Energy Efficient Service Embedding in IoT Networks

Haider Qays Al-Shammari ,Ahmed Lawey ,Taisir El-Gorashi ,and Jaafar M. H. Elmirghani
School of Electronic and Electrical Engineering ,University of Leeds ,United Kingdom

Abstract— **The Internet of Things (IoT) is anticipated to participate in performing diverse and complex tasks in the near future. IoT objects capable of handling multiple sensing and actuating functions will be the corner stone of future IoT systems in smart cities.**

In this paper, we present an energy efficient service embedding framework in IoT network by using mixed integer linear programming (MILP). This framework addresses a set of metrics such as scalability, flexible resource allocation, cost reduction, and efficient use of resources. We consider the event-driven paradigm of Service Oriented Architecture (SOA) in our framework in order to provide service abstraction of basic services which can be composed into complex services and exploited by the upper application layer. The results show that our optimized network can save an average of 27% and 36% of the processing and network power consumption, respectively, compared to an energy unaware service embedding scheme.

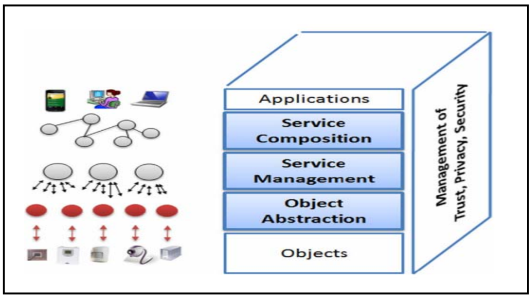
Keywords: IoT, SOA, Energy Efficiency, MILP, Smart city.

I. INTRODUCTION

In the near future, the majority of physical objects will be incorporated into sensors and actuators that have the ability to communicate, forming the basis for the Internet of Things (IoT) [1]. IoT attracted many global establishments to research and invest in this area and its promising applications in healthcare, transportation, and other smart city applications [2]. However, these promises of IoT come with great challenges. One of these challenges is the energy efficiency due to its impact on the environment and expenditure [3], [4]. Although each IoT device consumes low power, it is predicted that the number of IoT nodes will reach approximately 50 billion by the year 2020 [5], a massive number that can cause a high aggregate power consumption. Smart city applications [6] are expected to use a large number of IoT devices across cities, therefore, minimizing the energy consumed by such applications can play a significant role in reducing IoT total energy consumption.

This paper investigates the solutions that IoT nodes can introduce to enhance real world applications in the smart city. The smart city consists of a system for monitoring and controlling the applications of interest in the public areas of the city. The monitoring and controlling system consists of different types of sensors and actuators such as motion detectors, sound detectors, light detectors, smoke detectors, alarms, gates controllers among others. These sensors and actuators are connected by means of wireless nodes, and are placed on the city streets and buildings forming typically a mesh topology. In the smart city, there are distinct applications employing the same resources in the monitoring and controlling system. For example, security applications and energy saving applications employ motion detectors, RFID, and light detectors for data monitoring simultaneously. Furthermore, the Smart City concept will serve various applications, industries, service providers, or administrations, and will be applied in a mutual pattern for these sectors efficiently [7]. An essential phase towards the realization of the Smart City concept involves the improvement of the communication infrastructure. The IoT infrastructure provides the capability of collecting data from a massive assortment of distinct devices uniformly and seamlessly. The decentralized and heterogeneous properties of IoT devices that are capable of providing multiple functions require an efficient architecture that hides such heterogeneity from higher level applications and provides interoperability for information exchange with other IoT devices [8]. SOA is considered as a viable middleware between user's applications and the IoT physical layer and can support the interoperability between those heterogeneous IoT devices [9]. SOA enables the abstraction of IoT devices' functions that can then be translated into basic services which in turn can be composed into complex services and exploited by the upper application layer. Fig.1 depicts the SOA middleware for IoT which is composed of three sub-layers [1], [2], [10]: (i) Objects abstraction layer that enables IoT devices to provide their functions to the upper layers, (ii) Service management layer to enable dynamic object discovery, status monitoring and mapping of available services to the IoT devices' abstracted functions, (iii) Service composition layer where complex services; referred to as business process (BP) workflow; are created from basic services provided by the service management layer.

Fig. 1: SOA-based architecture for the IoT middleware [1].



Employing SOA, devices can be reused or upgraded individually; leading to several SOA advantages such as extensibility, scalability, and modularity plus the aforementioned interoperability among IoT devices [11]. Such features are essential for large-scale implementations such as smart city applications. The Authors in [6] summarized the main aspects of a 2020smart city vision. Due to the advantages of SOA, the authors in [12] presented an energy-centered and QoS-aware services selection algorithm (EQSA) for IoT services composition. They proposed a framework that selects the services by using a lexicographic optimization strategy and QoS constraints relaxation technique. The authors in [13] surveyed the recent development of SOA models for IoT and reviewed their fundamental technologies. The authors in [14] proposed a reference architecture for the smart city based on SOA concepts by integrating IoT, Cloud and Edge technologies with existing city infrastructure. The authors in [15] surveyed the recent development of energy-efficient solutions for wireless sensors networks and reviewed some existing topologies that allow trade-offs between multiple requirements to be achieved for efficient and sustainable sensor networks. The authors in [16] presented a QoS message scheduling algorithm in IoT network based SOA, which is more targeted towards service provisioning with the idea of service differentiation and classification into high priority and best effort messages. The aim of this paper is to evaluate the energy efficiency of embedding high-level application requests in the lower level IoT nodes in a smart city setting. These requests are implemented following the SOA in the form of business processes (BP). A BP is considered to be a virtual topology that consists of virtual nodes and virtual links where the virtual nodes encapsulate the request processing and location requirements, such as the requested sensing/actuating functions. The virtual links encapsulate the requests communication requirements such as traffic demands. The embedding operation maps the virtual nodes and links of each BP into the IoT layer. The goal is to find the optimal set of IoT nodes and links to embed the BPs virtual topology so that the IoT total power consumption (network plus processing) is minimized. This problem is formulated and analyzed using MILP.

This paper is organized as follows: In Section II, we introduce our framework of service embedding in IoT networks. Section III discusses the energy efficient service embedding evaluation and results. Finally, Section IV concludes the paper.

II. THE FRAMEWORK OF SERVICE EMBEDDING IN IOT NETWORKS

In this section we introduce the framework developed to embed services in IoT networks. The framework is based on a Mixed Integer Linear Programing (MILP) optimization model with the objective of minimizing the power consumption of service embedding in IoT networks. We benefit from our track record in virtual network embedding in core networks [17]. We model two layers, a physical layer that consists of IoT nodes, and a virtual layer that consists of several BPs. Fig. 2 shows a schematic where each IoT node is characterized by:

- A processing module hosting a CPU and RAM. We only take into account the power consumption of the CPU considering both idle and proportional power components.
- A network module hosting a Tx/Rx circuit and a Tx power amplifier. We consider the power consumption of all these elements while accounting for both idle and proportional power components.
- A function module that provides interfaces to a set of supported sensors and actuators. Furthermore, each IoT node can work as a control node, i.e. providing control functions through the CPU. We do not take into account the power consumption of the function module and its attached sensors and actuators.

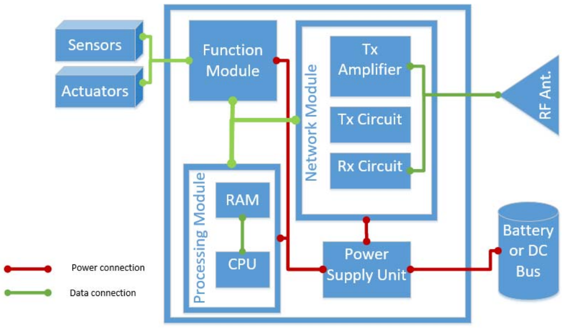


Fig. 2: Block diagram of IoT Node.

We only consider the processing and network power consumption as they dominate the IoT node power consumption. In addition, compared to the function module, and high power consuming actuators are usually externally powered by an independent power source.

Our virtualization framework is represented by a set of BP's, each BP is characterized by:

- Set of virtual nodes and links.
- A function for each virtual node.
- Virtual nodes processing and memory requirements.
- Virtual nodes geographical zone to be allocated.
- Virtual links traffic demands.

Given the above information, the framework responds by selecting the optimal IoT nodes and end to end routes to embed the BPs in the IoT network so that the total power consumption (network plus processing) of the IoT network is minimized. This is done by exploiting the selecting, virtualization and heterogeneity in IoT nodes resources and power consumption.

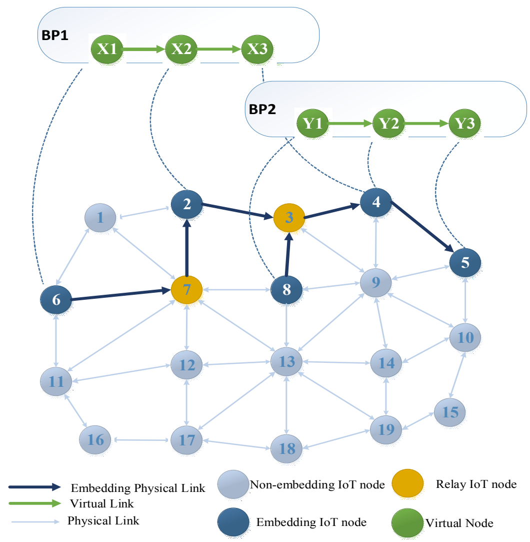


Fig. 3: Service embedding layers in IoT networks

Fig. 3 gives an example of embedding two BPs. The framework embeds the virtual nodes of BP1 (X1-X2-X3) in the physical IoT nodes (6-2-4), respectively; and chooses the optimal path (6-7-2-3-4). Each virtual node, e.g. X1 in BP1, is embedded into an IoT node that satisfies the virtual node's requirements (e.g. IoT node 6 to embed X1 in BP1). Each IoT node has been located on geographical zone and connected with subset of neighbors, e.g. IoT nodes 2, 6, and 7 are considered as neighbor's subset to node 1. Note that it is not necessary that neighboring node pairs in the virtual layer are embedded in neighboring node pairs in the physical later. For example, the neighboring virtual nodes X1 and X2 from BP1 are embedded in IoT nodes 6 and 2 which are not physical neighbors. This is acceptable as IoT nodes 6 and 2 can still communicate, and therefore preserve X1 to X2 communication, through IoT node 7 which works in this case as a wireless relay between IoT nodes 6 and 2. An IoT node that embeds a certain virtual node of a certain BP can at the same time work as a relay node for the traffic associated with another BP. This is shown in the second embedding example where IoT node 4 which is an embedding node for BP1 is working as a relay node for the traffic associated with BP2.

Due to the heterogeneous property exhibited by IoT nodes, various power consumption properties characterize IoT networks. Our framework optimizes the selection of IoT nodes in a manner that reduces the total power consumption. As discussed, the framework exploits the heterogeneous characteristics of the power consumption of each module in the selection of the IoT nodes as shown in Fig. 2. The power consumption of IoT nodes is mainly attributed to the processing and network modules because the sensing power is significantly lower than the processing power consumption and network power consumption [4], [18]. The framework performs the embedding operation through two parts as follows:

A. Embedding of virtual nodes:

$$\sum_{c \in P} P_{iac}^{NE} = 1 \qquad (1)$$
$$\forall i \in B, \quad \forall a \in V$$

$$\sum_{a \in V} P_{iac}^{NE} \leq 1 \qquad (2)$$
$$\forall i \in B, \forall c \in P$$

where P is set of IoT nodes, B is set of BP's, V is set of virtual nodes, P_{iac}^{NE} is binary variable indicate that virtual node a in BP i has been embedded in IoT node c .Constraints (1) and (2) ensure that each virtual node in a BP is embedded in a single IoT node only and states that each IoT node is not allowed to host more than one virtual node in each BP.

The framework selects the IoT nodes: i) by processing module, based on CPU and memory capacity constraints. The framework ensures that the embedded CPU and memory workloads in an IoT node do not exceed the CPU and memory capacities. ii) by function module, which ensures that the required function of each virtual node in BP is provided by its hosting IoT node. Finally iii) by the zone, so that the required zone of each virtual node in BP is matched to the zone of the hosting IoT node.

B. Embedding of virtual links:

$$P_{iac}^{NE} + P_{ibd}^{NE} = P_{iabcd}^{LE} + 2 \cdot W_{iabcd}^{LE} \qquad (3)$$
$$\forall i \in B, \forall a \in V, \forall b \in VN_{ia} : a \neq b, \forall c, d \in P : c \neq d$$

where VN_{ia} is the subset of virtual node's neighbors. Constraint (3) generates a binary variable P_{iabcd}^{LE} that indicate each neighboring virtual nodes pair (a and b) in any BP are also connected in the embedding IoT nodes (c and d), W_{iabcd}^{LE} is neglected variable

$$\sum_{i \in B} \sum_{a \in V} \sum_{b \in VN_{ia}} P_{iabcd}^{LE} \cdot V_{iab}^{TRFIC} = P_{cd}^{TRFP} \qquad (4)$$
$$\forall c, d \in P : c \neq d$$

Constraint (4) generates the path's traffic matrix P_{cd}^{TRFP}, where V_{iab}^{TRFIC} is the traffic demand between the virtual node pair (a, b) in BP i in kb/s. Using flow conservation constraints, the framework applies the flow conservation constraints to the traffic flows in the IoT network then generates the link's traffic matrix that describes the traffic between the neighboring IoT nodes e and f. The framework has constraints that state that the total traffic flows of the IoT node should not exceed the node capacity. Other constraints ensure that traffic splitting is prevented for each path between the embedding IoT nodes c and d, such that the maximum number of physical links between neighboring IoT nodes e and f is one.

III. ENERGY EFFICIENT SERVICE EMBEDDING IN IOT NETWORKS

The framework considers a smart city scenario where the physical layer is composed of 30 IoT nodes connected by 89 bidirectional wireless links, these IoT nodes are distributed across a city section of an area 500m × 600m, where the IoT nodes can carry various functions. The following assumptions are made:

- There is a set of 9 different functions, 4 sensing functions, one control function and 4 actuation functions. Each IoT node is capable of providing four functions only from this set while the virtual node requests are for one function only.
- The IoT nodes processing capability is uniformly distributed among CPU frequencies (4, 8, 16, 25, or 48 MHz) representing microcontrollers MSP430L09, MSP430F1, MSP430FR4, or MSP430F5 [19] as shown in Table 1.

Table1: Processing modules power specifications and power consumption in active mode

CPU Type	CPU CLK	RAM	Voltage	Current per MHz	Power per MHz
MSP 430L09	4 MHz	2 kB	0.9V – .65V	45 µA	40 µW
MSP 430F1	8 MHz	60 kB	1.8 – 3.6V	200 µA	600µW
MSP 430FR2	16 MHz	16 kB	1.8 – 3.6V	126 µA	378 µW
MSP 430FR4	16 MHz	16 kB	1.8 – 3.6V	126 µA	378 µW
MSP 430i2	16 MHz	32 kB	2.2V– 3.6V	350 µA	1050 µW
MSP 430FR5	16 MHz	64 kB	1.8 – 3.6V	100 µA	300 µW
MSP 430FR6	16 MHz	128 kB	1.8 – 3.6V	100 µA	300 µW
MSP 430G2	16 MHz	56 kB	1.8 – 3.6V	220 µA	660 µW
MSP 430F2	16 MHz	120 kB	1.8 – 3.6V	200 µA	600 µW
MSP 430F5	25 MHz	512 kB	1.8 – 3.6V	195 µA	585 µW
MSP 432P4	48 MHz	256 kB	1.62 -3.7V	95 uA	285 µW

- The processing demand of virtual nodes is random and uniformly distributed. It varies between 4 and 48 MHz.
- The IoT nodes are connected with neighboring nodes according to a mesh topology via HPZB01, HPZB01P or CC3100 RF transceiver modules [20]. These modules are

low cost, low power, and are compatible with the ZigBee protocol stack for IoT networks according to the standard of IEEE 802.15.4. The network demands of the virtual links vary from 50 to 250 kbps, random, uniform.

- There is a set of five geographical zones that represent the sub-districts of the smart city. The IoT nodes are distributed randomly and uniformly over these zones and each virtual node requests a location in one of these five zones.
- The virtual demands in the IoT network arrive at a single time point.

We evaluated the power consumption of two different scenarios. In the first scenario, dubbed energy unaware scenario, the framework embeds the virtual nodes of each BP into available IoT nodes that satisfy the virtual nodes and links requirements only without asserting a particular objective. While in the second scenario, dubbed energy aware scenario, the framework embeds the virtual nodes and links of each BP into available IoT nodes that satisfy the objective of power consumption minimization. The objective function of the energy aware framework is given as:

$$\textbf{Objective}: \textit{minimize } P^{TCP} + P^{TTP} \quad (5)$$

where P^{TCP} is the processing power consumption in the IoT network and given as:

$$P^{TCP} = \sum_{c \in P} P_c^{PMI} \cdot P_c^{IDLECP} + \sum_{c \in P} \sum_{i \in B} \sum_{a \in V} P_{iac}^{NE} \cdot P_c^{MAXCP} \cdot \frac{V_{ia}^{MCU}}{P_c^{MCU}} \quad (6)$$

where P_c^{PMI} is binary variable that indicate active processing module in IoT node c, P_c^{IDLECP} is idle processing power parameter of IoT node c in mW, P_{iac}^{NE} is binary variable indicates that virtual node a in BP i has been embedded in IoT node c, P_c^{MAXCP} is parameter of maximum CPU power consumption in each IoT node c in mW, V_{ia}^{MCU} is parameter of processing requirement of the virtual node a in BP i in MHz, and P_c^{MCU} is parameter of processing capability of the IoT node c in MHz.

While P^{TTP} is the network power consumption in the IoT network and given as:

$$P^{TTP} = \sum_{e \in P} P_e^{TMI} \cdot P_e^{IDLETP} + 2 \cdot \sum_{e \in P} \sum_{f \in PN_e} P_{ef}^{TRFIC} \cdot P_{ef}^{EPBT} + \sum_{e \in P} \sum_{f \in PN_e} P_{ef}^{TRFIC} \cdot (P_{ef}^{DIST})^2 \cdot P_{ef}^{FACTOR} \quad (7)$$

where f is neighbor IoT node of e and included in PN_e, PN_e is neighbors subset of IoT node e, P_e^{TMI} is binary variable that indicate active network module in IoT node, P_e^{IDLETP} is idle network power parameter of IoT node e, P_{ef}^{TRFIC} is variable of traffic between neighbouring IoT nodes (e, f) in kbps, P_{ef}^{EPBT} is parameter of energy per bit for each IoT link (e, f) in mW/kbps, P_{ef}^{DIST} is parameter of distance between the neighbouring IoT nodes pair (e, f) in meters, and P_{ef}^{FACTOR} is parameter of transmit amplifier factor [18] for each IoT link (e, f) in mW/kbps/m^2.

We run our framework at different number of BPs, varying from 2 to 10 BPs in step of two for both scenarios. We have assumed that each BP is composed of three virtual nodes with the sequential workflow of a sensor, a controller, and an actuator. The framework results are given for both scenarios.

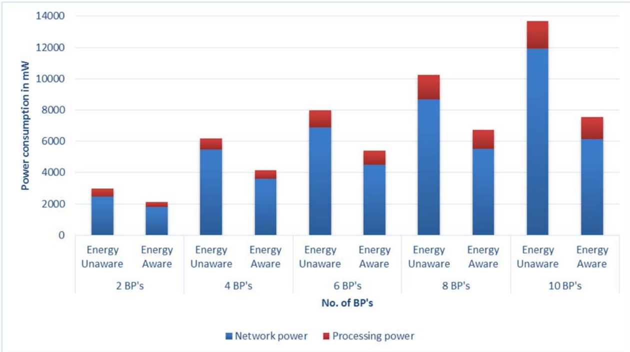


Fig. 4: Power consumption of service embedding.

Fig. 4 shows respectively the processing and network power consumption of the energy unaware embedding and energy aware embedding scenarios evaluated under a different number of BPs. The results show that for both scenarios, most of the power is consumed by the network module in IoT nodes. The power consumed by the network modules contributes about 86% (on average) of the total power consumption of the IoT network. We note also that the network power consumption increases at a rate higher than the processing power consumption (118.5 mW/BP compared to 15.25 mW/BP, respectively) as the network module has a steeper power curve compared to the processing module.

The processing power consumption of the energy unaware scenario was evaluated and compared with the processing power consumption of the energy aware scenario at different number of BPs. The results show that the energy aware scenario has an average saving 27% of processing power consumption compared with the energy unaware scenario. This is due to the ability to embed the highest number of virtual nodes in the minimal number of IoT nodes. Unlike the network power consumption results, the processing power consumption results show that the highest power saving (54.7%) was observed at two BPs, and the lowest power saving (16.4%) was observed at 10 BPs. This is because of the multiple requirements of virtual nodes that decrease the degrees of freedom in the choice space of IoT nodes, resulting in the IoT nodes being less capable of hosting more BPs virtual nodes (and thus conserving power) due to the lack of suitable resources matching all the requirements of each virtual node.

The framework results show that the energy aware scenario has saved 36% on average network power consumption compared with the energy unaware scenario. This saving is due to: firstly, selecting the shortest distance links in order to reduce Tx-amplifier power consumption (distance is reduced by 46% compared to the energy unaware embedding). Secondly, reducing the number of hops in order to reduce the number of relay IoT nodes (reduced by 35% compared to energy unaware embedding). Thirdly, efficiently utilizing the capacity of the links selected for the energy efficient routes in the IoT network. The lowest power saving is observed at 2 BPs case with power saving of 26%, while the highest power saving was observed at 10BPs with power saving of 48%. This is because with more BPs, the corresponding traffic demands can be consolidated more efficiently in a fewer number of paths in the IoT network.

IV. CONCLUSIONS

This paper has investigated the energy efficiency of service embedding in IoT networks of a smart city scenario. The services to be embedded are represented by a virtual topology (virtual nodes and links) which meets the demands of a business process workflow. We have developed a framework for optimizing the selection of IoT nodes and routes in the IoT network to meet the demands of the BPs virtual nodes and links with the goal of minimizing the IoT system total power consumption. The results show that we can save an average of 27% of the processing power consumption, and 36% of the network power consumption compared to energy unaware embedding. The results also show that the network power saving is proportional to the number of embedded BPs as increasing the number of BPs allows further consolidation of virtual links into physical links. On the other hand the processing power savings is inversely proportional to the number of embedded BPs as embedding multiple virtual nodes decreases the degrees of freedom in selecting the IoT nodes resulting in less efficient selection.

ACKNOWLEDGEMENT

The authors would like to acknowledge funding from the Engineering and Physical Sciences Research Council (EPSRC), INTERNET (EP/H040536/1) and STAR (EP/K016873/1) projects. Mr Haider Al-Shammari would like to thank the Higher Committee for Education Development (HCED) for funding his scholarship.

REFERENCES

[1] D. Giusto, A. Iera, G. Morabito, and L. Atzori, *The internet of things: 20th Tyrrhenian workshop on digital communications*. Springer Science & Business Media, 2010.

[2] K. Evangelos A, T. Nikolaos D, and B. Anthony C, "Integrating RFIDs and smart objects into a UnifiedInternet of Things architecture," *Advances in Internet of Things,* vol. 2011, 2011.

[3] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton, and T. Razafindralambo, "A survey on facilities for experimental internet of things research," *IEEE Communications Magazine,* vol. 49, no. 11, 2011.

[4] Z. T. Al-Azez, A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "Virtualization framework for energy efficient IoT networks," in *Cloud Networking (CloudNet), 2015 IEEE 4th International Conference on*, 2015, pp. 74-77.

[5] S. E. Collier, "The Emerging Enernet: Convergence of the Smart Grid with the Internet of Things," *IEEE Industry Applications Magazine,* vol. 23, no. 2, pp. 12-16, 2017.

[6] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the internet of things using big data analytics," *Computer Networks,* vol. 101, pp. 63-80, 2016.

[7] A. Cenedese, A. Zanella, L. Vangelista, and M. Zorzi, "Padova smart city: An urban internet of things experimentation," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on a*, 2014, pp. 1-6: IEEE.

[8] P. Sethi and S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering,* vol. 2017, 2017.

[9] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on industrial informatics,* vol. 10, no. 4, pp. 2233-2243, 2014.

[10] P. Kumar, "Some Observations on Dependency Analysis of SOA Based Systems," *International Journal of Information Technology and Computer Science (IJITCS),* vol. 8, no. 1, p. 54, 2016.

[11] W. Zhiliang, Y. Yi, W. Lu, and W. Wei, "A SOA based IOT communication middleware," in *Mechatronic science, electric engineering and computer (MEC), 2011 International conference on*, 2011, pp. 2555-2558: IEEE.

[12] M. E. Khanouche, Y. Amirat, A. Chibani, M. Kerkar, and A. Yachir, "Energy-centered and QoS-aware services selection for Internet of Things," *IEEE Transactions on Automation Science and Engineering,* vol. 13, no. 3, pp. 1256-1269, 2016.

[13] P. Spiess *et al.*, "SOA-based integration of the internet of things in enterprise services," in *Web Services, 2009. ICWS 2009. IEEE International Conference on*, 2009, pp. 968-975: IEEE.

[14] S. Clement, D. McKee, and J. Xu, "Service-Oriented Reference Architecture for Smart Cities," in *Service-Oriented System Engineering (SOSE), 2017 IEEE Symposium on*, 2017, pp. 81-85: IEEE.

[15] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Computer Networks,* vol. 67, pp. 104-122, 2014.

[16] S. Abdullah and K. Yang, "A QoS aware message scheduling algorithm in Internet of Things environment," in *Online Conference on Green Communications (GreenCom), 2013 IEEE*, 2013, pp. 175-180: IEEE.

[17] L. Nonde, T. E. El-Gorashi, and J. M. Elmirghani, "Energy efficient virtual network embedding for cloud networks," *Journal of Lightwave Technology,* vol. 33, no. 9, pp. 1828-1849, 2015.

[18] J. Huang, Y. Meng, X. Gong, Y. Liu, and Q. Duan, "A novel deployment scheme for green internet of things," *IEEE Internet of Things Journal,* vol. 1, no. 2, pp. 196-205, 2014.

[19] (Accessed 26-06). *MSP430L092 MSP430L092 Mixed Signal Microcontroller | TI.com* [Datasheet]. Available: http://www.ti.com/product/MSP430L092?keyMatch=MSP430L09&tisearch=Search-EN-Everything

[20] (2017). *SimpleLink Wi-Fi CC3100 BoosterPack Reference Design*. Available: http://www.ti.com/tool/cc3100boost-rd?keyMatch=CC3100%20RF&tisearch=Search-EN-Everything