Algorithm Design for Data Communications in Duty-Cycled Wireless Sensor Networks: A Survey*

Kai Han
School of Computer Science
Zhongyuan University of Technology
Zhengzhou, China
Email: hankai@ntu.edu.sg

Jun Luo
School of Computer Engineering
Nanyang Technological University
Singapore
Email: junluo@ntu.edu.sg

Yang Liu
School of Information Sciences and Engineering
Henan University of Technology
Zhengzhou, China
Email: enjoyang@gmail.com

Athanasios V. Vasilakos
Department of Electrical and Computer Engineering
National Technical University of Athens
Athens, Greece
Email: vasilako@ath.forthnet.gr

***Abstract*—Though duty-cycling has long been a critical mechanism for energy conservation in wireless sensor networks, it is only recently that research efforts have been put to design data communication protocols that perform efficiently in *Duty-Cycled Wireless Sensor Networks* (DC-WSNs). In this article, we survey these research problems, aiming at revealing insights into the following three key questions: i) what are the meaningful (algorithm design) problems for DC-WSNs? ii) which problems have been studied and which have not? and iii) what are the essential techniques behind the existing solutions? All these insights may serve as motivations and inspirations for further developments in this field.**

I. INTRODUCTION

As *Wireless Sensor Networks* (WSNs) are deemed as powerful tools in monitoring physical events, research topics related to WSNs have been heavily investigated in the last decade. Among all these topics, energy efficiency of WSNs is arguably the hottest one and hence has attracted a lot of attention. *Duty-cycling* (DC) has been considered as one of the most important techniques for energy conservation in WSNs [1], and actually it was born almost at the same time as WSNs [2], [3]. In a nutshell, DC temporarily "shuts down" sensor nodes from time to time and thus saves their energy dissipations. As it was generally assumed that most WSNs do not carry heavy data traffics, applying DC was believed to have no effect on communication protocols and was thus considered rather independently from the algorithm design in the network layer of WSNs.

However, the assumption of no heavy traffic in WSNs is becoming increasingly unreasonable lately, especially due to the more demanding applications of WSNs (e.g., multimedia contents). In fact, applying DC leads to a time-varying network topology and a WSN may even become disconnected at certain points in time. In particular, the so called *Wireless Multicast Advantage* (WMA) is significantly weakened by DC, as one local broadcast cannot reach all neighboring nodes due to DC. As a result, all performance objectives (e.g., capacity, delay, and energy efficiency) of a data communication protocol are affected by DC; this necessitates the incorporation of algorithm design for data communications in DC-WSNs into the research agenda [4]. While quite a few efforts have been made in the past few years on the related topics, the whole area of data communication protocol design for DC-WSNs is still at the early stage,[1] compared with other relatively mature areas in WSNs. Therefore, we believe that it is the right time to review the existing design methodologies and to motivate future research directions.

II. CLASSIFICATIONS OF MODELS AND PROBLEMS

Given the many aspects of network protocol design in DC-WSNs, we introduce two ways of classifying them. We first look at different models for DC in WSNs, then we describe different problems that may potentially be re-defined due to the application of DC. The terminologies discussed in this section will be carried throughout the remaining of this paper.

A. Duty-Cycle Modeling

In DC-WSNs, a wireless sensor node switches between *active* and *dormant* states *periodically*, and the *working periods* of different sensor nodes usually have the same length, say, a constant T. A node can wake up its transceiver to transmit a packet at any time, but can only receive a packet when it is active. These are widely adopted assumptions in the literature, which are rather realistic. However, the exact models of DC taken by the existing proposals for DC-WSNs may differ in terms of synchronization levels, duty-cycling behaviors, link reliability and etc..

---

*This work is supported in part by the Start-up Grant of NTU and AcRF Tier 2 Grant ARC15/11, as well as National Natural Science Foundation of China under Grant no. 61103007 and 61003052.

[1] Quite different from the earlier research on DC whose goal was about how to deploy DC in WSNs, this new body of work focuses more on how to make algorithms work efficiently along with DC.

TABLE I CLASSIFICATION OF ALGORITHMS FOR DATA COMMUNICATIONS IN DC-WSNS

	Energy-efficiency	Latency	Lifetime	Capacity	Delivery ratio
Unicast	[5], [6]	[5], [6]		[6]	
Anycast	[4], [7]	[4], [7], [8]	[8]		[4], [7]
Broadcast	[9]–[12]	[9], [10], [12], [13]			[9], [10]
Multicast	[14], [15]	[14]			
Convergecast		[16]	[16], [17]		[17]

1) Synchronized vs. Non-synchronized DC: If global or local synchronization is assumed for a DC-WSN [2], the model is amenable to graph-theoretical characterizations because one may augment the original connectivity graph of the WSN by associating a binary (active or dormant) state with each node. As a result, most combinatorial solutions make such an assumption [4], [7], [9], [11], [13]–[15]. If one wants to eliminate the overhead for synchronization, the working periods of different nodes may not align with each other [3]. Therefore, the residual active/dormant time is a random variable, and hence a stochastic modeling technique has to be used [5], [8].

2) Generalized vs. Simplified DC: Though a duty-cycling model often assumes an identical and fixed working period T, each node in general can determine its active/dormant schedule without any constraints. Such a generalized DC model is frequently adopted, as indicated in the literature [4], [9], [15]. However, simplified DC model is also considered to facilitate algorithm design; such simplifications often impose certain restrictions on the active/dormant schedules within a working period. For example, a single-active-time-slot model is used in [5], [11], [13], i.e., there exists only one active time slot in a working period of any node. Both [14] and [6] allow for multiple active time slots but [14] requires the (variable number of) active time slots to be consecutive in a working period while [6] assumes that the proportion of active time slots in a working period of any node equals to a predefined constant. Following the convention of stochastic analysis, the work in [8] assumes that each node wakes up independently according to the Poisson process.

3) Static vs. Dynamic DC: Many algorithms we discuss in this paper assume that the active/dormant time slots of any network node are static, i.e., the active/dormant schedule is pre-determined for each node and can not be changed by the algorithms [4], [5], [8]–[16]. However, there also exist algorithms adopting dynamic duty cycling models, such as [6], [7], [8][2], [17]. The common idea behind these proposals is that the active/dormant schedules of sensor nodes can be dynamically controlled such that the nodes are awaken only when they are needed, hence more energy can be conserved because the power consumption for idle-listening is reduced and retransmissions caused by collisions are limited. Besides energy conservation, other optimization goals affected by dynamic DC (such as latency and capacity) are also considered in these proposals.

4) Reliable vs. Unreliable Links: As with conventional WSNs, link reliability issue persists in DC-WSNs. However, considering both DC and link reliability issue may significantly complicates the problem. Therefore, many research proposals neglect the latter, with some exceptions [4], [7], [9], [10], [16], [17]. In particular, most combinatorial approaches tends to avoid the link reliability issue [11], [13]–[15], as it simply adds the dimension of the resulting problems.

B. Problem Categorization

As the focus of this survey is the design of data communication algorithms, is natural to categorize the problems in DC-WSNs according to their respective functionalities. This leads to the following five categories: *unicast* [5], [6], *anycast* [4], [7], [8], *broadcast* [9]–[13], *multicast* [14], [15] and *convergecast* [16], [17]. From application point of view, unicast/anycast are often used for information exchanges within pairs of sensor nodes, broadcast/muticast are needed for disseminating commands or codes to sensor nodes, and convergecast serves mainly for data collection from sensor nodes.

For each type of communication protocols, there can be several performance objectives, such as *energy-efficiency*, *latency*, *lifetime*, *capacity* and *delivery ratio*. Each proposal in the literature can often optimize one or two such objectives. These observations leads to a "two-dimension" view on the data communication protocol design in DC-WSNs, as we illustrated in Table I. For those "crossing points" between problems and objectives that are filled with references, we will give detailed discussions in Section III. The remaining "empty region" leaves us spaces for future work.

III. SURVEY OF DATA COMMUNICATION PROTOCOLS IN DC-WSNS

A. Unicast Algorithms

In unicasting, each forwarding node must properly select one neighboring node as its next-hop *relaying node*, and transmit data packets to this relay at right time slots, such that no data are sent to sleeping nodes while certain objectives are optimized. These objectives include latency, capacity and energy-efficiency [5], [6], and they may conflict with each other when duty-cycling is involved.

When the proportion of active time slots in a working period of WSN nodes gets lower, more energy can be conserved, but it also increases unicast latency and decreases unicast capacity. This trade-off was observed by Guha *et al.* [6], and they aim to optimize the unicast latency and capacity while maintaining energy conservation at a certain level. They propose a sleep scheduling method called Green-Wave Sleep Scheduling, or

---

[2] The work in [8] actually consists of two parts: it considers both the static and dynamic DC model. We will see this in Section III-B.

GWSS, whose idea is borrowed from the theory of optimizing traffic lights. In essence, GWSS tries to make the active time slots of any node coincide with the time slots it receives unicast data packets. Whereas theoretical bounds of unicast latency and capacity are derived for GWSS, these bounds are based on special network topologies such as grid networks and random Poisson-distributed networks.

When unicasting in DC-WSNs is employed under a geometric routing framework, another trade-off arises between the delay of waiting for a next-hop node to wake up and the progress (in distance) that a packet can make towards the destination if sending it to this next-hop node. Observing this trade-off, Naveen *et al.* [5] study the problem of selecting next-hop relaying nodes such that the average packet delay is minimized subject to a constraint on the average packet progress. They formulate this problem as a Markov Decision Process (MDP) and find the optimal solution. The asynchronous duty-cycling model is adopted in [5].

B. Anycast Algorithms

In the unicast algorithms described above, each forwarding node designates one neighboring node as its next-hop relaying node, and it thus has to wait for the designated relaying node to wake up to receive a data packet. However, the extra delay and energy consumption resulted from duty-cycling in such unicast algorithms could be unacceptable, especially when the link reliability is taken into account or time synchronization is not in position. To solve this problem, an anycast scheme for data forwarding in DC-WSNs have been proposed [4], [8], where each forwarding node maintains a set of candidate neighboring nodes (we denote it by *relaying set*) as its next-hop relaying nodes, and the single-hop data forwarding is regarded successful if any node in the relaying set receives the data. The key problems in such a scheme are how to select the relaying sets and how to prioritize the nodes in the relaying sets to achieve specific optimization objectives.

Gu *et al.* [4] consider the link reliability issue and propose an anycast framework called Dynamic Switch-based Forwarding (DSF) for end-to-end data transmissions. They model a DC-WSN as a *time expanded network* (see an example shown in Fig. 1), and propose several dynamic programming algorithms for selecting relaying sets to reach different optimization objectives including the expected data delivery ratio, the expected communication delay and the expected energy consumption. The nodes in any relaying set are prioritized according to their wake-up times, and each forwarding node transmits a data packet to its relaying nodes one by one until one successful transmission occurs. However, such a transmission scheme requires local synchronization, because each forwarding node needs to be fully aware of its neighbors' wake-up times. The performance of DSF is tested both by simulations and by a testbed running MicaZ motes.

Similar to [4], the work in [7] also prioritizes relaying nodes according to their wake-up times. However, [7] adopts a dynamic DC model; the nodes' wake-up times are dynamically adjusted so that proper relaying sets are selected to reduce retransmissions (hence energy consumption), as well as to improve the data delivery ratio under real-time constraints.

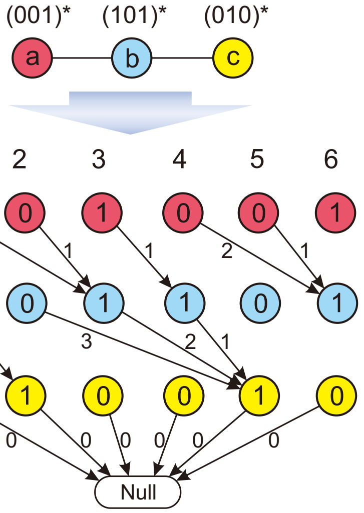

Fig. 1. The top figure shows the original network topology: a linear network of three nodes. The regular expressions denote the active/dormant schedules of the nodes, where 0 and 1 represent the dormant and active states, respectively. The bottom figure is a time-expanded network constructed from the top one: each node (a, b, or c) is expanded to multiple copies, with numbers shown in the top line represent a sequence of time slots. There is a directed link with delay $j - i$ between node $u : u \in \{a, b, c\}$ at time slot i and node $v : v \in \{b, c\}$ at time slot j iff: (i) u and v are adjacent in the original topology and (ii) j is the first active time slot of node v after time slot i. Finally, all the time expanded nodes of the destination node c are connected to a null node with edges of delay 0.

The work of Kim *et al.* [8] has adopted an asynchronous duty-cycling model, but with a constraint that each node wakes up independently according to the Poisson process. The minimum-delay anycast problem under their duty-cycling model is translated into an instance of the *stochastic shortest path* problem, and a dynamic programming-based scheme is presented to find the optimal relaying sets as well as the prioritizations of the relaying nodes. Based on this result, they also study the trade-off between network lifetime and anycast delay, and propose an anycast algorithm that can maximize the network lifetime by adjusting the wake-up rates of the nodes subject to a constraint on the expected end-to-end packet-delivery delay.

C. Broadcast Algorithms

In always-active WSNs, the WMA feature is heavily leveraged to design efficient broadcast algorithms for various optimization goals such as latency and energy efficiency; the resulting combinatorial problems are usually NP-hard (e.g., [18]). In DC-WSNs, designing efficient broadcast algorithms is more tricky due to the lack of WMA, and in the worst case one can only use unicast operations to complete a broadcast request: imagine an extreme case where no nodes have common active time slots. Fortunately, in an average case, nodes may have common active time slots, and a forwarding node can schedule its transmissions at its neighboring nodes' common active time slots to get a better broadcast performance. We call this idea as *Partial WMA* (PWMA). Actually, most of current broadcasting algorithms for DC-WSNs are designed based on PWMA to reach different optimization goals such as energy-efficiency, latency and delivery ratio [9], [11]–[13].

From an algorithmic point of view, we classify the current broadcast algorithms for DC-WSNs into two categories: i) designing approximate algorithms with provable approximation ratios that bound the worst-case performance of the approximation algorithms[3] [11], [13], and ii) designing intelligent heuristics without providing theoretical performance bounds [9], [10], [12].

1) Approximate Broadcast Algorithms: Jiao *et al.* [13] study the collision-free minimum-latency broadcast scheduling problem in DC-WSNs under the *graph-based interference model*, where a node cannot receive anything if more than one of its neighboring nodes send it messages at the same time. They prove the NP-hardness of the problem, and then propose several centralized approximation algorithms with provable approximation ratios for one-to-all broadcasting, all-to-all broadcasting without aggregation, and all-to-all broadcasting with aggregation [20], respectively. The idea behind these algorithms is a *Maximum Independent Set* (MIS) based link scheduling, similar to [20]. In their algorithms, a shortest path tree based on the active time slots of the nodes is first constructed, then some MIS's are constructed layer by layer. A broadcast tree is constructed by selecting some connector nodes to join these MIS's. Based on the broadcast tree, they schedule the transmissions to avoid interference, as well as to leverage on PWMA to reduce the broadcast latency.

In contrast to the latency-efficient broadcast algorithms in [13], Hong *et al.* [11] focus on minimizing the number of total transmissions in a one-to-all broadcast session. They observed that, the better a broadcast tree utilizes PWMA, the more transmissions can be reduced by the broadcast tree (as illustrated by a simply example in Fig. 2). To build a minimum-transmission broadcast tree, they first group the wireless nodes according to their active time slots, then find a small number of connected nodes that can cover the nodes in each group. Their algorithm can be run in a distributed manner with a constant approximation ratio and $\mathcal{O}(|V|)$ time and message complexities.

A common limitation of the work in [11], [13] is that they all adopt the single-active-time-slot model, which could be uncommon in real WSNs. Adapting their work to a more general duty-cycling model can be non-trivial.

2) Heuristic-based broadcast algorithms: As illustrated before, a broadcasting node may need to transmit a packet multiple times to reach its neighboring nodes due to duty-cycling. This situation gets even worse when the link-unreliability is taken into account, because a transmission can fail simply because of the poor qualities of wireless links. In [9], Guo *et al.* aim to design efficient broadcast algorithms to resist the joint impact of duty-cycling and link-unreliability. They argue that pure tree-like broadcasting can be inefficient in such an environment, and use additional links outside a broadcast tree for transmitting data if it makes a data packet reach a tree node statistically earlier than the expected receiving time by using the broadcast tree, thus the transmission redundancy and delay can be reduced. Similar to [4], local synchronization is required in [9].

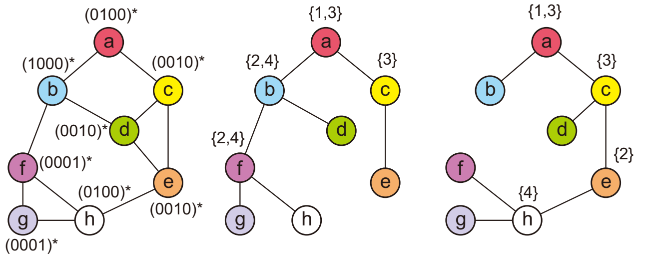

Fig. 2. Given the network topology and active/dormant schedules shown in the left figure, the broadcast tree chosen in the central figure ends up with 7 transmissions starting in the source node a, whereas the broadcast tree chosen in the right figure yields only 5 transmissions. The numbers associated with each tree node in the middle figure and the right figure indicate the time slots in a working period that the tree node needs to transmit a data packet, subject to the broadcast tree it belongs. It is straightforward to see a similar phenomenon persists in a multicast session, as we will discuss in Section III-D.

Unlike [9], the ADB (Asynchronous Duty-cycle Broadcasting) protocol proposed by Sun *et al.* [10] supports broadcasting in DC-WSNs with asynchronous duty-cycling, and does not rely on any broadcast trees. To avoid blind flooding, ADB augments the MAC-layer packets to exchange extra information between neighboring nodes, such as link qualities and current packet-receiving status. With these information, the transmissions of neighboring nodes can be coordinated, and a local election process is conducted to select proper nodes for next-hop transmissions. Consequently, ADB has the potential to reduce the energy consumption and delivery latency in broadcasting.

We note that although no theoretical performance bounds are presented in [9], [10], they have considered more realistic settings of WSNs compared with the algorithms in [11], [13], including more general duty-cycling models, unreliable wireless links and MAC-level details. The performance of the algorithms presented in [9], [10] are tested on real WSN platforms running TinyOS on MicaZ motes, as opposed to simulations done in, for example, [11], [13].

Hybrid-cast [12] is another heuristic-based broadcast algorithm for asynchronous DC-WSNs. In Hybrid-cast, a node sends out a beacon message when it wakes up, and a forwarding node defers its transmitting after hearing the first beacon message in order to accommodate more neighboring nodes in one broadcast. However, compared with [9], [10], link reliability issue is not considered by [12], and the asynchronous sleep-scheduling models adopted in [12] are more restricted, such as the single-active-time-slot model.

---

[3] For the concepts on approximation algorithms, the interested readers are kindly referred to [19].

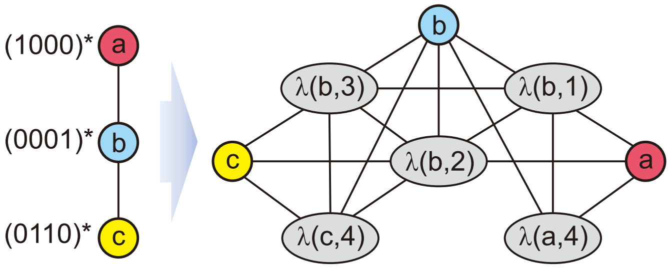

Fig. 3. The left figure shows the original network topology: a network which is similar to the linear network in Fig. 1. The right figure is the network graph transformed from the left one. A node $\lambda(u,i)$ is created for node $u : u \in \{a,b,c\}$ iff any neighboring node of u is active during the i-th time slot in a working period. As a result, both time and connectivity are multiplexed into one graph. Based on this time modulated connectivity graph, the nodes in the transformed graph is further connected by a special method to facilitate the algorithm design in [15].

D. Multicast Algorithms

Designing multicast algorithms in DC-WSNs may often be harder than designing broadcasting algorithms, since broadcast can be seen as a special case of multicast, and the transmissions to non-destination nodes in multicast should be reduced to the greatest extent. Currently, only a few multicast algorithms have been designed for DC-WSNs [14] [15].

Observing a phenomenon similar to that shown in Fig. 2 for multicast, Su *et al.* [14] present optimal algorithms to minimize the energy consumption for multicasting in DC-WSNs. The idea of [14] is to transform the original network graph into a directed auxiliary graph where the transmitting time slots of each node are represented by some“widget” nodes. Consequently, the Minimum Energy Multicasting (MEM) problem is converted into the problem of finding a Directed Steiner Tree (DST) in the auxiliary graph. In finding the final DST, a dynamic programming approach is adopted to gradually expand and merge small intermediate trees.

Han *et al.* [15] later propose a polynomial-time approximation algorithm with provable approximation ratio for the MEM problem in DC-WSNs. They adopt a generic duty-cycling model where each node can determine its active/dormant time slots without any constraints. Again, their algorithm is based on graph transformation, but their graph transformation method is more delicate than that in [14] (see an example shown in Fig. 3). Taking advantage of the special structure of their transformed network graph, they design an order-optimal approximation algorithm with the approximation ratio being a logarithm of the maximum node degree, and the time complexity of their algorithm is $\mathcal{O}(|V|^2)$.

Both [14] and [15] use graph transformation methods to tackle the MEM problem in DC-WSNs. However, [14] adopts a restricted duty-cycling model where the active time slots of any node must be consecutive. Moreover, the dynamic programming based algorithm presented in [14] has an exponential time complexity, so it only works for the scenarios where the number of multicast destinations are small (e.g., a few sinks). These drawbacks are eliminated in [15] by slightly sacrificing optimality.

E. Convergecast Algorithms

Convergecast is a representative traffic pattern of WSNs, due to the need for collecting sensory data at a few number of (if not one) sinks. Roughly speaking, convergecast can be further classified based on whether or not data are aggregated at the relaying nodes. Clearly, introducing DC will inevitably affect the performance of convergecast algorithms such as latency and lifetime.

An interesting problem for data aggregation in DC-WSNs is the trade-off between lifetime and latency, i.e., when a node waits a longer time for aggregating more data before one transmission, it may prolong its lifetime but at the cost of higher data-delivery latency. The work in [16] has studied this problem, and proposed heuristics for prolonging network lifetime under certain data-delivery latency bounds. A highlight of the heuristics provided in [16] is that each node adjusts its aggregation holding time in an adaptive manner by communicating with its neighboring nodes, hence the lifetime of the nodes in the whole network can be gradually adjusted to a balanced status and the nodal lifetime bottlenecks are reduced. However, a static DC model is adopted in [16].

In contrast with [16], DISSense [17] provides data collection services and exploits the power of dynamic DC to prolong the network lifetime. In [17], the length of each node's active phases is dynamically adjusted to guarantee both a very low duty cycle and a high data delivery ratio. We note that both [16] and [17] are heuristics-based approaches and use testbeds to evaluate their algorithms. However, the work in [17] also provides extensive simulations using the TOSSIM simulator, which further proves its suitability for real applications such as long-term environmental monitoring.

IV. FUTURE DIRECTIONS

Though many algorithms have been designed for data communications in DC-WSNs, bearing objectives ranging from energy-efficiency to delivery ratio, there are still many unexplored algorithm design issues for DC-WSNs. Again based on Table I, we list a few possible future research topics in the following.

A. Capacity Scaling Laws

Since the seminal work of Gupta and Kumar [21], the capacity scaling laws of always-active WSNs have been extensively studied in the literature. However, little work has been done for DC-WSNs. Although Guha *et al.* [6] has made a first step towards this direction, we believe more work needs to be done, e.g., what is the broadcast/multicast capacity bounds in DC-WSNs with arbitrary or random topologies? how these capacities can be affected by adjusting the active/dormant schedule of each node? We believe that, rather than sticking to the convention information theoretical approach, the combinatorial techniques (e.g., [4], [15]) may play a role in this front.

B. Multi-Criteria Approximation Algorithms

Most existing efforts on designing approximation algorithms with provable approximation ratios for DC-WSNs have only one optimization objective. To meet the demands of practical applications, designing multi-criteria approximation algorithms that optimize multiple QoS goals simultaneously needs further investigation. For example, designing a multicast algorithm with provable worst-case bounds on multiple performance targets (e.g., energy consumption and latency, or latency and lifetime) would be of interest for the future research.

C. Further Research on Convergecast Algorithms

Though convergecast is a critical operation in WSNs and corresponding algorithms have been extensively studied in always-active WSNs, little work has been done to understand the theoretical performance bounds for convergecast algorithms (with or without aggregation) in DC-WSNs, as we have discussed in Section III-E. We believe that further research work on this topic is necessary to tackle various optimization objectives such as latency, capacity, energy-efficiency and load balancing.

D. Joint Load Balancing and Routing

Load balancing is, to a large extent, related to lifetime [22], and both have not been well studied under DC-WSNs. A possible approach is to assume that the routing is already fixed, so the load balancing problem becomes a stand-alone problem. However, it would be more intriguing but challenging if routing (for all types of communication listed in Table I) is jointly optimized with transmission schedules for load balancing.

V. CONCLUSIONS

In this article, we have surveyed the algorithms designed for data communication in DC-WSNs. We have discussed different duty-cycling models used by existing approaches, as well as data communication protocols proposed for DC-WSNs under five categoris: unicast, anycast, broadcast, multicast, and convergecast. The main insight revealed by our survey is twofold: i) a well characterized problem space, distinguishing well-studied aspects from potential future research directions, and ii) a concise exposition of the key techniques behind the proposed algorithms.

REFERENCES

[1] G. Anastasi, M. Conti, M. D. Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, 2009.

[2] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless wensor networks," in *Proc. IEEE INFOCOM*, 2002, pp. 1567–1576.

[3] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. ACM SenSys*, 2004, pp. 95–107.

[4] Y. Gu and T. He, "Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 12, pp. 1741–1754, 2011.

[5] K. Naveen and A. Kumar, "Tunable locally-optimal geographical forwarding in wireless sensor networks with sleep-wake cycling nodes," in *Proc. IEEE INFOCOM*, 2010, pp. 920–928.

[6] S. Guha, P. Basu, C.-K. Chau, and R. Gibbens, "Green wave sleep scheduling: Optimizing latency and throughput in duty cycling wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1595–1604, 2011.

[7] F. Yang and I. Aug-Blum, "Delivery ratio-maximized wakeup scheduling for ultra-low duty-cycled wsns under real-time constraints," *Computer Networks*, vol. 55, no. 3, pp. 497–513, 2011.

[8] J. Kim, X. Lin, N. B. Shroff, and P. Sinha, "Minimizing delay and maximizing lifetime for wireless sensor networks with anycast," *IEEE/ACM Transactions on Networking*, vol. 18, pp. 515–528, 2010.

[9] S. Guo, Y. Gu, B. Jiang, and T. He, "Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links," in *Proc. ACM MobiCom*, 2009, pp. 133–144.

[10] Y. Sun, O. Gurewitz, S. Du, L. Tang, and D. B. Johnson, "Adb: an efficient multihop broadcast protocol based on asynchronous duty-cycling in wireless sensor networks," in *Proc. ACM SenSys*, 2009, pp. 43–56.

[11] J. Hong, J. Cao, W. Li, S. Lu, and D. Chen, "Minimum-transmission broadcast in uncoordinated duty-cycled wireless ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 307–318, 2010.

[12] S. Lai and B. Ravindran, "On multihop broadcast over adaptively duty-cycled wireless sensor networks," in *Proc. Distributed Computing in Sensor Systems(DCOSS)*, 2010, pp. 158–171.

[13] X. Jiao, W. Lou, J. Ma, J. Cao, X. Wang, and X. Zhou, "Duty-cycle-aware minimum latency broadcast scheduling in multi-hop wireless networks," in *Proc. IEEE ICDCS*, 2010, pp. 754–763.

[14] L. Su, B. Ding, Y. Yang, T. F. Abdelzaher, G. Cao, and J. C. Hou, "Ocast: optimal multicast routing protocol for wireless sensor networks," in *Proc. IEEE ICNP*, 2009, pp. 151–160.

[15] K. Han, Y. Liu, and J. Luo, "Duty-cycle-aware minimum-energy multicasting in wireless sensor networks," *IEEE/ACM Transactions on Networking*, 2012, in press. [Online]. Available: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6280608

[16] Z. Li, Y. Peng, D. Qiao, and W. Zhang, "Lba: lifetime balanced data aggregation in low duty cycle sensor networks," in *Proc. IEEE INFOCOM*, 2012, pp. 1844 –1852.

[17] U. Colesanti, S. Santini, and A. Vitaletti, "Dissense: an adaptive ultralow-power communication protocol for wireless sensor networks," in *Proc. Distributed Computing in Sensor Systems(DCOSS)*, 2011, pp. 1–10.

[18] R. Gandhi, A. Mishra, and S. Parthasarathy, "Minimizing broadcast latency and redundancy in ad hoc networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 4, pp. 840–851, 2008.

[19] D. Hochbaum, *Approximation algorithms for NP-hard problems*. PWS Publishing Company, 1997.

[20] P.-J. Wan, S. C.-H. Huang, L. Wang, Z. Wan, and X. Jia, "Minimum-latency aggregation scheduling in multihop wireless networks," in *Proc. ACM MobiHoc*, 2009, pp. 185–194.

[21] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[22] J. Luo and J.-P. Hubaux, "Joint sink mobility and routing to increase the lifetime of wireless sensor networks: the case of constrained mobility," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 871–884, 2010.

Kai Han received the B.S. and Ph.D. degrees from University of Science and Technology of China, Hefei, China, in 1997 and 2004, respectively, both in computer science. From 2005 to 2008, he has been a post-doctoral research fellow at the School of Mathematics, University of Science and Technology of China, Hefei, China. He is now an associate professor with the School of Computer Science, Zhongyuan University of Technology, China. His research interests include wireless ad hoc and sensor networks, distributed computing, combinational optimization, and applied operations research. This work is done when he is working as a research fellow at the School of Computer Engineering, Nanyang Technological University, Singapore.

Jun Luo got his PhD degree in computer science from EPFL (Swiss Federal Institute of Technology in Lausanne), Lausanne, Switzerland, in 2006. From 2006 to 2008, he has worked as a post-doctoral research fellow in the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada. In 2008, he joined the faculty of the School of Computer Engineering, Nanyang Technological University in Singapore, where he is currently an Assistant Professor. His research interests include wireless networking, distributed systems, multimedia protocols, network modeling and performance analysis, applied operations research, as well as network security. He is a Member of both IEEE and ACM. More information can be found at http://www3.ntu.edu.sg/home/junluo/.

Yang Liu received the M.Sc. and Ph.D. degrees from Northwestern Polytechnical University, Xian, China, in 2003 and 2007, respectively, both in computer science. She is currently an associate professor with the School of Information Sciences and Engineering, Henan University of Technology, Zhengzhou, China. Her research interests include distributed algorithms, algorithmic game theory, grid computing and cloud computing.

Athanasios V. Vasilakos is currently a visiting professor at National Technical University of Athens (NTUA), Athens, Greece. He has authored or co-authored over 200 technical papers in major international journals and conferences. He is author/coauthor of five books and 20 book chapters in the areas of data communications. Prof. Vasilakos has served as General Chair, Technical Program Committee Chair for many international conferences. He served or is serving as an Editor or/and Guest Editor for many technical journals, including IEEE JSAC and IEEE Communications Magazine. He is founding Editor-in-Chief of the International Journal of Adaptive and Autonomous Communications Systems (IJAACS, http://www.inderscience.com/ijaacs) and the International Journal of Arts and Technology (IJART, http://www.inderscience.com/ijart). He is the Chairman of the Council of Computing of the European Alliances for Innovation.