Resilient IoT Architectures Over Dynamic Sensor Networks With Adaptive Components

Sharief M.A. Oteafy, *Member, IEEE*, and Hossam S. Hassanein, *Senior Member, IEEE*

***Abstract*—As competing industries delve into the Internet of Things (IoT), a growing challenge of interoperability and redundant deployments is magnified. Specifically, as we augment more "things" in the IoT fabric, how will these components interact across their heterogeneity, let alone collaborate. In this paper, we address the core issue of component interaction and operation under the IoT umbrella. We present our contribution in the framework of wireless sensor networks (WSNs), as a founding block in the IoT. More importantly, we present a novel paradigm in the design of WSNs, to build a resilient architecture that decouples operational mandates from the nodes. We abstract IoT things as wirelessly interfaced components, which introduce functionality physically decoupled from their devices; boosting resilience, dynamicity, and resource utilization. This approach dissects the study of any IoT nodal capacity to its "connected" components, and empowers dynamic associativity between things to serve varying functional requirements and levels. It also enables reintroducing only the components required to suffice for network operation, or only those needed to meet a new requirement. More importantly, critical resources in the network will be shared within their neighborhoods. Thus network lifetime will relate to functional cliques of dynamic IoT nodes, rather than individual networks. We evaluate the cost effectiveness and resilience of our paradigm via simulations.**

***Index Terms*—Dynamic components, dynamic topology, heterogeneous architecture, intelligent things, Internet of Things (IoT), novel paradigm, parallel-assignments, resilient protocols, sensor networks.**

I. Introduction

THE proliferation of the Internet of Things (IoT) is contingent on how efficiently its components will interact. More specifically, how will "things" communicate, coordinate, and most importantly collaborate to achieve IoT operational goals. In the broad sense, IoT will bridge sensors, actuators, and machine-to-machine communication to enable real-time sense-making services. This entails the operation of wireless sensor networks (WSNs), radio identification systems (RFID) and ultimately access networks that will enable varying access schemes to communicate these devices (e.g., WiFi, ZigBee, ANT+, NFC, LiFi, etc.).

On a more specific scale, how will heterogeneous components interact in the IoT, given their operational mandates which might not be in alignment with other IoT components produced by different proprietaries? Today, these things span smart devices, sensing nodes (SNs), wearable technologies, and all the software and hardware architectures that will support their communication. While we were successful in standardizing many communication standards, such as the IEEE 802 family, many of the core protocols that govern IoT technologies are neither standardized nor interoperable. Even if we transcend standardization, we have a fundamental challenge of functional representation for IoT components, as each vendor is manufacturing their own components in disregard to what is already deployed or viably accessible in a region of deployment. Simply assuming that a communication standard (e.g., 6LowPAN) will bring together IoT components is neither scalable nor realistic in the current market of diverse things.

We address the challenge of IoT proliferation by leveraging the resilience and coordination of interaction between things. Specifically, as WSNs form the founding block of IoT, we will elaborate on a novel component-based design, which enables resource sharing and resilient operation between WSN components. This paper targets a foundational block in IoT proliferation, as we present a framework for adaptive association between functional components (things) in the grand scheme of building sensing applications. This component-based framework will encompass sensing, communication, and control components that realize the foundation of a scalable and truly synergetic view of IoT. We hereafter label this paradigm as a dynamic WSN (D-WSN) framework.

The D-WSN paradigm introduced in this paper presents a threefold contribution. 1) Assigning network functionality to individual components that dynamically associate with active sensor nodes, to augment their capabilities as needed. 2) Re-engineering WSN operation in the IoT to accommodate for dynamic architectures that evolve over time to boost resilience and lifetime, based on individual components rather than static WSN nodes. 3) Present a deterministic model for WSN functional lifetime in the IoT, tightly coupled with functional capacity rather than individual nodes. Fig. 1 shows an overview of the D-WSN paradigm in the IoT framework.

The contribution of D-WSN contrasts itself to novel mainstreams in WSN research, hence significant emphasis is presented in Section II to identify these mainstreams and elaborate on their evolution. Section III follows with an explanation of the components of the architecture, with an overarching theme of synergy. This is followed by a description of how these components work and communicate, and an analysis of resilience

Manuscript received January 1, 2016; revised July 19, 2016; accepted October 24, 2016. Date of publication October 26, 2016; date of current version April 28, 2017. This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant STPGP 479248.

The authors are with the School of Computing, Queen's University, Kingston, ON K7L 2N8, Canada (e-mail: oteafy@cs.queensu.ca; hossam@cs.queensu.ca).

Digital Object Identifier 10.1109/JIOT.2016.2621998

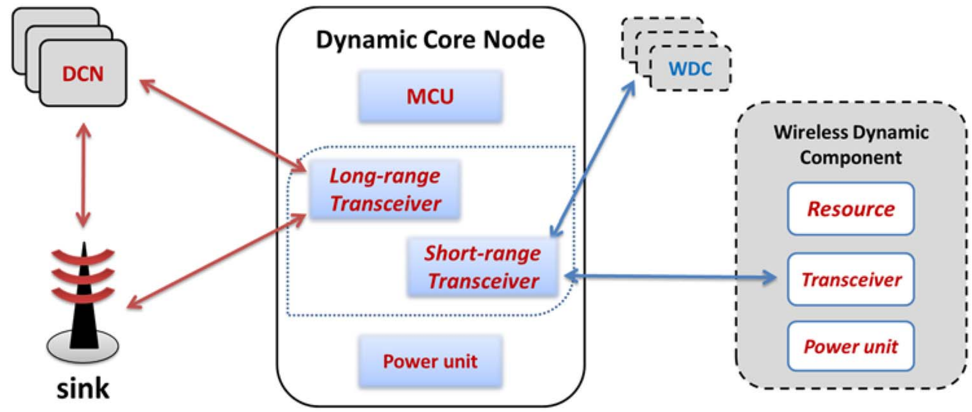

Fig. 1. D-WSN paradigm under the IoT—overview of the interfacing components.

metricized by functionality sustenance. Our experimentation and performance evaluation are presented in Section IV and this paper is concluded with future work and remarks in Section V.

II. Motivation and Background

This section covers the main streams of research upon which this paradigm is built. It is important to note the evolution of WSNs from the basic principles and requirements set over a decade ago [1] to where this paper proposes to set future architectures. Knowing the inherited architectures and protocols from mobile ad hoc networks to WSNs and further evolutions, brings great insights as to where bottlenecks are identified, and sets a framework for scalable WSN proliferation under IoT operational mandates.

A. Building WSN for Heterogeneous IoT Operation

In all fairness, WSNs were application specific from their inception, and the idea of a generic platform/protocol was clearly eliminated early on in literature due to various tradeoffs. For example, prolonging network lifetime came at the price of time-latency constraints for sensing and communication [1], [2]. Then, as load balancing was researched to marginalize the tradeoff, control-overhead became a discriminatory metric [3].

This mainstream approach to WSNs suffers from two main bottlenecks. At one end, the dominant application-specific drive hinders WSN synergy with ubiquitous networks. At the other, the notion of an SN as a single entity with static operational goals and functional parameters. This is a by-product of aiming for one-time installations of sensing architectures. Two simple notions hence followed: nodes, once deployed, are static in terms of functionality, and the lifetime of all components on a node are mostly capped by the failure of the first one; being the transceiver, sensor or even the memory unit [2]. Understanding the boundaries of defining an erroneous node is very crucial, yet quite commonly assumed in WSN literature to encompass any fault in the node (software, hardware, communication, etc.).

This is a fundamental hindrance to evolution in WSN protocols and operation, especially as applications tap into urban and harsh environments where nodes (the collection of components on a chip) are more prone to failures at any given time. Failures as simple as unaccounted for clock drifts could significantly impact node operation; not only for synchronization, but for in-node scheduling of events and processor operation [4]. Moreover, there is a near consensus on the "no free lunch" design of sensor nodes. A node simply cannot perform all possible tasks required, while maintaining homogeneity across nodes and prolonging network lifetime, without sacrificing other design parameters and functionalities. This becomes more of a problem as the requirements for WSN applications increase in complexity and diversity; let alone change post-deployment.

As such, this paper presents the D-WSN paradigm. It serves operational capacity from the design phase, and introduces the dynamicity of self-adapting sensor nodes capable of coping with targeted components. These components will hold both communication interface and specific functionalities, which are to be mapped to the requirements for the whole WSN. Their locations would be adaptive to application requirements, and they could be introduced at network deployment and/or later on as a measure of maintenance as the need arises. Thus, the "dynamic" component of D-WSN spans both functional variation through network lifetime, and the (re)association of components with nodes post-deployment. For example, a high-end sensor could be probed by multiple nodes in parallel, instead of mandating a separate installation on specific nodes.

Previous efforts in literature have presented the notion of platforms with multiple components [5], and others focused mainly on multiple transceivers/antennas for boosting communication and evaluated their performance [3]. Other studies investigated the possibility of having multilevel duty cycles, to allow a node to operate in different states based on available resources [4], [6]. Nevertheless, these notions are static by nature and are predesigned to cater for fixed application requirements.

Here we approach four directions that have inspired evolution in WSN architectures; yet with their varying constraints.

The first capitalizes on redundancy in components, as a measure of resilience, and a selective mechanism to choose the most appropriate for a given task. The second and third probe resources from the public, via smart devices, to carry out sensing tasks. The fourth, though now a dated contender, exploited the simple fact that not all components need to function at maximum capacity all the time (on a single sensor node), and hence each components was presented in multiple operational states, each tagged with a functional capacity and power expenditure. The following sections elaborate on these directions.

B. Architectures With Redundant Components

As requirements for more sophisticated sensor nodes increased, researchers investigated adding more components—possibly redundant ones—to boost performance [3], [5]. The extent and deviations varied significantly according to the available platform, compatibility of components, and design requirements. The intent varied for introducing the additional components on a board. For adding sensing accuracy and data fidelity, sensing boards were equipped with multiple sensors for each phenomenon or high end ones. Dedicated processors for data aggregation and filtering were also introduced.

C. Public Sensing and LTE-A Architectures

A new paradigm of sensing has emerged in a domain called public sensing. It builds upon research in mobile computing, cellular networks, and WSNs. The main idea is depending on users with smartphones, or specially supplied devices, to carry out sensing tasks and reporting it back to a database. Many solutions, such as Pachube [10], have been launched thereafter. The proliferation of device to device (D2D) communication, as well as promising directions in enabling cellular access to heterogeneous resources, are empowering technologies in our design directions. Recent surveys on D2D empowerment under LTE-A [11] provide further insights into the potential of D2D.

However, it is important to note that public sensing is not a mainstream WSN paradigm. It lends itself to literature on data aggregation and fidelity checking, yet the core concepts of how the two paradigms operate are different. For one, reporting is a function of when the users (whether passively or actively) report their findings. This could be based on dedicated hardware, generic smartphones with dedicated applications, or simply text reporting. Most of public sensing research takes place under the participatory sensing paradigm.

D. Current Drivers in IoT Interoperability

Significant research efforts in IoT proliferation are attempting to utilize smart devices and low power sensors in the ubiquitous operation of IoT. The core challenge of interoperability between all these devices remains an open issue, especially as new architectures are presented everyday with varying resources and attached constraints.

To this end, recent efforts on bridging these devices are gaining traction, whether on functional interoperability, or integrating collected data and information fusion. Most of the recent work has been directed in the latter category, sine functional interaction across heterogeneous devices proves increasingly complex, as devices grow more diverse. Although current standardization efforts and industry alliances are attempting to bridge operational mandates for IoT devices [11].

A more promising approach lies in IoT interoperability over information fusion, mostly via a hub or gateway approach [13]. The core argument for this approach lies in adopting a staged approach to interoperability, whereby devices interact directly with a hub, which acts as a bridge to other devices managed by other IoT hubs. Blackstock and Lea [14] adopted a JSON-based catalogue for IoT devices, named HyperCat IoT, and present in [12] a specification for IoT hubs and devices.

E. Participatory Sensing Networks

The notion of enticing the crowds to carry out sensing tasks has been approached in many ways; most dominantly now in the domain of participatory sensing networks. Incentive schemes that promote either "reputation" or rewards based on monetary or credit systems, have been seen in many proposals. Although there is merit in the claim of crowd-intelligence, and the dependency on ubiquitously available devices, there are many challenges that hinder the wide scale adoption of participatory sensor networks (PSNs).

Xie *et al.* [18] investigated bargain-based mechanisms to remedy the intrinsic tendency of nodes not to take part in PSNs. This is a growing concern as PSN systems take a toll on smartphones when the users activate their applications, and little consensus has been seen in establishing fairness metrics in reporting and respective rewards [13], [14]. In fact, the case of large scale deployments (province, country, continent, etc.) it is often impossible to ensure predetermined trajectories and expected paths for mobile nodes taking part in the PSN, and their localization schemes remain a privacy issue.

F. Contrasting Network Function Virtualization

The premise of decoupling hardware from operational capacities has been heavily investigated under network function virtualization (NFV) and software defined networks (SDNs). Liang and Yu [15] presented a detailed survey on recent directions specific to Wireless network virtualization. While NFV focuses on virtualizing network functions to enable rapid developments on the software side, we argue in D-WSN for a hardware driven approach to associate sensors with core nodes to derive varying sensing applications. While the network model is described in Section III, it is important to highlight that our goal is not to abstract the functionality of underlying resources for service orchestration, but to focus on resiliently reassigning sensing resources to sensing/core nodes to maintain network functionality post-failures, and enable a dynamic load distribution of functionalities and failure mitigation post-deployment.

However, it is important to note the potential of adopting an NFV view of D-WSN, to facilitate service orchestration and software-driven management of sensing tasks that are built on a resilient infrastructure. This presents an open research issues and will further propagate the design advantages of D-WSN. This could further extend to capitalize on SDNs [12], where the underlying D-WSN architecture could be viewed as a service enabler, albeit with intrinsic resilience capabilities, potentially fine-tuned by the network management, and control planes.

G. Nodes With Multioperational Levels

As the cost of individual MEMS components (e.g., transceivers and sensors) dropped, a new feasible possibility came to be. Introducing multiple components on the same node, i.e., redundant ones, and experimenting with switching individual components on an off, in studied combinations, to conserve power. Not only would it serve power conservation for network longevity [4], but it also enables introducing higher end nodes that have multiple capabilities, switched on upon need.

DMULD presented, in [6], a deterministic operational mandate for a decentralized network of duty cycled nodes. Yet the duty cycling took place on the component level. The operation of nodes, intra-node coordination, and decentralized approach highlighted the potential of nodes with multiple capacities, to metamorphically adapt to applications and network failures.

III. Component Based D-WSN Architecture

The architecture of our D-WSN is intrinsically different than traditional WSNs.[1] The core difference is how functionality (of components) are decoupled from the main platform of SNs. Thus, performing a task now is a utilization and virtual coupling problem, involving multiple entities and less resources over the whole network.

D-WSN has three core goals. First, to boost dynamicity and generic design as a paradigm shift in WSNs. Second, potentiate a broader platform for application independent components that scale over time. Third, establish a utility-based quantifier to the choice of resources matched to each functional request. That is, establishing a paradigm that would allow different resources to compete for carrying a given task, whereby the SN would choose among them.

The following sections dissect the D-WSN paradigm and present the three main components, namely the dynamic sensing (core) nodes, resources dubbed dynamic components, and components with remote-wakeup capability. These components are presented in contrast to traditional WSN components, eliciting the core differences in paradigms, and mode of operation.

A. Network Model

The D-WSN will be comprised of three components. First, dynamic core nodes (DCNs) which will form the topology of the communicating network. Each DCN will attach itself to one or more wireless dynamic component (WDC). Thus, forming a star-like network association with neighboring WDCs. Finally, DCNs will communicate with each other, relaying their data back to a sink (or multiple sinks). Thus the network is formed of two types of nodes, and heterogeneous in that sense. However, the decisions of associations between DCNs and WDCs are all made locally within their vicinities, in a decentralized and homogeneous manner.

[1]An abbreviated version of this model appeared in [14].

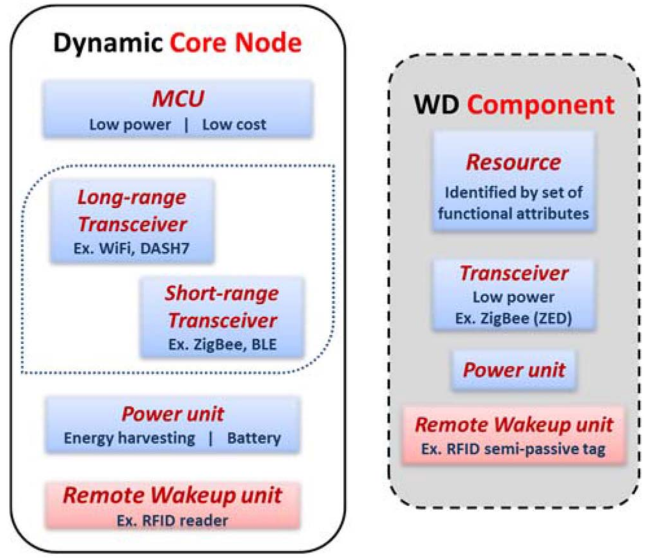

Fig. 2. Design and components of a wireless DCN and a WD component, highlighting the auxiliary remote wakeup unit which enable “dormant” mode operation for WDCs in the IoT.

B. DCNs

The DCN will form an anchor for multiple operations. It will carry out regular sensing and communication tasks, as per the mandate of the governing application(s). In addition, it will interface to WDCs for one of two reasons. Adopting a functionality that it requires but does not have, or saving its battery/resources and “outsourcing” the required functionality from a neighboring WDC. Imperatively, a utility function will dictate the benefit in attaching to a neighboring WDC for a given functionality, if the current DCN already has that capability.

As depicted in Fig. 2, each DCN will encompass the typical micro-controller unit (MCU) and a power unit. The latter could have an energy harvesting component, as this is a growing trend in current sensor node designs. In addition, the DCN will have two transceiver units. The first will enable long-range communication, between DCNs and each other, and DCN to sink.

Two viable candidates are WiFi or DASH7, as both could sustain a reasonably long range communication, with varying power demands [9]. For example, a typical DASH7-compliant transceiver would achieve a range of 1000 m, since it operates on a lower frequency band; 433 MHz.

The second unit will be a short range transceiver, which would establish a parent-child relationship with neighboring WDCs. This would typically be a ZigBee protocol stack,

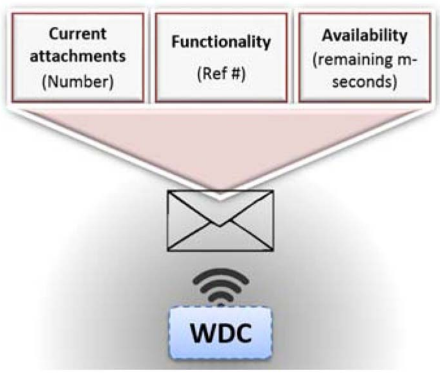

Fig. 3. WDC broadcasting its availability to neighboring DCNs, detailing the contents of the join message.

as it operates in low-power mode, and enables communication under the parent–child paradigm.

By design, DCNs communicate with each other over a multihop architecture. At this level, many routing and MAC protocols could handle data communication between DCNs, thus it is an inconsequential factor for this D-WSN paradigm.

C. Wireless Dynamic Components

The core task of a WDC is to provide functionality to its neighboring DCNs. It could associate with one or more DCNs, depending on its functional resources, remaining energy and current attachments. That is, how many DCNs is it already serving? The components of a WDC are depicted in Fig. 2. Most importantly, WDCs are equipped with short-range low-power transceivers, enabling only direct communication with DCNs. As such, a typical choice would be a ZigBee protocol stack, whereby the WDC would function as a ZigBee end device if the DCN is a ZigBee router [12].

A WDC would have a functionally distinct description of its resources, as a deterministic set of attributes, as described in [7]. All DCNs and WDCs will share a unique pool of resource identifiers, enabling a 1–1 association between what the WDC offers and what the DCN needs. For example, the WDC would offer a camera with a known resolution, bitrate, and capturing speed. We will assume that a table containing all these identifiers and descriptors are known by the application governing the operation of the network, and each functional identifier would have a reference number. This will be communicated by the DCN to its neighboring DCNs, as depicted in Fig. 3.

A WDC intrinsically serves neighboring DCNs, thus it needs to broadcast its availability periodically. While this operation is detailed in Section IV, it is important to note that WDCs will switch to a dormant state when it serves no DCNs, with time wakeup timers enabling it to probe DCNs again. This range-limited broadcasted "join" message is shown in Fig. 3.

D. Remote Wakeup

Generally, sensing nodes are deemed useless when their batteries die. Thus, maintenance protocols in WSNs aim to replace their functionality by introducing new ones, or leveraging operation via high-density deployments to start with. In our D-WSN paradigm, we incorporate an important advancement in novel designs. Recent advances in RFID systems, especially semipassive ones, enable tags to store a small amount of data (typically 56 bytes), and report it back when interrogated by readers. As such, we cater for the capability of high-end DCN designs to hold short-range RFID readers. Similarly, for WDCs to be equipped with semi-passive tags that could store aggregated information from its resources before it runs out of battery. As such, after a WDC loses communication with its neighboring DCNs, and cold no longer sustain that level of operation, it would switch into operate and store mode. Thus, enabling a DCN with reader capabilities to interrogate it at a later time when it comes into its range, and extract information that has been stored over time.

We thus dub the WDC as "proactive" in its former state, and refer to it as dormant after it drops in battery power and transfers to the latter state. This operation and switching are further detailed in Section IV.

IV. D-WSN in Operation: The Synergy of Dynamic Sensing

A core motivation for D-WSN is the overarching synergy in its operations. The notion of a single-application WSN no longer holds prospect, nor does that of static functionality. More importantly, associations of nodes to functional components require a dynamic paradigm to improve resilience and service delivery on the long run.

We hereby detail the operation of our D-WSN paradigm, both in terms of nodal operation, and interactions within vicinities. The remainder of this section elaborates on the operation of DCNs and WDCs post-deployment. Moreover, the decentralized coordination between these components to mitigate intermittent and permanent failures is presented.

It is important to note that we assume that all data would be routed back to a sink, which will mandate the operation of nodes. For larger scale deployments, WLOG we assume that multiple sinks will dissect the operational grid to smaller regions, whereby a single sink would manage data collection and the dissemination of application updates.

A. Operation of D-WSN

As in any WSN, there is a mandate for a functional description of an application. That is, functional requirements with spatial and temporal mandates, and predetermined QoS measures. In D-WSN, we adopt the functional descriptors of application requirements as detailed in [1]. In addition, DCNs and WDCs have predetermined resources that are static in their attributes.

For example, a DCN would have a transceiver, with predefined specifications at known dB levels, power consumption at each level, data rate, protocol stack, etc. Thus, mapping a functional requirement from an application to the known resources in the network is a sheer assignment problem. In D-WSN, we establish the architecture that realizes this assignment, and the interactions of the components that render its dynamic functional capabilities.

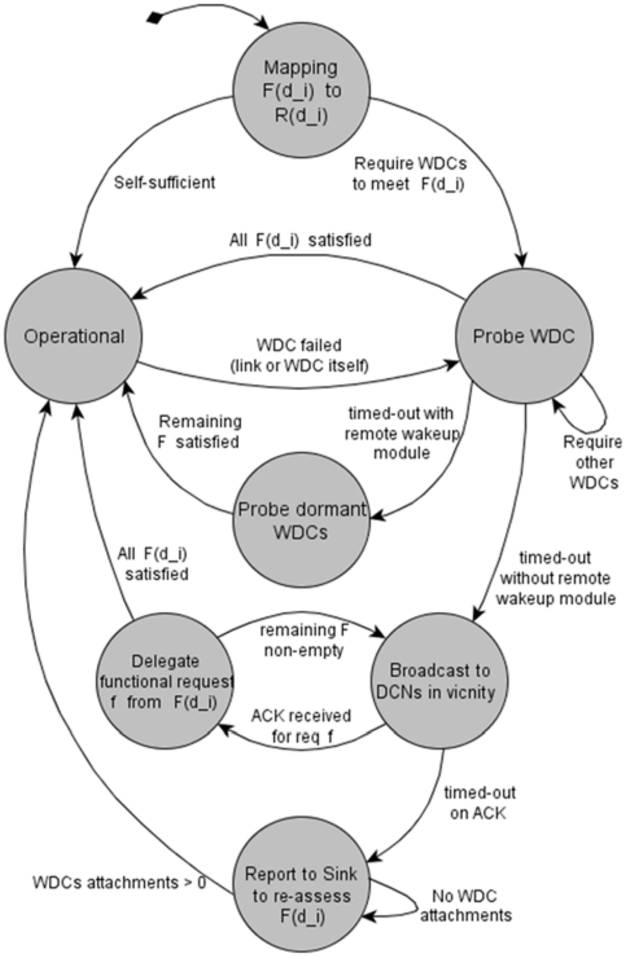

Fig. 4. Deterministic FSM for a DCN in operation.

B. DCN in Operation

The operations of the DCN are depicted as a deterministic finite state machine (FSM) in Fig. 4. We denote the set of DCNs as **D**, where $|\boldsymbol{D}| > 0$ is known by the sink, and the location of each $d_i \in \boldsymbol{D}$ has a predetermined set of attributes $\text{Attr}(d_i)$. After deployment, and depending on the locations of each d_i, the sink would multicast to each d_i a set of functional requirements to be carried out in its region, denoted as $F(d_i)$. Since we adopt a homogeneous operation for DCNs, the remainder of this section will refer to a single DCN in operation WLOG.

When d_i receives $F(d_i)$, it will probe its own local resources, denoted as $R(d_i)$, to attempt to serve them. If local resources suffice, it will settle for that and transition into operation mode. That is, performing its functional requirements. If not, it will start probing its neighboring WDCs, denoted as $W(d_i)$ and represented by

$$W(d_i) = \bigcup_{j\,\in\,W} w_j = \left\{j : w_j \text{ active } \wedge\ w_j \text{ in range } d_i\right\} \tag{1}$$

where all WDCs $w_j \in W$ that are currently in their proactive state, and within the transmission range of the short-range transceiver of d_i. We further introduce the notion of resources that are not in the first tier of neighborhood of d_i, yet reachable through $W(d_i)$ within a hop limit of $\boldsymbol{k}$, computed as

$$W^k(d_i) = W(d_i) \bigcup \left(\bigcup_{j\,\in\,W(d_i)} W^{k-\mathbf{1}}(d_j) \right). \tag{2}$$

Fig. 4 details the operations thereafter, since space limitation hinders a detailed description. However, it is important to note that if neither $R(d_i)$ nor $W(d_i)$ could serve $F(d_i)$, then d_i would report back to the sink for reassessment of the assigned functional requirements $F(d_i)$.

In this case, D-WSNs present a significant edge. That is, the sink assigns tasks based on location, and DCNs decide in a decentralized fashion the optimal assignment of neighboring resources to their respective $F(d_i)$. Thus, the sink need not encompass global knowledge of the viable resources in the network, only the locations of current DCNs.

Hence, if a shortage of resources arises, all the application would require is deploying WDCs in the regions of interest, and their governing DCNs would attach to them and resume operation. Moreover, if functional requirements change, this is a decentralize method for assessing precise need for resources, instead of random dense deployments.

C. WDC in Operation

The operation of WDCs is a major contributor to the dynamic dimension of this paradigm; D-WSNs. A WDC is placed at the time of network deployment to meet initial functional requirements, and reintroduced later on to mitigate failures and leverage new application requirements. As such, WDCs play an important role in the total resource pool of the network, enabling multiapplications to run concurrently.

At any given point, there will be W WDCs in the network, where $|W| \geq 0$ and could vary; incremented by new deployments or reduced by failures. We note that the functional requests served by W are in fact greater than $|W|$, since each $w_j \in W$ could serve more than one DCN, depending on its resource attributes.

Fig. 5 details the operation of each w_j. The overarching duty of a w_j is to serve neighboring DCNs. Upon deployment, it would broadcast its availability via a Join message, depicted in Fig. 3, announcing how many more DCNs it could serve, and the remaining time it would spend in the proactive state. Both metrics are broadcasted to allow DCNs in arbitrating should more than one w_j offer a needed resource. When a w_j reaches its maximal allowed attachments, it would turn off its periodic broadcasting mechanism, and return to it only when a DCN releases that connection (due to failure, change of requirements, etc.). After all connections are released, the WDC would go into a dormant state of sense and store, at an increasing sleep timer till it is depleted (for future physical data extraction), or await in a passive wakeup mode if it is equipped with a remote wakeup module. Dedicated timers dictate linger time in each state before a deterministic transition occurs (i.e., triggering the transitioning).

D. Resilience Model

Any component in a WSN is prone to failure. The core objective of D-WSN is designing a network that is resilient to

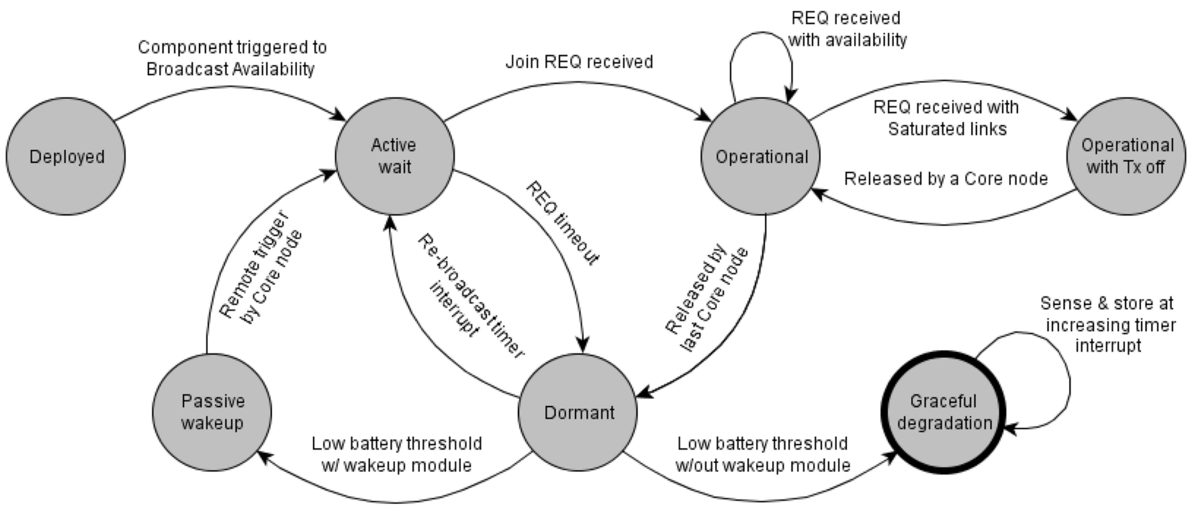

Fig. 5. FSM detailing the deterministic operation of a WDC $w_j \in W$ after deployment.

various types (and durations) of failures, and establishing a formal model for failure mitigation and recovery. We formally define two types of failures in the D-WSN model, component failure and network failure. We define a component failure as the inability of a component to adhere to its functional requirement, including the loss of communication link to an associated DCN or WDN. For example, an MCU that ran out of memory could no longer process data (due to failed memory module, failing bus, etc.), yields a component failure in D-WSN.

We define network failure in D-WSN as the instant when DCNs can no longer establish a backbone network to connect all components to the sink (i.e., network partitioning stage) or when the network cannot meet all designated functional requests given the underlying WDCs and DCNs. More formally, the D-WSN network has failed when at least one DCN reaches the "report to sink to reassess $\mathbf{F}(\boldsymbol{d_i})$" state.

It is important to note that in both intermittent and permanent failures, the network is designed to adapt its functionality through periodical reassignments of tasks to WDC resources. Failed DCNs will release the attached WDCs, opening them up for use by other DCNs. Upon recovery from intermittent failures, new attachments will be made as per the state machine in Fig. 4. Similarly for WDCs, a failure will release is attachment to DCNs, which would then await broadcasting from neighboring WDCs.

To expand on the resilience of D-WSN, we present in Fig. 6 two elaborate scenarios for functional operations that were disrupted by different types of failures. In the base scenario, we assume a single D-WSN with three DCNs and ten WDCs, of which two WDCs are currently not associated with any DCNs, and remain in a dormant state (DCNs 3 and 10). The base case depicted at the top demonstrates the functional stage of the D-WSN, after all components have converged to their operational states (as per the state machines depicted in Figs. 3 and 4).

In Scenario A, some failure (e.g., fire or circuit failure) result in component-level failures for WDCs 4 and 6. As such, their associations with DCNs B and A, respectively, are severed. The caption below scenario A depicts the resilience of D-WSN in reacting to these component failures, and expands on the reassociations and state transitions for DCNs and WDCs.

On the other hand, scenario B captures the case when a core DCN node fails, with the ensuing impact on the core network topology as well as attached WDCs. Also, this type of failure is mitigated and the reaction of D-WSN is explained in the caption.

V. Performance Evaluation

The performance evaluation of D-WSN was carried out via simulations. We contrast our results to two dominant paradigms in WSNs; namely mainstream homogeneous sensing networks (HSNs) and participatory sensing networks (PSNs). We contend that HSNs sustain advantages in resilience (by sheer density variation) and that PSNs benefit from a dynamic architecture, based on the heterogeneity of participating nodes.

D-WSNs on the other hand reap advantages from both paradigms. The distribution of tasks between nodes is homogeneous, yet the network architecture is heterogeneous and dynamic in its changing associations; between DCNs and WDCs. The remainder of this section details the simulation setup for this performance evaluation, and elaborates on metrics and parameters eliciting the performance of these three paradigms.

A. Simulation Environment and Experiment Setup

Our simulations have been carried out over MATLAB. In all three paradigms, the SNs/components are randomly distributed over a fixed grid with uniform density. The grid is 500 m $\times$ 500 m. The core challenge in experiment setup was managing equivalent scenarios. That is, ensuring that varying network parameters would not infringe the representativeness of the metrics.

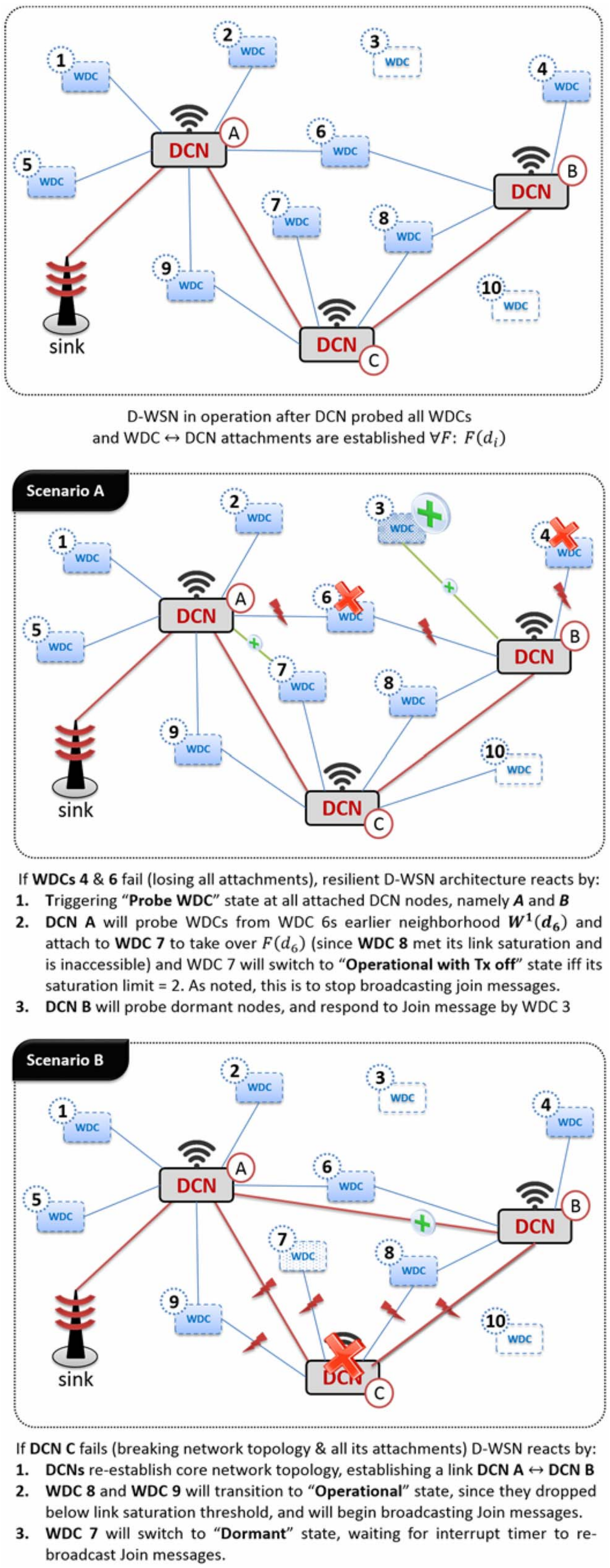

Fig. 6. Two independent scenarios detailing the resilience of D-WSN to failures of DCNs and WDCs, and the entailing reactions.

The ensuing experiments were carried on a network with 30 DCNs and 60 WDCs under the D-WSN model, and with an equivalent 100 nodes under the HSN and PSN models.

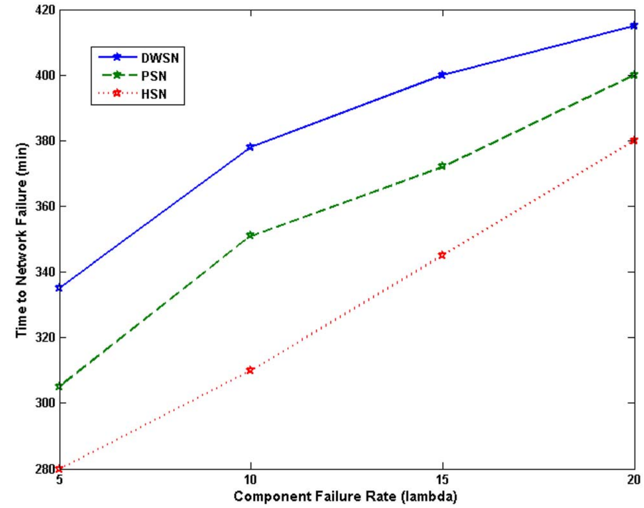

Fig. 7. Impact of failure rate (shown in decreasing occurrence, i.e., increasing average arrival rate) on functional network lifetime (minutes).

Each WDC was triggered for association/de-association as per energy changes or failures (detailed in Section V-C) and following the FSM presented in Fig. 5.

B. Performance Metrics and Network Parameters

We vary network parameters of the three paradigms under study (HSN, PSN, and D-WSN), to achieve equivalent scenarios in terms of functional capacity. We carry out our experiments to evaluate the performance of these paradigms over network-wide goals. Namely, we experiment with metrics for energy conservation, resilience to failures and faults, and the overall cost of the network; detailed as follows.

1) Total Energy: We measure the total energy consumed by the network to carry out a set of functional tasks. That is, given a certain application with a predetermined set of functional requirements, what would be the operational energy impact on each of the networks. We then vary the load to experiment with network scalability, and report the total energy used in Joules.

2) Nodal Resilience to Failures: Given a functional deployment for all three networking paradigms, we vary the failure rates of components and nodes to measure the resilience of the network as a whole. We adopt the definition of failures highlighted in Section IV-D.

3) Deployment Cost: Each component in any network has a cost. It contributes to the price of individual nodes, and overall network design as a constraint. We thus vary application requirements, in terms of functionalities, and study the impact on network cost to fulfill these requirements. This is intrinsically a monetary value, measured in units.

C. Performance Results

We report the results obtained for the three performance metrics highlighted in the previous section, under the aforementioned experiment setup. In Fig. 7, we demonstrate the impact of increasing the functional requirements, in terms of application functional requests (a control variable across the

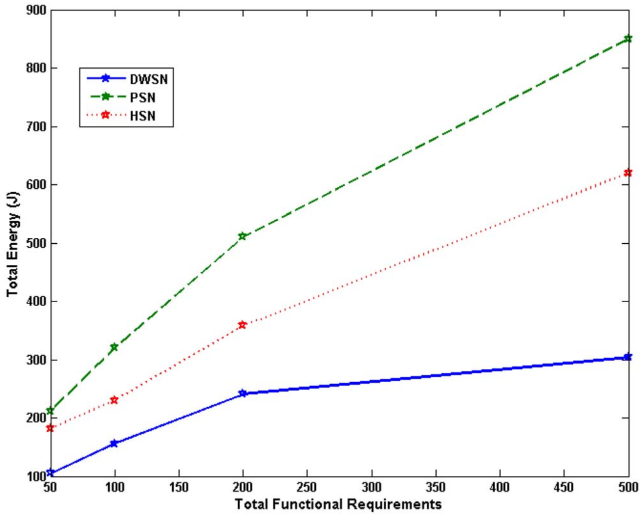

Fig. 8. Effect of increasing the functional requirements on network energy consumption.

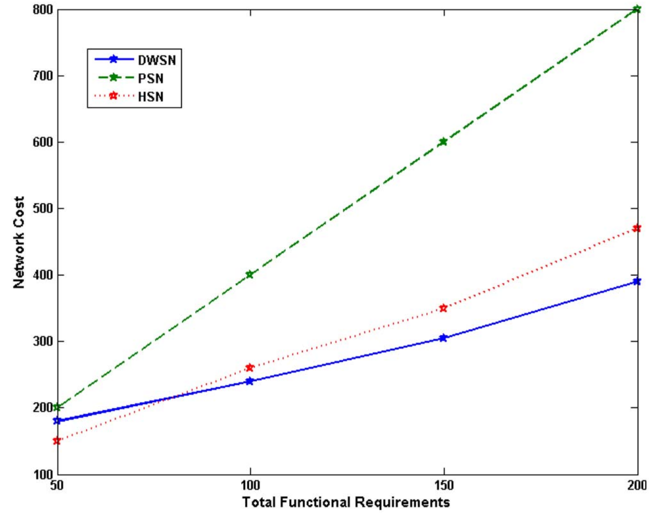

Fig. 9. Cost of deployment (monetary value) to sustain increasing functional requirements.

three paradigms) on the energy dissipation in the network. Evidently they all show an increase in energy consumption, yet PSNs and HSNs suffer the most due to the coupling between nodal load and functional requirement. Thus, to perform more functions, more nodes need to be switched on to operate, causing the spike in power usage.

In Fig. 8, we demonstrate how nodal/component failures impact network lifetime. We define network lifetime as the time it takes before the network could no longer fulfill all functional requests, as adopted by a thorough study in [2]. We vary the occurrence (arrival) rate of faults, with exponential interarrival times following a Poisson distribution, with averages (λ) denoted on the x-axis. Evidently, the HSN paradigm eventually loses momentum as the density of operational nodes falls. Similarly, PSNs lose functionality due to the tendency of public nodes to drop out (fail) as they leave the network, and the varying density impacts availability in zones of need.

Finally, we demonstrate the hiking costs of requesting more functional requirements in static networks. That is, in HSNs where nodes have a 1–1 matching of functionality to density, and PSNs where nodes may have varying capabilities, but we still have a constraint on the number of functional requests allowed per node. We have assigned a unit cost to each functional component (e.g., Tx 2 units, MCU 1 unit, Humidity sensor and light sensors 2 units), and contrasted the results as depicted in Fig. 9.

VI. Conclusion

We argued for a dynamic paradigm that integrates IoT things in a real-time association model. With a growing abundance of wireless technologies that enable sensing and communication, and interact over multiple access mediums, it is imperative to reassess our view of what a WSN is, and how its interplay with IoT should manifest. In the near future, most of the sensing applications, especially in urban settings, will not rely on dedicated and overpriced WSNs. In fact, sensing systems provided by smart vehicles and smartphone are already changing our view of WSN capabilities. However, a major hindrance in the latter technologies is their isolated operation.

We presented a paradigm that enables dynamic nodes to change their functional mandates post deployment, enabling a WSN that can change its application span over time. We contend that the presented DCNs will be easily replaceable with smartphones, and WDCs will evolve from resources offered by a myriad of wirelessly enabled devices. We realize a true opportunity for synergy in the IoT, and hence presented this paradigm to shift the operation of WSNs from its isolated progression to a ubiquitous IoT enabler.

References

[1] S. Oteafy and H. Hassanein, *Dynamic Wireless Sensor Networks*. Hoboken, NJ, USA: Wiley, 2014.

[2] I. Dietrich and F. Dressler, “On the lifetime of wireless sensor networks,” *ACM Trans. Sensor Netw.*, vol. 5, no. 1, 2009, Art. no. 5.

[3] A. B. M. A. Al Islam, M. S. Hossain, V. Raghunathan, and Y. C. Hu, “Backpacking: Energy-efficient deployment of heterogeneous radios in multi-radio high-data-rate wireless sensor networks,” *IEEE Access*, vol. 2, pp. 1281–1306, 2014.

[4] Y. Gu, L. Cheng, J. Niu, T. He, and D. H.-C. Du, “Achieving asymmetric sensing coverage for duty cycled wireless sensor networks,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 12, pp. 3076–3087, Dec. 2014.

[5] K. Mikhaylov, J. Petäjäjärvi, M. Mäkeläinen, A. Paatelma, and T. Hänninen, “Demo: Modular multi-radio wireless sensor platform for IoT trials with plug&play module connection,” in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, Paris, France, 2015, pp. 188–189.

[6] S. M. A. Oteafy, H. M. Aboelfotoh, and H. S. Hassanein, “Decentralized multi-level duty cycling in sensor networks,” in *Proc. IEEE Glob. Telecommun. Conf.*, New Orleans, LA, USA, 2008, pp. 1–5.

[7] S. M. A. Oteafy and H. S. Hassanein, “Re-usable resources in wireless sensor networks: A linear optimization for a novel application overlay paradigm over multiple networks,” in *Proc. Glob. Telecommun. Conf.*, Houston, TX, USA, 2011, pp. 1–5.

[8] G. Degirmenci, J. P. Kharoufeh, and O. A. Prokopyev, “Maximizing the lifetime of query-based wireless sensor networks,” *ACM Trans. Sensor Netw.*, vol. 10, no. 4, pp. 1–24, 2014.

[9] S. Lin *et al.*, “Toward stable network performance in wireless sensor networks: A multilevel perspective,” *ACM Trans. Sensor Netw.*, vol. 11, no. 3, pp. 1–26, 2015.

[10] A. Munir, A. Gordon-Ross, and S. Ranka, “Multi-core embedded wireless sensor networks: Architecture and applications,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 6, pp. 1553–1562, Jun. 2014.

[11] J. Liu, N. Kato, J. Ma, and N. Kadowaki, "Device-to-device communication in LTE-advanced networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1923–1940, 4th Quart., 2015.
[12] Z. Sheng *et al.*, "A survey on the IETF protocol suite for the Internet of Things: Standards, challenges, and opportunities," *IEEE Wireless Commun.*, vol. 20, no. 6, pp. 91–98, Dec. 2013.
[13] J. Matias, J. Garay, N. Toledo, J. Unzilla, and E. Jacob, "Toward an SDN-enabled NFV architecture," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 187–193, Apr. 2015.
[14] M. Blackstock and R. Lea, "IoT interoperability: A hub-based approach," in *Proc. Int. Conf. Internet Things (IOT)*, Cambridge, MA, USA, 2014, pp. 79–84.
[15] C. Liang and F. R. Yu, "Wireless network virtualization: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 358–380, 1st Quart., 2015.
[16] S. Faria and V. Kostakos, "A scalable sensor middleware for social end-user programming," in *Mobile Context Awareness*. London, U.K.: Springer, 2012, pp. 115–131.
[17] "ZigBee specification," ZigBee Alliance, Davis, CA, USA, ZigBee Doc. 053474r13, pp. 344–346, 2006.
[18] X. Xie, H. Chen, and H. Wu, "Bargain-based stimulation mechanism for selfish mobile nodes in participatory sensing network," in *Proc. IEEE Sensor Mesh Ad Hoc Commun. Netw. (SECON)*, Rome, Italy, 2009, pp. 1–9.
[19] S. M. A. Oteafy and H. S. Hassanein, "Component-based wireless sensor networks: A dynamic paradigm for synergetic and resilient architectures," in *Proc. IEEE 38th Conf. Local Comput. Netw. (LCN)*, Sydney, NSW, Australia, 2013, pp. 735–738.
[20] I. Koutsopoulos, "Optimal incentive-driven design of participatory sensing systems," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Turin, Italy, 2013, pp. 1402–1410.
[21] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Clarke, "Middleware for Internet of Things: A survey," *IEEE Internet Things J.*, vol. 3, no. 1, pp. 70–95, Feb. 2016.

Sharief M.A. Oteafy (S'08–M'13) received the Ph.D. degree in 2013 from Queen's University, focusing on adaptive resource management in next-generation sensing networks, introducing the notion of organic WSNs that adapt to their environment and scale in functionality with resource augmentation from Queen's University, Kingston, ON, Canada, in 2013.

He is an Adjunct Assistant Professor with the School of Computing, Queen's University. He co-authored *Dynamic Wireless Sensor Networks* (Wiley). His current research focuses on dynamic architectures for enabling large-scale synergy with the Internet of Things; encompassing dynamic resource management across multiple platforms, in addition to managing the proliferation of big sensed data.

Dr. Oteafy is actively engaged in the IEEE Communications Society. He has been an ACM member since 2008. He is an active member of the IEEE ComSoc Standards Association as the AHSN Standards Liaison and a member in the ComSoc Tactile Internet standard WG. He has presented over 40 peer-reviewed publications in sensing systems and IoT. He has co-chaired a number of IEEE workshops, in conjunction with IEEE ICC and IEEE LCN conferences, and served on the TPC of numerous IEEE and ACM symposia. He has delivered tutorials on big sensed data management in IEEE ICC, IEEE CAMAD, and IEEE Globecom conferences. He serves as an Associate Editor for IEEE ACCESS.

Hossam S. Hassanein (S'86–M'90–SM'06) received the Ph.D. degree in computing science from the University of Alberta, Edmonton, AB, Canada, in 1990.

He is a leading authority in the areas of broadband, wireless and mobile networks architecture, protocols, control, and performance evaluation. His record spans more than 500 publications in journals, conferences, and book chapters, in addition to numerous keynotes and plenary talks at flagship venues. He is also the Founder and Director of the Telecommunications Research Lab, School of Computing, Queen's University, Kingston, ON, Canada, with extensive international academic and industrial collaborations.

Dr. Hassanein is an IEEE Communications Society Distinguished Speaker (Distinguished Lecturer 2008–2010). He is the past Chair of the IEEE Communication Society Technical Committee on Ad hoc and Sensor Networks. He was the recipient of several recognitions and Best Papers Awards at top international conferences, and has led a number of symposia in flagship ComSoc conferences.