# Towards energy and resource efficient Internet-of-Things: a design principle combining computation, communications and protocols

Zhengguo Sheng[1], Daxin Tian[2], and Victor C.M. Leung[3]

[1]Department of Engineering and Design, University of Sussex, UK
[2]School of Transportation Science and Engineering, Beihang University, China
[3]Department of Electrical and Computer Engineering, The University of British Columbia, Canada

***Abstract*—Advances in future computing and communications to support Internet-of-Things (IoT) are becoming more important as the need to better utilize resources and make them energy efficient. As a result, it is predicted that intelligent devices and networks, including wireless sensor networks (WSN), will become the new interfaces to support future IoT applications. However, many open challenges remain, which are mostly due to the resource constraints imposed by various hardware platforms and complex characteristics of applications wishing to make use of IoT systems. Thus, it becomes critically important to study how the current approaches incorporating both computing and communications in this area can be improved, and at the same time better understand the opportunities for the research community to utilize the proposed approaches. To this end, this article presents an overview of our latest research results in sensor edge computing and lightweight communication protocols as well as their potential applications.**

## I. Introduction

With the development of Internet-of-things (IoT), wireless sensing devices will be massively deployed in a wide range of application environments. Such devices are usually constrained by limited battery and memory resources, processing capability, radio communication range and reliability, etc. In order to provide real-time applications without direct human interactions, IoT systems should be able to provide reliability and efficiency without consuming significant resources.

IoT can be categorized into varies application verticals, such as smart city, smart home and smart transportation [1]. It is clear that although a wide range of intelligent and tiny sensing devices have been massively deployed, they all share a common architecture and network elements. Despite IoT architectures are proposed and discussed in various organizations, such as the oneM2M Global Initiative is to develop one globally agreed specifications for common service architecture and IEEE IoT initiative is to address a reference model cross different IoT application verticals, we can come up with an IoT architecture with following four layers:

1) Sensor device: IoT uses various wireless devices to capture events or monitor statuses of different things. In a multi-hop wireless sensor network, sensor data is relayed through peer nodes to the gateway via wireless, wired, or hybrid networks.
2) Data connectivity: It connects sensor devices, as sources of data, with a cloud platform, which processes the data. It actually behaves as a gateway to translate the captured event from the sensor layer into a standard format and deliver it through broadband or wireless networks in the IoT architecture.
3) Cloud platform: The key idea of cloud computing is to create a pool of centralized computing resources across networks, which can deliver on demand services to users over the Internet. Moreover, it can provide a unified set of common operation functions such as management, protocol conversion, route forwarding for service operators.
4) Applications and services: Mashup applications can be further developed via application programming interfaces (APIs) provided by the cloud platform and delivered as cloud-based services.

As can be seen, the existing IoT architecture is designed for traditional web applications, rather than future Internet applications running on various mobile and sensor nodes. Today, with the development of wireless communications and the advancement of more powerful and low cost sensor platforms, the emerging dissemination of wireless sensor networks and cloud computing has brought new opportunities of sensor and cloud integration. Although various sensor cloud schemes have been developed to increase bandwidth efficiency [2], the sensor node is usually assumed as data collecting point and there is lack of understanding of its computing capability and the potential benefits of being as a computing edge. Such edge computing component can be fully utilized as large scale IoT applications would have trouble on sending data to cloud, due to the stringent real-time requirements. Moreover, the wireless communications and networking as a resource in cloud computing has never been exploited. Thanks to the

This research was supported in part by Asa Briggs Visiting Fellowship from University of Sussex, Royal Society- Newton Mobility Grant (IE160920) and The Engineering, and Physical Sciences Research Council (EPSRC) (EP/P025862/1).

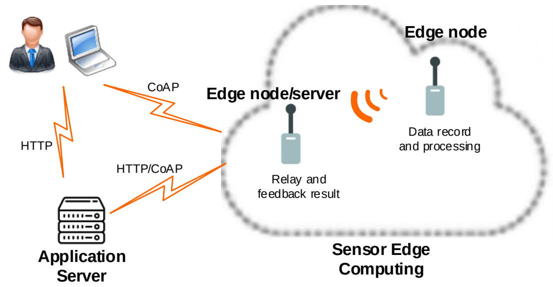


Fig. 1. An illustration of sensor edge computing

development in IoT protocol suite [3], the routing protocol for low power and lossy networks (RPL) [4] is shown to be a promising solution for multi-hop wireless sensor networks. The latest implementation of the application protocol, i.e., Constrained Application Protocol (CoAP) [5], for accessing applications and services on wireless resource constrained networks has enabled the newly emerging Sensor-as-a-Service (SaaS) paradigm.

Fig. 1 gives an example of the sensor edge computing where wireless sensor networks form a cooperative cloud[1] and can serve clients' service requests via standard IP networks. On the way to design and develop such an IoT system, there is a considerable need to understand its limitations and benefits, and its inter-dependence with computing and communications functions. Distinct from the existing literatures where a single design criterion is usually considered, the contribution of this tutorial seeking is two fold.

- One is to pursue new design of sensor edge computing by fully exploring the application characteristics, computing and communications capabilities, which provides the foundation and theoretical understanding of energy efficiency.
- The second step is from the upper layer and practical aspect, where the lightweight IETF protocols for IoT shall be applied and coexist with existing state-of-the-art solutions, e.g., standard TCP/IP.

Obviously our design principle combines the two together not just because the ultimate goal of energy and resource efficiency of IoT is only possible with the accomplishment of the both two parts, but also because the interaction between the two parts is the key for effective system design.

The remainder of this paper is organized as follows. Section II discusses the IoT application characters, modeling of computation and communications, protocols and outlines new design principles in achieving energy and resource efficiency. Moreover, some evaluation results are also supplemented to demonstrate positive gain of the proposed solutions. Section III summarizes the emerging applications of the proposed approach and their challenges. Conclusion is given in Section IV.

## II. IoT design principles

### A. Application Model

We adopt a well known canonical model [6] to capture the essentials of a typical application, which can be abstracted into the following two parameters:

- Processing data size $L$: the total number of data bits for executing an IoT application. In a distributed computing system, such processing data can be partitioned from the main code and scheduled to a peer sensor for remote execution.
- Application completion deadline $T$: the maximum time that an IoT application must be completed. $t$ is discrete time index ranging from $t = 1...T$.

It is worth noting that an application is a program that performs a computation on an input file, such as calculating the minimum temperature from a period of history record. Similar to the model applied in MapReduce [7] which has been shown as an effective solution to process and generate big data with a parallel and distributed strategy, an application can be breakable into small tasks which do not exhibit dependencies across partitions of its input. We consider that all sensor nodes are capable of executing a same application without need to transfer executable files for operation, thus only the input partitions are transmitted to other sensor nodes for parallel executions. Although there are cases that some tasks cannot be broken into smaller pieces and can only be executed on a single node due to the dependencies in its input, there are still concurrency benefits when many such tasks are executed in batches.

In essence, the energy consumption of an application is highly related to these two parameters. For example, with a large size of input data and stringent completion deadline, a sensor node may consume extensive energy. In the following, we denote such an application as $A(L, T)$ and use it to characterize the energy consumption of computation and communication, respectively.

### B. Computation

The energy consumption of computation is directly determined by the CPU workload of a sensor node. According to [8], the workload can be measured by the number of CPU cycles required by an application, which is related to the data size and computation complexity, and can be defined as $W = LX$, where $W$ is the number of CPU cycles, $L$ is the processing data size and $X$ is the computation algorithm which can be characterized as a random variable with Gamma distribution.

Although a number of factors consume CPU power, such as short circuit power and dynamic power, etc., the energy consumption is dominated by dynamic power which can be minimized by configuring the clock frequency of the chip via the dynamic voltage scaling technology [9]. In CMOS circuits, the computation energy per operation cycle $\epsilon_c$ is proportional to $V^2$, where $V$ is the supply voltage to the chip. When an operation is at low voltage, such as in wireless sensor networks,

[1] It is defined as a single application processed with helps of multiple edges, e.g., peer sensors.

the clock frequency, $f$, can be treated as a linear function of the voltage supply. Therefore, the total energy consumption of computation can be expressed as $E_c = \sum_{w=1}^{W} \kappa f_w^2$, where $\kappa$ is the effective switched capacity determined by the chip architecture and $f_w$ is the clock-frequency which is scheduled in the next CPU cycle given the number of $w$ CPU cycles have been completed.

Intuitively, the CPU can reduce its energy consumption by scheduling low clock frequency. However, as a practical real-time implementation, the application has to meet a delay deadline. We adopt the statistical CPU scheduling model [10] which assumes the application should satisfy the soft real-time requirement, in which the application completion needs to meet its deadline with the probability $p$ by allocating $W_p$ CPU cycles. Hence, the total energy consumption can be derived as $E_c = \kappa \sum_{w=1}^{W_p} F_W^c(w) f_w^2$, where $F_W^c(w)$ is the complementary cumulative distribution function (CCDF) that the application has not completed after $w$ CPU cycles.

According to [6], by optimizing the clock-frequency scheduling for each CPU cycle $f_w$ and ensure the application completion time is less than the deadline $T$, we can derive the minimum value of computation as [11]

$$E_c = \frac{KL^3}{T^2}\,. \tag{1}$$

where $K$ is a constant factor determined by $\kappa$ and $p$.

### C. Communications

The power consumption of communications is determined by the number of bits being transmitted and the current draw of the electrical circuits that implement the physical communication layer which includes idle, transmit and receive modes. According to IEEE 802.15.4 transceiver which is widely used in IoT, the power consumption is dominated by the transmit or receive modes and their costs are approximately the same. We use an empirical transmission energy model [12] to characterize communication cost which includes both transmission and reception of processing tasks, but do not consider the small output results[2] from the node. The required energy $E_t$ to transmit $L$ bits within a time slot is governed by a convex monomial function

$$E_t = \rho \frac{L^n}{g}\,. \tag{2}$$

where $\rho$ denotes the energy coefficient, $g$ denotes channel state and $n$ denotes the order of monomial with value $1 \leq n \leq 5$. The choice of $n$ depends on the bit scheduler policy, with a large value of $n$, the scheduler will transmit equal number of bits at every time slot regardless of the channel state [12]. In this tutorial, we focus on the opportunistic scheduling by having $n = 1$ in which the transmission only depends on the channel state and is completed in one time slot. The motivations behind this scenario are following: First, for energy constrained nodes, it is not desirable to divide a single data unit across a number of consecutive time slots because of energy consumption from extra overhead associated with each slot. Second, since we impose an application deadline, the transmission time should be relatively small compared to $T$, such that the time offset between local and remote executions can be negligible. Third, the transmission time should be minimized to avoid channel fluctuation caused by node mobility.

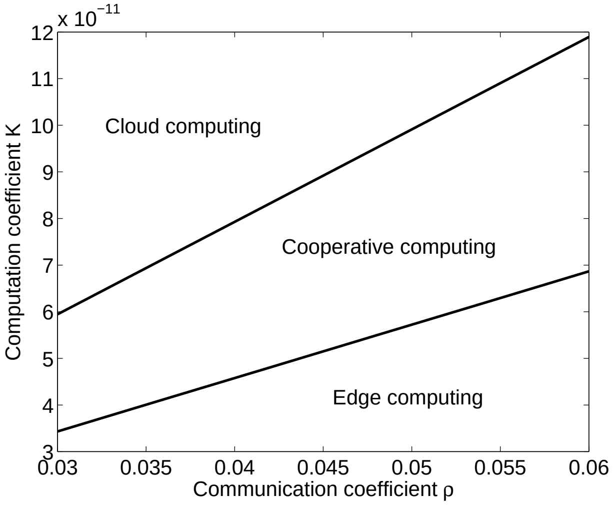


Fig. 2. An illustration of energy optimal computing rules [11]

The first design principle to achieve energy efficiency is to find an optimal partition to minimize the total energy consumption of processing an application given that a target completion deadline $T$ is satisfied by using the above computation and communication modelling, and can be derived as

***Design principle 1***: By defining the application processing speed as $\upsilon = \frac{L}{T}$, we have the equivalent energy optimal computing rules [11]

$$\begin{cases} \text{Edge computing,} & \text{if} \quad 0 < \upsilon \leq \sqrt{\frac{2\rho}{3Kg}} \\ \text{Cooperative computing}^3 & \text{if} \quad \sqrt{\frac{2\rho}{3Kg}} < \upsilon \leq \sqrt{\frac{2\rho}{\sqrt{3}Kg}} \\ \text{Cloud computing,} & \text{if} \quad \sqrt{\frac{2\rho}{\sqrt{3}Kg}} < \upsilon \end{cases} \tag{3}$$

Fig. 2 shows a result of the energy optimal computing rules for the case of $L = 1024\,\text{bits}$, $T = 30\,\text{ms}$ and $g = 0.5$ is the channel state between peer sensors. To be consistent with the real energy measurements [8] and specifications of IEEE 802.15.4, we consider the computation coefficient in the order of $10^{-11}$, the communication coefficient in the order of $10^{-2}$. The result reveals that with the application profile and system coefficients, we can quickly decide the best strategy to process an IoT application.

[2] This is a reasonable assumption for sensor networks since most of IoT applications come with simple results of warning or image detection indication, etc.

[3] Here we consider a simple case where only one peer node is used for cooperation. That is, a single task is partitioned into two parallel subtasks and is processed by two nearby sensor nodes.

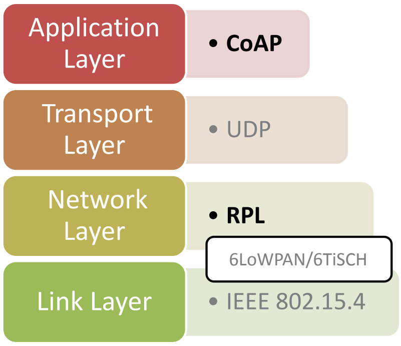


Fig. 3. An overview of IETF protocol stack ranging from Physical layer up to Application layer

It is worth noting that the above computing decision considered both application complexity and communications. We observe that with better computation efficiency (smaller $K$) and higher communication cost (larger $\rho$), the optimal partition tends to allocate more processing task locally at the edge. Moreover, with a relaxed completion deadline (large $T$), the local execution is more preferable to save energy by reducing processing speed. As the input data size increases, the cooperative computing can ensure optimal with better energy efficiency than the edge computing. Moreover, the cloud computing is only applied when high processing speed is required.

## D. Protocol

The computation and communications that we have discussed so far are more focusing on physical and hardware aspects of design principle. The IoT nodes sitting on the network edge are usually with very limited power, storage and communication bandwidth. Beside the resources taken by the data processing, the software based protocol design is also challenging and important for achieving resource efficient IoT. The IETF protocol stack used in wireless sensor networks [3] is a promising candidate to achieve the goal. Fig. 3 illustrates the protocol stack ranging from physical layer up to application layer. For example, IETF 6TiSCH [13] including IEEE 802.15.4 TSCH and 6LoWPAN is one promising radio technology standard for low power and large scale IoT applications. The medium access protocol (MAC) of 6TiSCH is built based on the IEEE 802.15.4 time slotted channel hopping (TSCH) link layer and can be directly incorporated with upper IPv6 application layers.

Among these protocol layers, RPL and CoAP are two key protocols to support a large scale IoT transmission and enable the newly emerging Sensor-as-a-Service (SaaS) paradigm.

*1) RPL:* It is a distance vector routing protocol. It does not have predefined topology but will be generated through the construction of Destination-Oriented Directed Acyclic Graphs (DODAGs). Directed Acyclic Graphs (DAGs) describe tree shaped structures. The DODAG, with sink node or the node providing default routing to the Internet as the root node, is a direction-oriented graph.

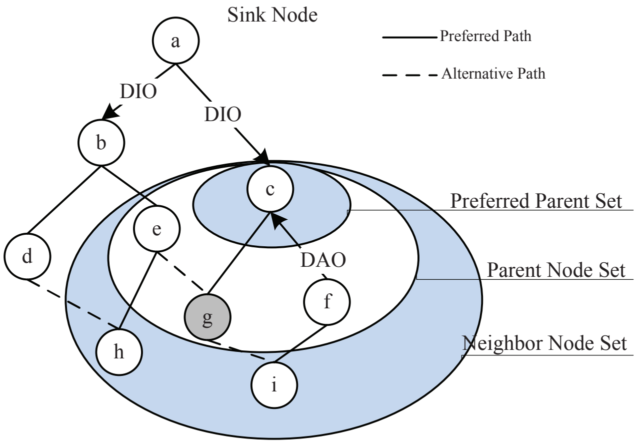


Fig. 4. An example of RPL and node set relationships.

The construction of network topology is controlled by three types of control message - DODAG Information Object (DIO), DODAG Information Solicitation (DIS) and Destination Advertisement Object (DAO) messages. They all belong to RPL control message, which is an Internet Control Message Protocol (ICMP) information message type with type value 155. DIO message is used for upward routing construction, which is essential for establishing communication from non-sink nodes (or multiple points) to the sink node (one point). Such Multipoint-to-point (MP2P) mode is dominating the RPL applications. The construction of upward route of RPL is realized by DIOs. The sink node will first broadcast DIOs, the nodes receiving the DIO directly from the sink node become its neighbours. By setting the sink node as their parent nodes, those neighbour nodes will re-broadcast DIOs to further nodes. The similar step will repeat in such way that the DODAG topology is constructed through handling DIOs and building parent sets. DIS message is used for soliciting the sending of DIO in order to make immediate response to network inconsistency. DAO message is used for downward routing construction (Point-to-Point and Point-to-multipoint). There are two modes of downward routing - storing and non-storing modes, which indicate that the routing table information is stored in intermediate nodes (non-root and non-leaf nodes) and root node, respectively.

It is worth noting that in order to construct a valid RPL routing, firstly, candidate neighbour node set must be the subset of nodes that can be reached through link local multicast. Secondly, parent set is the subset of candidate neighbour set which satisfies specific limitation conditions. Thirdly, preferred parents are those with optimal path characteristics. If there exist a group of nodes with equivalent rank and preferred extent regarding the metrics calculation, there can be more than one preferred parent nodes. Fig. 4 illustrates logical relationships of candidate neighbour node set, parent node set, and preferred parent node of the node.

*2) CoAP:* It is a specialized web transfer protocol for resource constrained nodes and networks. CoAP conforms to the representational state transfer (REST) style, which means it abstracts all the objects in the network as resources. Each resource corresponds to a unique Universal Resource Identifier (URI) from which the resources can be operated stateless, including GET, PUT, POST, DELETE and so on.

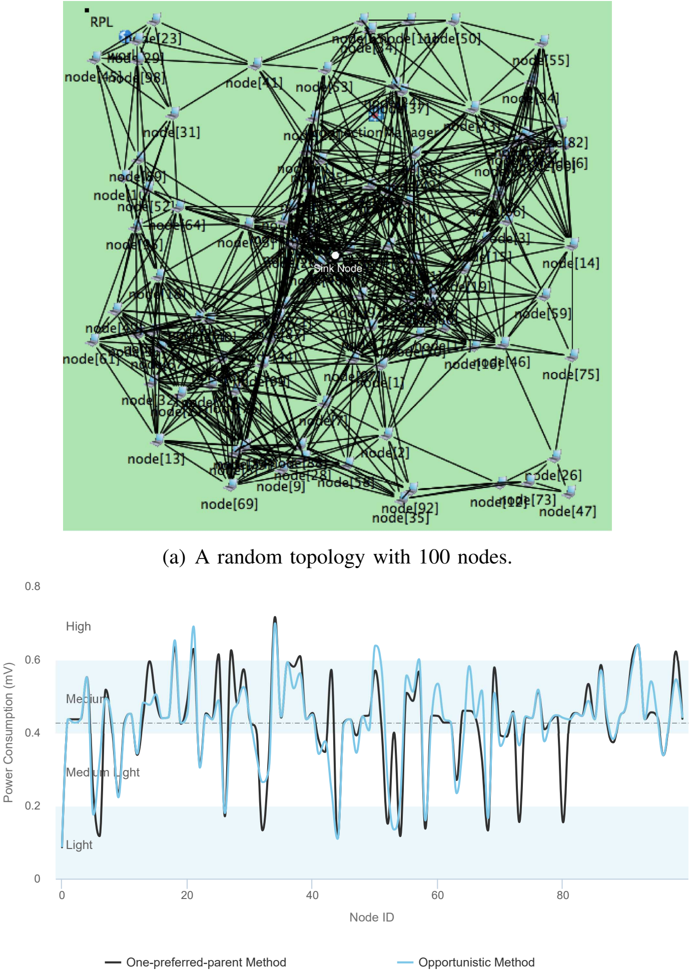
Strictly speaking, CoAP is not a HTTP compression protocol. On the one hand, CoAP realizes a subset of HTTP functions and is optimized for constrained environments. On the other hand, it offers features such as built-in resource discovery, multicast support and asynchronous message exchange. Unlike HTTP, CoAP adopts datagram-oriented transport protocols, such as UDP. The CoAP is based on the exchange of short messages which, by default, are transported over UDP. The protocol has a registered scheme of $<$ coap : // $\sim>$ with a default port of 5683. CoAP messages are encoded in a simple binary format.

Each application function can be abstracted as a recall process to conduct with resources on sensor device, thus the RESTful approach provided by the CoAP protocol can be adopted as a lightweight method to access from application servers to sensor devices. Fig. 1 shows an example of HTTP-CoAP mapping to translate between lightweight IoT protocol and standard HTTP protocol.

***Design principle 2***: The IETF protocol stack is a promising solution for IoT, particularly the use of RPL and CoAP can further enable the large scale IoT deployment and the lightweight RESTful Sensor-as-a-Service (SaaS).

Fig. 5(b) shows the average energy consumption of 100 nodes in Fig.5(a) using the two RPL based methods. The network topology is generated randomly with the sink node sitting at the center. Nodes are connected by log-normal shadowing and transmitting fixed UDPApp payload size of 60 bytes. The random seed has been applied to generate the data. The data is obtained from our self-developed OMNeT++ simulation platform[4] and we run the simulation with a duration of 100 s during which the topology construction and packet forwarding have been finished and nodes with zero energy will quit the topology immediately. As can be seen, the opportunistic forwarding method has a lower standard deviation (12.235) than the standard method (13.485) while the total consumption of the both are closed, which indicates that RPL can maintain nodes' energy cost in medium and light levels.

To demonstrate the overall energy performance of the proposed IoT design principles, we provide a real implementation of the edge computing with the lightweight protocols for remote acoustic monitoring, i.e., ACI calculation [15]. We consider a simple network scenario with two wireless sensor nodes. Each node is equipped with CC2530 MCU with 8051 CPU core running at 32MHz, 8KB SRAM and 256KB flash block to support IEEE 802.15.4-compliant radio transceiver. To support IPv6 connectivity and multi-hop transmission, all nodes are running Contiki v3.0 operating system with implementation of 6LoWPAN, IPv6 and RPL based on IEEE 802.15.4. The calculated ACI value will be packed as a network resource and sent back to users via CoAP protocol. In order to support audio recording, each node also consists of an audio board, a microphone, a SD card and a battery in a waterproof case.



(a) A random topology with 100 nodes.

(b) Energy consumption of 100 nodes with standard RPL method and opportunistic method [14].

Fig. 5. A simulation result developed by OMNeT++

Instead of two fully equipped recorders (nodes), as illustrated in Fig. 1, we use one node to record and process the data, and another node, to relay data only. Recording and computation of acoustic indices are carried out directly on edge nodes built from low noise primo condenser microphones. Each node can be individually configured to a user-defined recording schedule and execute onboard processing tasks, which prevents the need to save the audio files. Only the calculated acoustic indices are sent to the server in a multi-hop fashion, so that no data collection or external processing are required. The test schedules a 1 minute recording every 10 minutes between 4 AM to 9 AM and 5 PM to 10 PM. The audio file size is 5325 Byte. The energy consumption is measured with an USB digital power meter and all nodes keep awake during the whole observation, i.e., Idle mode when there is no data recording.

[4] The source code and basic setting of simulator are made available at https://github.com/qqbzg/rpl_omnet.

Tab. I shows the total energy consumption of the test using the proposed edge computing approach and transmission only approach, respectively.

TABLE I
TOTAL ENERGY CONSUMPTION BETWEEN EDGE COMPUTING APPROACH AND TRANSMISSION ONLY APPROACH

| Schedule (1 min every 10 min) | Total 24h<br>(4 AM to 9 AM,<br>5 PM to10 PM) | Energy cost for edge computing | Energy cost for trans. only |
|---|---|---|---|
| Recording | 10h<br>x 6 recordings/h | 86 mAh | 86 mAh |
| ACI calculation | 10h<br>x 6 calculations/h | 50.4 mAh | 0 |
| Transmission | 10h<br>x 6 transmit/h | 10.8 mAh | 375.6 mAh |
| Idle mode | 1349 min | 1940 mAh | 1940 mAh |
| Total energy consumption | | 2087.2 mAh | 2401.6 mAh |

The "transmission only" includes only the energy consumption for the transmission of the file (to the server), the external audio processing on the remote server is not considered in the calculation. As a comparison, the edge computing includes an onboard ACI calculation plus the transmission of the calculated acoustic index. It is clear that the proposed design principle can reduce the overall energy consumption. If we compare the ACI processing and transmission alone, the energy consumption can be saved by 83.7%.

## III. APPLICATIONS AND CHALLENGES

Such a joint design principles can be applied in a wide scope of IoT application areas including urban networks, building automation, industrial automation, and home automation. In different use cases, adaptation of design principles need to be considered to ensure optimized network performance.

### A. *Wireless body area networking and computing for eHealth*

Novel sensors, wearable and embedded devices enable the creation of imaginative pervasive computing applications to assist eHealth in everyday home, office and mobile environments, including environmental control applications, health-related applications of long term monitoring of people's health conditions, alarming for abnormal and life-threatening events such as heart attack, fall detection, and statistics collection and trending applications to integrate the smart eHealth into the greater society. These devices should include the mechanisms to deliver the collected intelligent data according to the ambient conditions to appropriate external parties in a timely manner.

However, there are two inseparable challenges in developing such systems: 1) Sensor devices are normally small and inexpensive, which puts several constraints in communications, including energy, storage, and bandwidth. These constraints pose a number of unique challenges in the design of wireless body area communication networks, including resource constraints, channel dynamics, interference, devices heterogeneity and security, etc. 2) Collected data need to be processed and further transferred to a remote cloud computing facility via various Internet access channels for detailed analysis and feedback on how to control the application. However, existing cloud computing models are designed for traditional web applications, rather than future Internet applications running on various mobile and sensor devices. Moreover, public clouds, as they exist in practice today, are far from the idealized utility computing model. This is particularly true for applications that are developed for a particular provider's platform and run in data centres that exist at singular points in space. This makes their network distance too far from many users to support highly latency-sensitive healthcare.

We believe that such an emerging dissemination of lightweight protocols for wireless body sensor networks and cooperative computing can bring new opportunities of sensor and cloud integration, which will facilitate clients not only to monitor and collect data from users but also to execute and output eHealth applications using its own processing capabilities.

### B. *IoT for environment and biodiversity Assessment*

Environment and biodiversity assessment is a central and urgent task in contemporary biology, not only for scientific research but also in areas of applied conservation biology, such as land-management and industrial planning. As a new paradigm for assessment, the relationship between soundscape and ecosystem structure is predicted by evolutionary biology (sound now being recognised as a key functional dimension in ecospace) and verified by an increasing number of experimental and observational studies. The current approach usually relies on a set of microphone recorders that are spatially distributed over the environment and manually collected in a regular basis.

The emerging wireless acoustic sensor networks, which are usually equipped with powerful processors and communication transceivers, have opened new possibilities for the design of sound signal processing and increased capabilities of remote monitoring. However, important challenges also arise in such a context, such as power consumption, intelligent data analysis and data routing, etc. The proposed sensor edge computing can be applied to achieve the extraction and evaluation of ecologically-meaningful soundscape components locally on sensor nodes. Meanwhile, the lightweight networking protocols can ensure a long lasting and reliable connectivity.

### C. *Automotive IoT*

The future network infrastructure for vehicular environments will increase the pervasiveness of the Internet and the overall connectivity by integrating every object (e.g., passengers' smart phones, vehicles' sensors, infrastructures and external management platform) forming an intelligent vehicular transportation system. IoT supported by vehicular networks will be a key enabler for new geo-spatial information mapping communities. Using the advanced communication and sensor capabilities hosted by vehicles, these IoT communities have the potential to reach a wide range of objectives at more cost effective ways. Applications for such a promising combination of information production range from providing road safety and driver assistance to mapping road status.

However, automotive IoT encounters many challenges, particularly data security is identified as a major technical problem and needs to be solved before large scale deployment. IEEE 1609.2 is the basic security protocol for V2X communications. Each sending message should be signed with an attached certificate. A vehicle using dedicated short-range communications (DSRC) to transmit a message needs to create a key pairs and send a public key to a certificate authority (CA) which is usually located on a remote server, then CA uses the public key to issue a certificate. Due to privacy and security concern, a DSRC unit will request hundreds or thousands of certificates at the same time. Such cryptographic operations at CA is immense if tens of thousand vehicles running at the same time. Therefore, a distributed key encryption and exchange method should be proposed in order to support real-time vehicle applications. The concept of mobile edge computing (MEC), which is similar to our idea, is a promising solution to establish a hieratical CA structure and push the cryptographic operations to the edge. Moreover, due to the high mobility of vehicle networks, a lightweight networking protocol is highly needed. Therefore, the proposed design principles can also be applied into vehicle networks to support security.

## IV. Conclusions

This tutorial describes two IoT design principles ranging from computing, communications to networking protocols. We have mainly characterized the IoT system from both theoretical and practical aspects, and analyzed the sensor edge computing strategy and lightweight routing and application protocols for IoT. In our view, these benefits make resource constrained IoT capable of maintaining energy efficiency and meeting future application requirements of complexity and real-time. The acquired new insights on the network performance could also provide a precise guideline for the efficient designs of practical and reliable communication systems. Hence these results will potentially have a broad impact across a range of areas, including embedded systems and computing, wireless communications, and network protocols.

## V. Acknowledgment

The authors would like to thank Xiyuan Liu and Saskia Pfersich who conducted the experiments, and anonymous reviewers who provided valuable comments to us.

## References

[1] Z. Sheng, C. Mahapatra, C. Zhu, and V. C. M. Leung, "Recent advances in industrial wireless sensor networks toward efficient management in iot," *IEEE Access*, vol. 3, pp. 622–637, 2015.

[2] A. Alamri, W. S. Ansari, M. M. Hassan, M. S. Hossain, A. Alelaiwi, and M. A. Hossain, "A Survey on Sensor-Cloud: Architecture, Applications, and Approaches," in *Int'l J. of Distributed Sensor Networks*, vol. 9, 2013, p. 18.

[3] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. McCann, and K. Leung, "A survey on the IETF protocol suite for the internet of things: standards, challenges, and opportunities," *IEEE Wireless Commun.*, vol. 20, no. 6, pp. 91–98, December 2013.

[4] Routing Over Low power and Lossy networks (roll), IETF, *Available at: http://datatracker.ietf.org/wg/roll/charter*, Accessed on Aug. 2017.

[5] Z. Shelby, K. Hartke, and C. Bormann, "Constrained Application Protocol (CoAP)," *Internect Draft, Available at: http://datatracker.ietf.org/wg/core/charter*, Accessed on Aug. 2017.

[6] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. on Wireless Commu.*, vol. 12, no. 9, pp. 4569–4581, September 2013.

[7] B. Heintz, A. Chandra, R. K. Sitaraman, and J. Weissman, "End-to-end optimization for geo-distributed mapreduce," *IEEE Transactions on Cloud Computing*, vol. 4, no. 3, pp. 293–306, July 2016.

[8] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proc. of the 2nd USENIX Conf. on Hot Topics in Cloud Computing*, ser. HotCloud'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 4–4.

[9] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits (2nd Edition)*. Prentice Hall, 2002.

[10] W. Yuan and K. Nahrstedt, "Energy-efficient CPU scheduling for multimedia applications," *ACM Trans. Comput. Syst.*, vol. 24, no. 3, pp. 292–331, 2006.

[11] Z. Sheng, C. Mahapatra, V. Leung, M. Chen, and P. Sahu, "Energy efficient cooperative computing in mobile wireless sensor networks," *IEEE Transactions on Cloud Computing*, vol. PP, no. 99, pp. 1–1, 2015.

[12] J. Lee and N. Jindal, "Delay constrained scheduling over fading channels: Optimal policies for monomial energy-cost functions," in *Proc. IEEE Int'l Conf. on Commu. (ICC).*, June 2009, pp. 1–5.

[13] T. Watteyne, P. Tuset-Peiro, X. Vilajosana, S. Pollin, and B. Krishnamachari, "Teaching communication technologies and standards for the industrial iot? use 6tisch!" *IEEE Communications Magazine*, vol. 55, no. 5, pp. 132–137, May 2017.

[14] M. Zhao, H. Y. Shwe, and P. H. J. Chong, "Cluster-parent based rpl for low-power and lossy networks in building environment," in *Proc. 2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, Jan 2015, pp. 779–784.

[15] N. Pieretti, A. Farina, and D. Morri, "A new methodology to infer the singing activity of an avian community: The acoustic complexity index (aci)," *Ecological Indicators*, vol. 11, no. 3, pp. 868 – 873, 2011.

PLACE
PHOTO
HERE

**Zhengguo Sheng** received the B.Sc. degree from University of Electronic Science and Technology of China, in 2006 and M.S. and Ph.D. degrees from Imperial College London, UK, in 2007 and 2011, respectively. He is currently a Lecturer with University of Sussex, UK. Previously, he was with UBC as a Research Associate and with Orange Labs as a Senior Researcher. He has more than 70 publications. His research interests cover IoT, vehicular communications, cloud/edge computing.

PLACE
PHOTO
HERE

**Daxin Tian** is a professor in the School of Transportation Science and Engineering, Beihang University, Beijing, China. His current research interests include mobile computing, intelligent transportation systems, vehicular ad hoc networks, and swarm intelligent.

PLACE
PHOTO
HERE

**Victor C. M. Leung** is a Professor of Electrical and Computer Engineering and holder of the TELUS Mobility Research Chair at the University of British Columbia. He has co-authored more than 1000 technical papers in the areas of wireless networks and mobile systems, in addition to 37 book chapters and 12 book titles. Dr. Leung is a Fellow of the Royal Society of Canada, the Canadian Academy of Engineering and the Engineering Institute of Canada.