This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

On Networking of Internet of Things: Explorations and Challenges

Huadong Ma, *Member, IEEE,* Liang Liu, Anfu Zhou, and Dong Zhao

***Abstract*—Internet of Things (IoT), as the trend of future networks, begins to be used in many aspects of daily life. It is of great significance to recognize the networking problem behind developing IoT. In this article, we first analyze and point out the key problem of IoT from the perspective of networking: how to interconnect large-scale heterogeneous network elements and exchange data efficiently. Combining our on-going works, we present some research progresses on three main aspects: the basic model of IoT architecture, the internetworking model, and the sensor-networking mode. Finally, we discuss two remaining challenges in this area.**

***Index Terms*—Internet of Things, architecture, internetworking, sensor-networking, data exchange.**

I. Introduction

Internet of Things (IoT), which is closely related to the Internet, mobile communication networks and wireless sensor networks, has become a hot research topic recently. Based on the traditional information carriers such as the Internect, telecommunication networks and so on, IoT is a network that interconnects ordinary physical objects with the identifiable addresses so that provides intelligent services [1].

As a new type of network, IoT is characterized by the large-scale heterogeneous network elements. According to a Gartner report there will be over 26 billion devices in IoT by 2020 [2]. Compared with the traditional networks, IoT pursues more extensive interconnection and more intensive information collection, so as to provide more comprehensive intelligent service. Along with its explosive scale-growth, the network elements in IoT are becoming more and more heterogeneous. IoT consists of not only strong network elements such as computers, smart phones and laptops, but also more and more weak network elements such as RFID tags, sensors and other emerging small devices. Different from strong network elements, weak network elements in IoT are manufactured based on different technologies and have different consideration in cost. As a result, these network elements usually have much weaker functions than strong network elements, in terms of computing, communication, power, storage, etc, as shown in Fig. 1.

Therefore, the most important problem of IoT is networking, i.e., *how to interconnect large-scale heterogeneous network elements and exchange data efficiently*. Nowadays, there is still no complete theory guiding us to deal with this problem, which seriously constrains the development and application of IoT. Mainly with the support of China's National 973 Program[1] "Basic research on the architecture of Internet of Things", we have worked on this key problem from three main aspects: the basic model of IoT architecture, the internetworking model, and the sensor-networking mode. In this article, we present the main results of our research and then conclude the remaining challenges of IoT networking. Specially, we first build up a functional model, which combines the layering and the object-oriented methods, to describe the architecture of IoT. Then, we explore both the internetworking model and the sensor-networking mode from two different routes, respectively, i.e., address-centric/information-centric models for internetworking and proactive-deploying/crowdsensing modes for sensor-networking.

II. Basic Model of IoT Architecture

Combining the layering and the object-oriented solutions, we build up a functional architecture model for IoT, as shown in Fig. 2, which includes four layers: object sensing and controlling layer, data exchange layer, information integration layer, and application service layer.

***Object sensing and controlling layer*.** The most significant feature of IoT is its close relations with the physical space. IoT acquires the information and controls the behavior of the objects in the physical space. The functional entities of this layer are *sensors* and *controllers*. Sensors[2] transform the physical state into electrical signals and transmit them to the upper layer, and controllers perform specific functional behavior to the physical object by understanding the control information from the upper layer. For sensors, the attributes mainly include ID, type, and electrical phenomenon, and the corresponding functions include signal conversion and data transmission. For controllers, the attributes include ID, control modes, trigger conditions, and control activities, and the functions include data receiving and command execution.

This work was supported by the National Natural Science Foundation of China under Grants No. 61332005 and No. 61133015, the Key Technologies R&D Program of China under Grant No. 2012BAH39F07, the Funds for Creative Research Groups of China under Grant No. 61421061, the National 973 Project of China under Grant No.2011CB302700, and the Beijing Training Project for the Leading Talents in S&T (ljrc201502).

H.-D. Ma, L. Liu, and D. Zhao are with the Beijing Key Lab of Intelligent Telecomm. Software and Multimedia, Beijing University of Posts and Telecomm., Beijing, 100876, China. E-mail: mhd@bupt.edu.cn, liangliu.bupt@gmail.com, dzhao@bupt.edu.cn.

A.-F. Zhou is with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, China. E-mail: zhouanfu@ict.ac.cn.

[1]The National Basic Research Program (also called 973 Program) is China's on-going national keystone basic research program, which objectives are to strengthen the original innovations and to address the important scientific issues concerning economic and social development.

[2]In our model, the entity Sensor includes extensive devices which acquire the sensing data of physical objects by using the technologies of sensors, cameras, RFID, etc.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

	RFID	Sensor	Smartphone	Computer
Quantity (by 2020)	Over 20 billions		~6 billion	~1 billion
Size	~mm to ~cm	~cm	~cm to ~dm	~dm to ~m
Computing	No	Limited	strong	strong
Communication	passive	local	global	global
Sensing	No	Yes, usually with single sensing function	Yes, with rich sensing function	no
Power	Harvested	Battery	Battery and recharged	Power supply
Storage	~KB	~KB to ~MB	~GB	~TB

Fig. 1. Comparison between heterogeneous network elements in IoT.

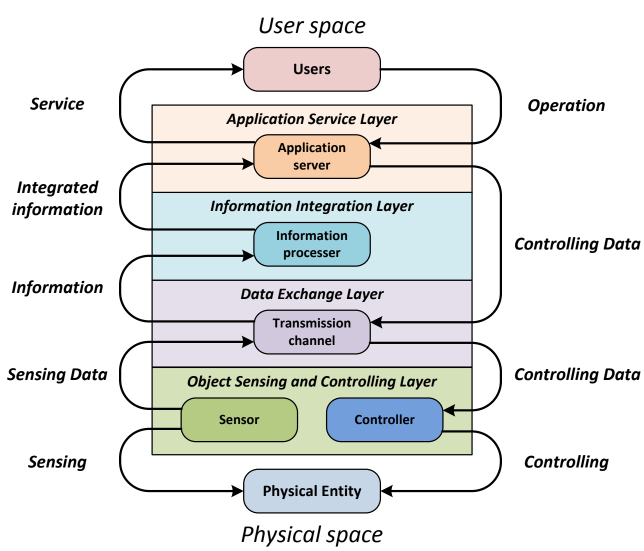


Fig. 2. Four-layer architecture model.

Data exchange layer. Data exchange is the key function of a network system. Data exchange layer interacts with object sensing and controlling layer by using structured data. The attributes of data include ID, type, length, life time, etc. Structured data is transmitted through wired or wireless, single hop or multiple hops, always-linked or intermittent connection ways. The functional entity of this layer is the wired or wireless *transmission channel*, and the common attributes include bandwidth, protocol, etc. Wired channels, including twisted pair, coaxial cable, optical fiber, and so on, are the main transmission channels of the core networks in IoT. As for wireless channels, satellite and 3G/4G/5G support the remote data exchange of IoT, and WiFi, Zigbee and Bluetooth support efficient data exchange from short range applications. The corresponding functions mainly include data routing and transmission control.

Information integration layer. In this layer, the sensing data is integrated into the semantic information. The structured information has the attributes of information ID/type and behaviors of processing/storing. The functional entity of this layer is the *information processor* with attributes of computing resources and storage resources. The main functions of the information processor are data fusion, data mining, information query, privacy protection, etc. The intergraded information can be divided into discrete information (such as scalar data) and continuous information (such as audio and video). Information integration layer shields the details of lower networks for the developers of the application services.

Application service layer. This layer provides services directly to users or the devices in the user space. According to the specific requirements of users, the application service layer provides different kinds of services by using integrated information from the information integration layer. Thus, this layer interacts with the information integration layer by using integrated information, and interacts with users by using services whose attributes are service ID and type. The main functional entity is the *application server*, whose attributes include server ID and location, and the functions mainly include service release, authorization, and management. There are three main service categories: information publication service, sensing-controlling (CPS) service, and physical object retrieval service.

The entities, attributes, and functions for each layer are summarized in Table I. From description of this four-layer functional model, we know that the networking problem of IoT is across the two lower layers. In the object sensing and controlling layer, the functional entities, especially for the sensors, need to form local networks for transmitting and collecting data efficiently. In the data exchange layer, amounts of edge sensor networks need to interconnect to the Internet for extensive internetworking. Therefore, based on this model, we explore the IoT networking problem from internetworking model and sensor-networking mode, respectively.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

TABLE I
ENTITIES, ATTRIBUTES, AND FUNCTIONS OF FOUR LAYERS

Layers	**Entities**	**Main Attributes**	**Main Functions**
Application service layer	Application servers	Server ID, location	Service release, authorization, management
Information integration layer	Information processors	Computing resources	Data mining, information query, information fusion, the privacy protection
Data exchange layer	Transmission channel	Bandwidth, protocol	Data routing, transmission control
Object sensing and controlling layer	Sensor	Name, type	Signal conversion, data transmission
	Controller	Name, control modes	Data receiving, command execution

III. INTERNETWORKING MODEL: ADDRESS CENTRIC OR INFORMATION CENTRIC

Building efficient and scalable internetworking model, which is the basis for interconnecting the edge network (sensor networks) to the core network (Internet), is probably the most important fundamental to support extensive interconnection among a huge amount of heterogeneous network elements in IoT. To better understand this, we compare IoT with Internet. The design principle of Internet is the classic "end-to-end principle", which states that application-specific functions ought to reside in the end devices rather than in intermediary nodes. This networking mode, being address (named host) centric, focuses on connections between ends. Core protocols of Internet, including IP and TCP, are designed by following the end-to-end principle. As noted before, IoT is a very huge system consisting of billions of heterogeneous elements. Besides strong network elements like computers and smart phones, there are more weak network elements like RFID tags and sensors on IoT, for which the traditional address centric model, represented as IP, is not suitable anymore.

More importantly, IP is designed to locate devices to enable an end-to-end conversation between two devices. While in IoT, more and more applications are interested in obtaining desired information rather than connecting to a specific device. For example, in video applications, one may want a specific video chip no matter where it is located. Compared to the address centric networking where accessing information and services requires mapping from the *what* that users care about to the network's *where*, information centric networking, which is a kind of internetworking models built on named data, is received more and more attentions in the field of future Internet architecture.

A. Global Addressability for Address Centric Networking

Effective addressing mechanism is required to enable global addressability of devices and further global interaction and co-operation among devices. As we know, IP protocol is a global addressability mechanism and has achieved great success in the Internet. Therefore, IP seems to be a nature choice for enabling global addressability of the IoT. We find that although IP is suitable for a portion of IoT devices (indeed, strong and static devices such as computer, laptops) and applications, it is not a good choice for weak/mobile devices and emerging applications on IoT. First, IP protocol has large overhead with regard to energy constraints of most weak devices on IoT, such as sensors and RFID tags, and may not be able to run on these devices. Second, IP protocol is originally designed for stationary devices (where an IP address is not only the identifier, but also the locator of a device), which does not handle mobility easily. While on IoT, many devices such as mobile phones, RFID tags used in logistics have high spatial mobility and intermittent connectivity, and hence require a new addressing protocol with better mobility support.

To deal with the challenge, our work is carried out in two approaches. The first approach is to connect weak devices to Internet through simplified IPv6 compatible with weak devices. The second is to design a new ID/Locator separation architecture for efficient addressability of mobile devices and information.

(1) Simplified IPv6

For the first approach, it is well known that the number of IPv4 addresses is definitely not enough for IoT devices. Therefore, IPv6 is proposed for integrating sensor nodes into Internet [4]. However, a straightforward deployment of IPv6 on sensor nodes is not feasible, mainly due to the incompatibility between IPv6 and two-layer protocols such as IEEE 802.15.4 currently used in sensor networks [3]. For example, IPv6 requires support of packet sizes much larger than the largest IEEE 802.15.4 frame size. In order to integrate IEEE 802.15.4 into the IPv6 architecture, we have done work in three aspects. First, we propose a simplified IPv6 address allocation method, which rapidly allocates IPv6 address to mobile devices, and reduces the complexity of address translation between IPv6 and IEEE 802.15.4 MAC address. Second, we design a new encapsulation and header compression mechanism that allows IPv6 packets to be sent to and received from IEEE 802.15.4 based networks. The mechanism also supports redundant information removal and fragment retransmission under multicast scenarios. Third, taking into account low-power devices on IoT, the energy-aware routing algorithm can setup and use network paths with the lowest energy consumption, while satisfying traffic demands [5].

(2) ID/Locator separation approach

After simplification, IPv6 can run on weak devices. However, it still has many IP-inherited disadvantages in terms of mobility support [6]. To overcome such disadvantages of IP approach, we propose new network architecture that enables the coexistence of IP-based and non-IP based devices and applications. In this separation approach, we assign a global unique ID for each interested entity (a device or a piece of information) on IoT, which is independent of the device's locators such as IP address, IEEE 802.15.4 MAC address or RFID identifier. Note that by naming information, the architecture enables information addressability, that is, one can retrieve

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

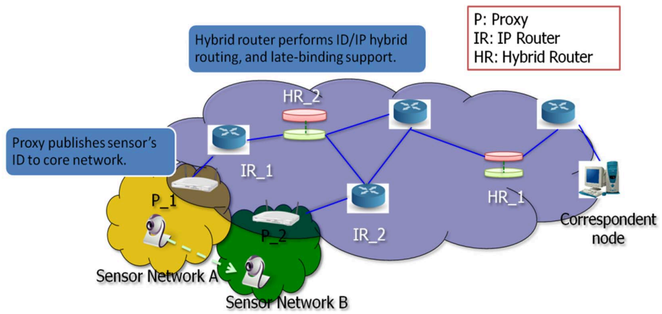


Fig. 3. ID/locator separation architecture for IoT.

named information without specifying a host location. When the information request goes into the IoT, the routing service of the architecture will route the request to the nearest location with the information. Furthermore, through this information naming, the network becomes information-aware and hence can do in-network caching and traffic engineering so as to optimize network.

In addition, compared with IP approach, the security of IoT is enhanced by utilizing a public key security technique. Specifically, a device's ID is actually a public key, and the device also holds a private key corresponding to the public key. Before any data exchange, a sender device should sign a digital certificate with its private key, then the receiver can verify the authentication of the sender through the certificate. The data integrity could also be inherently guaranteed through such public key security techniques.

Then we illustrate how a mobile device could be accessed by using the ID/locator separation approach, by taking a sensor as an example. Here we assume that the sensor is too light-weighted to run IP stack, then the sensor will not be reachable outside its local sensor network. However, we will find that ID/locator separation architecture guarantees the global access despite of mobility of the sensor.

A high-level overview of the ID/locator separation architecture is shown in Fig. 3. Compared with current Internet architecture, two network entities are added: proxy and hybrid router. A proxy is like a gateway in a sensor network. When a sensor enters the coverage of a proxy, it will register its ID to the proxy. However, a proxy performs one more work than a gateway, that is, it will publish the sensor's ID to the core network, specifically, to the hybrid router on IoT. In fact, due to this publishing operation of the proxy, a sensor becomes visible and can be directly accessed by other parts of IoT.

A hybrid router has two main functions. The first is the ID/IP hybrid routing mechanism, in which a packet could also be routed according to its destination ID when the destination IP address cannot be determined at the moment. Correspondingly, the second function is the late-binding support. When a packet without a destination IP address arrives at an appropriate hybrid router, it is the responsibility of the hybrid router to perform the ID-to-IP lookup and then to forward the packet to the current network location of the destination device.

Now we see how a node communicates with the sensor, while the sensor is moving between different sensor networks, as shown in Fig. 3. Suppose that at the beginning, the sensor is in sensor network A. Then the Proxy in this network, P_1, will publish the sensor's ID to the near hybrid router HR_2. In this publishing, the network location of the sensor is determined to be the IP address of P_1. Note that packets from the node just contain the ID of the sensor, without knowing the sensor's location. According to the ID/IP hybrid routing mechanism supported by the network architecture, these packets will be routed to HR_2. At this moment, HR_2 will perform the late-binding support, that is, it will resolve the sensor's ID to the address of P_1. After this resolving operation, packets will be routed to P_1, and consequently be delivered to the sensor.

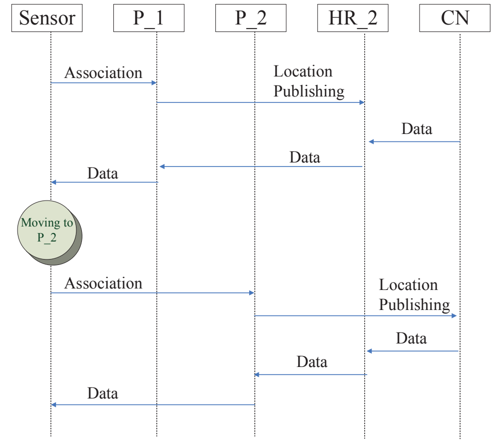


Fig. 4. The procedure that a sensor moves from sensor network A to sensor network B.

TABLE II
COMPARISON BETWEEN ADDRESSING PROTOCOLS

	ISO 18000 (RFID)	IEEE 802.15.4 (Sensor)	Simplified IPv6	ID/Locator Separation
Layer	PHY	MAC	Network	Network
Global addressing	No	No	Yes	Yes
Mobility support	No	No	Limited	Yes
Information support	No	No	No	Yes
Security support	No	Limited	Limited	Yes

When the sensor moves into sensor network B, proxy P_2 will update the ID-IP mapping of the sensor on HR_2, that is, change the network location to the IP address P_2. When subsequent packets from the node arrive at HR_2, HR_2 will resolve the sensor's ID to the new network location, i.e., the address of P_2. Consequently, packets will be routed to P_2 and then be delivered to the sensor in sensor network B. To sum up, the ID/Locator separation network architecture supports seamless mobility. The whole procedure is illustrated as Fig. 4.

To sum up, we summarize and compare the characteristics of different addressing mechanisms for IoT, including ISO 18000 for RFID, IEEE 802.15.4 for Sensor, simplified IPv6 and ID/Locator separation in TABLE II. Clearly, the ID/Locator separation approach has important advantages on mobility, information and security support. We believe that it is the most promising approach to enable global addressability and interaction for the IoT.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

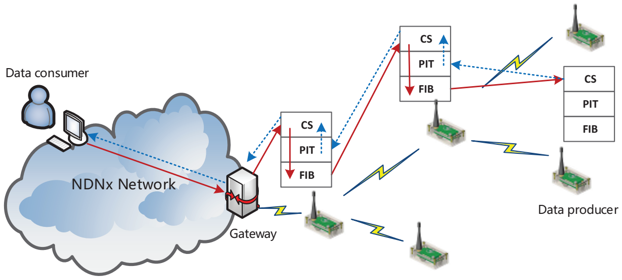


Fig. 5. Information-centric networking across the core network (NDNx network which is an implementation of NDN.) and the edge network (sensor network).

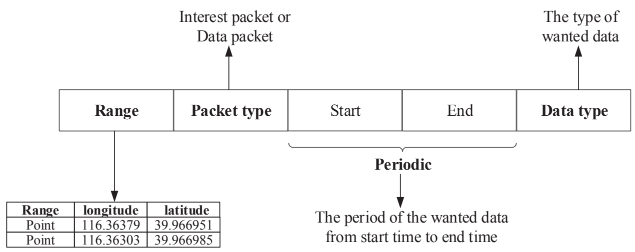


Fig. 6. Packet structure.

B. Interconnection between Strong and Weak Network Elements for Information-centric Internetworking

Inspired by the Named Data Networking (NDN) [7], [8], in our model the communication is also driven by client (data consumers) through the exchange of Interest and Data. A consumer puts the name of a desired piece of data into an Interest packet and sends it to the network. Routers use this name to forward the Interest toward the data producer. Once the Interest reaches a node that has the requested data, the node will return a Data packet that contains both the name and the content (see Fig. 5). This Data packet follows in reverse the path taken by the Interest to get back to the requesting consumer.

Since most network elements in edge networks are resource-constrained, we design three lightweight modules [9] for weak network elements: 1) *Content Store* (CS) is used to cache the content and responsible for keeping the timeliness of the content; 2) *Pending Interest Table* (PIT) records the source node and the route of the coming Interest; and 3) *Forwarding Information Base* (FIB) maintains the forwarding information and routing items. The edge networks communicate with information-centric core networks which exploit the NDN architecture through the gateways.

As shown in Fig. 6, The network packet contains four parts:

- *Range* indicates where (geographical location) user want to get the data. Node will transmit the packet according to the range and FIB.
- *Packet Type* indicates the packet is Interest packet or Data packet.
- *Periodic* containing two timestamps – start and end – indicates the period when the wanted data is collected.
- *Data Type* indicates the type of wanted data, such as temperature, light, and humidity.

As shown in Fig. 7(a), after the tree topology created, the son node sends its range to its parent. The initial value of the range is the node's position. The parent node receives the ranges, and then calculates its own range which is a bigger area containing all son nodes and itself. This process will last till all nodes' ranges are stable. After each node gets the corresponding range, the son node sends their current range to the parent node for filling its FIB (see Fig. 7(b)). Items in FIB record source node and value of the range. FIB plays the role of route table, and the node transmits the Interest packet according to FIB.

Fig. 8(a) illustrates a routing process of Interest packet. User who want to get the data creates an Interest packet. After receiving the Interest packet, the sink node extracts the range and the periodic information in the interest packet, and then queries its FIB. The sink node finds that the wanted data is located in the Node 1's range. Thus, it transfers the Interest packet to Node 1. The node received the Interest packet queries its CS at first. If the matched item is not found, the node will repeat the above process. Finally, the Interest packet is transmitted to Node 5 in this case. If a node finds matched item in the CS after receiving the Interest packet, it no longer transfers the Interest packet but creates a Data packet. The nodes who transmitted the Interest packet store the packet's source and content in its PIT. The function of the PIT likes "bread crumbs" strategy, and Data packet will return back according to PIT. Fig. 8(b) illustrates the process of returning back a Data packet in the network.

IV. Sensor-networking Mode: Proactive Deploying or Crowdsensing

In the last ten years, sensor network as an important embedded networked sensing technology has largely been applied in relatively homogeneous rural areas, where researchers or users proactively deploy sensor nodes in specific areas such as forests, vineyards, and glaciers. We call this sensor-networking mode as proactive deploying mode which becomes an important way for exploring the physical world (see Fig. 9(a)). However, deploying a large number of sensor nodes in a specific area lead to high networking cost, inconvenient system maintenance, and inflexible service.

On the other hand, mobile devices equipped with onboard sensors become popular. As two typical categories of mobile devices, smartphones and vehicles become extremely popular recently and are potentially important sources of sensing data. A large amount of mobile devices, constituting a mobile sensing network, can sense hotspots of human activity ubiquitously (see Fig. 9(b)). Through intentional or unintentional collaboration, the mobile devices can execute many complex sensing tasks which are impossible for a single sensor node or a small group of sensor nodes [11]. We call it crowdsensing mode which is complementary to the proactive deploying mode [12]. Overall, crowdsensing, coming up with the new ubiquitous concept where every person and vehicle in urban areas can be used as a sensing component, has brought us to a new field of IoT.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

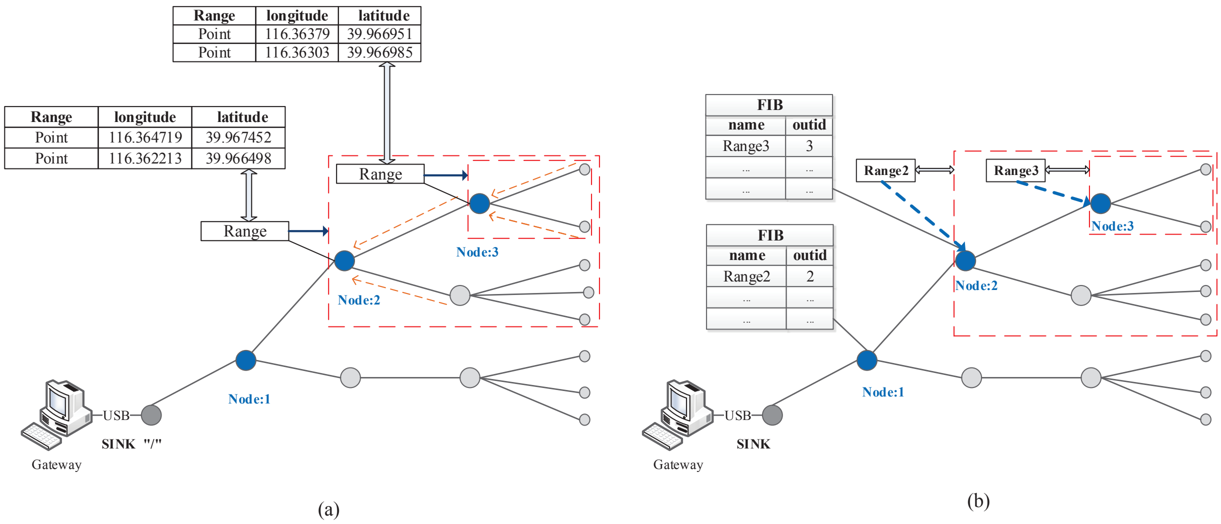


Fig. 7. Routing discovery. (a) Each node gets the corresponding range. (b) Generation of FIB.

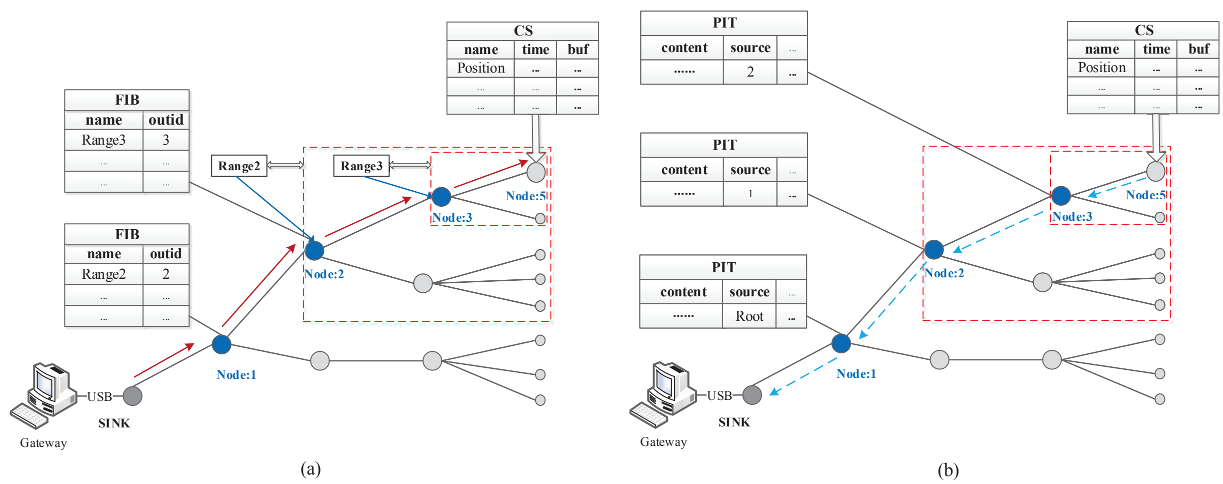


Fig. 8. (a) Routing process of Interest packet. (b) Return process of Data packet.

A. Proactive Deploying Mode

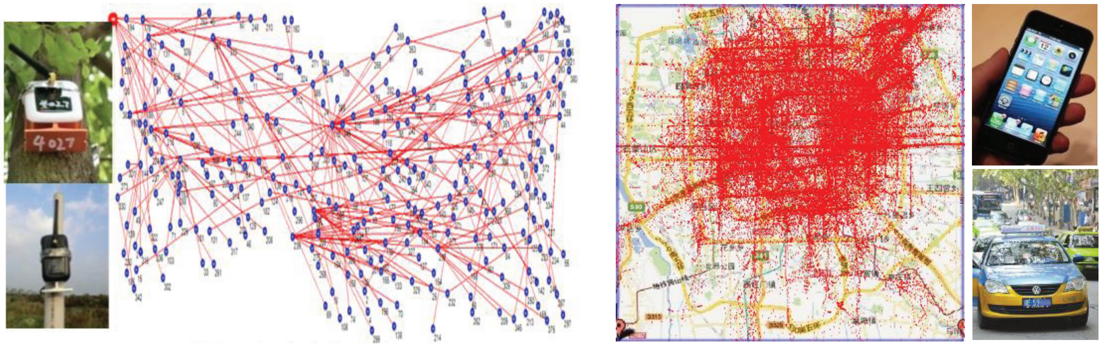
The classical representative of proactive deploying mode is Wireless Sensor Network (WSN) in which the pre-deployed sensor nodes form an aggregation-structure network and the sensing data is collected to a sink node. Many IoT applications require persistent long-term sensing data collection. However, sensor nodes are often battery powered and deployed in harsh environments. Hence, the networking strategy must be carefully designed to reduce energy consumption on sensor nodes, so as to prolong the network lifetime as much as possible. Moreover, WSN networking also faces the problems of high data redundancy, heavy network load, and low data reliability. These problems are the main bottlenecks for developing large-scale WSNs.

We utilize two ways to deal with the above problems. The first one is to utilize the readings of a small subset of nodes to recover all the others at sink node as shown in Fig. 10(a). This is an approximate data collection method in fact. However, data recovery inevitably incurs errors. Thus, how to guarantee the accuracy of recovered data is a key challenge. The other way is to fuse the sensing data at intermediate nodes step by step as shown in Fig. 10(b). Because data fusion needs to process the sensing data at each intermediate node, how to ensure the tolerant delay and desired reliability of data transmission are two challenges.

(1) Approximate Data Collection Oriented Networking

The key idea of our networking strategy for Approximate Data Collection (ADC) is to divide a sensor network into clusters, discover local data correlations on each cluster head, and perform global approximate data collection on the sink node according to model parameters uploaded by cluster heads[13]. Specifically, our approach can be divided into two steps: local estimation and adaptive data approximation. During the first step, on the purpose to reduce the communication cost

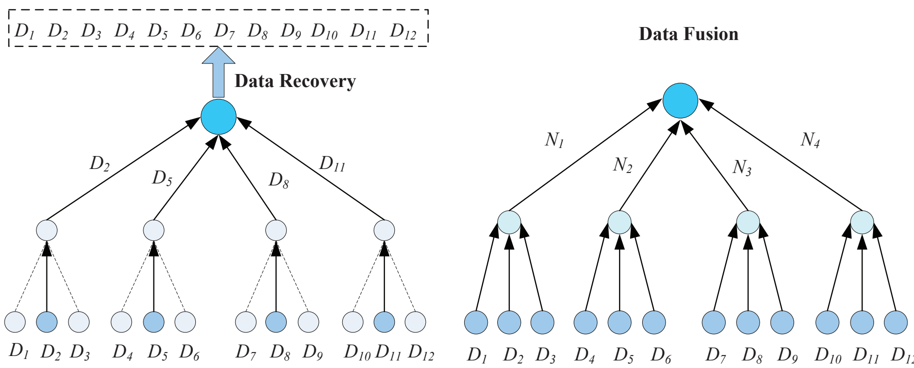
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal



(a) Proactive deploying mode

(b) Crowdsensing mode

Fig. 9. Two networking modes in IoT. (a) Proactively deployed sensor networks. This figure comes from the GreenObs project [10]. (b) Crowdsensing networks.



(a)

(b)

Fig. 10. Two ways for decreasing data redundancy and network load. (a) Data recovery. (b) Data fusion.

among sensor nodes and their cluster heads, we formulate the following data model through which a sensor node can estimate a newly generated reading $p_i(t)$ based on its historic data:

$$p_i(t) = m_i(t) + \alpha(v_i(t-1) - m_i(t-1)) + \beta(v_i(t-2) - m_i(t-2)) + \gamma(v_i(t-3) - m_i(t-3)),$$

where $m_i(t) = a + bt$ is a linear trend component that grows over time t (a and b are two constants), α, β, and γ are three real constants with empirical values which satisfy $\alpha+\beta+\gamma < 1$, $v_i(t)$ is the reading of sensor s_i at time t.

Each sensor node only needs to send the parameters of its local estimation data to its cluster head, rather than the raw sensor readings. If the difference between the estimated value and the original value does not exceed a given threshold, the sensor node does not upload its data to its cluster head. As a result, the communication cost is reduced. The update message is sent only when the difference between the estimated value and the original value exceeds the prespecified threshold. Due to that sensor readings of nearby sensor nodes are very similar, thus during the second step, we define the following new metric $D_{ij}(t)$ to measure the sensor reading similarity between sensor node s_i and s_j based on their local estimation:

$$D_{ij}(t) = |p_i(t) - p_j(t)|\,.$$

Utilizing this metric, we employ a novel tool called correlation graph to describe the spatial correlation among sensor nodes based on the sensor reading information provided by the local estimation. Based on the correlation graph, the sink can elaborately select a small portion of sensor nodes as dominating set, and recover sensor readings of all sensor nodes with the local estimation data of senor nodes in the dominating set, while guaranteeing that the errors of recovered data are within the prespecified error bound.

(2) Data Fusion Oriented Networking

To save more energy, in-network processing such as data fusion is a widely used technique [14], [15], [16]. Traditional schemes aim at providing reliable transmission to individual data packets from source node to the sink, but seldom offer the desired reliability to a data fusion tree. To deal with this problem, we explore the problem of Minimum Energy Reliable Information Gathering (MERIG) when performing data fusion [17]. Our purpose is to design a reliability-guaranteed networking mechanism over the existing routing structure with data fusion to achieve energy-efficient and reliable information collection.

By adaptively using redundant transmission on fusion routes without acknowledgments, packets with more information are delivered with higher reliability. In order to compute the optimal number of transmissions for each node, firstly we

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

mathematically describe the problem of MERIG as finding a set of transmission reliability $\{R_i\}$, which can minimize the total energy cost:

$$E = (1+\beta)DC\log_r \prod_{i=1}^{N}(1-R_i) + E^{FS},$$

where E is the total energy cost; R_i denotes the transmission reliability of node i; N is the total number of nodes; D is the size of data packet in bytes; C is the unit transmission cost for each bytes; β is a constant; r denotes the packet error rate, and satisfies $0 < r < 1$. After obtaining $\{R_i\}$, we can easily find the set of the optimal number of transmissions at each node $\{t_i\}$ using the following equation:

$$t_i = \log_r(1-R_i).$$

In some applications, WSNs are required to provide real-time information. Thus, the sink nodes (called controllers also) need to accurately estimate the process state with the estimates obtained from the sensors under rigid delay constraints. To deal with this issue, we propose a novel in-network estimation approach in multi-hop WSNs [18]. More specifically, we address the problem through jointly designing in-network estimation operations and an aggregation scheduling algorithm. Our in-network estimation operation performed at relays not only optimally fuses the estimates obtained from the different sensors but also predicts the upper stream sensors' estimates, which cannot be aggregated to the sink before deadlines. Our estimate aggregation scheduling algorithm is interference-free and able to aggregate as much estimate information as possible from the network to the sink within delay constraints. Fig. 11 illustrates the summary of our in-network estimation. In an aggregation tree, each node takes the following actions. For the leaf node, it samples the dynamical process state at time t, performs a local estimation based on its own measurements obtained before time t, and then, waits for being scheduled. If a leaf node is scheduled, it transmits the local estimate to its parent directly. For the relay node, firstly it also does sampling and performs local estimation as the leaf node does, then it performs an optimal information fusion based on the estimates received from its child nodes and its own local estimate. If the estimates cannot be received from its child nodes in the current scheduling period, the relay node predicts the estimates based on the previously received estimates from its child nodes,and use the predicted estimates to calculate the fused one. When a relay node is scheduled, it transmits the fused estimate to the next-hop node. For the sink node, it continuously collects the estimates from its child nodes. At the deadline of every scheduling period, the sink predicts the estimates based on the previously received estimates from its child nodes, and then calculates an optimal estimate based on the previously received information and the predicted estimates. We also prove the unbiasedness of in-network estimation, and theoretically analyze the optimality of our approach.

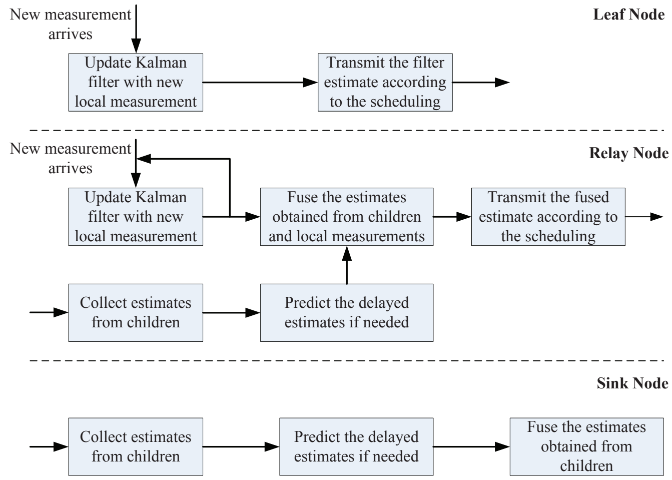


Fig. 11. The summary of in-network estimation.

B. Crowdsensing Mode

Human mobility plays an important role in networking under crowdsensing mode, as it can be exploited to improve sensing coverage and data transmission. On one hand, mobile nodes can sense the surroundings wherever their holders arrive opportunistically, enabling large-scale urban sensing applications. On the other hand, opportunistic contacts among mobile nodes can be exploited to transmit sensory data in networks with intermittent connections based on the store-carry-and-forward paradigm. Meanwhile, human mobility also brings the following three problems. First, *how to evaluate the impacts of human mobility on sensing quality and realize network planning reasonably*? At present, the sensing quality evaluation and network planning have been extensively studied under proactive deploying mode, but little work focuses on them under crowdsensing mode, especially at the city-scale. Second, *how to model the data dissemination law and exploit it to further improve the opportunistic data forwarding performance*? At present, the data dissemination laws and opportunistic forwarding schemes have been extensively studied in opportunistic networks or delay-tolerant networks (DTNs), but they only consider the sharing or dissemination of data interesting to individual mobile users instead of the data collection under crowdsensing mode, and thus fail to consider the spatial-temporal correlation among sensor data and its impact on the network performance. Third, *how to model the human sociality and exploit it to further improve the opportunistic data forwarding performance*? It is known that human sociality has a key impact on human mobility, since it decides the spatial properties of human mobility. Moreover, many studies have verified that human sociality can be exploited to improve the opportunistic data forwarding performance. However, most existing social-based opportunistic forwarding schemes use the traditional approaches in social networks or ego networks to evaluate social metrics, which are difficult and time-consuming to calculate mainly because of the transient node contact and the intermittently connected environment. In the following, we will introduce some of our explorations to cope with the above three problems.

(1) Sensing Quality Evaluation and Network Planning

Under proactive deploying mode, the coverage ratio is commonly used to measure the sensing quality of WSNs, and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

the coverage problem has been extensively studied for network planning. In contrast, the sensing quality under crowdsensing mode is variant due to human mobility. The traditional metric of coverage ratio is not suitable for crowdsensing networks. Toward this end, we explore the new metrics for measuring sensing quality from spatial-temporal dimensions, and further realize network planning reasonably.

First, we consider the spatial dimension. For most urban environment monitoring applications, measurements from sensing nodes are utilized to generate a sensing image[3] over a region according to spatial interpolation methods. This is because both natural phenomenon over a region and images can be expressed as a two-dimensional (2D) signal. Similarly, the smartphones/vehicles equipped corresponding sensing modules are able to record the urban phenomenon (e.g. temperature, CO_2 concentration, and noise) as the form of sensing images. This kind of sensing system seems like a "urban camera", and a vast number of smartphones/vehicles form the "CCD (Charge-Coupled Device) sensor" of this camera. In conventional image systems, *resolution* is an important parameter that describes the detail an image holds. Higher/lower resolution means more/less image detail. Learning from the concept of resolution, we proposed a new metric – *urban resolution* – to measure the quality of the sensing images and the sensitivity of urban sensing systems for environment monitoring applications [19]. Different from the commonly used definition of resolution for digital images and cameras, the resolution of urban camera is not simply the "pixel" (smartphone/vehicle) count. Because the pixels of digital camera form a fine grid and the pixels of urban camera are with the scattered and dynamic distribution.

Towards this end, we first present a Delaunay triangulation based method to generate sensing image, and apply the correlation coefficient to defining the quality of sensing image (QoI). Then, we propose a new metric, urban resolution, to measure the quality of urban sensing for environment monitoring applications. To the best of our knowledge, urban resolution is the first metric which directly gives insight of *how sensitivity the city-scale sensing system could achieve*. We utilize three 2D signals with different variation degrees to study the relationship between resolution r and sensing node number s via Monte Carlo simulations. From the simulation results, we reveal the *linear growth relationship* between $\sqrt{r}$ and $\sqrt{s}$: $\sqrt{r} = a\sqrt{s} + b$. We also give the reference values of the parameters that $a = 0.9, b = -0.8$ for the uniform random distribution model of sensing nodes and $a = 0.5, b = 0.6$ for the human/vehicles mobility model. By using this linear growth relationship, on one hand, it is easy to infer the urban sensing quality according to the scale of urban sensing system; on the other hand, this result can guide the deployment of urban sensing system, i.e., determine how many smartphone/vehicles needed for participating in urban sensing application for a given resolution requirement. We find the distribution model of human/vehicles is able to be described by a *truncated Pareto distribution*. Combining the truncated Pareto distribution and the linear function of $\sqrt{s}$, we derive the probability that the urban resolution, r, in an unit grid is larger than a given value γ is given by:

$$\mathbf{Pr}\{r > \gamma\} = \mathbf{Pr}\{s \neq 0\} \times \frac{L^{\lambda}\left(4^{-\lambda}(\sqrt{\gamma} - 0.6)^{-2\lambda} - H^{-\lambda}\right)}{1 - L^{\lambda}H^{-\lambda}}.$$

where L, H, and λ are the parameters of truncated Pareto distribution. This result can be used to calculate the radio of the sub-regions which satisfy the required resolution to the whole region for a specific application.

Second, we consider the temporal dimension. The coverage under crowdsensing mode is time-variant due to human mobility. Therefore, we propose a new metric, *inter-cover time (ICT)*, to characterize the opportunity with which a subregion is covered. Specifically, we divide the whole urban sensing area into grid cells, and define the ICT as the time elapsed between two consecutive coverage of the same grid cell, to characterize the opportunity with which each grid cell is covered. Obviously, shorter ICT results in better sensing quality for a grid cell. In order to explore the pattern of ICT in realistic scenarios, we perform empirical studies on real mobility traces of thousands of taxis collected in Beijing and Shanghai, two of the largest cities in China. We find that the distribution of the aggregated ICT follows a truncated Pareto distribution regardless of the taxi number or the grid cell size. Next, we propose another metric, *opportunistic coverage ratio*, to characterize the relationship between the sensing quality and the number of nodes. The *opportunistic coverage ratio* is defined as the expected ratio of grid cells that can be covered during the specific time interval τ, which can be derived as a function of the distribution of the aggregated ICT:

$$f(\tau) = \frac{\sum_{i=1}^{m} F_i(\tau; n)}{m},$$

where $F_i(\tau; n)$ denotes the ICT distribution of grid cell g_i under the condition with n nodes, and m is the number of grid cells. We have provided detailed analysis and computing method of opportunistic coverage ratio in [21]. Since $f(\tau)$ increases monotonically with n and τ, we can easily derive the required number of nodes to achieve the specific opportunistic coverage ratio during a specific time interval, thus realizing network planning reasonably. For example, based on our analysis on the real datasets, it is needed to deploy at least 5800 and 6300 taxis in Beijing and Shanghai, respectively, so that the opportunistic coverage ratios in a region of 900 km^2 are not less than 90 percent during the time interval of one hour.

(2) Data Dissemination Modeling and Cooperative Data Forwarding

In practice, it is beneficial to integrate opportunistic forwarding schemes with data fusion by considering the spatial-temporal correlation for two reasons: first, users may be interested only in the aggregated results of sensory data (e.g., the average temperature or noise level); second, sensory data collected in close proximity or time periods may be highly correlated, and data fusion can effectively eliminate redundancy and hence reduce network overhead.

[3] In traditional environment monitoring applications, GIS platforms are commonly used to visualize the sensing data in real space and construct maps of layered information [20]. The way to display sensing data is sensing map. In this paper, GIS is not used, then we use the word "sensing image" instead of "sensing map".

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

First, we integrate the most basic and classic opportunistic forwarding scheme, epidemic routing, with data fusion. The approach is simple and straightforward: Let $X = \{x_1, x_2, \cdots, x_i\}$ and $Y = \{y_1, y_2, \cdots, y_j\}$ denote two packets having already fused i and j original packets respectively, and $0 \leq i \leq j$. When any node u_1 with packet X encounters another node u_2 with packet Y, both of them obtain a new fused packet by using a fusion function $f(X, Y)$. The detailed forwarding rules can be found in [22]. We call this new scheme as Epidemic Routing with Fusion (ERF). Although the idea seems simple and straightforward, we still face the first important challenge: how to model the data dissemination laws? Previous studies on data dissemination laws of opportunistic forwarding schemes assume that all the data packets are propagated independently. However, the spatial-temporal correlation among sensor data should be considered in the forwarding process with data fusion, which results in a more complex data dissemination process. In order to cope with this challenge, we derive an ordinary differential equation [22] to model the dissemination law of correlated data, which could serve as fundamental guidelines on integrating opportunistic forwarding schemes with data fusion for improving data forwarding performance. Specially, we leverage the S-I-R model, one classic model for analyzing the propagation process of infectious diseases, to analyze the data dissemination process. Let $F(t)$ denote the probability that a given node has a copy of one specific packet at time t, then we have

$$F(t) = \frac{1}{1 + (N-1)e^{-\beta N t}},$$

where N is the number of nodes in the network, and β is the contact rate of the pairwise nodes. Let $I_i(t)$ denote the number of nodes which have a packet fusing i of K original packets under the ERF scheme at time t, then we have

$$I_i(t) = \binom{K}{i} F^i(t)(1 - F(t))^{K-i} N.$$

We can derive the transmission overhead of the ERF scheme by the following system of rate equations:

$$\begin{aligned} \frac{dC_f(t)}{dt} &= \sum_{0 \leq i < j \leq K} \left(2 - \frac{\binom{j}{i}}{\binom{K}{i}}\right) I_i(t) I_j(t) \beta \\ &\quad + \sum_{0 < i = j \leq K} \left(1 - \frac{1}{\binom{K}{i}}\right) I_i^2(t) \beta, \\ C_f(0) &= 0. \end{aligned}$$

Moreover, we theoretically prove that the ERF scheme has the same delivery delay with the traditional epidemic routing scheme while significantly reducing the transmission overhead.

On the other hand, the Binary Spary-and-Wait (BSW) scheme is considered to be one of the most efficient schemes to reduce the large transmission overhead of the epidemic routing scheme without incurring significant delay penalties. However, while considering to integrate the BSW scheme with data fusion, we still face a challenge: how many forwarding tokens should be assigned to the nodes for the new fused data packet? In order to cope with this challenge, we design a series of rules to assign proper number of forwarding tokens to the nodes, and theoretically prove that this new scheme, called as Binary Spary-and-Wait with Fusion (BSWF), can reduce both the delivery delay and transmission overhead of the BSW scheme.

Through trace-driven simulations, we verify that our designed two cooperative data forwarding schemes can achieve better tradeoff between delivery delay and transmission overhead. Specially, the ERF scheme can reduce the transmission overhead by 78% compared with the epidemic routing scheme; the BSWF scheme can increase the delivery ratio by 16%, and reduce the delivery delay and transmission overhead by 5% and 32% respectively, compared with BSW scheme.

(3) Human Sociality Modeling and Social-based Data Forwarding

It is important to develop a lightweight approach to exploiting human sociality for improving opportunistic data forwarding performance. We are interested in the following question: can we explore some stable social attributes to quantify the centrality and similarity of nodes in a lightweight way?

Taking GPS traces of human walks from the real world, we find that there exist three known phenomena: a) people always move around some popular locations, namely *public hotspots*, b) each individual always shows preference for some particular locations, namely *personal hotspots*, c) both the public and personal hotspots have two key features enabling a lightweight scheme, namely *burstiness*, implying that there are only a small number of hotspots required to exchange among mobile nodes, and *stability*, implying that only infrequent updating of hotspots is required.

Motivated by this observation, we present Hotent (HOTspot-ENTropy), a novel forwarding metric to improve the performance of opportunistic routing [23]. Let ω_i denote the weight of the i-th hotspot, $\omega_{p_i}^j$ denote the weight of the j-th hotspot influenced by the i-th node, random variable X_i denote the distribution of personal hotspot of node i, and random variable Y denote the distribution of public hotspots. Then we have $Y = \omega_1, \omega_2, \ldots, \omega_k$, and $X_i = \omega_{p_i}^1, \omega_{p_i}^2, \ldots, \omega_{p_i}^k$. The Hotent metric can be calculated by the following five steps. First, we use the relative entropy between the public hotspots and the personal hotspots to compute the *betweenness centrality* of nodes:

$$C_b^i = \left(\sum_{j=1}^{k} \omega_{p_i}^j \log(\omega_{p_i}^j / \omega_j) \right)^{-1}.$$

Second, we utilize the inverse symmetrized entropy of the personal hotspots between two nodes to compute the *similarity* between them:

$$Sim(i, j) = (Sim(i/j) + Sim(j/i))^{-1},$$

where $Sim(i/j)$ is the relative entropy of node i about node j, and $Sim(j/i)$ is the relative entropy of node j about node

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2015.2493082, IEEE Internet of Things Journal

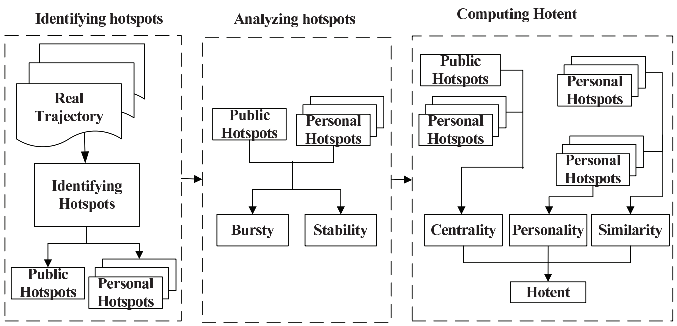


Fig. 12. The framework of designing the Hotent metric.

i:

$$Sim(i/j) = \sum_{l=1}^{k} \omega_{p_i}^l \log(\omega_{p_i}^l / \omega_{p_j}^l),$$

$$Sim(j/i) = \sum_{l=1}^{k} \omega_{p_j}^l \log(\omega_{p_j}^l / \omega_{p_i}^l).$$

Third, we utilize the entropy of personal hotspots of a node to characterize its *personality*:

$$Per_i = -\sum_{l=1}^{k} \omega_{p_i}^l \log(\omega_{p_i}^l).$$

Fourth, we exploit the law of universal gravitation to incorporate the centrality and similarity, using the metaphor of mass for node centrality and distance for the similarity between two nodes. We have the gravitation between node i and j as follows:

$$g_{i,j} = G \frac{C_b^i C_b^j}{Sim(i,j)^2},$$

where G is the gravitation constant. Finally, we integrate the three factors into the Hotent metric as follows:

$$Hotent_{i,j} = Per_i \times g_{i,j}.$$

The framework of designing the Hotent metric is illustrated in Fig. 12.

The Hotent metric can be directly used for the opportunistic forwarding scheme. Take node i and j for example. When node i encounters node j, for any data packet that node i carries, if its destination is node j, node i delivers the packet to node j and removes the packet from its buffer. Otherwise, if node j does not hold this data packet, they swap their Hotent metrics. If $Hotent_{i,d}$ is smaller than $Hotent_{j,d}$, node i delivers the packet to node j and removes the packt from its buffer, namely that it adopts a single-copy scheme. Trace-driven simulation results show that our proposed Hotent scheme largely outperforms other state-of-the-art solutions, especially in terms of packet delivery ratio and the average number of hops per message.

V. Challenges

A. Internetworking: How to Merge IP and Non-IP?

As noted before, IoT is a very huge system consisting of billions of heterogeneous network elements (including strong and weak devices). Internetworking of IoT faces the following situation: the strong network elements in the core network (Internet) exploit IP while the weak network elements in the edge networks (such as RFID network, sensor networks) exploit light-weight IP and non-IP technologies including ID, named data, and so on. This is because that IP has large overhead with regard to capability constraints of weak network elements. Even in the Internet, IP exposes many problems, such as mobility-support, network security, imbalance of network flow. We explored two technical routes of non-IP, ID/locator separation architecture and information-centric model, to build efficient and scalable internetworking models for extensive interconnection among a huge amount of heterogeneous network elements.

However, an inconvenient truth is that it is impossible to discard IP which has achieved great success in the Internet. Therefore, a novel internetworking model which merges IP and non-IP is the key challenge for designing a practical IoT architecture. The ID/locator separation architecture we proposed provides a start point for merging IP and non-IP. In this architecture, a global unique ID is assigned to each interested entity in IoT. The interested entity can be a device or a piece of information. For a device, the ID is independent of the device's locators such as IP address, IEEE 802.15.4 MAC address or RFID identifier. For a piece of information, the ID enables information addressability, that is, one can retrieve named information without specifying a host location.

B. Sensor-Networking: How to Integrate the Crowdsensing Mode and the Proactive Deploying Mode?

For the proactive deploying mode, we have explored the way to combine networking with in-network processing. However, we just studied several typical in-network processing methods, such as approximate data collection and data fusion using simple functions (e.g., averaging, summation, voting, and max/min). In fact, for further improving network performance in different application scenarios, we also need to explore how to combine networking with more complex in-network processing methods. Furthermore, it is necessary to build an uniform theoretical framework to deeply reveal the relationship between in-network processing and network transmission. This framework will provide the basis for networking proactively deployed sensor nodes.

For the crowdsensing mode, the evaluation of sensing quality, as the guiding factor for sensor-networking, is a very complex problem. We have analyzed it mainly considering human mobility from both time and space dimensions, but it is necessary to consider more other factors, such as human social psychology, sensor availability and reliability. Due to various factors, different mobile users have heterogeneous data quality at different times. Thus, it is important to evaluate users' data quality, and get rid of malicious and low-quality data. Moreover, under the crowdsensing mode, the sensing data is transmitted with diverse patterns. It is still a big challenge to design adaptive data transmission scheme under varied network connectivity for achieving a good balance between transmission quality and network resource consumption. It is also an important research issue to leverage the correlation among sensing data to improve the data transmission performance.

Compared to sensor networks, the number of mobile devices participated in the sensing task is much huger, the device types are more diverse, and the sensing area is much broader. But for the places where few people access to, the crowdsensing mode cannot satisfy the quality of sensing, i.e., there will exist some sensing blank areas. Therefore, the IoT needs not only crowdsensing networks, but also proactive deployed sensor networks. The huge difference between the two networking modes causes the high difficulty of designing an integration scheme. There needs some pioneering works to reveal the complementary relationship between the crowdsensing mode and the proactive deploying mode, so as to satisfy the sensing requirement of IoT. One basic idea is to leverage the relationship between sensing quality and human mobility to decide an optimal set of locations for deploying static sensors, and then design an efficient scheme enabling the cooperation between a number of mobile sensors with these static sensors.

VI. Conclusions

In this article, we concluded our research progresses on IoT networking. To interconnect large-scale heterogeneous network elements in IoT and exchange data efficiently, we built up a 4-layer functional model, and then explored both the internetworking model and the sensor-networking mode from two different routes: address-centric/information-centric models for internetworking and proactive-deploying/crowdsensing modes for sensor-networking. In the future work, we will further focus on the two fundamental IoT networking issues: 1) how to merge IP and non-IP? and 2) how to integrate the crowdsensing mode and the proactive deploying mode? Solving these problems will forcefully promote the development of IoT technologies.

References

[1] H.-D. Ma, “Internet of things: Objectives and scientific challenges,” *Journal of Computer science and Technology*, 26(6): 919-924, 2011.
[2] http://www.gartner.com/newsroom/id/2636073
[3] http://www.ipso-alliance.org
[4] G. Hu, K. Xu, J. Wu, Y. Cui, and F. Shi, “A general framework of source address validation and traceback for IPv4/IPv6 transition scenarios,” *IEEE Network*, 27(6): 66-73, 2013.
[5] S. Wang, Y. Cui, S. K. Das, W. Li, and J. Wu, “Mobility in IPv6: Whether and How to Hierarchize the Network?” *IEEE Transactions on Parallel and Distributed Systems*, 22(10): 1722-1729, 2011.
[6] J. Li, Y. Zhang, K. Nagaraja, and D. Raychaudhuri, “Supporting efficient machine-to-machine communications in the future mobile Internet,” *IEEE Wireless Communications and Networking Conference Workshops (WCNCW)2012*, 2012: 181-185, 2012.
[7] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, 2009: 1-12, 2009.
[8] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. Claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, “Named Data Networking,” *ACM SIGCOMM Computer Communication Review (CCR)*, July 2014.
[9] Y.-N. Song, H.-D. Ma, and L. Liu, “Content-centric internetworking for resource-constrained devices in the Internet of Things,” *2013 IEEE International Conference on Communications (ICC)*, 2013: 1742-1747, 2013.
[10] Y. Liu, Y. He, M. Li, J. Wang, K. Liu, and X.-Y. Li, “Does wireless sensor network scale? A measurement study on GreenOrbs,” *IEEE Transactions on Parallel and Distributed Systems*, 24(10): 1983-1993, 2013.
[11] R. Ganti, F. Ye, H. Lei, “Mobile crowdsensing: Current state and future challenges,” *IEEE Communications Magazine*, 2011, 49(11): 32-39.
[12] H.-D. Ma, D. Zhao, P.-Y. Yuan, “Opportunities in Mobile Crowd Sensing,” *IEEE Communications Magazine*, 2014, 52(8): 29-35.
[13] C. Wang, H.-D. Ma, Y. He, and S.-G. Xiong, “Adaptive approximate data collection for wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, 23(6): 1004-1016, 2012.
[14] H. Luo, J. Luo, Y. Liu, S. Das, “Adaptive Data Fusion for Energy Efficient Routing in Wireless Sensor Networks,” *IEEE Transactions on Computers*, 55 (10): 1286-1299.
[15] H. Luo, Y. Liu, S. Das, “Routing Correlated Data with Fusion Cost in Wireless Sensor Networks,” *IEEE Transactions on Mobile Computing*, 5(11): 1620-1632.
[16] Y. Sun, H. Luo, S. Das, “A Trust-Based Framework for Fault-Tolerant Data Aggregation in Wireless Multimedia Sensor Networks,” *IEEE Transactions on Dependable and Secure Computing*, 9(6): 785-797.
[17] H. Luo, H. Tao, H.-D. Ma, and S. K. Das, “Data fusion with desired reliability in wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, 22(3): 501-513, 2011.
[18] H.-T. Zhang, H.-D. Ma, X.-Y. Li, and S.-J. Tang, “In-network estimation with delay constraints in wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, 24(2): 368-380, 2013.
[19] L. Liu, W.-Y. Wei, D. Zhao, and H.-D. Ma, “Urban Resolution: New Metric for Measuring the Quality of Urban Sensing,” *IEEE Transactions on Mobile Computing*, 2015.
[20] D. Cuff, M. Hansen, and J. Kang, “Urban Sensing: Out of the Woods,” *Comm. of the ACM*, vol. 51, no. 3, March 2008.
[21] D. Zhao, H.-D. Ma, L. Liu, and X.-Y. Li, “Opportunistic coverage for urban vehicular sensing,” *Computer Communications*, 2015.
[22] D. Zhao, H.-D. Ma, S.-J. Tang, and X.-Y. Li, “COUPON: A Cooperative Framework for Building Sensing Maps in Mobile Opportunistic Networks,” *IEEE Transactions on Parallel and Distributed Systems*, 26(2): 392-402, 2015.
[23] P.-Y. Yuan, H.-D. Ma, and H.-Y. Fu, “Hotspot-entropy based data forwarding in opportunistic social networks,” *Pervasive and Mobile Computing*, 16, Part A: 136-154, 2015.