm for wireless sensor networks," in *Green Computing and Communications (GreenCom), 2011 IEEE/ACM International Conference on*. IEEE, 2011, pp. 120 –123.
[16] L. Militano and G. Araniti, "Introducing fairness-efficiency trade-off for energy savings in wireless cooperative networks," *Wireless Personal Communications*, vol. 76, no. 1, pp. 3–21, 2014.
[17] D. Guinard, V. Trifa, F. Mattern, and E. Wilde, "From the internet of things to the web of things: Resource-oriented architecture and best practices," in *Architecting the Internet of Things*. Springer, 2011, pp. 97–129.
[18] B. Silverajan and J. Harju, "Developing network software and communications protocols towards the internet of things," in *Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middlewaRE*. ACM, 2009, p. 9.
[19] G. Colistra, V. Pilloni, and L. Atzori, "The problem of task allocation in the internet of things and the consensus-based approach," *Computer Networks*, vol. 73, pp. 98–111, 2014.
[20] iCore Project. (2015) iCore: D3.2 real object awareness and association.
[21] V. Pilloni, M. Franceschelli, L. Atzori, and A. Giua, "Deployment of applications in wireless sensor networks: A gossip-based lifetime maximization approach," *IEEE Transactions on Control Systems Technology*, 2016.
[22] A. Pilloni, M. Franceschelli, A. Pisano, and E. Usai, "Recent advances in sliding-mode based consensus strategies," in *Variable Structure Systems (VSS), 2014 13th International Workshop on*. IEEE, 2014, pp. 1–6.