ARTICLE IN PRESS

Journal of King Saud University – Computer and Information Sciences xxx (xxxx) xxx

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

A review of IoT network management: Current status and perspectives

Moussa Aboubakar [a,b,*], Mounir Kellil [a], Pierre Roux [a]

[a] *Université Paris-Saclay, CEA, List, F-91120 Palaiseau, France*

[b] *Sorbonne Universités, Université de Technologie de Compiègne, CNRS, HEUDIASYC UMR 7253, CS 60319, 60203 Compiègne Cedex, France*

ARTICLE INFO

Article history:
Received 9 October 2020
Revised 11 February 2021
Accepted 17 March 2021
Available online xxxx

Keywords:
IoT
Network management
Device management
IoT network management protocols
IoT networks management frameworks

ABSTRACT

During this last past decade, the Internet of Things (IoT) has gained much attention because it encompass intelligent devices such as smart sensors and actuators, which enable a wide range of applications that improve our daily life (e.g. smart agriculture). However, due to the presence of an important number of heterogeneous and resources constrained devices (in terms of memory, CPU and bandwidth) communicating over error-prone and lossy radio channels and often deployed in hostile environments (e.g. war zone), IoT networks are experiencing various network performance problems (e.g. excessive energy consumption resulting from network device failure). In this context, an efficient management of IoT networks is needed in order to ensure good network performances. This has fueled the development of different protocols and frameworks for management of IoT networks. In this paper we present a comprehensive study of representative works on IoT network management. The paper analyzes existing solutions for IoT low power networks management and presents a taxonomy of those solutions. Moreover, this paper also compares existing research proposals on management of IoT low power networks based on different requirements. At the end, this survey identifies remaining challenges for an efficient management of IoT low power networks.

Contents

* Corresponding author at: Université Paris-Saclay, CEA, List, F-91120 Palaiseau, France.
E-mail addresses: moussa.aboubakar@hds.utc.fr (M. Aboubakar), mounir.kellil@cea.fr (M. Kellil), pierre.roux@cea.fr (P. Roux).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2021.03.006
1319-1578/© 2021 The Authors. Production and hosting by Elsevier B.V. on behalf of King Saud University.

Please cite this article as: M. Aboubakar, M. Kellil and P. Roux, A review of IoT network management: Current status and perspectives, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2021.03.006

1. Introduction

Internet of Things (IoT) has attracted a lot of attention these recent years from both researchers and industrials. Coined by Kevin Ashton (Ashton et al., 2009), the term IoT refers to a paradigm where the physical objects of our daily life (e.g. sensors, actuators, home appliances and so forth) are connected to the Internet and are able to communicate in an intelligent fashion. The IoT aims at making our daily life agreeable, more connected and more productive. The recent technological advances in low power devices contributed to foster the development of IoT applications ranging over smart healthcare, smart agriculture, smart transportation, factory of the future and so forth.

Nowadays, IoT environments are characterized by the presence of a large number of heterogeneous and resource constrained devices often massively deployed in an area of interest to enable an IoT application. Moreover, IoT networks have experienced the development and standardization of a wide range of protocols in order to enable a wide range of IoT applications. This includes wireless communication technologies (e.g. Zigbee, BLE, LoraWAN and Sigfox (Palattella et al., 2016)), lightweight network management protocols (e.g. LWM2M (Klas et al., 2014), CoMI (Veillette et al., 2017)), communication protocols for resource constrained devices (e.g. 6LowPAN (Hui et al., 2010)), routing protocols for resource constrained devices (e.g. RPL (Vasseur et al., 2011)). However, because of their constraints (e.g. heterogeneity, resource limitations, etc.), IoT networks are experiencing many problems that affect their performance. These problems include: link quality deterioration, network congestion, failure of devices, and contribute to a significant reduction of the performance of IoT networks. In this context, it is therefore important to perform an efficient management of IoT networks in order to ensure good network performances (e.g. low end-to-end delay, energy efficiency and so forth).

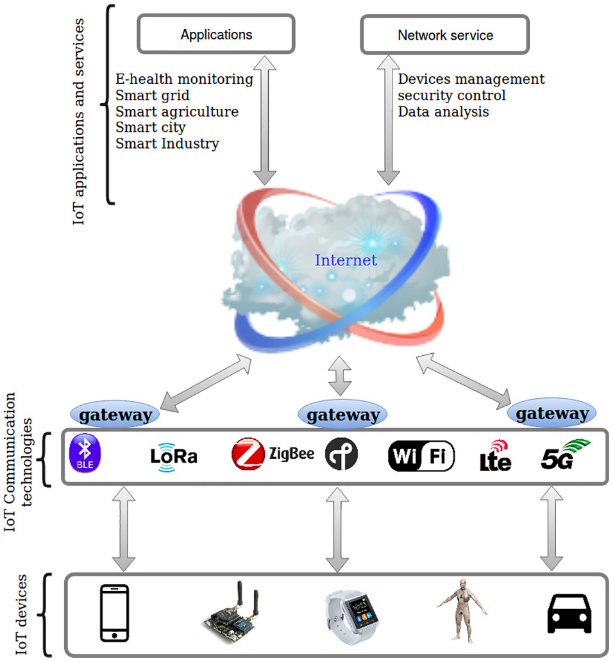

Fig. 1. IoT network architecture.

Basically, IoT networks management enables functionalities such as authenticating, provisioning, configuring, monitoring, routing, and device software management (e.g. firmware update, bug fix, and so forth). These functionalities allow to maintain good network performances and they are generally provided in an IoT environment as a network service as shown in Fig. 1. In the literature, different papers have investigated on IoT networks management solutions from different perspectives (Sheng et al., 2015; Younis et al., 2014; Paradis and Han, 2007; Alamri et al., 2013; Bizanis and Kuipers, 2016; Ndiaye et al., 2017; Haque and Abu-Ghazaleh, 2016; Thoma et al., 2014; Wang et al., 2017; Mao et al., 2018; Sinche et al., 2019).

Table 1 shows a set of solutions for IoT network management covered by this survey and the related surveys. These solutions for IoT network management include: IoT networks management protocols, Cloud based frameworks, SDN based frameworks, Semantic based frameworks and Machine learning based frameworks. Interestingly, none of the literature surveys on IoT network management encompass a large view on different existing solutions for resource-constrained networks. Therefore, we present in this paper an exhaustive literature review on IoT low power network management, while identifying and discussing the limitations of existing solutions and emphasizing research challenges.

Our contribution in this paper can be summarized as follows:

- Compared to existing literature survey on IoT networks management, we provide an exhaustive overview on existing solutions for the management of IoT low power networks.
- We present a classification of existing approaches for management of IoT low power networks based on their objectives.
- We define the requirements for an efficient management of IoT low power networks.
- We propose a detailed review of the literature on IoT low power networks management solutions and a comparative analysis of those solutions based on different requirements.
- Based on the prior-art, we identified a number of challenges and open issues relating to management of IoT low power networks.

The remainder of this paper is organized as follows. In Section 2, we provide an overview on IoT network management and a classification of existing solutions for resource-constrained networks. In addition, we define the requirements for an efficient management of IoT low power networks. In Section 3, we present the review of the state of the art proposals along the taxonomy of IoT low power networks management. Moreover, we provide a comparison of those solutions based on different requirements. In Section 4, we present challenges and open research issues on IoT low power network management. The conclusion and research directions for future work are presented in Section 5.

Table 1
Comparison of this survey to other related survey papers on IoT network management.

IoT network management solutions	This paper	(Sinche et al., 2019)	(Sheng et al., 2015)	(Younis et al., 2014; Paradis and Han, 2007)	(Alamri et al., 2013)	(Ndiaye et al., 2017; Haque and Abu-Ghazaleh, 2016)	(Thoma et al., 2014)	(Wang et al., 2017; Mao et al., 2018)
IoT network management protocols	√	√	√	√	–	–	–	–
Cloud-based frameworks	√	√	√	–	√	–	–	–
SDN-based frameworks	√	–	–	–	–	√	–	–
Semantic-based frameworks	√	–	–	–	–	–	√	–
Machine learning based frameworks	√	–	–	–	–	–	–	√

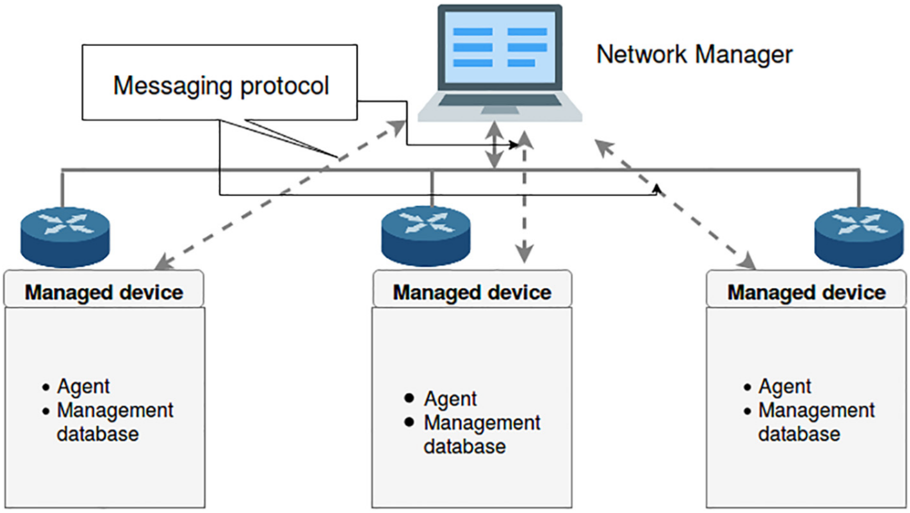

Fig. 2. Network management entities overview.

2. Overview of IoT network management

2.1. Traditional network management

Network management consists in performing operations such as devices monitoring, routing management and security management in order to ensure a good network performance (e.g. low latency, low energy consumption, low packet loss, etc.). Basically, a typical network management is based on three logical elements: network manager, managed devices and agents. Fig. 2 gives an overview of the different functional elements involved in network management. The "network manager" represents the device used to manage a group of managed nodes. A "Managed device" refers to a network device exposing a number of parameters (e.g. IP address, CPU usage, residual battery, etc.) that are managed (through read/write operations) by the network manager. The "agent" refers to the software which runs on managed device. It collects raw data from the managed device to transfer it, in a comprehensible or exploitable format, to the network manager. The "management database" contains information concerning the managed device parameters. The "messaging protocols" can be used to exchange information between the network manager and the managed devices. This allows the network manager to get parameters from managed devices and accordingly take appropriate decision concerning the reconfiguration of network devices.

Typically, a network management system needs to support the following operations:

- **Network configuration management:** It refers to the process that helps setting a network in order to meet a desired objective (e.g. level of security, low latency, etc.). It encompasses different operations related to the configuration and reconfiguration of all the (writable) network device parameters.
- **Topology management:** It corresponds to a set of operations that help to maintain the network connectivity while providing good network performance.
- **Security management:** This operation prevents unauthorized access to an intruder. For this purpose, it includes a wide range of operations such as encryption (key distribution techniques), threat detection and recovery.
- **QoS management:** It refers to a mechanism that helps to configure the network so as to obtain a desirable network performance in term of data latency, packet loss, throughput.
- **Fault management:** It corresponds to a mechanism that helps detecting, isolating and resolving network problems without affecting the proper functioning of the network.
- **Network maintenance:** It refers to a set of operations to perform in order to maintain the network running. It encompasses operations such as software maintenance (e.g. firmware update and bug fixes) and troubleshooting network problems.

To manage traditional networks, various network management protocol such as SNMP (Mauro and Schmidt, 2005), CMIP (Hunt, 1997), NETCONF (Enns et al., 2011), RESTCONF (Watsen and Protocol, 2016), CWMP (Rachidi and Karmouch, 2011) and OMA-DM (Alliance, 2010) have been proposed.

- ***Simple Network Management Protocol (SNMP)***
  SNMP is a network protocol developed by IETF (Internet Engineering Task Force) for remote monitoring of IP devices. It supports a set of operations including monitoring, configuring and/or reconfiguring network device parameters. SNMP involves the three elements of network devices management (agents, nodes and manager) described above. It relies on Structure of Management Information (SMI) and Management Information Base (MIB). MIB designates the database used for managing network devices while SMI defines the structure and types of objects stored in the MIB.
- ***Common Management Information Protocol (CMIP)***
  CMIP is a network protocol responsible for the communication between the network manager and the managed devices. CMIP enables various network management operations such as fault management, security management, performance monitoring and so forth. CMIP was designed to be used on Open Systems Interconnection (OSI) and it extends the capability of SNMP. Nevertheless CMIP has not been widely adopted because of slowness in the process of its standardization.
- ***Network Configuration Protocol (NETCONF)***
  NETCONF has been introduced to improve SNMP. It introduces new features in network management such as multiple configuration data stores (candidate, running, startup), distinction between configuration and state data. NETCONF uses Extensible Markup Language (XML) based data encoding for both the configuration data and the protocol messages. NETCONF uses the YANG model which is a data modeling language used to model configuration and state data manipulated by the NETCONF protocol (Bjorklund, 2010).
- ***RESTCONF***
  RESTCONF protocol has been designed with the goal of extending NETCONF protocol in order to enable possibility of performing network management operations through web applications. Concretely, RESTCONF provides a way to perform CRUD (Create, Retrieve, Update, Delete) operations through execution of HTTP methods to access to configuration data defined in YANG, using the datastore concepts defined in NETCONF.
- ***CPE WAN Management Protocol (CWMP)***
  CWMP is a protocol defined by Broadband Forum in TR-069 Technical report in order to remotly manage customer-premises equipment (CPE) connected to an Internet Protocol (IP) network. This protocol allows performing tasks such as auto-configuration, software or firmware image management, software module management, status and performance management, and diagnostics.
- ***OMA-DM***
  OMA-DM is a secure device management protocol specified by the Open Mobile Alliance (OMA) Device Management (DM) Working Group and the Data Synchronization (DS) Working Group. It enables performing management tasks such as device provisionning, device configuration, software upgrade and fault management.

Nevertheless, the above network management protocols were designed before the emergence of IoT paradigm, and it was rather obvious that those protocols did not consider a number of IoT characteristics and constraints (e.g. devices resource constraints) that raise technical barriers for their applicability in the IoT environment.

2.2. Requirements of IoT low power network management

Designing an IoT network management solution is not an easy task because of the intrinsic constraints of IoT networks such as devices heterogeneity, variable network topologies, scarce resources and variable/unreliable radio link quality. Thereby, in order to operate under a good performance, IoT networks need to satisfy requirements as: scalability, fault tolerance, energy efficiency, Quality of Service (QoS) and security (Ersue, 2015; Ndiaye et al., 2017).

2.2.1. Scalability

A scalable IoT low power network corresponds to a network where new devices or services can be added without negatively affecting the network performance. As the current deployment of IoT low power networks low power is characterized by the presence of billion of resource-constrained devices, the scalability requirement need to be satisfy in order to avoid having poor network performance.

2.2.2. Fault tolerance

Fault tolerance is the ability of a system to continue operating in the event of failure of any of its components (Chouikhi et al., 2015). This requirement is necessary in order to guarantee that the network will fulfill its expected functioning in presence of fault (e.g. node fault, network fault, sink fault, software fault) in the network. In particular, this requirement is important for IoT low power networks since they may be subject to failure of devices because of their characteristics (limited battery, memory and CPU) and/or the environment in which they are deployed (e.g. war zone, pipeline, chemical spill area).

2.2.3. Quality of Service (QoS)

QoS refers to the measurement of overall performance of service in order to assess user satisfaction. This performance is evaluated using these metrics: packet loss, latency, bandwidth and end-to-end delay in the network. Concretely, the level of QoS in IoT low power networks depends on the type of application. For example, IoT applications such as smart metering are delay tolerant while another IoT applications like forest fire detection are not. Therefore, to avoid having poor network performance, it is important to consider the QoS requirement when designing the network.

2.2.4. Energy efficiency

One of the main requirement of IoT low power is the energy efficiency (Rault et al., 2014). An energy efficient network is a network that has the capability to execute operations with a minimum energy consumption so that the network lifetime can be maximized. This requirement is particularly desirable in an IoT low power network since its composed of devices powered with battery which has a limited lifetime and often cannot be replaced. Moreover, if the energy of the resource constrained devices is consumed quickly, the network may experience a loss of connectivity which may cause an interruption of the network.

2.2.5. Security

Security is an important concerns for IoT networks as reviewed in (Granjal et al., 2015). In fact, having a secure network may help to prevent the potential risks for tampering the communication data by unauthorized entity. As a result, a secure IoT network help to guarantee the security of data exchanged by the different devices involved in the network. Nevertheless, in IoT low power networks, more attention should be paid because the mechanisms for security developed for traditional network are not always suitable for resource-constrained devices (Kouicem et al., 2018).

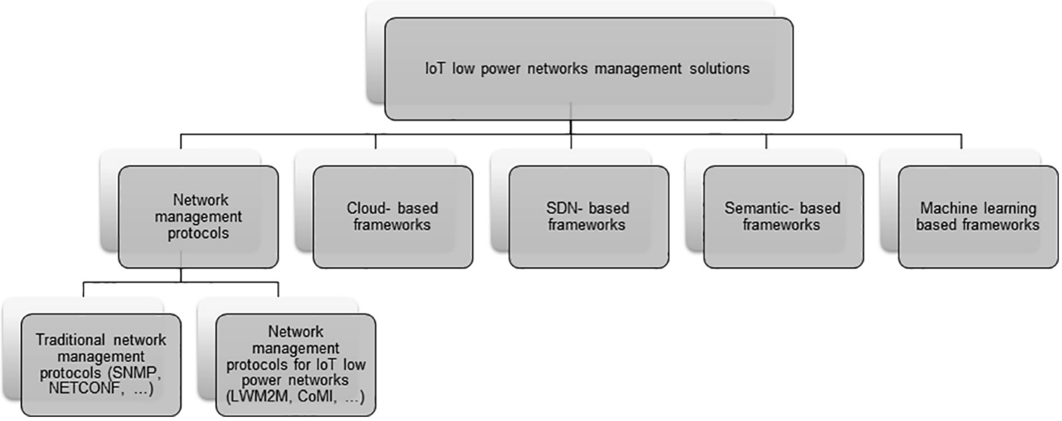
IoT low power networks management solutions
Network management protocols
Cloud- based frameworks
SDN- based frameworks
Semantic- based frameworks
Machine learning based frameworks
Traditional network management protocols (SNMP, NETCONF, …)
Network management protocols for IoT low power networks (LWM2M, CoMI, …)

Fig. 3. Classification of solutions for management of IoT low power networks.

2.2.6. Self configuration

This requirement refers to the capability of IoT low devices to adapt their behavior according to the network state. In fact, self configuration is important for IoT low power networks since these networks are subject to frequent update caused by the traffic patterns, the mobility of devices, the failure of devices and so forth. Moreover, this requirement is necessary because it is not realistic to perform manual configuration of billion of IoT low power devices in a dynamic network. Thereby, having a self configurable IoT low power network can help to avoid human error due to manual configuration, and thus ensure a good network performance.

2.3. Classification of IoT low power networks management solutions

In order to fulfill the above requirements of IoT low power networks, different network management solutions for resource-constrained devices have been proposed in the literature. Based on their design objectives, these network management solutions can be classified into several categories (cf. Fig. 3), notably network management protocols for IoT low power networks, SDN-based frameworks, Cloud-based frameworks, Semantic-based frameworks and machine learning based frameworks.

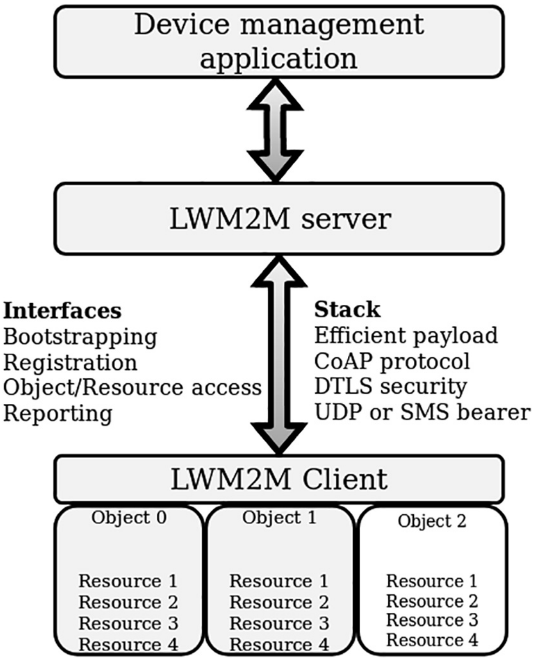

Fig. 4. LWM2M architecture.

Network management protocols for IoT low power networks have been designed in order enable and optimization of the network performance while using small resources for network management operations. Cloud based frameworks for IoT networks management have been proposed in order to cope with the issue of limited resources of IoT devices by enabling management of those connected devices through cloud platforms. SDN-based frameworks for IoT networks management have been proposed in order to centralize network management operations on a central entity and so, reduce computational operations on IoT devices. Semantic-based frameworks have been proposed in order to leverage the knowledge generated by data collected from the network devices. Machine-learning based frameworks have been proposed in order to cope with the increasing complexity of IoT networks (caused by nodes mobility, dynamic nature of the network traffic and so forth).

It is worth to note that above approaches for IoT networks management are often associated (Huang et al., 2015; Corici et al., 2015) in order to satisfy requirements of IoT networks management mentioned in Section 2.2.

3. Management of IoT low power networks

In this section, we present the state of the art on IoT low power network management according to the classification presented in Section 2.3.

3.1. Network management protocols for IoT low power networks

In the literature, different network management protocols have been proposed in order to remotely manage resource-constrained devices. These protocols include: LWM2M, CoMI, NETCONF light and 6LowPAN-SNMP.

- ***LWM2M***
  LWM2M is a client-server protocol developed for the management of IoT low power devices. This protocol has been designed by Open Mobile Alliance (OMA) and is based on protocol and security standards from the IETF. It provides several features such as connectivity monitoring, resources monitoring, firmware upgrade. In Fig. 4, we depict a high-level view of LWM2M architecture. LWM2M server is located at the network manager device and LWM2M client are typically located on managed devices. IoT device resources are organized into objects (e.g.

Table 2
Messaging protocols used in IoT networks.

Protocol	Suitability for Constrained devices	Messaging type	Architecture	QoS	QoS Level	Interoperability
XMPP (Saint-Andre, 2011)	+	Request/ Response Publish/subscribe	Client/server	No	-	Yes
MQTT-SN (Stanford-Clark and Truong, 2013)	++	Request/ Response Publish/subscribe	Client/Broker	Yes	QoS 0 (fire and forget) QoS 1 (delivered at least once) QoS 2 (delivery exactly once)	-
DDS (Hakiri et al., 2015)	+	Publish/ subscribe	Brokerless	Yes	23 levels of QoS	Yes
CoAP(Sheng et al., 2015)	+++	Publish/ subscribe Request/Response	Client/server	Yes	Confirmable message Non confirmable message	Yes
MQTT(Naik, 2017)	++	Request/ Response Publish/subscribe	Client / broker	Yes	QoS 0 (fire and forget) QoS 1 (delivered at least once) QoS 2 (delivery exactly once)	partial
AMQP (Bhimani and Panchal, 2018)	+	Publish/ subscribe	Client/Server	Yes	At-most-once At least once Exactly once	Yes

+++ Excellent, ++ Fair, + Poor

Table 3
A comparison of IoT low power networks management protocols.

Network management protocol	Scalability	Fault tolerance	Energy efficient	QoS	Security	Self configuration
6LowPAN-SNMP (Choi et al., 2009)	–	–	√	–	√	–
NETCONF light (Schoenwaelder et al., 2012)	–	–	√	–	√	–
LWM2M (Rao et al., 2015)	–	–	√	–	√	–
CoMI (Veillette et al., 2020)	–	–	√	–	√	–

√ The requirement can be handle by the Network management protocol.

location object which contains all resources that provide information about the location of IoT devices). A description of the implementation of that protocol is provided in (Rao et al., 2015).

- ***CoAP Management Interface(CoMI)***
  CoMI is a management interface dedicated for IoT low power devices and networks. This network management protocol enables performing management operations on IoT device resources specified in YANG, or SMIv2 converted to YANG by accessing them through the CoAP protocol. The specification of that protocol is given in (Veillette et al., 2020).
- ***6LowPAN-SNMP***
  6LowPAN-SNMP is an adaptation of SNMP for IPv6 Low-Power Wireless Personal Area Network (6LowPAN) (Choi et al., 2009). It has been designed to work in resource constrained networks and offers the possibility to perform SNMP operations over IPv6 Low-Power Wireless Personal Area Networks. To achieve that, a mechanism of compression of SNMP header is performed in order to reduce the number of SNMP messages generated. The compatibility with standard SNMP is ensured by using a proxy forwarder that helps to convert SNMP messages into 6LowPAN-SNMP messages.
- ***NETCONF light***
  NETCONF light is a network management protocol developed by IETF (Schoenwaelder et al., 2012) in order to extend NETCONF to enable the management of resource-constrained devices. It provides tools to install, manipulate, and delete the configuration of network devices by using only a set of NETCONF operations.

It is worth mentioning that network management protocols are often associated with messaging protocols in order to enable the management of resource-constrained devices (Lindholm-Ventola and Silverajan, 2014; Scheffler and Bonneß, 2017). These messaging protocols include: CoAP (Constrained Application Protocol) (Sheng et al., 2015), XMPP (Extensible Messaging and Presence Protocol) (Saint-Andre, 2011), DDS (Data-Distribution Service for Real-Time Systems) (Hakiri et al., 2015), MQTT (Message Queuing Telemetry Transport) (Naik, 2017), MQTT-SN (MQTT for Sensor Networks) (Stanford-Clark and Truong, 2013) and AMQP (Advanced Message Queuing Protocol) (Bhimani and Panchal, 2018). We provide a comparison of those messaging protocols in Table 2.

In Table 3, we summarized a comparison of different protocols for management of resource-constrained networks according to the requirement formulated in Section 2.2. Nevertheless, these network management protocols are not able to satisfy all the requirements of IoT low power networks mentioned earlier. To achieve that, it is necessary to associate those protocols with other mechanisms to fulfill the requirement such as self configuration and scalability. In the next section we will discuss about those mechanisms.

3.2. Cloud based frameworks for management of IoT low power networks

Cloud computing refers to a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort, often over the Internet (Mell et al., 2011). These services are provided through cloud platforms.

Due to the resources limitation of IoT low power networks, several frameworks based on cloud platforms for management of resource-constrained devices have emerged. This has been motivated by the fact that cloud platforms can provide the needed resources to perform various network management operations (e.g. firmware update). Table 4 presents some cloud platforms for

Table 4
Cloud of Things platforms features.

IoT cloud platform	Protocols for data collection	Management protocol	Configuration management	Device Monitoring	Communication technologies	Resource constrained devices
Azure (Microsoft, 2017)	MQTT	LWM2M	√	√	–	√
IBM IoT (IBM, 2017)	MQTT, HTTP	LWM2M	√	–	–	√
Artik Cloud (SAMSUNG, 2017)	REST/HTTP, websockets, MQTT, CoAP	LWM2M	√	–	–	√
Mbed (Arm, 2018)	HTTP, HTTPS, CoAP, MQTT	LWM2M	√	–	BLE, Thread, 6LowPAN, Wi-Fi, LoRa	√
Arkessa (Arkessa, 2018)	–	–	√	√	–	–
Thethings.io (thethings.io, 2018a)	HTTP, CoAP, MQTT, WebSockets	–	√	√	Sigfox, Wi-Fi, LoRa, GSM	√
Arrayent (Systems, 2018)	–	–	√	–	–	–
ThingWorx (ThingWorx, 2018)	–	–	√	√	–	–
Carriots (Carriots, 2018)	HTTP, MQTT	–	√	√	Zigbee, Z-Wave, Bluetooth, WiFi, Ethernet	–
Echelon (ECHELON, 2018)	–	–	√	–		–
KAA (KAA, 2018)	MQTT, CoAP, XMPP, TCP, HTTP	–	√	√	Z-Wave, Zigbee, LoRa, Bluetooth, WiFi, 6LoWPAN	√
Ayla IoT Platform (Networks, 2018)	–	–	√	–	Wi-Fi, Ethernet, Zigbee	√
Thinger.io (thinger.io, 2018b)	MQTT, CoAP and HTTP	–	√		Sigfox	√
SiteWhere (SiteWhere, 2018)	MQTT, AMQP, Stomp, WebSockets, and direct socket connection	–	√	–	–	–
Google cloud plateform (Cloud, 2020)	MQTT	–	√	√	–	√
Autodesk Fusion Connect (Autodesk, 2020)	CoAP, HTTP, XMPP, DDS, MQTT	LWM2M	√	√	–	–
Amazon Website Site (AWS) IoT (Amazon, 2020)	MQTT	–	√	√	–	√

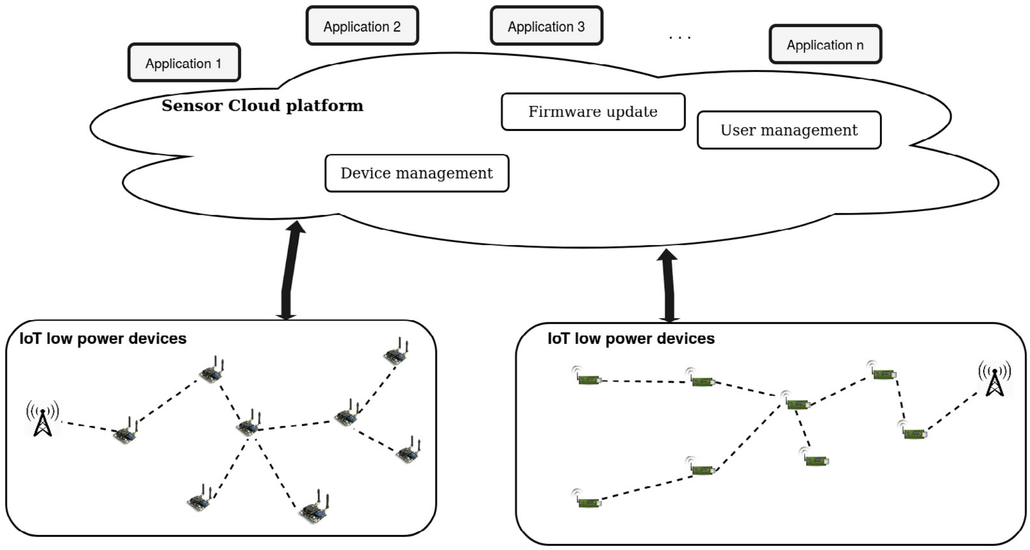

Fig. 5. Example of architecture for management of IoT devices over a sensor cloud infrastructure.

management of IoT low power networks present in the IoT market. Generally, the architecture for management of IoT low power networks over a cloud plateform consists in three levels: 1) the first level is composed of resource-constrained devices, 2) the second level is composed by cloud infrastructure and 3) the third level is composed of IoT applications. We provide an example of such architecture in Fig. 5.

Yuriyama and Kushida (2010) proposed a management solution for sensor networks based on a cloud infrastructure. In the proposed solution, physical sensor devices are virtualized in order to enable the management of heterogeneous resource-constrained devices over a cloud platform infrastructure. Likewise, Xu and Helal (2015) proposed an architecture for management of IoT devices called Cloud-Edge-Beneath (CEB). This proposal leverage the benefits of cloud platforms in order to provide a management solution for large-scale and dynamic IoT networks. Similarly, Ojha et al. (2014) proposed a solution for management of wireless sensor networks based on a cloud platform. The proposed solution enables dynamic scheduling of duty in order to extend the network lifetime. Along the same lines, Kim et al. (2014) proposed a routing scheme called H-SMSR (hierarchical scalable multipath source routing) in the context of IoT low power devices managed over a cloud platform called Agriculture Sensor-Cloud Infrastructure (ASCI). The proposed routing protocol includes hierarchical source routing (HSR) and aggregation gradient routing (AGR) in order to increase the network lifetime. Das et al. (2017) proposed an energy

Table 5
A comparison of network management frameworks based on Cloud.

Network management framework	Scalability	Fault tolerance	Energy efficient	QoS	Security	Self configuration
Yuriyama and Kushida (2010)	–	–	√	–	–	√
Suciu et al. (2013)	√	–	√	–	√	√
Ojha et al. (2014)	–	–	–	√	–	√
Kim et al. (2014)	√	–	√	–	–	√
Xu and Helal (2015)	√	–	√	–	–	√
Das et al. (2017)	–	–	√	–	–	–

√ The requirement can be handle by the network management framework.

efficient model for the management of IoT low power devices over a cloud platform. This solution includes a predictive model that helps to reduce the network transmission overhead. Suciu et al. (2013) proposed a framework based on a cloud platform to enable management of IoT low power devices in the context smart cities. The proposed framework allows an improvement of the network traffic quality through autonomic management.

The above frameworks for management of IoT low power networks exhibit different functionalities. We summarized those frameworks in Table 5 and evaluated them against the requirements of IoT low power networks formulated in Section 2.2. From the Table 5, we see that none of the existing solutions fulfill all the requirements of IoT low power networks. Therefore, additional mechanisms may be required in order to meet requirements of IoT low power networks.

3.3. SDN based frameworks for management of IoT low power networks

Over the last decade, the number of resource-constrained devices present in the IoT ecosystem has increased dramatically. These devices are often running many events which may imper on the network performance. To cope with this issue, Software Defined Network (SDN) has been used in order to achieve energy efficient management of IoT low power networks (Ndiaye et al., 2017). According to Kim and Feamster (2013), SDN is a paradigm where a central software program called a controller dictate the overall network behavior. SDN advocates separating control plane of the network (where decisions about how packets should flow through the network is taken) from the data plane of the network (traffic forwarding plan). Fig. 6 gives an overview of an SDN architecture. Network devices are considered as simple packet routing devices (data plan) and the control logic is implemented at the controller (control plane). Southbound interface is the relay between programmable switches and the software controller. Several southbound API has been proposed in literature (Sezer et al., 2013), notably: OpenFlow, ForCES, PCEP, etc. Openflow is considered as the most common southbound SDN interfaces (Lara et al., 2013). Openflow exists in several software releases (Shalimov et al., 2013): NOX, POX, Beacon, Floodlight, MuL, Maestro, Ryu, etc. Northbound interface enables communication among the higher-level components. In fact, northbound interface allows exchange of information between network and application running on top of it.

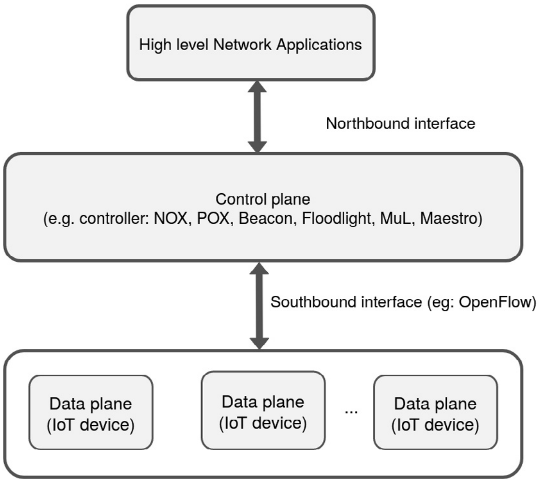
High level Network Applications
Northbound interface
Control plane
(e.g. controller: NOX, POX, Beacon, Floodlight, MuL, Maestro)
Southbound interface (eg: OpenFlow)
Data plane
(IoT device)
Data plane
(IoT device)
...
Data plane
(IoT device)

Fig. 6. SDN architecture.

De Gante et al. (2014) proposed a centralized architecture for the management of wireless sensor networks. The proposed architecture leverages the benefits of SDN paradigm, notably it allows prolonging the network lifetime. In the same vein, Costanzo et al. (2012) and Jacobsson and Orfanidis (2015) proposed network management solutions for resource-constrained devices based on SDN. Moreover, Orfanidis (2016) proposed an architecture for management of IoT low power networks based on SDN with a machine learning model. Similarly, Bera et al. (2016) proposed a centralized network management scheme called software-defined wireless sensor network architecture (Soft-WSN) in order to configure IoT low power networks according to policies defined by the network management entity. Huang et al. (2015) proposed a framework for management of IoT low power networks based on SDN and reinforcement learning. The proposed framework enables reduction of the overhead of control traffic by filtering redundancy and performing a load-balancing routing mechanism according to data flows with the required QoS. Additionally, Wu et al. (2016) proposed a framework based on SDN to mitigate security attack in wireless sensor networks. The proposed framework enable dynamic reaction against unknown attacks. However, since these solutions are centralized, in a large network, they may suffer from scalability problem. To cope with this issue, Olivier et al. (2015) proposed an architecture called software-defined clustered sensor networks (SDCSN). The proposed network management framework uses clustering technique to organize the network in clusters where each cluster head plays the role of the controller. In the same line, De Oliveira et al. (2015) proposed an implementation of a scalable framework for management of wireless sensor networks based multiple SDN controller.

We provide a comparison of the above frameworks in Table 6 according to the requirement formulated in Section 2.2. It is worth mentioning that SDN needs to be associated with another mechanisms such as machine learning in order to fulfill the requirements of IoT low power networks (Matlou and Abu-Mahfouz, 2017).

3.4. Semantic based frameworks for management of IoT low power networks

The presence of billion of heterogeneous and resource-constrained devices in IoT environment raises the need for

Table 6
A comparison of network management frameworks based on SDN.

Reference	Scalability	Fault tolerance	Energy efficient	QoS	Security	Self configuration
Costanzo et al. (2012)	–	–	√	–	–	–
De Gante et al. (2014)	–	–	√	–	–	–
Jacobsson and Orfanidis (2015)	–	–	√	–	–	–
Olivier et al. (2015)	√	√	√	√	√	–
De Oliveira et al. (2015)	√	–	√	–	–	–
Huang et al. (2015)	–	√	√	√	–	√
Wu et al. (2016)	–	–	√	–	√	√
Bera et al. (2016)	–	√	√	–	–	√
Orfanidis (2016)	–	√	–	–	–	√

√ The requirement can be handle by the network management framework.

handling heterogeneity of devices management solutions. For this purpose, semantic technology has been used to cope with IoT devices heterogeneity while ensuring good network performance.

Katasonov et al. (2008) present a middleware for self management of heterogeneous IoT devices. This middleware is based on agent technologies and it enables interoperability by using semantic technologies. Vlacheas et al. (2013) proposed a framework for management of IoT devices deployed in context of smart cities applications. The proposal is based on the concept of cognition and proximity and provides mechanisms to face heterogeneity of connected things. In the same vein, Ismail et al. (2018) proposed a framework based on semantic technology in order to enable management of IoT devices. The proposed framework ease automatic management of IoT devices by using ontology to enable management of heterogeneous network devices. Likewise, Sahlmann et al. (2017) proposed a framework based on the oneM2M ontology (a structured vocabulary that describes a certain domain of interest) in order to ease automatic configuration of heterogeneous IoT devices. The proposed solution uses NETCONF and MQTT protocols for management of resource-constrained devices. Further, Datta et al. (2015) proposed a framework based on semantic technologies to enable management of heterogeneous IoT devices. The proposed framework includes automatic discovery of the mobile devices, provisioning of sensors and IoT domains, semantic reasoning on sensor data and actuation based on the suggestions. The authors claim that their proposal can help to efficiently use the resources of IoT devices. In the same vein, Aissaoui et al. (2020) proposed an extension of SAREF ontology in order to manage heterogeneous IoT devices. The proposed model enables cross-system data interoperability and knowledge enrichment through reasoning.

Based on this state of the art on management of IoT low power networks based on semantic, we observed that existing solutions focused on enabling automatic management of heterogeneous IoT devices. Therefore, in order to meet the requirement of IoT low power networks formulated in Section 2.2, those solutions should be enhanced. We summarize in Table 7 a comparison of these solutions according to different requirements.

3.5. Machine learning based frameworks for management of IoT low power networks

Nowadays, IoT networks generate a huge amount of data due to the dynamic nature of these networks and/or the number of resource-constrained devices. In order to leverage the benefits of those data, machine learning techniques have been used in order to help in taking decision of network management (Aboubakar, 2020a,b; Alsheikh et al., 2014; Kumar et al., 2019; Wang et al., 2017). Machine Learning (ML) refers to the process that gives a computer the ability to mimic the human brain in order to perform complex tasks based on their knowledge. It has been useful for IoT network management because it provides predictive mechanisms that help taking decision such as routing table reconfiguration, network scheduling, parameter adaptation according to the current states of the network. In general, ML techniques can be divided into supervised learning, unsupervised learning and reinforcement learning. Supervised Learning is a ML method which provides a way to predict the outcome of unseen values by using classification or regression with pre-labelled data. It is based on two steps namely training (phase which involves dataset training and designing of classification model) and testing (which involves classification of unseen value). The common supervised learning algorithms used for IoT network management include: Support Vector Machine (SVM), regression tree, neural network, Convolutional neural network (CNN), Deep Belief Network (DBN) and Recurrent Neural Network (RNN). Unlike supervised learning, unsupervised learning is not based on pre-labeled. It uses instead unlabeled dataset to perform classification of data into cluster by discovering common pattern within those unlabeled dataset. The common unsupervised learning algorithms used for network management include: K-MEANS, Autoclass, Deep Belief Network and Deep Boltzmann machine. Reinforcement learning is another approach of ML that enables to find the ideal behavior in particular context by machines and software agent in order to maximize performance. Basically, the reinforcement learning is described as a Markov Decision Process (MDP). Fig. 7 shows a high level overview of a reinforcement learning model. The agent can visit a set of finite

Table 7
A comparison of network management frameworks based on Semantic.

Network management framework	Scalability	Fault tolerance	Energy efficient	QoS	Security	Self configuration
Katasonov et al. (2008)	–	–	–	–	–	√
Vlacheas et al. (2013)	–	–	–	–	–	√
Datta et al. (2015)	–	–	√	–	–	√
Sahlmann et al. (2017)	–	–	–	–	–	√
Ismail et al. (2018)	–	–	–	–	–	√
Aissaoui et al. (2020)	–	–	–	–	–	√

√ The requirement can be handle by the network management framework.

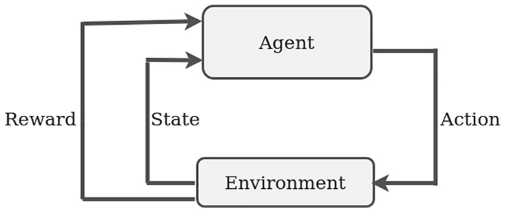

Fig. 7. Reinforcement learning model.

environment states S by performing actions. In visiting a state, a numerical reward will be collected in order to measure the success or failure of an agent's actions in a given state. The common reinforcement algorithms used for IoT network management include: Sarsa, Q-learning and Policy Gradient.

Generally, the usage of ML for solving networking problem is done by following specific steps as shown in Fig. 8 (Wang et al., 2017). These different steps include problem formulation, data collection, data analysis, model construction, model validation, deployment and inference.

In the following, we present some existing solutions for management of IoT low networks based on ML.

1) ***IoT low power networks management solutions based on supervised learning***

Wang et al. (2006) proposed a framework based on decision tree learners, a supervised learning algorithm, in order to predict the link quality in IoT low power networks. The proposed solution aims to optimize the network performance by taking routing decision that helps improving the data delivery rate and data latency. Likewise, Liu and Cerpa (2011) proposed a framework called 4C, to estimate the link quality in IoT low power networks. The proposed framework is based on logistic regression and uses PHY parameters of the last received packets and packet reception rate (PRR) to estimate the link quality. The authors claim that very little data (5–7 links for a couple of minutes) are needed in order to train the models in the environments tested. In the same vein, Feo-Flushing et al. (2014) presented a scheme to perform an online learning using a supervised learning algorithm in order to predict the link quality in a given wireless sensor network. The authors claim that strategies that keep balanced the set of training samples in terms of ranges of target values provide better accuracy and faster convergence. Further, Kaplantzis et al. (2007) proposed a centralized intrusion detection system (IDS) based on Support Vector Machines(SVMs) and sliding windows for wireless sensor networks. The proposed IDS uses only 2 features to detect selective forwarding and black hole attacks.

2) ***IoT low power networks management solutions based on unsupervised learning***

Barbancho et al. (2006) proposed a solution called Intelligent Wireless Sensor Network (IWSN) in order to manage data route by IoT devices. The proposed solution is based on a neural network which allows the selection of the route that optimize the network performance in presence of node failure. Additionally, Moustapha and Selmic (2008) proposed a dynamic model for fault detection in wireless sensor networks. The proposed framework includes neural network modeling approach for sensor node identification and fault detection in the network. Further, Branch et al. (2013) proposed a method for outlier detection method in WSNs using k-nearest neighbors. The authors claim that their proposal is well suited for applications in which the confidence of an outlier rating may be calculated by either an adjustment of sliding window size or by the number of neighbors used in a distance-based outlier detection technique. Another framework for management of IoT low power networks was proposed by Chang et al. (2018). The proposed framework aims at controlling the topology of ultra-dense wireless sensor networks. The proposed framework is based on K-Means, an unsupervised learning algorithm, and it enables an optimization of the network lifetime by balancing energy consumption.

3) ***IoT low power networks management solutions based on reinforcement learning***

Stampa et al. (2017) proposed a framework based on Deep-Reinforcement Learning (DRL) agent for routing optimization. The proposed framework helps to define tailored configurations that minimize the network delay. Additionally, Shah and Kumar (2007) proposed a framework called Distributed Independent Reinforcement Learning (DIRL), for resource/task management in wireless sensor networks. The proposed framework is based on Q-learning and it allows each sensor device to autonomously schedule its tasks and allocate its resources by a learning process. Another framework for management of IoT low power networks based on reinforcement learning was proposed by Mihaylov et al. (2012). The proposed framework enables scheduling the wake-up cycles of nodes in a wireless sensor network according to their interactions with neighbouring nodes. Further, a framework based on reinforcement learning for routing management in wireless sensor has been proposed by Wang and Wang (2006). The proposed framework called AdaR (Adaptive Routing) uses Least Squares Policy Iteration (LSPI) and allows sensor nodes to learn the optimal routing strategy regarding a set of optimization goal. Furthermore, Förster and Murphy (2011) proposed a framework called Feedback ROuting to Multiple Sinks (FROMS), to optimize routing selection in wireless sensor networks. The proposed framework based on reinforcement learning helps to define the optimal multicast routes using different cost metrics (e.g. hops, geographic distance, latency and remaining battery). Moreover, FROMS enables quick recovery in case of failures and sink mobility.

We summarized frameworks for IoT low power networks management based on machine learning in Table 8 and compared them according to the requirement of IoT low power formulated in Sec-

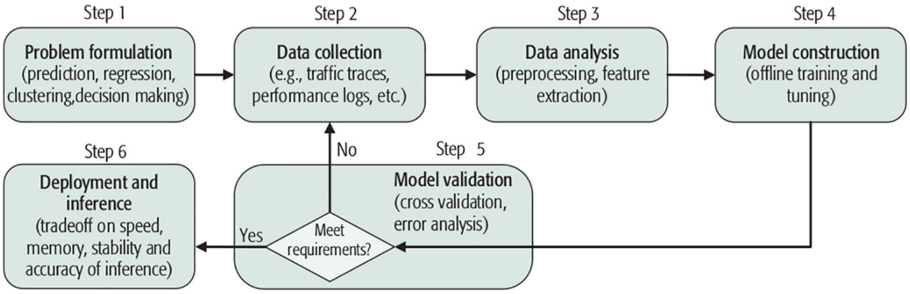
Step 1
Problem formulation
(prediction, regression, clustering,decision making)
Step 2
Data collection
(e.g., traffic traces, performance logs, etc.)
Step 3
Data analysis
(preprocessing, feature extraction)
Step 4
Model construction
(offline training and tuning)
Step 6
Deployment and inference
(tradeoff on speed, memory, stability and accuracy of inference)
No
Step 5
Model validation
(cross validation, error analysis)
Yes
Meet requirements?

Fig. 8. Workflow of machine learning for networking (Wang et al., 2017).

Table 8
A comparison of network management frameworks based on Machine learning.

Network management framework	Scalability	Fault tolerance	Energy efficient	QoS	Security	Self configuration
			Supervised Learning			
Wang et al. (2006)	–	√	–	√	–	√
Kaplantzis et al. (2007)	–	–	–	–	√	√
Liu and Cerpa (2011)	–	√	–	–	–	√
Feo-Flushing et al. (2014)	–	√	–	–	–	√
			Unsupervised Learning			
Barbancho et al. (2006)	–	√	√	√	–	√
Moustapha and Selmic (2008)	–	√	–	–	–	√
Branch et al. (2013)	–	–	–	–	√	√
Chang et al. (2018)	–	–	√	–	–	√
			Reinforcement Learning			
Wang and Wang (2006)	–	–	√	–	–	√
Shah and Kumar (2007)	–	–	–	√	–	√
Förster and Murphy (2011)	√	√	√	–	–	√
Mihaylov et al. (2012)	–	–	√	–	–	√
Stampa et al. (2017)	–	–	–	√	–	√

√ The requirement can be handle by the network management framework.

tion 2.2. From this study, we can see that additional effort are needed to develop efficient solutions in order to meet the requirement of IoT low power networks.

4. Challenges and future research directions

There have been extensive efforts by the research community to propose new network management solutions in order cope with resource constraints of IoT networks and fulfill the requirements formulated in Section 2.2. These solutions include: lightweight network management, cloud based frameworks, SDN based frameworks, Semantic based frameworks and machine learning based frameworks for management of IoT low power networks.

Concerning the network management protocols, LWM2M and CoMI are two main network management protocols for resource-constrained IoT devices. These network management protocols are often associated with other solutions such as cloud platform frameworks (e.g. Azure(Microsoft, 2017), IBM IoT (IBM, 2017), etc.) or SDN based frameworks in order to satisfy the requirement of IoT network management mentioned in section 2.2.

We provided a comparative analysis of existing solutions for management of IoT low power networks in Table 3, Table 5, Table 6, Table 7 and Table 8. Overall, these solutions are still not able to correctly address the requirements of IoT low power networks. As consequence, some research challenges can be identified:

- *Efficient use of resources:* a typical IoT network would comprises an important number of embedded devices with limited resources (battery, memory, and processing resources and so forth) that are running various network services. Having a good performance in a such IoT network is challenging because of resource limitations. In fact, in a resource-constrained IoT network, the self configuration capability is needed in order to deal with the mobility of devices, the failure of devices and so forth. To enable that capability, IoT network management frameworks based on machine learning techniques can be used. However, since machine learning based frameworks generally need significant storage and computational resources, it is important to paid attention on how to develop those frameworks for management of resource-constrained networks while not introducing a computational burden and/or unneccessary network traffic during their operations. This is particularly challenging for distributed management of IoT low power networks. In the future, researchers need to adequately address this issue.
- *Scalability:* due to the exponential increase in terms of number of IoT devices, the scalability represents a critical requirement for IoT networks management solutions. However, most of the solutions for IoT low power networks management reviewed in this work do not address well the scalability. Therefore, future works need to focus on development of scalable solutions for IoT networks management in order to accommodate with the rapid growth of IoT networks. In particular, for time-sensitive IoT applications (e.g. telemedicine), centralized frameworks for IoT low power networks management such as cloud-based frameworks are inefficient. In order to cope with this issue, a tempting idea could be to explore a new technology called fog computing (Atlam et al., 2018). This technology helps to leverage cloud and edge resources in order to accelerate reconfiguration of IoT low power networks (Dastjerdi and Buyya, 2016).
- *Real time network management:* there is a need for high availability of IoT low power networks for some IoT applications (e.g. smart healthcare). To achieve that, it is necessary to perform a realtime network management so as to enable good network performances. However, this task is not straightforward due to resources limitations of IoT low power networks. Moreover, this task may introduce a network traffic overhead due to frequent network management operations. Future works need to provide networks management solutions that help to perform real network management while minimizing the overhead of network management operations.
- *Heterogeneity:* with the emergence of various wireless technologies especially for low power networks, the heterogeneity of communication links has even more increased. Efforts towards IPv6-based, or at least IPv6-centric, IoT communications are progressing thanks to the emergence of various standards to enable interworking among different radio technologies (e.g., ZigBee (Alliance, 2012), BLE (Decuir et al., 2010), NB-IoT (Zayas and Merino, 2017), ISA100.11a (Wang, 2011), etc.). In order to manage these heterogeneous IoT low power networks, cloud-based or SDN-based or semantic-based frameworks can be used with IoT gateways. These gateways enable communications between non-IP and IP-based IoT devices (Zhu et al., 2010; Kim et al., 2015). Nevertheless, IoT gateway does not offer a wide range of QoS guarantees (Al-Fuqaha et al., 2015). Future research works need to focus on how to manage heterogeneous IoT devices while providing a good QoS. This could be achieve by the development of a common standard for IoT management architecture (Sinche et al., 2019).

- *Security and privacy:* security is an important concern in IoT low power networks as shown in the literature (Granjal et al., 2015; Kouicem et al., 2018). For this reason, IoT low power networks management solutions need to support secure process in order to guarantee the protection of sensible data. However, due the inherent characteristic of IoT low power networks (e.g. resource limitations), having a network management solution that help to avoid the leak of sensible data is challenging. Consequently, future research should address this issue.

5. Conclusion

Since IoT low power networks are being exponentially deployed both in public (smart cities, smart buildings, etc.) and private areas (smart homes, smart factories, etc.), network management becomes the cornerstone of IoT low power networks to achieve the best network performance and maintain a high level of network availability. In this survey paper, we discussed the state-of-the-art solutions for IoT low power networks management. We identified some requirements for an efficient management of IoT low power networks and proposed a classification of existing solutions into five categories, notably: network management protocols for IoT low power networks, SDN-based frameworks, Cloud-based frameworks, Semantic-based frameworks and machine learning based frameworks. Moreover, we performed a comparative analysis of existing solutions for management of IoT low power networks based on different requirements.

The shortcomings of existing solutions for IoT low power networks management clearly call for further investigation in order to design efficient solutions for management IoT low power networks so as to support scalability, efficient resource utilization and the capabilty to handle the heterogeneity IoT networks while ensuring security and privacy. We believe that future research works need to investigate on hybrid solutions (solutions that encompass at least two types of approaches for IoT low power networks management mentioned in this paper).

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Aboubakar, M. et al., 2020. Using Machine Learning to Estimate the Optimal Transmission Range for RPL Networks. NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium, 1–5. https://doi.org/10.1109/NOMS47738.2020.9110297.

Aboubakar, M. et al., 2020. An Efficient and Adaptive Configuration of IEEE 802.15.4 MAC for Communication Delay Optimisation. 11th International Conference on Network of the Future (NoF), 1–7. https://doi.org/10.1109/NoF50125.2020.9249218.

Aïssaoui, F., Berlemont, S., Douet, M., Mezghani, E., 2020. A semantic model toward smart iot device management. In: Workshops of the International Conference on Advanced Information Networking and Applications. Springer, pp. 640–650.

Al-Fuqaha, A., Khreishah, A., Guizani, M., Rayes, A., Mohammadi, M., 2015. Toward better horizontal integration among iot services. IEEE Commun. Mag. 53 (9), 72–79.

Alamri, A., Ansari, W.S., Hassan, M.M., Hossain, M.S., Alelaiwi, A., Hossain, M.A., 2013. A survey on sensor-cloud: architecture, applications, and approaches. Int. J. Distrib. Sens. Netw. 9, (2) 917923.

Alliance, O.M., 2010. Oma device management representation protocol. Approved Version 1 (1).

Alliance, Z., 2012. Zigbee specification. URL: https://www.zigbee.org/download/standards-zigbee-specification/.

Alsheikh, M.A., Lin, S., Niyato, D., Tan, H.-P., 2014. Machine learning in wireless sensor networks: algorithms, strategies, and applications. IEEE Commun. Surveys Tutor. 16 (4), 1996–2018.

Amazon, 2020. Aws iot device management. URL: https://aws.amazon.com/iot-device-management/?c=i&sec=srv.

Arkessa, 2018. Arkessa services. URL: http://www.arkessa.com/iot-m2m-connectivity-services/.

Arm, 2018. Mbed iot platform. URL: https://os.mbed.com/users/coisme/notebook/google-cloud-iot-from-mbed-os-device/.

Ashton, K. et al., 2009. That 'internet of things' thing. RFID J. 22 (7), 97–114.

Atlam, H.F., Walters, R.J., Wills, G.B., 2018. Fog computing and the internet of things: a review. Big Data Cogn. Comput. 2 (2), 10.

Autodesk, 2020. Autodesk Fusion Connect. URL: https://www.iotone.com/software/autodesk-fusion-connect/s269.

Barbancho, J., Leon, C., Molina, J., Barbancho, A., 2006. Giving neurons to sensors. qos management in wireless sensors networks. In: 2006 IEEE Conference on Emerging Technologies and Factory Automation. IEEE, pp. 594–597.

Bera, S., Misra, S., Roy, S.K., Obaidat, M.S., 2016. Soft-wsn: Software-defined wsn management system for iot applications. IEEE Syst. J. 12 (3), 2074–2081.

Bhimani, P., Panchal, G., 2018. Message delivery guarantee and status update of clients based on iot-amqp. In: Intelligent Communication and Computational Technologies. Springer, pp. 15–22.

Bizanis, N., Kuipers, F.A., 2016. Sdn and virtualization solutions for the internet of things: a survey. IEEE Access 4, 5591–5606.

Bjorklund, M., 2010. Yang-a data modeling language for the network configuration protocol. RFC6020 [Z]. IETF.

Branch, J.W., Giannella, C., Szymanski, B., Wolff, R., Kargupta, H., 2013. In-network outlier detection in wireless sensor networks. Knowl. Inf. Syst. 34 (1), 23–54.

Carriots, 2018. Carriots – internet of things platform. URL: https://carriots.com/.

Chang, Y., Yuan, X., Li, B., Niyato, D., Al-Dhahir, N., 2018. A joint unsupervised learning and genetic algorithm approach for topology control in energy-efficient ultra-dense wireless sensor networks. IEEE Commun. Lett. 22 (11), 2370–2373.

Choi, H., Kim, N., Cha, H., 2009. 6lowpan-snmp: Simple network management protocol for 6lowpan. In: High Performance Computing and Communications, 2009. HPCC'09. 11th IEEE International Conference on. IEEE, pp. 305–313.

Chouikhi, S., El Korbi, I., Ghamri-Doudane, Y., Saidane, L.A., 2015. A survey on fault tolerance in small and large scale wireless sensor networks. Comput. Commun. 69, 22–37.

Cloud, G., 2020. Présentation technique de l'internet des objets. URL: https://cloud.google.com/solutions/iot-overview#top_of_pa.

Corici, A.A., Shrestha, R., Carella, G., Elmangoush, A., Steinke, R., Magedanz, T., 2015. A solution for provisioning reliable m2m infrastructures using sdn and device management. In: 2015 3rd International Conference on Information and Communication Technology (ICoICT). IEEE, pp. 81–86.

Costanzo, S., Galluccio, L., Morabito, G., Palazzo, S., 2012. Software defined wireless networks: Unbridling sdns. In: 2012 European Workshop on Software Defined Networking. IEEE, pp. 1–6.

Das, K., Das, S., Darji, R. K., Mishra, A., 2017. Energy efficient model for the sensor cloud systems. In: 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT). IEEE, pp. 373–375.

Dastjerdi, A.V., Buyya, R., 2016. Fog computing: Helping the internet of things realize its potential. Computer 49, 112–116.

Datta, S.K., Gyrard, A., Bonnet, C., Boudaoud, K., 2015. onem2m architecture based user centric iot application development. In: 2015 3rd International Conference on Future Internet of Things and Cloud. IEEE, pp. 100–107.

De Gante, A., Aslan, M., Matrawy, A., 2014. Smart wireless sensor network management based on software-defined networking. In: Communications (QBSC), 2014 27th Biennial Symposium on. IEEE, pp. 71–75.

De Oliveira, B.T., Gabriel, L.B., Margi, C.B., 2015. Tinysdn: Enabling multiple controllers for software-defined wireless sensor networks. IEEE Latin Am. Trans. 13 (11), 3690–3696.

Decuir, J. et al., 2010. Bluetooth 4.0: low energy. Cambridge Silicon Radio SR plc 16, Cambridge, UK.

ECHELON, 2018. Izot platform. URL: https://www.echelon.com/izot-platform.

Enns, R., Bjorklund, M., Schoenwaelder, J., Bierman, A., 2011. Network configuration protocol (netconf).

Ersue, M. et al., 2015. Management of networks with constrained devices: problem statement and requirements. IETF, RFC 7547.

Feo-Flushing, E., Kudelski, M., Nagi, J., Gambardella, L.M., Di Caro, G.A., 2014. Link quality estimation—a case study for on-line supervised learning in wireless sensor networks. In: Real-World Wireless Sensor Networks. Springer, pp. 97–101.

Förster, A., Murphy, A.L., 2011. Froms: a failure tolerant and mobility enabled multicast routing paradigm with reinforcement learning for wsns. Ad Hoc Netw. 9 (5), 940–965.

Granjal, J., Monteiro, E., Silva, J.S., 2015. Security for the internet of things: a survey of existing protocols and open research issues. IEEE Commun. Surveys Tutor. 17 (3), 1294–1312.

Hakiri, A., Berthou, P., Gokhale, A., Abdellatif, S., 2015. Publish/subscribe-enabled software defined networking for efficient and scalable iot communications. IEEE Commun. Mag. 53 (9), 48–54.

Haque, I.T., Abu-Ghazaleh, N., 2016. Wireless software defined networking: a survey and taxonomy. IEEE Commun. Surveys Tutor. 18 (4), 2713–2737.

Huang, R., Chu, X., Zhang, J., Hu, Y.H., 2015. Energy-efficient monitoring in software defined wireless sensor networks using reinforcement learning: a prototype. Int. J. Distrib. Sens. Netw. 11, (10) 360428.

Hui, J., Thubert, P., et al., 2010. Compression format for ipv6 datagrams in 6lowpan networks. draft-ietf-6lowpan-hc-13 (work in progress).

Hunt, R., 1997. SNMP, SNMPv2 and CMIP-the technologies for multivendor network management. Comput. Commun. 20 (2), 73–88.

IBM, 2017. Ibm watson iot platform. URL: https://internetofthings.ibmcloud.com/#/.

Ismail, H., Hamza, H., M. Mohamed, S., 2018. Semantic enhancement for network configuration management. pp. 1–5.

Jacobsson, M., Orfanidis, C., 2015. Using software-defined networking principles for wireless sensor networks. In: 11th Swedish National Computer Networking Workshop (SNCNW 2015) Karlstad, May 28-29, 2015.

KAA, 2018. Iot cloud platform the internet of things solutions and applications that set the standard. URL: https://www.kaaproject.org/.

Kaplantzis, S., Shilton, A., Mani, N., Sekercioglu, Y.A., 2007. Detecting selective forwarding attacks in wireless sensor networks using support vector machines. In: 2007 3rd International Conference on Intelligent Sensors, Sensor Networks and Information. IEEE, pp. 335–340.

Katasonov, A., Kaykova, O., Khriyenko, O., Nikitin, S., Terziyan, V.Y., 2008. Smart semantic middleware for the internet of things. ICINCO-ICSO 8, 169–178.

Kim, H., Feamster, N., 2013. Improving network management with software defined networking. IEEE Commun. Mag. 51 (2), 114–119.

Kim, K., Lee, S., Yoo, H., Kim, D., 2014. Agriculture sensor-cloud infrastructure and routing protocol in the physical sensor network layer. Int. J. Distrib. Sens. Netw. 10, (3) 437535.

Kim, S.-M., Choi, H.-S., Rhee, W.-S., 2015. Iot home gateway for auto-configuration and management of mqtt devices. In: Wireless Sensors (ICWiSe), 2015 IEEE Conference on. IEEE, pp. 12–17.

Klas, G., Rodermund, F., Shelby, Z., Akhouri, S., Höller, J., 2014. Lightweight m2m: enabling device management and applications for the internet of things. White Paper, February.

Kouicem, D.E., Bouabdallah, A., Lakhlef, H., 2018. Internet of things security: a top-down survey. Comput. Netw. 141, 199–221.

Kumar, D.P., Amgoth, T., Annavarapu, C.S.R., 2019. Machine learning algorithms for wireless sensor networks: a survey. Inf. Fusion 49, 1–25.

Lara, A., Kolasani, A., Ramamurthy, B., 2013. Network innovation using openflow: a survey. IEEE Commun. Surveys Tutor. 16 (1), 493–512.

Lindholm-Ventola, H., Silverajan, B., 2014. Coap-snmp interworking iot scenarios.

Liu, T., Cerpa, A.E., 2011. Foresee (4c): Wireless link prediction using link features. In: Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks. IEEE, pp. 294–305.

Mao, Q., Hu, F., Hao, Q., 2018. Deep learning for intelligent wireless networks: a comprehensive survey. IEEE Commun. Surveys Tutor. 20 (4), 2595–2621.

Matlou, O.G., Abu-Mahfouz, A.M., 2017. Utilising artificial intelligence in software defined wireless sensor network. In: IECON 2017–43rd Annual Conference of the IEEE Industrial Electronics Society. IEEE, pp. 6131–6136.

Mauro, D., Schmidt, K., 2005. Essential SNMP: Help for System and Network Administrators. O'Reilly Media Inc.

Mell, P., Grance, T., et al., 2011. The nist definition of cloud computing.

Microsoft, 2017. Overview of device management with iot hub. URL: https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-device-management-overview.

Mihaylov, M., Le Borgne, Y.-A., Tuyls, K., Nowé, A., 2012. Decentralised reinforcement learning for energy-efficient scheduling in wireless sensor networks. Int. J. Commun. Networks Distrib. Syst. 9 (3–4), 207–224.

Moustapha, A.I., Selmic, R.R., 2008. Wireless sensor network modeling using modified recurrent neural networks: application to fault detection. IEEE Trans. Instrum. Meas. 57 (5), 981–988.

Naik, N., 2017. Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. In: Systems Engineering Symposium (ISSE), 2017 IEEE International. IEEE, pp. 1–7.

Ndiaye, M., Hancke, G., Abu-Mahfouz, A., 2017. Software defined networking for improved wireless sensor network management: a survey. Sensors 17 (5), 1031.

Networks, A., 2018. IoT Software — IoT Platform — Ayla Networks. URL: https://www.aylanetworks.com/.

Ojha, T., Bera, S., Misra, S., Raghuwanshi, N.S., 2014. Dynamic duty scheduling for green sensor-cloud applications. In: 2014 IEEE 6th International Conference on Cloud Computing Technology and Science. IEEE, pp. 841–846.

Olivier, F., Carlos, G., Florent, N., 2015. Sdn based architecture for clustered wsn. In: 2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. IEEE, pp. 342–347.

Orfanidis, C., 2016. Ph. d. forum abstract: Increasing robustness in wsn using software defined network architecture. In: 2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN). IEEE, pp. 1–2.

Palattella, M.R., Dohler, M., Grieco, A., Rizzo, G., Torsner, J., Engel, T., Ladid, L., 2016. Internet of things in the 5g era: Enablers, architecture, and business models. IEEE J. Sel. Areas Commun. 34 (3), 510–527.

Paradis, L., Han, Q., 2007. A survey of fault management in wireless sensor networks. J. Network Syst. Manage. 15 (2), 171–190.

Rachidi, H., Karmouch, A., 2011. A framework for self-configuring devices using tr-069. In: 2011 International Conference on Multimedia Computing and Systems. IEEE, pp. 1–6.

Rao, S., Chendanda, D., Deshpande, C., Lakkundi, V., 2015. Implementing lwm2m in constrained iot devices. In: 2015 IEEE Conference on Wireless Sensors (ICWiSe). IEEE, pp. 52–57.

Rault, T., Bouabdallah, A., Challal, Y., 2014. Energy efficiency in wireless sensor networks: a top-down survey. Comput. Netw. 67, 104–122.

Sahlmann, K., Scheffler, T., Schnor, B., 2017. Managing iot device capabilities based on onem2m ontology descriptions. In: Proceedings of the 16. GI/ITG KuVS Fachgespräch Sensornetze (Technical Reports).

Saint-Andre, P., 2011. Extensible messaging and presence protocol (xmpp): Address format. Tech. rep., RFC 6122, March.

SAMSUNG, 2017. IoT Cloud Platform — Samsung ARTIK cloud services. URL: https://artik.cloud/.

Scheffler, T., Bonneß, O., 2017. Manage resource-constrained iot devices through dynamically generated and deployed yang models. In: Proceedings of the Applied Networking Research Workshop, ACM, pp. 42–47.

Schoenwaelder, J., Watsen, K., Ersue, M., Perelman, V., 2012. Network configuration protocol light (netconf light). In: Working Draft, IETF Secretariat, Internet-Draft draft-schoenw-netconf-light-01.

Sezer, S., Scott-Hayward, S., Chouhan, P.K., Fraser, B., Lake, D., Finnegan, J., Viljoen, N., Miller, M., Rao, N., 2013. Are we ready for sdn? Implementation challenges for software-defined networks. IEEE Commun. Mag. 51 (7), 36–43.

Shah, K., Kumar, M., 2007. Distributed independent reinforcement learning (dirl) approach to resource management in wireless sensor networks. In: 2007 IEEE International Conference on Mobile Adhoc and Sensor Systems. IEEE, pp. 1–9.

Shalimov, A., Zuikov, D., Zimarina, D., Pashkov, V., Smeliansky, R., 2013. Advanced study of sdn/openflow controllers. In: Proceedings of the 9th central & eastern european software engineering conference in russia. ACM, p. 1.

Sheng, Z., Mahapatra, C., Zhu, C., Leung, V.C., 2015. Recent advances in industrial wireless sensor networks toward efficient management in iot. IEEE access 3, 622–637.

Sheng, Z., Wang, H., Yin, C., Hu, X., Yang, S., Leung, V.C., 2015. Lightweight management of resource-constrained sensor devices in internet of things. IEEE Internet Things J. 2 (5), 402–411.

Sinche, S., Raposo, D., Armando, N., Rodrigues, A., Boavida, F., Pereira, V., Silva, J.S., 2019. A survey of iot management protocols and frameworks. IEEE Commun. Surveys Tutor. 22 (2), 1168–1190.

SiteWhere, 2018. The open platform for the internet of things. URL: http://www.sitewhere.org/.

Stampa, G., Arias, M., Sánchez-Charles, D., Muntés-Mulero, V., Cabellos, A., 2017. A deep-reinforcement learning approach for software-defined networking routing optimization. arXiv preprint arXiv:1709.07080.

Stanford-Clark, A., Truong, H.L., 2013. Mqtt for sensor networks (mqtt-sn) protocol specification. In: International business machines (IBM). Corporation version 1.

Suciu, G., Vulpe, A., Halunga, S., Fratu, O., Todoran, G., Suciu, V., 2013. Smart cities built on resilient cloud computing and secure internet of things. In: Control Systems and Computer Science (CSCS), 2013 19th International Conference on. IEEE, pp. 513–518.

Systems, P., 2018. Prodea's Arrayent IoT Services Platform. URL: https://prodea.com/iot-services-platform/.

thethings.io, 2018. The IoT platform to monitorize your devices. URL: https://thethings.io/.

thinger.io, 2018. Open source iot platform. URL: https://thinger.io/.

ThingWorx, 2018. ThingWorx Industrial IoT Platform. URL: https://www.ptc.com/en/products/iot/thingworx-platform.

Thoma, M., Braun, T., Magerkurth, C., Antonescu, A.-F., 2014. Managing things and services with semantics: a survey. In: Network Operations and Management Symposium (NOMS), 2014 IEEE. IEEE, pp. 1–5.

Vasseur, J., Agarwal, N., Hui, J., Shelby, Z., Bertrand, P., Chauvenet, C., 2011. Rpl: The ip routing protocol designed for low power and lossy networks. Internet Protocol for Smart Objects (IPSO) Alliance 36.

Veillette, M., der Stok, P. V., Pelov, A., Bierman, A., Dec. 2017. CoAP Management Interface. Internet-Draft draft-ietf-core-comi-02, Internet Engineering Task Force, work in Progress. URL: https://datatracker.ietf.org/doc/draft-ietf-core-comi/.

Veillette, M., der Stok, P. V., Pelov, A., Bierman, A., Petrov, I., Mar. 2020. CoAP Management Interface. Internet-Draft draft-ietf-core-comi-09, Internet Engineering Task Force, work in Progress. URL: https://datatracker.ietf.org/doc/html/draft-ietf-core-comi-09.

Vlacheas, P., Giaffreda, R., Stavroulaki, V., Kelaidonis, D., Foteinos, V., Poulios, G., Demestichas, P., Somov, A., Biswas, A.R., Moessner, K., 2013. Enabling smart cities through a cognitive management framework for the internet of things. IEEE Commun. Mag. 51 (6), 102–111.

Wang, G., 2011. Comparison and evaluation of industrial wireless sensor network standards isa100. 11a and wirelesshart.

Wang, M., Cui, Y., Wang, X., Xiao, S., Jiang, J., 2017. Machine learning for networking: Workflow, advances and opportunities. IEEE Network.

Wang, P., Wang, T., 2006. Adaptive routing for sensor networks using reinforcement learning. In: Computer and Information Technology, 2006. CIT'06. The Sixth IEEE International Conference on. IEEE, pp. 219–219.

Wang, Y., Martonosi, M., Peh, L.-S., 2006. A supervised learning approach for routing optimizations in wireless sensor networks. In: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality. ACM, pp. 79–86.

Watsen, K., Protocol, R., 2016. Network working group a. bierman internet-draft yumaworks intended status: Standards track m. bjorklund expires: April 30, 2017 tail-f systems.

Wu, J., Ota, K., Dong, M., Li, C., 2016. A hierarchical security framework for defending against sophisticated attacks on wireless sensor networks in smart cities. IEEE Access 4, 416–424.

Xu, Y., Helal, A., 2015. Scalable cloud–sensor architecture for the internet of things. IEEE Internet Things J. 3 (3), 285–298.

Younis, M., Senturk, I.F., Akkaya, K., Lee, S., Senel, F., 2014. Topology management techniques for tolerating node failures in wireless sensor networks: a survey. Comput. Netw. 58, 254–283.

Yuriyama, M., Kushida, T., 2010. Sensor-cloud infrastructure-physical sensor management with virtualized sensors on cloud computing. In: Network-Based Information Systems (NBiS), 2010 13th International Conference on. IEEE, pp. 1–8.

Zayas, A.D., Merino, P., 2017. The 3g pp nb-iot system architecture for the internet of things. In: Communications Workshops (ICC Workshops), 2017 IEEE International Conference on. IEEE, pp. 277–282.

Zhu, Q., Wang, R., Chen, Q., Liu, Y., Qin, W., 2010. Iot gateway: Bridgingwireless sensor networks into internet of things. In: Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on. IEEE, pp. 347–352.