Ad Hoc Networks 7 (2009) 537–568

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Energy conservation in wireless sensor networks: A survey

Giuseppe Anastasi [a], Marco Conti [b,*], Mario Di Francesco [a,*], Andrea Passarella [b]

[a] Department of Information Engineering, University of Pisa, via Diotisalvi 1, 56122 Pisa, Italy
[b] Institute for Informatics and Telematics (IIT), National Research Council (CNR), via G. Moruzzi 1, 56124 Pisa, Italy

ARTICLE INFO

Article history:
Received 26 February 2008
Accepted 30 June 2008
Available online 29 July 2008

Keywords:
Wireless sensor networks
Survey
Energy efficiency
Power management

ABSTRACT

In the last years, wireless sensor networks (WSNs) have gained increasing attention from both the research community and actual users. As sensor nodes are generally battery-powered devices, the critical aspects to face concern how to reduce the energy consumption of nodes, so that the network lifetime can be extended to reasonable times. In this paper we first break down the energy consumption for the components of a typical sensor node, and discuss the main directions to energy conservation in WSNs. Then, we present a systematic and comprehensive taxonomy of the energy conservation schemes, which are subsequently discussed in depth. Special attention has been devoted to promising solutions which have not yet obtained a wide attention in the literature, such as techniques for energy efficient data acquisition. Finally we conclude the paper with insights for research directions about energy conservation in WSNs.

1. Introduction

A wireless sensor network consists of sensor nodes deployed over a geographical area for monitoring physical phenomena like temperature, humidity, vibrations, seismic events, and so on [5]. Typically, a sensor node is a tiny device that includes three basic components: a sensing subsystem for data acquisition from the physical surrounding environment, a processing subsystem for local data processing and storage, and a wireless communication subsystem for data transmission. In addition, a power source supplies the energy needed by the device to perform the programmed task. This power source often consists of a battery with a limited energy budget. In addition, it could be impossible or inconvenient to recharge the battery, because nodes may be deployed in a hostile or unpractical environment. On the other hand, the sensor network should have a lifetime long enough to fulfill the application requirements. In many cases a lifetime in the order of several months, or even years, may be required. Therefore, the crucial question is: "*how to prolong the network lifetime to such a long time?*"

In some cases it is possible to scavenge energy from the external environment [59] (e.g., by using solar cells as power source). However, external power supply sources often exhibit a non-continuous behavior so that an energy buffer (a battery) is needed as well. In any case, energy is a very critical resource and must be used very sparingly. Therefore, energy conservation is a key issue in the design of systems based on wireless sensor networks.

In this paper we will refer mainly to the sensor network model depicted in Fig. 1 and consisting of one *sink node* (or *base station*) and a (large) number of sensor nodes deployed over a large geographic area (*sensing field*). Data are transferred from sensor nodes to the sink through a multi-hop communication paradigm [5]. We will consider first the case in which both the sink and the sensor nodes are static (static sensor network). Then, we will also discuss energy conservation schemes for sensor networks with mobile elements in Section 6, in which a sparse sensor network architecture – where continuous end-to-end paths between sensor nodes and the sink might not be available – will be accounted as well.

* Corresponding authors. Tel.: +39 050 315 3062; fax: +39 050 315 2593 (M. Conti).
E-mail addresses: giuseppe.anastasi@iet.unipi.it (G. Anastasi), marco.conti@iit.cnr.it (M. Conti), mario.difrancesco@iet.unipi.it (M. Di Francesco), andrea.passarella@iit.cnr.it (A. Passarella).

1570-8705/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.adhoc.2008.06.003

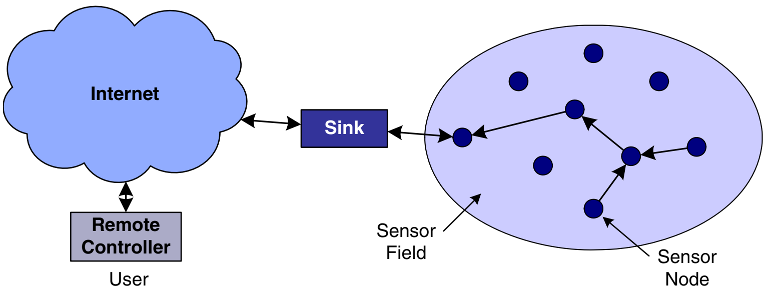

Fig. 1. Sensor network architecture.

Experimental measurements have shown that generally data transmission is very expensive in terms of energy consumption, while data processing consumes significantly less [108]. The energy cost of transmitting a single bit of information is approximately the same as that needed for processing a thousand operations in a typical sensor node [103]. The energy consumption of the sensing subsystem depends on the specific sensor type. In many cases it is negligible with respect to the energy consumed by the processing and, above all, the communication subsystems. In other cases, the energy expenditure for data sensing may be comparable to, or even greater than, the energy needed for data transmission. In general, energy-saving techniques focus on two subsystems: the networking subsystem (i.e., energy management is taken into account in the operations of each single node, as well as in the design of networking protocols), and the sensing subsystem (i.e., techniques are used to reduce the amount or frequency of energy-expensive samples).

The lifetime of a sensor network can be extended by jointly applying different techniques [10]. For example, energy efficient protocols are aimed at minimizing the energy consumption during network activities. However, a large amount of energy is consumed by node components (CPU, radio, etc.) even if they are idle. Power management schemes are thus used for switching off node components that are not temporarily needed.

In this paper we will survey the main enabling techniques used for energy conservation in wireless sensor networks. Specifically, we focus primarily on the networking subsystem by considering duty cycling. Furthermore, we will also survey the main techniques suitable to reduce the energy consumption of sensors when the energy cost for data acquisition (i.e. sampling) cannot be neglected. Finally, we will introduce mobility as a new energy conservation paradigm with the purpose of prolonging the network lifetime. These techniques are the basis for any networking protocol and solution optimized from an energy-saving point of view. Due to the fundamental role of these enabling techniques, we will stress the design principles behind them and their features instead of presenting a complete set of networking protocols for wireless sensor networks. For a survey on these aspects, the reader is referred to [39 and 99].

The rest of the paper is organized as follows. Section 2 discusses the general approaches to energy conservation in sensor nodes, and introduces the three main approaches (duty-cycling, data-driven, and mobility). In Section 3 we break down this high-level classification, and highlight the schemes that will be then described in detail in the following sections. Specifically, Section 4 deals with schemes related to the data-driven approach. Section 5 presents approaches related to the data-driven approach. Section 6 discusses schemes related to the mobility-based approach. Finally, conclusions and open issues are discussed in Section 7.

2. General approaches to energy conservation

Before discussing the high-level classification of energy conservation proposals, it is worth presenting the network- and node-level architecture we will refer to.

From a sensor network standpoint, we mainly consider the model depicted in Fig. 1, which is the most widely

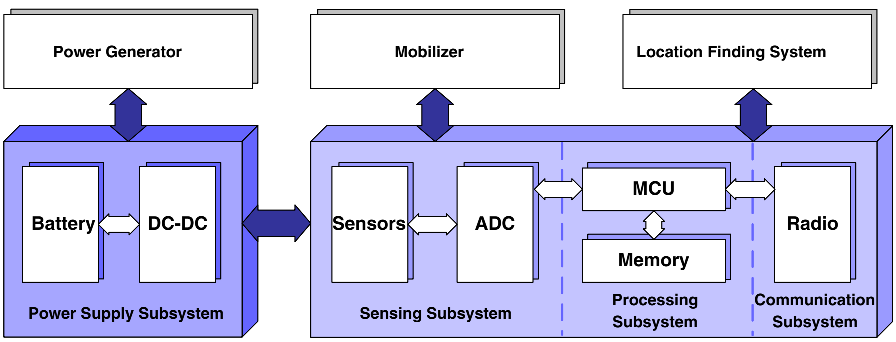

Fig. 2. Architecture of a typical wireless sensor node.

adopted model in the literature. On the other side, Fig. 2 shows the architecture of a typical wireless sensor node, as usually assumed in the literature. It consists of four main components: (i) a *sensing subsystem* including one or more sensors (with associated analog-to-digital converters) for data acquisition; (ii) a *processing subsystem* including a micro-controller and memory for local data processing; (iii) a *radio subsystem* for wireless data communication; and (iv) a *power supply unit*. Depending on the specific application, sensor nodes may also include additional components such as a *location finding system* to determine their position, a *mobilizer* to change their location or configuration (e.g., antenna's orientation), and so on. However, as the latter components are optional, and only occasionally used, we will not take them into account in the following discussion.

Obviously, the power breakdown heavily depends on the specific node. In [108] it is shown that the power characteristics of a Mote-class node are completely different from those of a Stargate node. However, the following remarks generally hold [108].

- The communication subsystem has an energy consumption much higher than the computation subsystem. It has been shown that transmitting one bit may consume as much as executing a few thousands instructions [103]. Therefore, communication should be traded for computation.
- The radio energy consumption is of the same order of magnitude in the reception, transmission, and idle states, while the power consumption drops of at least one order of magnitude in the sleep state. Therefore, the radio should be put to sleep (or turned off) whenever possible.
- Depending on the specific application, the sensing subsystem might be another significant source of energy consumption, so its power consumption has to be reduced as well.

Based on the above architecture and power breakdown, several approaches have to be exploited, even simultaneously, to reduce power consumption in wireless sensor networks. At a very general level, we identify three main enabling techniques, namely, *duty cycling*, *data-driven approaches*, and *mobility*.

Duty cycling is mainly focused on the networking subsystem. The most effective energy-conserving operation is putting the radio transceiver in the (low-power) sleep mode whenever communication is not required. Ideally, the radio should be switched off as soon as there is no more data to send/receive, and should be resumed as soon as a new data packet becomes ready. In this way nodes alternate between active and sleep periods depending on network activity. This behavior is usually referred to as *duty cycling*, and *duty cycle* is defined as the fraction of time nodes are active during their lifetime. As sensor nodes perform a cooperative task, they need to coordinate their sleep/wakeup times. A *sleep/wakeup scheduling algorithm* thus accompanies any duty cycling scheme. It is typically a distributed algorithm based on which sensor nodes decide when to transition from active to sleep, and back. It allows neighboring nodes to be active at the same time, thus making packet exchange feasible even when nodes operate with a low duty cycle (i.e., they sleep for most of the time).

Duty-cycling schemes are typically oblivious to data that are sampled by sensor nodes. Hence, *data-driven approaches* can be used to improve the energy efficiency even more. In fact, data sensing impacts on sensor nodes' energy consumption in two ways:

- *Unneeded samples*. Sampled data generally have strong spatial and/or temporal correlations [137], so there is no need to communicate the redundant information to the sink.
- *Power consumption of the sensing subsystem*. Reducing communication is not enough when the sensor itself is power hungry.

In the first case unneeded samples result in useless energy consumption, even if the cost of sampling is negligible, because they result in unneeded communications. The second issue arises whenever the consumption of the sensing subsystem is not negligible. Data driven techniques presented in the following are designed to reduce the amount of sampled data by keeping the sensing accuracy within an acceptable level for the application.

In case some of the sensor nodes are mobile, *mobility* can finally be used as a tool for reducing energy consumption (beyond duty cycling and data-driven techniques). In a static sensor network packets coming from sensor nodes follow a multi-hop path towards the sink(s). Thus, a few paths can be more loaded than others, and nodes closer to the sink have to relay more packets so that they are more subject to premature energy depletion (*funneling effect*) [83]. If some of the nodes (including, possibly, the sink) are mobile, the traffic flow can be altered if mobile devices are responsible for data collection directly from static nodes. Ordinary nodes wait for the passage of the mobile device and route messages towards it, so that communication takes place in proximity (directly or at most with a limited multi-hop traversal). As a consequence, ordinary nodes can save energy because path length, contention and forwarding overheads are reduced as well. In addition, the mobile device can visit the network in order to spread more uniformly the energy consumption due to communications. When the cost of mobilizing sensor nodes is prohibitive, the usual approach is to "attach" sensor nodes to entities that will be roaming in the sensing field anyway, such as buses or animals.

All of the schemes we will describe in the following fall under one of the three general approaches we have presented. Specifically, we provide the complete taxonomy of the schemes we will describe hereafter in Fig. 3. As this taxonomy is fairly rich, the remainder of the survey analyzes it using a top-down approach.

3. High-level taxonomy

In this section we discuss the breakdown at the first levels of the taxonomy in Fig. 3. The rest of the taxonomy, along with concrete examples proposed in the literature, is presented in the next sections.

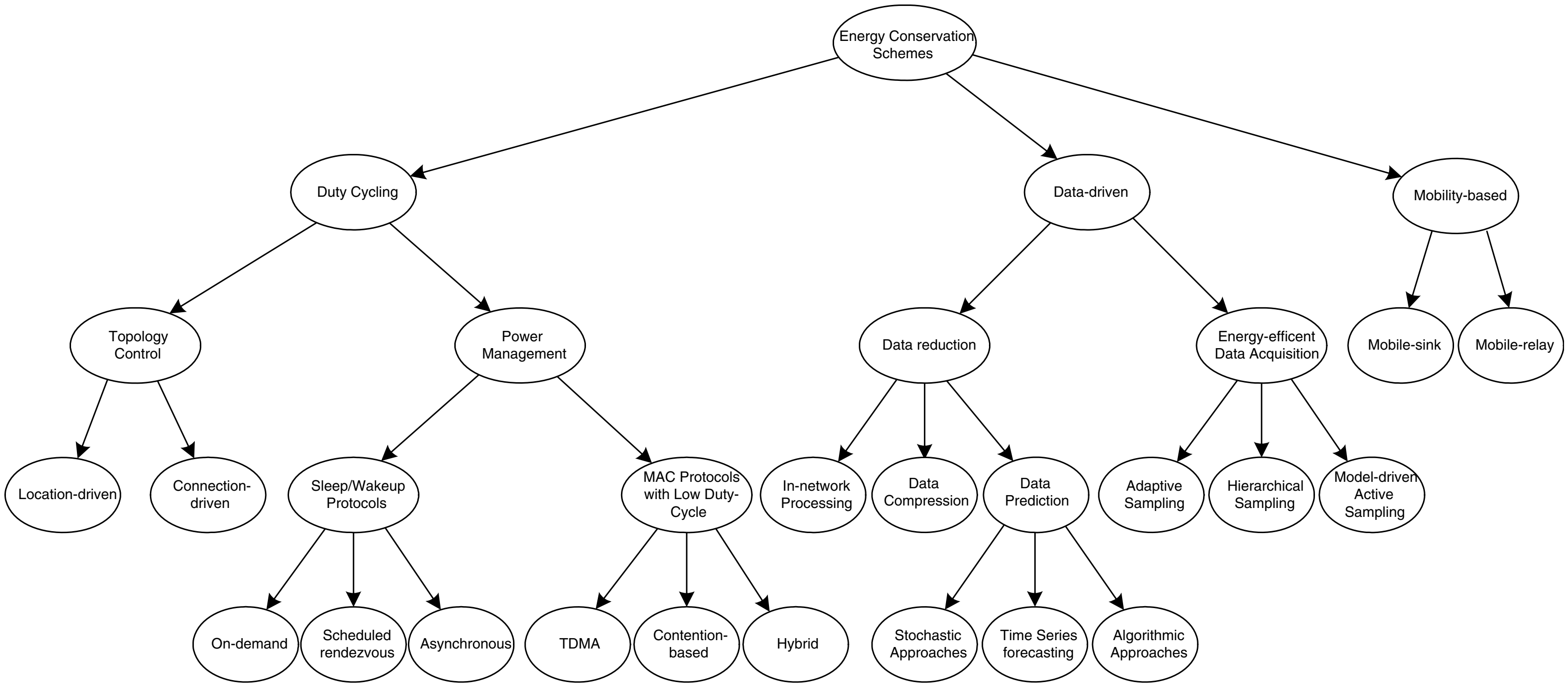

Fig. 3. Taxonomy of approaches to energy saving in sensor networks.

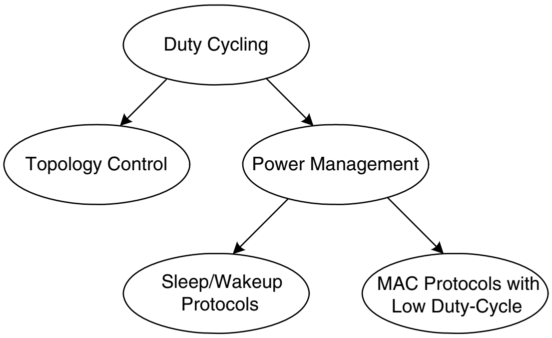

Fig. 4. Taxonomy of duty cycling schemes.

3.1. Duty-cycling

As shown in Fig. 4, *duty cycling* can be achieved through two different and complementary approaches. From one side it is possible to exploit node redundancy, which is typical in sensor networks, and adaptively select only a minimum subset of nodes to remain active for maintaining connectivity. Nodes that are not currently needed for ensuring connectivity can go to sleep and save energy. Finding the optimal subset of nodes that guarantee connectivity is referred to as *topology control*[1]. Therefore, the basic idea behind topology control is to exploit the network redundancy to prolong the network longevity, typically increasing the network lifetime by a factor of 2–3 with respect to a network with all nodes always on [41,92,140]. On the other hand, active nodes (i.e., nodes selected by the topology control protocol) do not need to maintain their radio continuously on. They can switch off the radio (i.e., put it in the low-power sleep mode) when there is no network activity, thus alternating between sleep and wakeup periods. Throughout we will refer to duty cycling operated on active nodes as *power management*. Therefore, topology control and power management are complementary techniques that implement duty cycling with different granularity. Power management techniques can be further subdivided into two broad categories depending on the layer of the network architecture they are implemented at. As shown in Fig. 4, power management protocols can be implemented either as independent sleep/wakeup protocols running on top of a MAC protocol (typically at the network or application layer), or strictly integrated with the MAC protocol itself. The latter approach permits to optimize medium access functions based on the specific sleep/wakeup pattern used for power management. On the other hand, independent sleep/wakeup protocols permit a greater flexibility as they can be tailored to the application needs, and, in principle, can be used with *any* MAC protocol.

The following breakdowns for topology-control schemes, independent sleep/wakeup schedules and MAC protocols with low duty cycle are presented in Sections 4.1, 4.2 and 4.3, respectively.

3.2. Data-driven approaches

Data-driven approaches (see Fig. 5) can be divided according to the problem they address. Specifically, data-reduction schemes address the case of unneeded samples, while energy-efficient data acquisition schemes are mainly aimed at reducing the energy spent by the sensing subsystem. However, some of them can reduce the energy spent for communication as well. Also in this case, it is worth discussing here one more classification level related to data-reduction schemes, as shown in Fig. 5. All these techniques aim at reducing the amount of data to be delivered to the sink node. However the principles behind them are rather different. *In-network processing* consists in performing data aggregation (e.g., computing average of some values) at intermediate nodes between the sources and the sink. In this way, the amount of data is reduced while traversing the network towards the sink. The most appropriate in-network processing technique depends on the specific application and must be tailored to it. As data aggregation is application-specific, in the following we will not discuss it. The interested reader can refer to [39] for a comprehensive and up-to-date survey about in-network processing techniques. *Data compression* can be applied to reduce the amount of information sent by source nodes. This scheme involves encoding information at nodes which generate data, and decoding it at the sink. There are different methods to compress data (see, e.g., [105,129,143,144]). As compression techniques are general (i.e. not necessarily related to WSNs), we will omit a detailed discussion of them to focus on other approaches specifically tailored to WSNs. *Data prediction* consists in building an abstraction of a sensed phenomenon, i.e. a model describing data evolution. The model can predict the values sensed by sensor nodes within certain error bounds, and resides both at the sensors and at the sink. If the needed accuracy is satisfied, queries issued by users

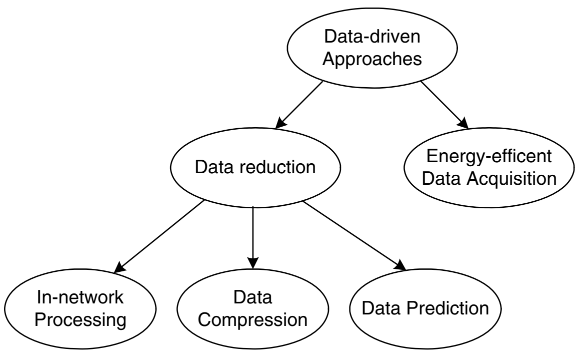

Fig. 5. Taxonomy of data-driven approaches to energy conservation.

[1] Before proceeding on, it may be worthwhile to point out that the term "topology control" has been used with a larger scope than that defined above. Some authors include in topology control also techniques that are aimed at super-imposing a hierarchy on the network organization (e.g., clustering techniques) to reduce energy consumption. In addition, the terms "topology control" and "power control" are often confused. However, power control refers to techniques that adapt the transmission power level to optimize a single wireless transmission. Even if the above techniques are related with topology control, in accordance with [115], we believe that they cannot be classified as topology control techniques. Therefore, in the following we will refer to topology control as one of the means to reduce energy consumption by exploiting node redundancy.

can be evaluated at the sink through the model without the need to get the exact data from nodes. On the other side, explicit communication between sensor nodes and the sink is needed when the model is not accurate enough, i.e. the actual sample has to be retrieved and/or the model has to be updated. On the whole, data prediction reduces the number of information sent by source nodes and the energy needed for communication as well.

The following levels in the classification for data-prediction and energy-efficient data acquisition techniques are presented in Sections 5.1 and 5.2, respectively.

3.3. Mobility-based schemes

As shown in Fig. 6, mobility-based schemes can be classified as *mobile-sink* and *mobile-relay* schemes, depending on the type of the mobile entity. They will be directly discussed in Section 6. It is worth pointing out here that, when considering mobile schemes, an important issue is the type of control the sensor-network designer has on the mobility of nodes. A detailed discussion on this point is presented in [12 and 68]. Mobile nodes can be divided into two broad categories: they can be specifically designed as *part of the network infrastructure*, or they can be *part of the environment*. When they are part of the infrastructure, their mobility can be fully controlled as they are, in general, robotized. When mobile nodes are part of the environment they might be not controllable. If they follow a strict schedule, then they have a completely predictable mobility (e.g., a shuttle for public transportation [23]). Otherwise they may have a random behavior so that no reliable assumption can be made on their mobility. Finally, they may follow a mobility pattern that is neither predictable nor completely random. For example, this is the case of a bus moving in a city, whose speed is subject to large variation due to traffic conditions. In such a case, mobility patterns can be learned based on successive observations and estimated with some accuracy.

4. Duty-cycling

In this section we will discuss the duty-cycling approaches as defined in the previous section. For convenience, we report in Fig. 7 an excerpt of the taxonomy referred to duty-cycling.

4.1. Topology control protocols

The concept of topology control is strictly associated with that of network redundancy. Dense sensor networks typically have some degree of redundancy. In many cases network deployment is done at random, e.g., by dropping a large number of sensor nodes from an airplane. Therefore, it may be convenient to deploy a number of nodes greater than necessary to cope with possible node failures occurring during or after the deployment. In many contexts it is much easier to deploy initially a greater number of nodes than re-deploying additional nodes when needed. For the same reason, a redundant deployment may be convenient even when nodes are placed by hand [41]. Topology control protocols are thus aimed at dynamically adapting the network topology, based on the application needs, so as to allow network operations while minimizing the number of active nodes (and, hence, prolonging the network lifetime).

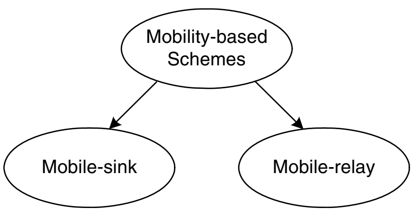

Fig. 6. Taxonomy of mobility-based energy conservation schemes.

Several criterions can be used to decide which nodes to activate/deactivate, and when. In this regard, topology control protocols can be broadly classified in the following two categories (Fig. 7). *Location driven* protocols define which node to turn on and when, based on the location of sensor nodes which is assumed to be known. *Connectivity driven* protocols, dynamically activate/deactivate sensor nodes so that network connectivity, or complete sensing coverage [78], are fulfilled.

A detailed survey on topology control in wireless ad hoc and sensor networks is available in [72 and 115]. In the following subsections we only review the main proposals for topology control in wireless sensor networks according to the above classification.

4.1.1. Location-driven

Geographical Adaptive Fidelity (GAF) [145] is a location-driven protocol that reduces energy consumption while keeping a constant level of routing fidelity. The sensing area where nodes are distributed is divided into small *virtual grids*. Each virtual grid is defined such that, for any two adjacent grids A and B, all nodes in A are able to communicate with nodes in B, and vice-versa (see Fig. 8). All nodes within the same virtual grid are equivalent for routing, and just one node at time need to be active. Therefore, nodes have to coordinate with each other to decide which one can sleep and how long.

Initially a node starts in the discovery state where it exchanges discovery messages with other nodes. After broadcasting the message, the node enters the active state. While active, it periodically re-broadcasts its discovery message. A node in the discovery or active state can change its state to sleeping when it detects that some other equivalent node will handle routing. Nodes in the sleeping state wake up after a sleeping time and go back to the discovery state. In GAF load balancing is achieved through a periodic re-election of the leader, i.e., the node that will remain active to manage routing in the virtual grid. The leader is chosen through a rank-based election algorithm which considers the nodes' residual energy, thus allowing the network lifetime to increase in proportion to node density [145]. GAF is independent of the routing protocol, so that it can be used along with any existing solution of that kind. In addition, GAF does not significantly affect the performance of the routing protocol in terms of packet loss and message latency. However, the structure imposed over

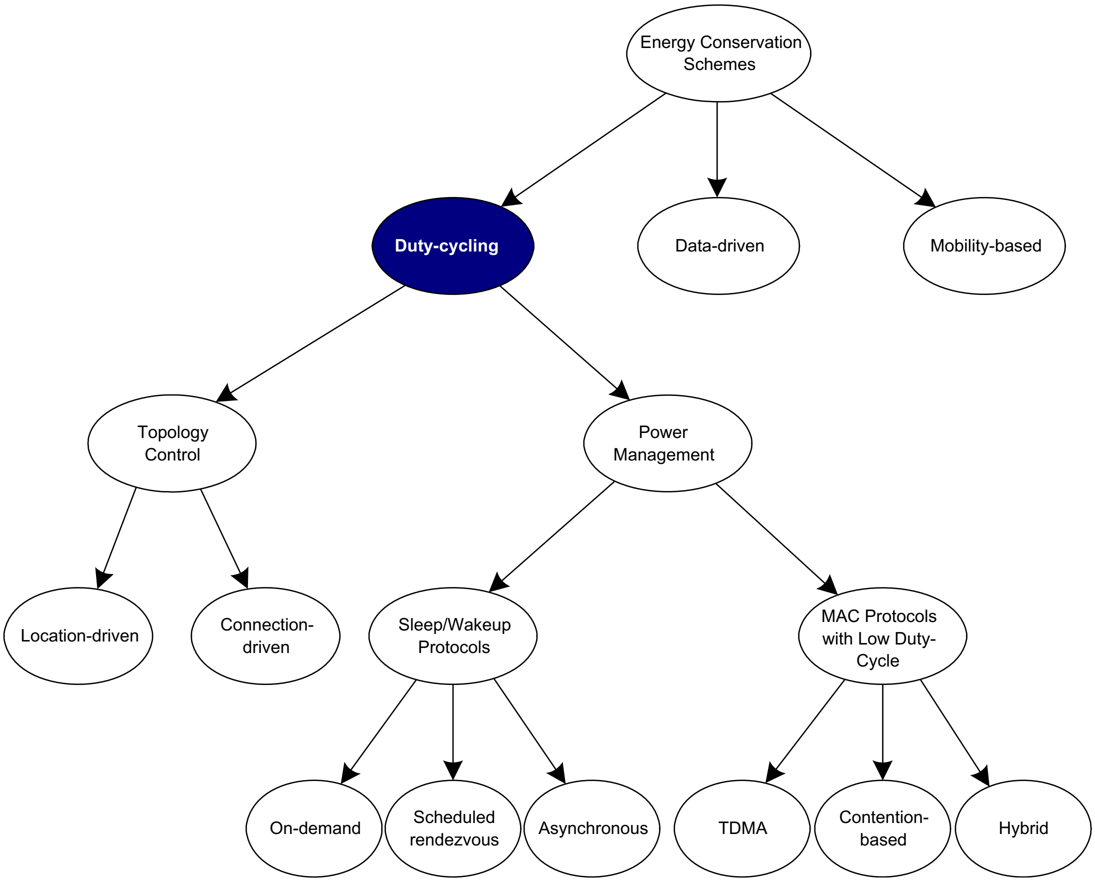

Fig. 7. Detailed taxonomy of duty cycling schemes.

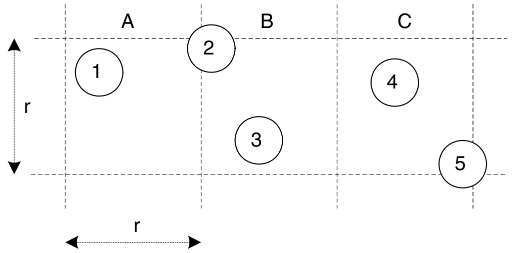

Fig. 8. Virtual grids in GAF.

the network may lead to an underutilization of the radio coverage areas. In fact, as all nodes within a virtual grid must be able to reach any node in an adjacent virtual grid, actually nodes are forced to cover less than half the distance allowed by the radio range.

Although being defined as a geographic routing protocol, Geographic Random Forwarding (GeRaF) [21,153,154] actually presents features which are in the direction of location-driven duty-cycled operations, which make use of both node position and redundancy. Nodes follow a given duty cycle to switch between awake (active) and sleep (inactive) states. Nodes periodically switch to the active state, starting with a listening time, so that they can participate to routing if needed. Data forwarding starts as soon as a node has a packet to send. In this case, the node becomes active and broadcasts a packet containing its own location and the location of the intended receiver. Then a receiver-initiated forwarding phase takes place. As a result, one of the active neighbors of the sender will be selected to relay the packet towards the destination. To this end, the main idea is that each active node has a priority which depends on its closeness to the intended destination of the packet. In addition to priority, a distributed randomization scheme is also used, in order to reduce the probability that many neighboring nodes are simultaneously sleeping. Specifically, the portion of the coverage area of the sender which is closer to the intended destination is split into a number of regions. Each region has its associated priority, and regions are chosen so that all nodes within a region are closer to the destination than any other node in a region with a lower priority (Fig. 9).

After the broadcast, nodes in the region with the higher priority contend for forwarding. If only one node gets the channel, it simply forwards the packet and the process ends. Otherwise, multiple nodes may transmit simultaneously, resulting in a collision. In this case, a resolution technique (i.e. a backoff) is applied in order to select a single forwarder. There may also be the case in which no node can forward the packet, because all nodes in the region are

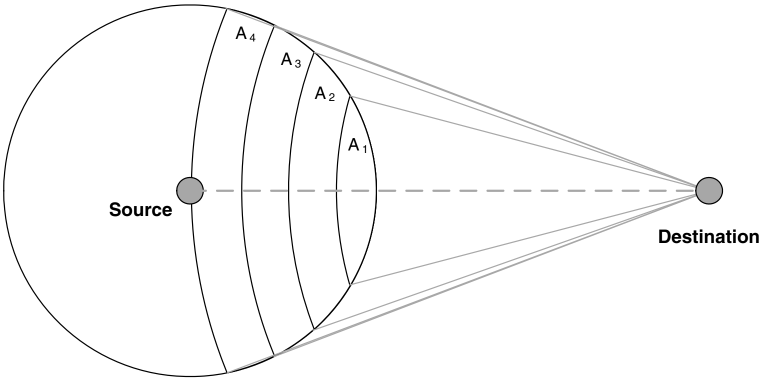

Fig. 9. Priority regions in GeRaF (with increasing priority from A_4 to A_1) [72].

sleeping. To this end, in the next transmission attempt, the forwarder will be chosen among nodes in the second highest-priority region and so on. Every time the relay selection phase will be repeated until a maximum number of retries will be reached. Eventually, after a hop-by-hop forwarding, the packet will reach the intended destination. Note that, as the relay selection is done a posteriori, GeRaF merely requires position information, thus it does not need topological knowledge nor routing tables.

4.1.2. Connectivity-driven

Span [26] is a connectivity-driven protocol that adaptively elects "coordinators" of all nodes in the network. Coordinators stay awake continuously and perform multi-hop routing, while the other nodes stay in sleeping mode and periodically check if it is needed to wake up and become a coordinator. To guarantee a sufficient number of coordinators Span uses the following *coordinator eligibility rule*: if two neighbors of a non-coordinator node cannot reach each other, either directly or via one or more coordinators, that node should become a coordinator. However, it may happen that several nodes discover the lack of a coordinator at the same time and, thus, they all decide to become a coordinator. To avoid such cases nodes that decide to become a coordinator defer their announcement by a random *backoff delay*. Each node uses a function that generates a random time by taking into account both the number of neighbors that can be connected by a potential coordinator node, and its residual energy. The fundamental ideas are that (i). nodes with a higher expected lifetime should be more likely to volunteer to become a coordinator; and (ii). coordinators should be selected in such a way to minimize their number. Each coordinator periodically checks if it can stop being a coordinator. In detail, a node should withdraw as a coordinator if every pair of its neighbors can communicate directly, or through some other coordinator. To avoid loss of connectivity, during the transient phase the old coordinator continues its service until the new one is available. The Span election algorithm requires to know neighbor and connectivity information to decide whether a node should become a coordinator or not. Such information is provided by the routing protocol, hence SPAN depends on it and may require modification in the routing lookup process.

Adaptive Self-Configuring sEnsor Networks Topologies (ASCENT [22]) is a connectivity-driven protocol that, unlike Span, does not depend on the routing protocol. In ASCENT a node decides whether to join the network or continue to sleep based on information about connectivity and packet loss that are *measured locally* by the node itself. The basic idea of ASCENT is that initially only some nodes are *active*, while all other ones are *passive*, i.e., they listen to packets but do not transmit. If the number of active nodes is not large enough, the sink node may experience a high message loss from sources. The sink then starts sending *help* messages to solicit neighboring nodes that are in the passive state (*passive neighbors*) to join the network by changing their state from passive to active (*active neighbors*). Passive neighbors have their radio on and listen to all packets transmitted by their active neighbors. However, they do not cooperate to forward data packets or exchange routing control information – they only collect information about the network status without interfering with other nodes. On the contrary, active neighbors forward data and routing (control) messages until they run out of energy. Active nodes can also send *help* messages when they find the local data loss at an unacceptable level. As soon as it joins the network, a node starts monitoring the network conditions and also signals its presence as an active node through a *neighbor announcement message*. This process continues until the number of active nodes is such that the message loss rate experienced by the sink is below a pre-defined application-dependent threshold. The process will re-start when some future network event (e.g. a node failure) or a change in the environmental conditions causes an increase in the message loss. As mentioned above, ASCENT is independent of the routing protocol. In addition, it limits the packets loss due to collisions because the node density is explicitly taken into account as a parameter (in the form of a neighbor threshold value). Finally, the protocol has good scalability properties. On the other side, energy saving does not increase proportionally with the node density because it actually depends on passive-sleep cycle rather than the number of active nodes.

A different class of approaches model the network as a random graph and exploit the percolation theory [46] to characterize the network connectivity when a duty-cycle is enforced on nodes. For instance, the authors in [42] propose Naps, a decentralized topology management protocol based on a periodic sleep/wakeup scheme. In Naps, time is split into time periods with duration *T*. Each node initially waits for a random amount of time t_v, uniformly distributed into the range [0,T). As from t_v, a node operates on the basis of *T* in the following way. First, it broadcasts a HELLO message to advertise its activation to neighbors. Then, it listens for HELLO messages sent by other nodes. The node can go to sleep until the next time period as soon as it receives *c* messages from its neighbors. Otherwise, it remains active for all the time period *T*. The authors analytically prove the connectivity properties of the protocol and also show by simulation that it is flexible and robust. A similar approach is exploited in [35], where the authors focus on time-critical monitoring applications. In detail, the impact of an asynchronous sleep/wakeup protocol on the network connectivity is investigated, and latency for reporting events is analytically derived. However – in contrast with [42] – the solution proposed in [35] requires the knowledge of the network density. To overcome this problem, the Degree-Dependent Energy Management Algorithm (DDEMA) is presented in [77], where the previous work of [35] is extended by considering only neighboring information.

4.1.3. Discussion

Location-driven topology control protocols obviously require that sensor nodes can somewhat know their position. This is generally achieved by providing sensors with a GPS unit. As the GPS is quite expensive and energy consuming, it is often unfeasible to install it on all nodes. In this case, it would be enough to equip only a limited subset of nodes with a GPS, and then derive the location of the other ones by means of other techniques [80]. Also a number of different technologies, i.e. exploiting radio or sound waves, can be used [93]. However, commonly available sensor platforms lack the hardware suitable to acquire location information. From the above discussion it emerges that connectivity-driven protocols are generally preferable, since they only require information which can be derived from local measurements.

In any case, as the energy efficiency of topology control protocols is tightly related to node density, also the achievable gain in terms of network lifetime depends on the actual density. It has been shown that topology control protocols can typically increase the network lifetime by a factor of 2–3 with respect to a network with nodes always on [41,92,140]. This value may be not acceptable for many practical applications. To this end, topology control protocols should be coupled with other kinds of energy conservation techniques, such as the ones presented in the following sections. However, the simultaneous application of multiple energy conservation schemes may lead to unforeseen consequences. In fact, although the combination of protocols should be transparent to the applications, actually the obtained results may be very different from what one would expect. Although some work – such as [91 and 94] – already explored the interactions between protocols, we think this area of research has not been fully explored yet.

4.2. Sleep/wakeup protocols

As previously discussed, sleep/wakeup schemes can be defined for a given component (i.e. the radio subsystem) of the sensor node, without relying on topology or connectivity aspects. In this section we will survey the main sleep/wakeup schemes implemented as independent protocols on top of the MAC protocol (i.e. at the network or the application layer). Independent sleep/wakeup protocols can be further subdivided into three main categories [15]: *on-demand*, *scheduled rendezvous*, and *asynchronous* schemes (Fig. 10).

On-demand protocols take the most intuitive approach to power management. The basic idea is that a node should wakeup only when another node wants to communicate with it. The main problem associated with on-demand schemes is how to inform the sleeping node that some other node is willing to communicate with it. To this end, such schemes typically use multiple radios with different energy/performance tradeoffs (i.e. a low-rate and low-power radio for signaling, and a high-rate but more power hungry radio for data communication).

An alternative solution consists in using a *scheduled rendezvous* approach. The basic idea behind scheduled rendezvous schemes is that each node should wake up at the same time as its neighbors. Typically, nodes wake up according to a wakeup schedule, and remain active for a short time interval to communicate with their neighbors. Then, they go to sleep until the next rendezvous time.

Finally, an *asynchronous* sleep/wakeup protocol can be used. With asynchronous protocols, a node can wake up when it wants and still be able to communicate with its neighbors. This goal is achieved by properties implied in the sleep/wakeup scheme, thus no explicit information exchange is needed among nodes.

We will survey the different classes in separate subsections below.

4.2.1. On-demand schemes

On-demand schemes are based on the idea that a node should be awaken just when it has to receive a packet from a neighboring node. This minimizes the energy consumption and, thus, makes on-demand schemes particularly suitable for sensor network applications with a very low duty cycle (e.g., fire detection, surveillance of machine failures and, more generally, all event-driven scenarios). In such scenarios sensor nodes are in the *monitoring state* (i.e., they only sense the environment) for most of the time. As soon as an event is detected, nodes transit to the *transfer*

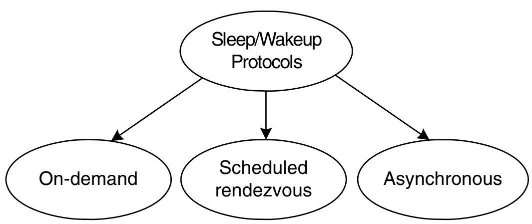

Fig. 10. Classification of sleep/wakeup protocols.

state. On-demand sleep/wakeup schemes are aimed at reducing energy consumption in the monitoring state while ensuring a limited latency for transitioning in the transfer state.

The implementation of such schemes typically requires two different channels: a data channel for normal data communication, and a wakeup channel for awaking nodes when needed. Although it would be possible to use a single radio with two different channels, all the proposals rely on two different radios. This allows not to defer the transmission of signal on the wakeup channel if a packet transmission is in progress on the other channel, thus reducing the wakeup latency. The drawback is the additional cost for the second radio. However, this additional cost is limited as the radio system typically accounts for a small percent of the entire cost of a sensor node (less than 15% for a MICA mote [118]). Another drawback is the possible mismatch between the coverage of the two radios.

Sparse Topology and Energy Management (STEM) [118] uses two different radios for wakeup signal and data packet transmissions, respectively. The wakeup radio is not a low power radio (to avoid problems associated with different transmission ranges). Therefore, an asynchronous duty cycle scheme is used on the wakeup radio as well. Each node periodically turns on its wakeup radio for T_{active} every T duration. When a source node (*initiator*) has to communicate with a neighboring node (*target*), it sends a stream of periodic beacons on the wakeup channel. As soon as the target node receives a beacon it sends back a wakeup acknowledgement, and turns on its data radio. If a collision occurs on the wakeup channel, any node that senses the collision activates its data radio "up" (no wakeup acknowledgement is sent in case of collision). The wakeup beacon transmission is repeated up to a maximum time unless a wakeup acknowledgement is received from the target node.

In addition to the above beacon-based approach, referred to as STEM-B, in [117] the authors propose a variant (referred to as STEM-T) that uses a wakeup tone instead of a beacon. The main difference is that in STEM-T all nodes in the neighborhood of the initiator are awakened.

Both STEM-B and STEM-T can be used in combination with topology control protocols. For example, in a practical case the combination of GAF and STEM can reduce the energy consumption to about 1% of that of a sensor network with neither topology control nor power management. This increases the network lifetime of a factor 100 [117]. However, STEM trades energy saving for path setup latency. In STEM the inter-beacon period is such that there is enough time to send the wakeup beacon and receive the related acknowledgement. Let T_{wakeup} and T_{wack} denote the time required to transmit a wakeup beacon and the related acknowledgement, respectively. Since nodes are not synchronized, the receiver must listen on the wakeup radio for a time T_{active} at least equal to 2 $T_{\text{wakeup}} + T_{\text{wack}}$ to ensure the correct reception of the beacon, i.e., $T_{\text{active}} \geqslant 2T_{\text{wakeup}} + T_{\text{wack}}$ (see also Section 4.2.3). Clearly T_{active} depends on the bit rate of network nodes. In low bit-rate networks the time between successive active periods (T) must be very large to allow a low duty cycle on the wakeup channel. This results in a large wakeup latency, especially in multi-hop networks with a large hop-count.

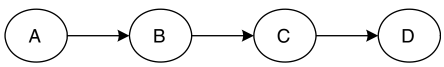

Fig. 11. String topology network.

To achieve a tradeoff between energy saving and wakeup latency, [146] proposes a *Pipelined Tone Wakeup* (PTW) scheme. Like STEM, PTW relies on two different channels for transmitting wakeup signals and packet data, and uses a wakeup tone to awake neighboring nodes. Hence, any node in the neighborhood of the source node will be awakened. Unlike STEM, in PTW the burden for tone detection is shifted from the receiver to the sender. This means that the duration of the wakeup tone is long enough to be detected by the receiver that turns on its wakeup radio periodically. The rationale behind this solution is that the sender only sends a wakeup tone when an event is detected, while receivers wake up periodically. In addition, the wakeup procedure is pipelined with the packet transmission so as to reduce the wakeup latency and, hence, the overall message latency. The idea is illustrated in Fig. 12 with reference to the string topology network depicted in Fig. 11.

Let's suppose that node A has to transmit a message to node D through nodes B and C. At time t_0 A starts the procedure by sending a tone on the wakeup channel. This tone awakens all A's neighbors. At time t_1 A sends a notification packet to B on the data channel to inform that the next data packet will be destined to B. Upon receiving the notification messages all A's neighbors but B learn that the following message is not intended for them. Therefore, they turn off their data radio. Instead, B realizes to be the destination of next data message, and replies with a wakeup acknowledgment on the data channel. Then, A starts transmitting the data packet on the data channel. At the same time, B starts sending a tone on the wakeup channel to awake all its neighbors. As shown in Fig. 12, the packet transmission from A to B on the data channel, and the B's

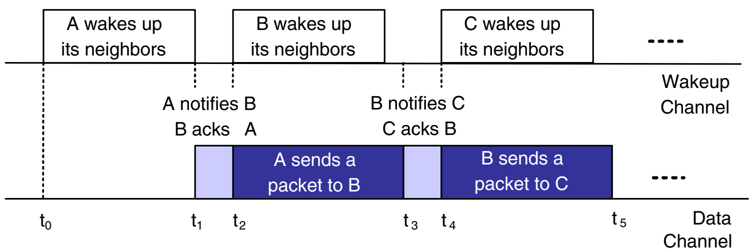

Fig. 12. Pipelined wakeup procedure in PTW.

tone transmission on the wakeup channel are done concurrently. As in STEM, the data transmission is regulated by the underlying MAC protocol. In [146] it is shown by simulation that, if the time spent by a sensor network in the monitoring state is greater than 10 min, PTW outperforms STEM significantly, both in terms of energy saving and message latency, especially when the bit rate of sensor nodes is low.

As the energy consumption of the wakeup radio is generally not negligible, both STEM and PTW use an asynchronous sleep/wakeup scheme for enabling a duty cycle on the wakeup radio as well. A different approach is using a low-power radio for the wakeup channel. The low-power radio is continuously in stand-by, and whenever receives a signal it wakes up the data radio [48,97,106,120]. The wakeup latency is thus minimized. The main drawback of this approach is that the transmission range of the wakeup radio is significantly smaller than that of the data radio. This may limit the applicability of such a technique as a node may not be able to wake up a neighboring node even if it is within its data transmission range. For example, in [120] the low power radio operates at 915 MHz (ISM band) and has a transmission range of approximately 332 ft in free space, while the IEEE 802.11 card operate at 2.4 GHz with a transmission range up to 1750 ft.

A side effect of using a second radio for the wakeup channel is the additional power consumption, which may not be negligible even when using a low-power radio. To overcome problems associated with the extra-energy consumed by the wakeup radio, a *Radio-Triggered Power Management* scheme is investigated in [47]. The basic idea is to use the energy contained in wakeup messages (e.g., STEM-B beacon) or signals (e.g., STEM-T and PTW tones) to trigger the activation of the sensor node. This approach is similar to the one used in active Radio Frequency Identification (RFID) systems [60]. The radio-triggered scheme, in its simplest form, is illustrated in Fig. 13. A special hardware component, a radio-triggered circuit, is used to capture the energy contained in the wakeup message (or signal), and uses such energy to trigger an interrupt for waking up the node. The radio-triggered approach is significantly different than using a stand-by radio to listen to possible wakeup messages from neighboring nodes. The stand-by radio consumes energy from the node while listening, while the radio-triggered circuit is powered by the wakeup message.

The main drawback of the radio-triggered approach is the limitation on the maximum distance from which the wakeup message can be sent. With the basic radio-triggered circuit proposed in [47], the maximum achievable distance is 3 m. This distance may be increased up to a few dozen meters at the cost of a more complex (and expensive) radio-triggered circuit and increased wakeup latency (due to limits on the electronics of the circuit). For instance, the Radio Trigger Wake-up with Addressing Capabilities (RTWAC) solution [13] can achieve a distance up to 7.5 m.

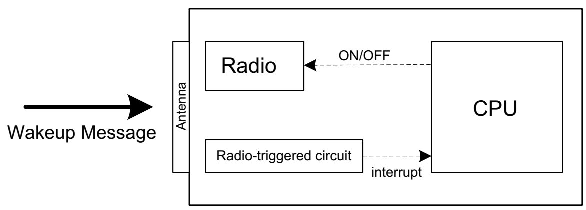

Fig. 13. Radio triggered power management.

4.2.2. Scheduled rendezvous schemes

Scheduled rendezvous schemes require that all neighboring nodes wake up at the same time. Typically, nodes wake up periodically to check for potential communications[2]. Then, they return to sleep until the next rendezvous time. The major advantage of such schemes is that when a node is awake it is guaranteed that all its neighbors are awake as well. This allows sending broadcast messages to all neighbors [15]. On the flip side, scheduled rendezvous schemes require nodes to be synchronized in order to wake up at the same time. Clock synchronization in wireless sensor networks is a relevant research topic. However, it is beyond the scope of the present paper. The reader can refer to [38 and 123] for detailed surveys on time synchronization techniques. In the following we will assume that nodes are synchronized by means of some synchronization protocol.

Different scheduled rendezvous protocols differ in the way network nodes sleep and wake up during their lifetime. The simplest way is using a *Fully Synchronized Pattern* [73]. In this case all nodes in the network wake up at the same time according to a periodic pattern. More precisely, all nodes wake up periodically every T_{wakeup}, and remain active for a fixed time T_{active}. Then, they return to sleep until the next wakeup instant. Due to its simplicity this scheme is used in several practical implementations including TinyDB [90] and TASK [19]. A fully synchronized wakeup scheme is also used in MAC protocols such as S-MAC [148] and T-MAC [28] (see Section 4.3). Even if simple, this scheme allows a low duty cycle provided that the active time (T_{active}) is significantly smaller than the wakeup period (T_{wakeup}). A further improvement can be achieved by allowing nodes to switch off their radio when no activity is detected for at least a timeout value [28]. In addition, due to the large size of the active and sleeping part, it does not require very precise time synchronization [90]. The main drawback is that all nodes become active at the same time after a long sleep period. Therefore, nodes try to transmit simultaneously, thus causing a large number of collisions. In addition, the scheme is not very flexible since the size of wakeup and active periods is fixed and does not adapt to variations in the traffic pattern and/or network topology.

The fully synchronized scheme applies equally well to both flat and structured sensor networks. To this end it may be worthwhile recalling that many routing protocols superimpose a tree or cluster-tree organization to the network by building a data-gathering tree (or routing tree) typically rooted at the sink node. Some sleep/wakeup schemes take advantage of the internal network organiza-

[2] Generally these schemes assume that an underlying contention-based MAC protocol is used for actual data transfer.

tion by sizing active times of different nodes according to their position in the data-gathering tree. The latter might change over time due to node failures, topology changes (node that joins or leaves), etc. In addition, it should be recomputed periodically by the routing protocol to achieve load balancing among nodes. However, under the assumption that nodes are static, the data-gathering tree is supposed to remain stable for a reasonable amount of time [85].

In the *Staggered Wakeup Pattern* [73], shown in Fig. 14, nodes located at different levels of the data-gathering tree wake up at different times. Obviously, the active parts of nodes belonging to adjacent levels must be partially overlapped to allow nodes to communicate with their children. Finally, the active parts of different levels are arranged in such a way that the portion of the active period a node uses to receive packets from its children is adjacent to the portion it uses to send packets to its parent (Fig. 14). This minimizes the energy dissipation for transitioning from sleep to active mode.

The staggered wakeup pattern shown in Fig. 14 is also called backward staggered pattern [73] as it optimizes packet latency in the backward direction i.e., from leaf nodes to the root (which is typically the sink node). It is also possible to arrange nodes' active periods in such a way to optimize the forward packet latency (i.e., from the root to leaves). The resulting scheme, called forward staggered pattern [73] is however not very used in practice, because in real networks most of data flows from sensor nodes to the sink. A combination of the backward and forward staggered pattern is also possible (see below).

The (backward) staggered scheme was first proposed in the framework of TinyDB [90] and TAG [89]. Due to its nice properties this scheme has been then considered and analyzed in several other papers ([20,84,85,95,125] among others) even if with different names. A staggered wakeup pattern is also used in D-MAC [85] (see Section 4.3).

With respect to the fully synchronized approach, the staggered scheme has several advantages. First, since nodes at different levels of the data-gathering tree wake up at different times, at a given time only a (small) subset of nodes in the network will be active. Thus, the number of collisions is potentially lower as only a subset of nodes contend for channel access. For the same reason, the active period of each node can be significantly shortened with respect to the fully synchronized scheme, thus resulting in energy saving. This scheme is also suitable for data aggregation. Parent nodes receive data from all their children before they forward such data to their own parent at the higher level. This allows parent nodes to filter data received from children, or to aggregate them.

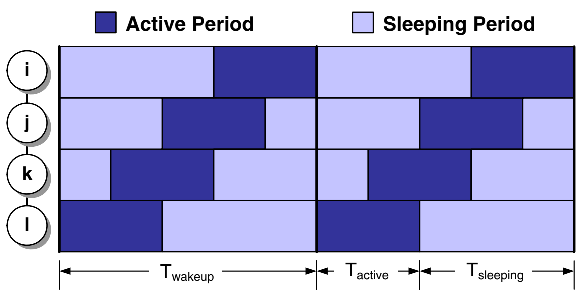

Fig. 14. Staggered sleep/wakeup pattern.

The staggered scheme has some drawbacks in common with the fully synchronized scheme. First, since nodes located at the same level in the data gathering tree wake up at the same time, collisions can potentially still occur. In addition, this scheme has limited flexibility due to the fixed duration of the active (T_{active}) and wakeup (T_{wakeup}) periods. The active period is often the same for all nodes in the network. For example, in [89] T_{active} is set to the duration of the wakeup period T divided by the maximum number of hops in the data gathering tree, while in [125] it is based on the delay to traverse a single hop.

Ideally, the active period should be as low as possible, not only for energy saving but also for minimizing the latency experienced by packets to reach the root node (see Fig. 14). In addition, since nodes located at different levels of the data-gathering tree manage different amounts of data, active periods should be sized based on individual basis. Finally, even assuming static nodes, topology changes and variations in the traffic patterns are still possible. The active period of nodes should thus adapt dynamically to such variations.

An adaptive and low latency staggered scheme is proposed in [9] (a somewhat similar approach is also taken in [85]). By setting the length of the active period to the minimum value consistently with the current network activity, this adaptive scheme not only minimizes the energy consumption but also provides a lower average packet latency with respect to a fixed staggered scheme. In addition, by allowing different lengths of the active period for nodes belonging to the same level, but associated with different parents, it also reduces the number of collisions [9].

A somewhat different approach, derived from the on-demand TDMA scheme, is taken in Flexible Power Scheduling (FPS) [56]. FPS takes a slotted approach, i.e. time is assumed to be divided in slots of duration T_s. Slots are arranged to form periodic cycles, where each cycle is made up of m slots and has a duration of $T_c = m\ T_s$. Each node maintains a power schedule of what operations it performs during a cycle. Obviously, a node must keep its own radio on only when it is has to receive/transmit from/to other nodes. Slotted schemes typically suffer from two common problems: they are not flexible and require a strict synchronization among nodes. To overcome the lack of flexibility, FPS includes a on-demand reservation mechanism that allows nodes to reserve slots in advance. As far as synchronization, slots are relatively large so that only coarse-grain synchronization is required. Twinkle, an improved version of FPS supporting broadcast traffic and sink to sensor communication, is presented in [55].

Several other sleep/wakeup schemes that still leverage the tree-network organization have been considered and analyzed [84,86]. The *Shifted Even and Odd Pattern* is derived from the Fully Synchronized Pattern by shifting the wakeup times of nodes in even levels by $T_{wakeup}/2$ (T_{wakeup} being the wakeup period). This minimizes the *overall* average packet latency, i.e., the average latency considering both the forward and backward directions, and also increases the network lifetime. Finally, the *Two-Staggered Pattern* and *Crossed Staggered Pattern* [73] are obtained as combinations of the Backward Wakeup Pattern and Forward Wakeup Pattern.

In [73] the authors also propose a multi-parent scheme that can be combined with any of the above sleep/wakeup patterns. The multi-parent scheme assigns multiple parents (with potentially different wakeup pattern) to each node in the network. This results in significant performance improvements in comparison with single-parent schemes.

4.2.3. Asynchronous schemes

Asynchronous schemes allow each node to wake up independently of the others by guaranteeing that neighbors always have overlapped active periods within a specified number of cycles.

Asynchronous wakeup was first introduced in [130] with reference to IEEE 802.11 ad hoc networks. The basic IEEE 802.11 Power Saving Mode (PSM) [57] has been conceived for single-hop ad hoc networks and thus it is not suitable to multi-hop ad hoc networks, where nodes may also be mobile. In [130] the authors propose three different asynchronous sleep/wakeup schemes that require some modifications to the basic PSM. More recently, Zheng et al. [150] took a systematic approach to design asynchronous wakeup mechanisms for ad hoc networks. Their scheme applies to wireless sensor networks, as well. They formulate the problem of generating wakeup schedules that rely upon asynchronous wakeup mechanisms as a combinatorial design problem [128]. Based on the optimal results derived from the theoretical framework, they design an Asynchronous Wakeup Protocol (AWP) that can detect neighboring nodes in a finite time without requiring slot alignment. The proposed asynchronous protocol is also resilient to packet collisions and variations in the network topology. The basic idea is that each node is associated with a Wakeup Schedule Function that is used to generate a wakeup schedule. For two neighboring nodes to communicate their wakeup schedules have to overlap, regardless of the difference in their clocks. The idea is illustrated in Fig. 15 by means of an example of asynchronous wakeup schedule for a set of 7 neighboring nodes. This example is based on a symmetric (7,3,1)-design of the wakeup schedule function. Symmetric means that all nodes have the same duty cycle, while (7,3,1)-design indicates that: (i). each schedule repeats every seven slots; (ii). each schedule has three active slots out of seven (dark slots); and (iii). any two schedules overlap for at most one slot. As shown in Fig. 15, by following its own schedule (i.e., by turning on the radio only during its active slots), each node is guaranteed to communicate with any other neighboring node.

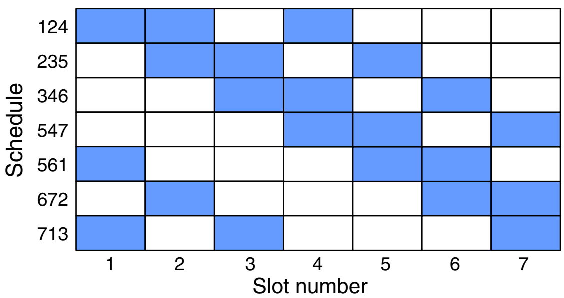

Fig. 15. An example of asynchronous schedule based on a symmetric (7,3,1)-design of the wakeup schedule function.

The above scheme ensures that each node will be able to contact any of its neighbors in a finite amount of time. However the packet latency introduced may be large, especially in multi-hop networks. In addition, it never happens that all neighbors are simultaneously active. Therefore, it is not possible to broadcast a message to all neighbors [15].

Random Asynchronous Wakeup (RAW) [101] takes a different approach as it leverages the fact that sensor networks are typically characterized by a high node density. This allows the existence of several paths between a source and a destination and, thus, a packet can be forwarded to any of such available paths. Actually, the RAW protocol consists of a routing protocol combined with a random wakeup scheme. The routing protocol is a variant of geographic routing. While in geographic routing a packet is sent to a neighbor that is closest to the destination, in RAW the packet is sent to any of the active neighbors in the *Forwarding Candidate Set*, i.e., the set of active neighbors that meet a pre-specified criterion. The basic idea of the random wakeup scheme is that each node wakes up randomly once in every time interval of fixed duration T, remains active for a predefined time T_a ($T_a \leqslant T$), and then sleeps again. Once awake, a node looks for active neighbors by running a neighbor discovery procedure. Suppose that a node S has to transmit a packet to a destination node D, and that in the forwarding set of S there are m neighbors as possible forwarders towards D. Then, the probability that at least one of these neighbors is awake along with S is given by

$$P = 1 - \left(1 - \frac{2 \cdot T_a}{T}\right)^m. \tag{1}$$

If the sensor network is dense, the number (m) of neighbors in the *Forwarding Candidate Set* is large and, by (1), the probability P to find an active neighbor to which forward the packet is large as well.

The random wakeup scheme is extremely simple and relies only on local decisions. This makes it well suited for networks with frequent topology changes. On the other side, it is not suitable for sparse networks. With RAW, it is not guaranteed that a node can find another active neighbor upon wakeup. Therefore, RAW does not guarantee the packet forwarding within one time frame (T), while AWP does. Nonetheless, it is very likely that some of the neighbors can be awake, due to the network density.

An alternative approach to ensure that an asynchronous node – typically a sender – finds its communication counterpart (i.e., the receiver) active when it wakes up, is forcing the receiver to listen periodically. The receiver wakes up periodically and listens for a short time to discover any potential asynchronous sender. If it does not detect any activity on the channel it returns to sleep, otherwise remains active to send/receive packets. Even if the receiver need to periodically wake up, this scheme falls in the category of asynchronous schemes because nodes do not need to be synchronized.

Two different variants are possible to discover asynchronous senders by periodic listening. We have already introduced these two variants with reference to the wake-up channel in STEM-B and PTW, respectively. However, their usage is more general. This is why we re-discuss them in this context.

In the first variant, depicted in Fig. 16 the asynchronous sender transmits a stream of periodic discovery messages (e.g., STEM-B beacons [118]). As anticipated in Section 4.2.1, to ensure the correct discovery of the sender, the receiver's listening time (T_{rx}) must be at least equal to $T_{on} + T_{idle} + T_{on}$, where, T_{on} is the time for transmitting a discovery message and T_{idle} is the time between the end of a discovery message and the start of the next one.

The second variant is illustrated in Fig. 17 and differs from the previous one in that the sender transmits a single long discovery message instead of a stream of periodic discovery messages. In this case the receiver listening time can be very short provided that the duration of the discovery message (T_{tx}) is, at least, equal to the listening period T_{rx}. This variant is used for enabling duty cycling on the wakeup channel in PTW. A similar scheme is also used in B-MAC [102] (see Section 4.3.2). In addition, both variants are very suitable for sensor networks where mobile nodes (data mules) are used to collect data [64,71]. Since the mule arrival time is usually unpredictable, static nodes typically use an asynchronous scheme, like the ones shown in Figs. 16 and 17, for mule discovery. This allows the timely discovery of the nearby mule without keeping the radio continuously on [64].

4.2.4. Discussion

Actually, the approach taken by on-demand protocols is the ideal one, because it maximizes energy saving as nodes remain active only for the minimum time required for communication. In addition, there is only a very limited impact on latency, because the target node wakes up immediately as soon as it realizes that there is a pending message. Unfortunately, the adoption of a radio triggered wakeup scheme is almost always impractical, because it can be only applied when the distance between nodes is very short indeed (a few meters). Introducing an additional wakeup radio is a more promising direction, especially suitable to event detection applications. However, the wakeup radio is costly and generally it is not shipped with commonly used sensor platforms. So, when a second radio is not available or convenient, other solutions – such as the scheduled rendezvous and the asynchronous wakeup schemes – can be used. Both of them trade energy saving for an increased latency experienced by messages to travel through several hops.

The scheduled rendezvous approach is convenient, because it is suitable to data aggregation and supports broadcast traffic. Unfortunately, it requires nodes to be synchronized, which in some cases can be difficult to achieve or expensive, in terms of additional protocol overhead for synchronization. On the other side, asynchronous wakeup protocols do not need a tight synchronization among network nodes. In addition, asynchronous schemes are generally easier to implement and can ensure network connectivity even in highly dynamic scenarios where synchronous (i.e., scheduled rendezvous) schemes become inadequate. This greater flexibility is compensated by a lower energy efficiency. In the asynchronous schemes nodes need to wake up more frequently than in scheduled rendezvous protocols. Therefore, asynchronous protocols usually result in a higher duty cycle for network nodes than their synchronous counterparts. In addition, the support to broadcast traffic is problematic.

Due to their wider applicability and their properties, scheduled rendezvous and asynchronous approaches seem to be the most promising solutions in the class of sleep/wakeup protocols. However, there is still room for improvements over the techniques discussed above. For instance, scheduled rendezvous protocols should relax the assumptions of clock synchronization among nodes, so that a coarse-grained time reference should be sufficient. Alternatively, they could embed a time synchronization solution as well, so that their timing requirements can be guaranteed without requiring a separate protocol. On the other side, exploiting cross-layer information seems to be a factor often neglected in the design of asynchronous protocols.

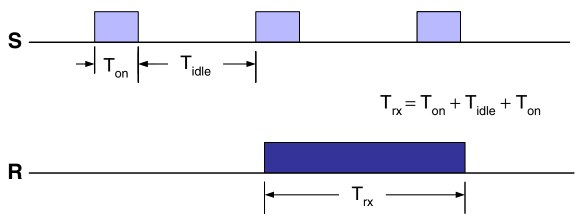

Fig. 16. Discovery of an asynchronous sender through periodic listening. The sender transmits a stream of periodic discovery messages.

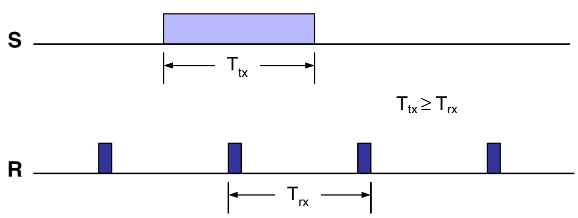

Fig. 17. Discovery of an asynchronous sender through periodic listening. The sender transmits a single long discovery message.

4.3. MAC protocols with low duty cycle

Several MAC protocols for wireless sensor networks have been proposed, and many surveys and introductory papers on MAC protocols are available in the literature (see, for example, [29,79,96 and 147]). In the following discussion we will focus mainly on power management issues rather than on channel access methods. Most of them implement a low duty-cycle scheme for power management. We will survey below the most common MAC protocols by classifying them according to the taxonomy illustrated in Fig. 18: TDMA-based, contention-based, and hybrid protocols.

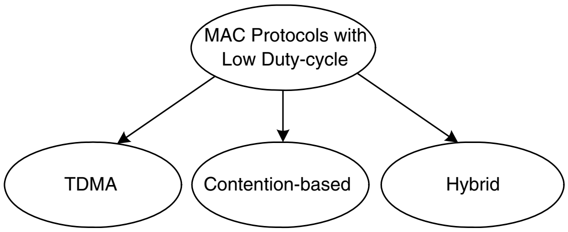

Fig. 18. Classification of MAC protocols based on a low duty-cycle.

Time Division Multiple Access (*TDMA*) schemes naturally enable a duty cycle on sensor nodes as channel access is done on a slot-by-slot basis. As nodes need to turn on their radio only during their own slots, the energy consumption is ideally reduced to the minimum required for transmitting/receiving data.

Contention-based protocols are the most popular class of MAC protocols for wireless sensor networks. They achieve duty cycling by tightly integrating channel access functionalities with a sleep/wakeup scheme similar to those described above. The only difference is that in this case the sleep/wakeup algorithm is not a protocol independent of the MAC protocol, but is tightly coupled with it.

Finally, *hybrid* protocols adapt the protocol behavior to the level of contention in the network. They behave as a contention-based protocol when the level of contention is low, and switch to a TDMA scheme when the level of contention is high.

4.3.1. TDMA-based MAC protocols

In TDMA-based MAC protocols [14,49,53,82,112] time is divided in (periodic) frames and each frame consists of a certain number of time slots. Every node is assigned to one or more slots per frame, according to a certain scheduling algorithm, and uses such slots for transmitting/receiving packets to/from other nodes. In many cases nodes are grouped to form clusters with a cluster-head which is in charge to assign slots to nodes in the cluster (as in Bluetooth [49], LEACH [53], and Energy-aware TDMA-based MAC [14]).

One of the most important energy-efficient TDMA protocol for wireless sensor networks is TRAMA [112]. TRAMA divides time in two portions, a random-access period and a scheduled access period. The random access period is devoted to slot reservation and is accessed with a contention-based protocol. On the contrary, the scheduled access period is formed by a number of slots assigned to an individual node. The slot reservation algorithm is the following. First, nodes derive two-hop neighborhood information, which are required to establish collision free schedules. Then, nodes start an election procedure to associate each slot with a single node. Every node gets a priority of being the owner of a specific slot. This priority is calculated as a hash function of the node identifier and the slot number. The node with the highest priority becomes the owner of a given slot. Finally, nodes send out a synch packet containing a list of intended neighbor destinations for subsequent transmissions. As a consequence, nodes can agree on the slots which they must be awake in. Unused slots can be advertised by their owners for being re-used by other nodes.

FLow-Aware Medium Access (FLAMA) [111] is a TDMA MAC protocol derived from TRAMA, and optimized for periodic monitoring applications. The main idea is to avoid the overhead associated to the exchange of traffic information. As the message flow in periodic reporting applications is rather stable, FLAMA first sets up flows and then uses a pull-based mechanism, so that data are transferred only after being explicitly requested.

As classical slot reservation algorithms tend to be complex and not very flexible. Some researchers have investigated simpler schemes which, at the same time, aim at achieving a good energy efficiency. For example, a low-complexity slot selection mechanism is adopted in [134], where a lightweight medium access protocol (LMAC) is proposed. The main goal of LMAC is to reduce the radio state transitions and the protocol overhead. To this end, data are not acknowledged and the actual slot assignment is based on a binary mask of occupied slot and a random selection among free ones. The main drawback of LMAC is the fixed length of the frame, which has to be specified prior to deployment, and may be problematic. To this end, in [24] an Adaptive Information-centric LMAC (AI-LMAC) is proposed, so that the slot assignment can be more tailored to the actual traffic needs.

4.3.2. Contention-based MAC protocols

Most of MAC protocols proposed for wireless sensor networks are contention-based protocols.

One of the most popular contention-based MAC protocols is B-MAC (Berkeley MAC) [102], a low complexity and low power MAC protocol which is shipped with the TinyOS operating system [54]. The goal of B-MAC is to provide a few core functionalities and an energy efficient mechanism for channel access. First, B-MAC implements basic channel access control features: a backoff scheme, an accurate channel estimation facility and optional acknowledgements. Second, to achieve a low duty cycle B-MAC uses an asynchronous sleep/wake scheme based on periodic listening (see Section 4.2.3) called Low Power Listening (LPL). Nodes periodically wake up to check the channel for activity. The period between consecutive wakeups is called *check interval*. After waking up, nodes remain active for a *wakeup time*, in order to properly detect eventual ongoing transmissions. While the wakeup time is fixed, the check interval can be specified by the application. B-MAC packets are made up of a long preamble and a payload. The preamble duration is at least equal to the check interval so that each node can always detect an ongoing transmission during its check interval. This approach does not require nodes to be synchronized. In fact, when a node detects channel activity, it just remains active and receives first the preamble and then the payload.

A well-known MAC protocol for multi-hop sensor networks is S-MAC (Sensor-MAC) [148], which adopts a scheduled rendezvous communication scheme. Nodes exchange *sync* packets to coordinate their sleep/wakeup periods. Every node can establish its own schedule or follow the schedule of a neighbor by means of a random distributed algorithm. Nodes using the same schedule form a virtual cluster. A node can eventually follow both schedules if they do not overlap, so that it can bridge communication between different virtual clusters. The channel access time is split in two parts. In the listen period nodes exchange sync packets and special control packets for collision avoidance (in a similar way to the IEEE 802.11 standard [57]). In the remaining period the actual data transfer takes place. The sender and the destination node are awake and talk to each other. Nodes not concerned with the communication process can sleep until the next listen period. To avoid high latencies in multi-hop environments S-MAC uses an adaptive listening scheme. A node overhearing its neighbor's transmissions wakes up at the end of the transmission for a short period of time. If the node is the next hop of the transmitter, the neighbor can send the packet to it without waiting for the next schedule. The parameters of the protocol, i.e. the listen and the sleep period, are constants and cannot be varied after the deployment. To this end, the authors of [28] propose an enhanced version of S-MAC called Timeout MAC (T-MAC) and specifically designed for variable traffic load.

Although duty-cycle based MAC protocols are energy efficient, they suffer sleep latency, i.e., a node must wait until the receiver wakes up before it can forward a packet. This latency increases with the number of hops. In addition, the data forwarding process from the nodes to the sink can experience an interruption problem. In fact, the radio sensitivity limits the overhearing range, thus nodes outside the range of the sender and the receiver can't hear the ongoing transmission and go to sleep. That's why in S-MAC and T-MAC the data forwarding process is limited to a few hops. D-MAC [85] is an adaptive duty cycle protocol optimized for data gathering in sensor networks where a tree organization has been established at the network layer. Specifically, in D-MAC the nodes' schedules are staggered according to their position in the data gathering tree, i.e., nodes' active periods along the multi-hop path are adjacent in order to minimize the latency (see Fig. 14). Each node has a slot which is long enough to transmit a packet. A node having more than one packet to transmit explicitly requests additional slots to their parent. In this way the length of the active periods can be dynamically adapted to the network traffic. Finally, D-MAC uses a data prediction scheme to give all children the chance to transmit their packets.

IEEE 802.15.4 [58] is a standard for low-rate, low-power Personal Area Networks (PANs). A PAN is formed by one PAN coordinator which manages the whole network, and, optionally, by one or more coordinators which manage subsets of nodes in the network. Other (ordinary) nodes must associate with a (PAN) coordinator in order to communicate. The supported network topologies are star (single-hop), cluster-tree and mesh (multi-hop). The IEEE 802.15.4 standard supports two different channel access methods: a *beacon enabled* mode and a *non-beacon enabled* mode. The *beacon enabled mode* provides an energy management mechanism based on a duty cycle. Specifically, it uses a superframe structure which is bounded by *beacons* – special synchronization frames generated periodically by coordinator nodes. Each superframe consists of an *active period* and an *inactive period*. In the active period devices communicate with the coordinator they associated with. The active period can be further divided in a contention access period (CAP) and a collision free period (CFP). During the CAP a slotted CSMA/CA algorithm is used for channel access, while in the CFP a number of guaranteed time slots (GTSs) can be assigned to individual nodes. During the inactive period devices enter a low power state to save energy. In the *non-beacon enabled* mode there is no superframe structure, i.e., nodes are always in the active state and use an unslotted CSMA/CA algorithm for channel access and data transmission. In this case, energy conservation is up to the above layers.

IEEE 802.15.4 beacon-enabled mode is suitable for single-hop scenarios. However, the beacon-based duty-cycle scheme has to be extended for multi-hop networks. In [95] the authors propose a maximum delay bound wakeup scheduling specifically tailored to IEEE 802.15.4 networks. The sensor network is assumed to be organized as a cluster tree. An optimization problem is formulated in order to maximize network lifetime while satisfying latency constraints. The optimal operating parameters for single coordinators are then obtained. Therefore, an additional extended synchronization scheme is used for inter-cluster communication.

4.3.3. Hybrid MAC protocols

The basic idea behind hybrid MAC protocols – i.e. switching the protocol behavior between TDMA and CSMA, depending on the level of contention – is not new. In the context of WLAN environments, a Probabilistic TDMA (PTDMA) approach had already been proposed in [37]. In PTDMA time is slotted, and nodes are distinguished in *owners* and *non-owners*. The protocol adjusts the access probability of *owners* and *non-owners* depending on the number of senders. By doing so it adapts the MAC protocol to a TDMA or a CSMA scheme depending on the level of contention in the network. However, PTDMA was conceived for a one-hop wireless scenario. Therefore, it does not take into account issues such as topology changes, synchronization errors, interference irregularities which are quite common in wireless sensor networks.

In the specific context of wireless sensor networks, one of the most interesting hybrid protocols is Z-MAC [113]. In order to define the main transmission control scheme, Z-MAC starts a preliminary setup phase. By means of the neighbor discovery process each node builds a list of two-hop neighbors. Then a distributed slot assignment algorithm is applied to ensure that any two nodes in the two-hop neighborhood are not assigned to the same slot. As a result, it is guaranteed that no transmission from a node to any of its one-hop neighbor interferes with any transmission from its two-hop neighbors. The local frame exchange is aimed at deciding the *time frame*. Z-MAC does not use a global frame equal for all nodes in the network. It would be very difficult and expensive to adapt when a topology change occurs. Instead, Z-MAC allows each node to maintain its own local time frame that depends on the number of neighbors and avoids any conflict with its contending neighbors. The local slot assignment and time frame of each node are then forwarded to its two-hop neighbors. Thus any node has slot and frame information about any two-hop neighbors and all synchronize to a common reference slot. At this point the setup phase is over and nodes are ready for channel access, regulated by the transmission control procedure. Nodes can be in one of the following modes: Low Contention Level (LCL) and High Contention Level (HCL). A node is in the LCL unless it has received an Explicit Contention Notification (ECN) within the last T_{ECN} period. ECNs are sent by nodes when they experience high contention. In HCL only the owners of the current slot and their one-hop neighbors are allowed to compete for accessing the channel. In LCL any node (both owners and non-owners) can compete to transmit in any slot. However, the owners have priority over non-owners. This way Z-MAC can achieve high channel utilization even under low contention because a node can transmit as soon as the channel is available.

4.3.4. Discussion

TDMA-based protocols are inherently energy efficient, as nodes turn on their radio only during their own slots and sleep for the rest of the time. By an appropriate design of the slot assignment algorithm, and a correct sizing of the protocol parameters, it is thus possible to minimize energy consumption. In addition, TDMA-based MAC protocols can solve problems associated with interference among nodes, as it is possible to schedule transmissions of neighboring nodes to occur at different times. However, in practice, TDMA-based protocols have several drawbacks that compensate the benefits in terms of energy saving [113]. First, they have limited flexibility and scalability, because in a real sensor network there may be frequent topology changes due to many factors (e.g. channel conditions, node failures etc.) and the slot allocation may be problematic, so that in many cases a centralized approach is adopted ([49] or LEACH [53]). Second, they need tight synchronization and they are very sensitive to interference [7]. Moreover, TDMA-based protocols perform worse than contention-based protocols in low traffic conditions. For all the above reasons, TDMA MAC protocols are not very frequently used in practical wireless sensor networks.

On the other side, contention-based MAC protocols are robust and scalable. In addition, they generally introduce a lower delay than TDMA-based ones and can easily adapt to traffic conditions. Unfortunately, their energy expenditure is higher than TDMA MACs because of contention and collisions. Duty-cycle mechanisms can help reducing the en-

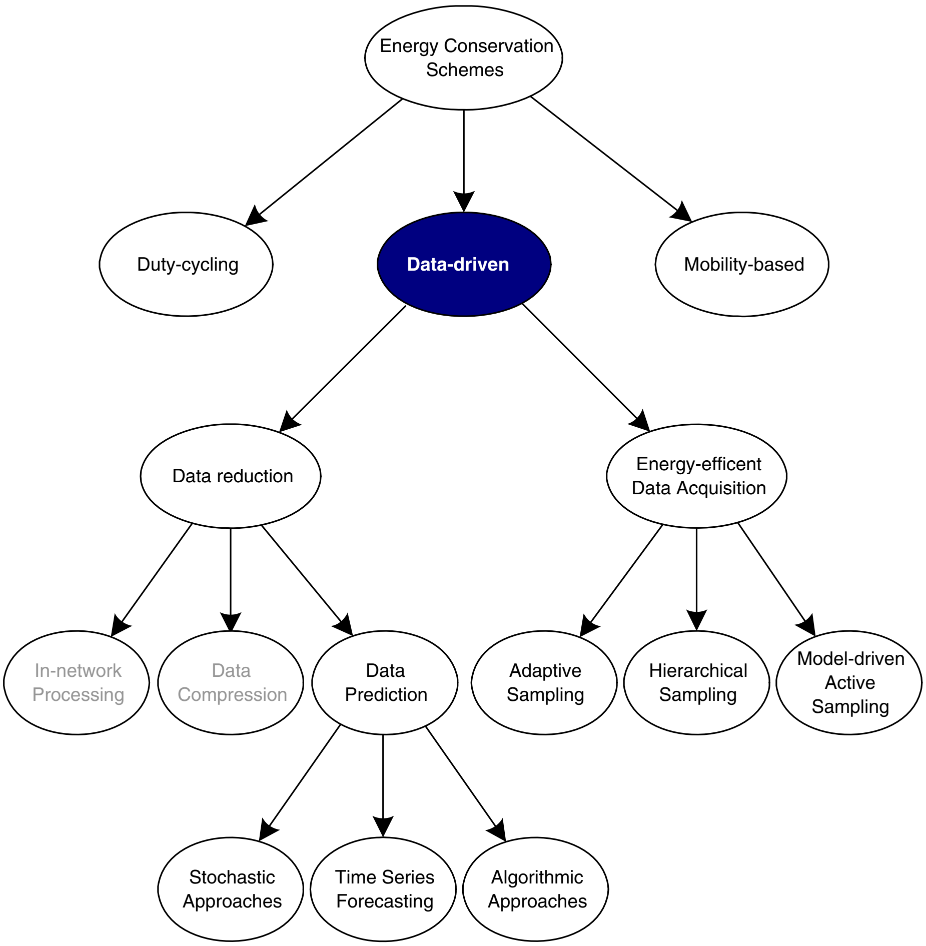

Fig. 19. Detailed taxonomy of data-driven schemes.

ergy wastage, but they need to be designed carefully to be adaptive and to be low latency. An emerging area of interest consists in using features of industry standards such as the IEEE 802.15.4. To this end, the core functions provided by the standard can be used as a basis for developing extensions targeted to a specific scenario, but based on the same specifications. In this way, a particular solution can be easily ported to different platforms, which can communicate under the umbrella of the underlying physical/packet level interface. We anticipate a growing interest in this area in the near future.

Finally, hybrid protocols try to combine the strengths of TDMA-based and contention-based MAC protocols while offsetting their weaknesses. However, these techniques seem to be too complex to be feasible in deployments with a high number of nodes. To this end, solutions such as [51 and 151] – providing simple slot allocation mechanisms and a low protocol overhead – represent a promising direction in the field of energy-efficient MAC protocols for wireless sensor networks.

5. Data-driven approaches

In this section we will survey the main proposal in the field of data-driven techniques for energy conservation, by following the line introduced in Section 3. For the sake of clarity, we report in Fig. 19 the relevant part of the taxonomy as regards data-driven solutions.

5.1. Data prediction

As discussed in Section 3, data prediction techniques build a model describing the sensed phenomenon, so that queries can be answered using the model instead of the actually sensed data. There are two instances of a model in the network, one residing at the sink and the other at source nodes (so that there are as many pairs of models as sources). The model at the sink can be used to answer queries without requiring any communication, thus reducing the energy consumption. Clearly, this operation can be performed only if the model is a valid representation of the phenomenon at a given instant. Here comes into play the model residing at source nodes, which is used to ensure the model effectiveness. To this end, sensor nodes just sample data as usual and compare the actual data against the prediction. If the sensed value falls within an application-dependent tolerance, then the model is considered valid. Otherwise, the source node may transmit the sampled data and/or start a model update procedure involving the sink as well. The features of a specific data prediction technique depend on the way the model is built. To this end, data prediction techniques can be split into three main classes (Fig. 20).

Techniques belonging to the first class derive a *stochastic characterization* of the phenomenon, i.e. in terms of probabilities and/or statistical properties. Two main approaches of this kind are the following. On the one hand, it is possible to map data into a random process described in terms of a probability density function (pdf). Data prediction is then obtained by combining the computed pdfs with the observed samples. On the other hand, a state space representation of the phenomenon can be derived, so that forthcoming samples can be guessed by filtering out a non-predictable component modeled as noise.

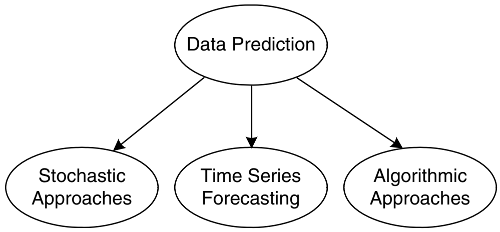

Fig. 20. Classification of data prediction approaches to energy management.

The second class of data prediction techniques is *time series forecasting*, where a set of historical values (the *time series*) obtained by periodical samplings are used to predict a future value in the same series. The main difference with respect to other statistical or probabilistic approaches is that time series analysis explicitly consider the internal structure of data. Generally, a time series can be represented as a combination of a pattern and a random error. The pattern, in turn, is characterized by its trend, i.e. its long-term variation, and its seasonality, i.e. its periodical fluctuation. Once the pattern is fully characterized, the resulting model can be used to predict future values in the time series.

Finally, the last class of data prediction techniques relies on a heuristic or a state-transition model describing the sensed phenomenon. Such *algorithmic approaches* derive methods or procedures to build and update the model on the basis of the chosen characterization.

In this section we will present the most relevant data prediction schemes according to the aforementioned taxonomy.

5.1.1. Stochastic approaches

Stochastic approaches exploit a characterization of the phenomenon in terms of a random process, so that a probabilistic model can be used to predict sensed values[3].

The Ken solution [27] well exemplifies this approach. The general scheme is the same already introduced at the beginning of the current section, i.e. there are a number of models, and each one is replicated at the source and at the sink. In this case, the base model is probabilistic, i.e. after a training phase a probability density function (pdf) referred to a set of attributes is obtained. When the model is not considered valid any more, the source node updates it and transmits a number of samples to the sink, so that the corresponding instance can be updated as well. Ken is flexible enough to use models tailored to a to a specific phenomenon and exploiting spatial or temporal correlations. For example, temporal correlations can be modeled

[3] A data prediction scheme based on a probabilistic model has been proposed first in [30]. However, this work is not limited to reducing data transmissions only, as it defines means to reduce the number of data acquisition as well. For this reason, it will be presented in Section 5.2.3.

as Markov processes. Spatial correlations are somewhat more difficult to handle, in the sense that they require to bring correlated data together at one node, which manages the model representing the evolution of the phenomenon in a certain area. Indeed nodes have to coordinate such that the communication cost is minimized. To this aim the authors propose a disjoint-clique organization, and derive an optimal solution in terms of energy efficiency by using a greedy approach. The authors of [63] exploit a similar approach, but use a Kalman filter as the core model for predictions.

An extension of [27] is given in [70], where a Dynamic Probabilistic Model (DPM) is exploited to implement a probabilistic database view, i.e. a consistent snapshot of data coming from a model with a user-friendly interface. The concept of bringing to the user such a hidden state of the sensor database can be actually implemented through model-based views [31]. The solution proposed in [70] obtains these views by means of a DPM. An interesting application of DPMs consists in deriving the internal (hidden) state of the sensed phenomenon through the available sampled data. For example, it is possible to get the operational state of a node (i.e. if it is working or failing) on the basis of its readings, even though a specific variable is not available in the system. In detail, the authors use a particle filter approach to store the output of a DPM as a set of weighted samples (the particles). The querying system converts queries referring to the DPM view into queries suitable to the particle-based representation. The resulting queries can also be optimized and can perform aggregates over requested data. Particles are updated to match the incoming data stream by performing particle filtering, a Monte Carlo algorithm allowing to estimate the state of DPMs.

5.1.2. Time series forecasting

A typical method to represent time series is given by Moving Average (MA), Auto-Regressive (AR) or a Auto-Regressive Moving Average (ARMA) models. These models are quite simple, but they can be used in many practical cases with good accuracy. More sophisticated models have been also developed (as ARIMA and GARCH), but their complexity does not make them suitable for wireless sensor networks.

PAQ [132] is based on a low-order AR model, with the aim of reducing the amount of computation to be performed by sensors. The first instance of the model is computed by sensor nodes using a set of sampled values. During this learning phase, nodes store the samples in a queue. When the queue is full they can get the model and send it to the sink. Note that the communication between nodes and the sink is limited to the parameters of the model (i.e. the coefficients of the AR model), and does not include sensor readings. Each model is associated to a user-specified error bound. When a predicted value falls within the error bound, the model is considered valid for the given sensed quantity. Otherwise one of the following cases can occur. (i) Sampled data are marked as outliers, for example because of a wrong reading. Outliers can be sent to the sink or simply ignored at the source. (ii) The model is marked as invalid, so that it has to be recomputed (in the same manner as in the learning phase) ad re-sent to the sink. This happens when many consecutive readings fall outside the acceptable error bound. In addition to the basic scheme, a distributed clustering scheme is proposed to group similar sensor nodes. In this case we denote nodes as similar when they are represented by the same model within a given user-specified threshold. Clustering reduces communication even more because the information exchange about models is limited to the cluster-heads and the sink.

SAF [133] improves the previous work in two aspects. First, the AR model is refined so that a trend component is included in the forecast as well. This leads to a better prediction of phenomena with sharp variations in their values. In addition, SAF can detect not only outliers, but also inconsistent data. This happens when nodes cannot compute a stationary model. In this case, nodes can improve model stability in two steps: (i) they can filter the data to smooth outliers; (ii) they can enlarge the size of used data to decrease the impact of the outliers. When such mechanisms are not enough to get a valid model, nodes can explicitly invalidate the model stored at the sink and start rebuilding a new model from scratch. Second, the authors present a centralized clustering scheme which is optimal in the number of clusters and has a complexity of $O(n \log n)$.

The approaches described above assume that a single model is used to represent a given quantity. The work in [81] extends the time series forecasting scheme with an adaptive multi-model selection mechanism. The main idea is the following: as an a-priori knowledge of the phenomenon could be not available, it would be better to let the system itself choose the right model automatically. To this end, all nodes keep a set of models but at a given instant only one of them is used for data prediction. Complex models can lead to a better prediction at the expense of a higher update cost, i.e. they need more parameters to be described properly. At every sampling instant all models are updated, but only the current one is used for prediction. If the error between sensed data and the current model is higher than the allowed threshold, then the current model is switched to the one satisfying the requested accuracy and minimizing the cost of the update. Then an update procedure is performed to ensure that both source and sink nodes are synchronized to the newly selected model. To save nodes' resources, poorly performing models are discarded over time by using a racing mechanism.

5.1.3. Algorithmic approaches

Several other kinds of models have been proposed for data prediction in wireless sensor networks. The common factor they share is the algorithmic approach used to get predictions, starting from a heuristic or behavioral characterization of the sensed phenomena. In the following we discuss the most important approaches of this kind.

Among the first works about data prediction, [43] applied to sensor networks an approach in analogy with video compression. In fact, at a given instant, a sensor network can be thought as an image where each "pixel" is represented by the data sensed at a given node. From this observation is apparent that it is possible to exploit the spatial correlation between samples. In addition, as sensed data generally vary over time, the evolution of readings can be seen as a "sensor movie". Hence the authors present a data prediction technique, called PREMON, which is inspired by the concepts of MPEG encoding. When the monitoring starts, sensor nodes send their initial readings to the sink. Then the sink computes the model by evaluating the correlations between macro-blocks and deriving a motion vector relative to each block. After the model has been obtained, it is sent back to the sensors. From this time on, sensors compare each sample with the prediction derived from the model. When sensed data are close to the prediction within a user-specified error, sensors do not transmit the data to the sink. The model is periodically invalidated, i.e. it is considered out of date and not representing sensed data any more. After the expiration, the process of data collection and model computation starts again from the beginning.

PREMON uses a centralized solution. The *buddy protocol* presented in [44] extends the PREMON approach by using a distributed scheme to exploit temporal correlations between sensed data. In detail, each node attempts to establish a buddy relationship with its neighbors. As a consequence, a number of buddy-groups (i.e. clusters) are formed so that only a single node is representative for all its buddies. This representative node (cluster-head) is responsible for monitoring and query processing, while the others can go to sleep. Within the buddy-group, the cluster-head is rotated so that the energy consumption is spread over all nodes in the group. Communication between ordinary nodes and the cluster-head can be one of default and PREMON. In the default mode, nodes simply send sampled data to the cluster-head. In the PREMON mode, nodes just send a model to the cluster-head, and data which do not fit the predictions, if the case. Each node decides whether to use default or PREMON mode based on an estimate of the energy cost associated to the specific operational mode. If the sampled phenomenon is rather stable, it is more convenient to use the PREMON mode, so that the number of exchanged packets is reduced. On the contrary, if sampled data change fast, the overheads associated to the PREMON mode (i.e. model construction and exchange) can be so high that the default mode can be more energy efficient in this case.

In [52] the authors take a different approach to data prediction, which we can refer to as behavioral, by means of Energy Efficient Data Collection (EEDC) mechanisms. Each node associates an upper and a lower bound, whose difference represents the accuracy of readings, to the actual value of the sensed data. These bounds are sent to the sink, which stores them for each sensor in the network. While acquiring the data, the sensors check the samples against the current bounds. If they fall outside the expected accuracy, the nodes send an update to the sink. This kind of interaction is called *source-initiated update*. On the other side, the sink receives queries from users with an associated requested accuracy. When the requested accuracy is lower than the actual accuracy provided by the value bounds, the sink can respond using the cached range. Otherwise the sink may request the real value and its new approximation to be used for subsequent queries directly to the sensor. This kind of interaction is called *consumer-initiated request and update*. Clearly the updates described above impact on the power consumption of nodes. In detail, they are related to two distinct aspects: the method to select ranges and the way sensor manage their state. The authors hence present a method to compute the optimal ranges to represent data. In addition, they discuss different data collection protocols – i.e. heuristics on how to manage transitions between different node states – to reduce the overall power consumption.

5.1.4. Discussion

The approach taken by the stochastic techniques is general and sound, and also provides means to perform high-level operations such as aggregation. The main drawback of this class of techniques is their rather high computational cost, which may be too heavy for current off-the-shelf sensor devices. To this end, stochastic approaches seem to be more convenient when a number of powerful sensors (e.g. Stargate nodes in a heterogeneous wireless sensor network [108]) are available. Possible improvements in this direction might focus on deriving simplified distributed models for obtaining the desired trade-off between computation and fidelity.

On the contrary, time series forecasting techniques can provide satisfactory accuracy even when simple models (i.e. low order AR/MA) are used. To this end, their implementation in sensor devices is simple and lightweight. In addition, most advanced techniques like [132] do not require the exchange of all sensed data until a model is available. Moreover, they provide the ability to detect outliers and model inconsistencies. However, using a specific type of model needs it to be actually suitable to represent the phenomenon of interest. This would require an a-priori validation phase, which may be not always feasible. An interesting direction involves the adoption of a multi-model approach as the one taken in [81]. As this kind of technique has not been fully explored, there is room to further research and improvements.

Finally, algorithmic techniques has to be considered case by case, because they tend to be more application specific. To this end, a research direction would focus on assessing if a specific solution is efficient for a certain class of applications in real scenarios, so that it can be taken as a reference for further study and possible improvements.

5.2. Energy efficient data acquisition

An emerging class of applications is actually sensing-constrained. This is in contrast with the general assumption that sensing in not relevant from the energy-consumption standpoint. In fact, the energy consumption of the sensing subsystem not only may be relevant, but it can also be greater than the energy consumption of the radio or even greater than the energy consumption of the rest of the sensor node [6]. This can be due to many different factors [107].

- *Power hungry transducers*. Some sensors intrinsically require high power resources to perform their sampling task. For example, sensing arrays such as CCDs or CMOS image sensors or even multimedia sensors [4] generally require a lot of power. Also chemical or biological sensors [32] can be power hungry as well.

- *Power hungry A/D converters*. Sensors like acoustic [121] and seismic transducers [141] generally require high-rate and high-resolution A/D converters. The power consumption of the converters can account for the most significant power consumption of the sensing subsystem, as in [116].
- *Active sensors*. Another class of sensors can get data about the sensed phenomenon by using active transducers (e.g. sonar, radar or laser rangers). In this case sensors have to send out a probing signal in order to acquire information about the observed quantity, as in [34].
- *Long acquisition time*. The acquisition time may be in the order of hundreds of milliseconds or even seconds, hence the energy consumed by the sensing subsystem may be high, even if the sensor power consumption is moderate.

In this case reducing communications may be not enough, but energy conservation schemes have to actually reduce the number of acquisitions (i.e. data samples). It should also be pointed out that energy-efficient data acquisition techniques are not exclusively aimed at reducing the energy consumption of the sensing subsystem. By reducing the data sampled by source nodes, they decrease the number of communications as well. Actually, many energy-efficient data-acquisition techniques have been conceived for minimizing the radio energy consumption, under the assumption that the sensor consumption is negligible.

We will group the approaches for energy-efficient data acquisition by following the classification presented in [107] and depicted in Fig. 21.

As measured samples can be correlated, *adaptive sampling* techniques exploit such similarities to reduce the amount of data to be acquired from the transducer. For example, data of interest may change slowly with time. In this case, temporal correlations (i.e. the fact that subsequent samples do not differ very much between each other) may be exploited to reduce the number of acquisitions. A similar approach can be applied when the investigated phenomenon does not change sharply between areas covered by neighboring nodes. In this case, energy due to sampling (and communication) can be reduced by taking advantage from spatial correlations between sensed data. Clearly, both temporal and spatial correlations may be jointly exploited to further reduce the amount of data to be acquired.

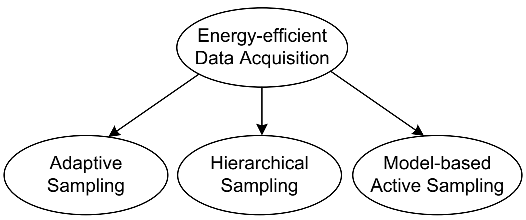

Fig. 21. Classification of energy-efficient data acquisition techniques to energy conservation.

The *hierarchical sampling* approach assumes that nodes are equipped with different types of sensors. As each sensor is characterized by a given resolution and its associated energy consumption, this technique dynamically selects which class to activate, in order to get a tradeoff between accuracy and energy conservation.

Last, *model-based active sampling* takes an approach similar to data prediction (see Section 5.1). A model of the sensed phenomenon is built upon sampled data, so that future values can be forecasted with a certain accuracy. Model-based active sampling exploits the obtained model to reduce the number of data samples, and also the amount of data to be transmitted to the sink – even though this is not their main goal.

In the following we will have a deeper look at energy-efficient data acquisition techniques.

5.2.1. Adaptive sampling

Adaptive sampling can reduce the number of samples by exploiting spatio-temporal correlations between data. The temporal analysis of sensed data is used in [6], where the authors propose an adaptive sampling scheme suitable to snow monitoring for avalanche forecast, though the presented approach is general. A periodically sampled parameter – i.e. the snow equivalent capacity – can be used to derive the actual signal. From the Nyquist theorem it is known that the sampling frequency needed for the correct reconstruction of the original signal should be $F_s \geqslant 2 \cdot F_{max}$ where F_{max} is the maximum frequency in the power spectrum of the considered signal. Unfortunately, choosing F_{max} is not trivial because (i) it cannot be known a priori, thus leading to choose an unnecessary high sampling frequency (*oversampling*), and (ii) it may vary over time (i.e., the process may be non-stationary). To overcome this problem the authors propose an adaptive algorithm that dynamically estimates the current maximum frequency F_{max}, according to the trend of measured data. The algorithm relies on a modified CUSUM test [17] to set the sampling rate. As computations are heavy, a centralized approach is taken, i.e. the algorithm is executed at the sink for each sensor node. The estimated sampling rates obtained by the sink are then notified to sensor nodes. A similar approach is proposed in [62], where the sampling rate is derived based on a Kalman filter. Also in this case, a centralized approach is used, i.e. the sink establishes the sampling rate of nodes. Furthermore, the adaptive sampling mechanism is coupled to a bandwidth reservation mechanism which guarantees that the overall traffic does not exceed the network capacity.

Spatial correlation is used, instead, in [142] where the authors propose the *backcasting* scheme. The main idea is that, nodes deployed with sufficient density do not have to sample the sensed field in a uniform way. In fact, more nodes have to be active in the regions where the variation of the sensed quantity is high. If we consider a given area, the process of activating the desired number of sensor can be done in two phases. In the first phase, called *preview*, only a subset of nodes are activated for sensing. This set of nodes can get a coarse-grained spatial distribution of the sensed phenomenon through a hierarchical estimation of the field. This estimation is performed in several steps. First, the sensors activated during the preview phase are exploited to recursively partition the sensing field in a number of sub-squares with non-uniform resolution (i.e. bigger subsquares corresponds to locations where the observed phenomenon has limited spatial variation and vice-versa). Then, the resulting information is used to group sensors in clusters, each managed by a cluster-head. Finally, a preliminary estimation of which sensors to activate is sent to a fusion center (i.e. the sink). Based on this initial estimation, in the second phase called *refinement*, the fusion center can activate additional sensors in the locations when the spatial correlation is low. To this end the fusion center "backcasts" an activation message to the cluster-heads residing in the smallest squares of the preview partition. These cluster-heads forward the received message to activate additional nodes in the cluster. Note that when the field is smooth, i.e. it has no region with sharp variations of the sensed quantity, the preview phase could be enough to get data with sufficient accuracy, so that the refinement phase is not executed at all.

Spatial correlation is also exploited in [138] to selectively reduce the number of nodes which have to report data to the sink. The authors define a spatial Correlation-based Collaborative MAC protocol (CC-MAC) that regulates sensor node transmissions so as to minimize the number of reporting nodes while achieving the desired level of distortion. To this end, the Iterative Node Selection (INS) algorithm, which resides at the sink, derives the correlation radius R_{corr}, given the maximum distortion that can be tolerated by the application. This information is then broadcast to sensor nodes during the network setup and it is used during the operational phase. Since a sensor node can be both a data source and a data forwarder, the CC-MAC protocol includes two different CSMA/CA-based components, *Event MAC (E-MAC)* and *Network MAC (N-MAC)*. E-MAC prevents the transmission of redundant information during the channel access phase. Initially, all nodes contend for accessing the medium. A node becomes the representative node of the area determined by the correlation radius as soon as it captures the channel, and all other nodes within that area stop their transmissions. Since redundant information are thus filtered out, packets injected into the network must be reliably delivered to the sink. Therefore, N-MAC manages the transmission of route-through packets by giving them a higher priority than newly generated packets. Although this proposal has been conceived for reducing the radio energy consumption, it can reduce the sensor consumption as well, as sensor nodes may switch off their sensing subsystem when they are not reporting data.

An application-specific approach to adaptive sampling is proposed in [152] where a flood warning system called FloodNet is presented. The system includes a grid-based flood predictor which can adjust the reporting rate of individual nodes. An important element of the system is the FloodNet Adaptive Routing (FAR). FAR optimizes the power consumption of nodes by jointly applying adaptive sampling and an energy aware routing based on interest diffusion. The routing algorithm uses two metrics for forwarders selection: priority and data importance. *Priority* is strictly related to energy consumption, i.e. the residual battery power and the energy needed for transmissions. On the other hand, *data importance* is related to the data reporting frequency, i.e. data with high sampling rates are more important because they are associated to critical zones (where variation in the phenomenon dynamics is high). The routing algorithm selects forwarders among nodes with higher priority and lower data importance. The main idea is to use first nodes with higher energy resources. Under the same priority, the routing algorithm selects the nodes which are less loaded by sampling tasks.

Also [110] takes an application-specific approach – exploiting both spatial and temporal correlation – for reducing the number of acquisitions. In the context of an environmental monitoring scenario, the authors use an actuation-enabled robotic sensor called Networked Info-mechanical System (NIMS). The structure of NIMS comprises a mobile node carrying meteorological sensors and an aerial infrastructure supporting the transport of the sensor itself. The horizontal and vertical position of the mobile sensor can be precisely set thanks to the suspended actuator. The sampling problem is addressed as a combination of different phases. At first a navigation criterion is defined, i.e. how the mobile sensor is moved along the field. The proposed solution takes into account both actuation and sampling costs, and characterizes the zones in which the phenomenon has a high variation. In this way the placement of the observations is tailored to the desired error, so that places with high error are covered more densely. Besides exploiting spatial correlation, the system also incorporates an adaptive parameters selection, so that temporal correlations between samples are exploited as well.

5.2.2. Hierarchical sampling

The *hierarchical sampling* approach consists in using sensor nodes equipped with different types of sensors. The quantity of interest can be obtained by all these sensors, each of them characterized by specific performance features, i.e. accuracy and power consumption. In most cases, simple sensors are energy efficient but have a very limited resolution. On the other hand, advanced/complex sensors can give a more detailed characterization of the sensed data at the expense of higher energy consumption. Accuracy can be traded off for energy efficiency by using the low-power sensors to get a coarse-grained information about the sensing field. Then, when an event is detected or a region has to be observed with greater detail, the accurate power hungry sensors can be activated.

For example, consider a target tracking application. Targets can be discovered using low power sensors such as magnetometers or passive acoustic energy detectors. Such sensors can actually detect targets, but they can lead to false positives. In addition, also when the detection is successful, they cannot be accurate enough to identify the type of target. In this case high resolution acoustic beam-forming [131] or image-capturing [109] sensors can help. Instead of keeping these power hungry sensors always on, the less accurate ones are used to detect possible targets. When a target is detected, the more accurate sensors are activated as long as the target has been completely characterized or tracked [116,104]. This kind of technique can also be referred to as *triggered sampling*. For instance, [75] presents a triggered sampling application for structure health monitoring and damage detection. The structure is split into zones containing sensors with different capabilities: m-nodes and μ-nodes. m-nodes are equipped with accelerometers and sample the environment periodically. On the other hand, μ-nodes are provided with strain gauges and they sleep for most of the time. When no problem is detected, m-nodes can sleep until the subsequent activation. Otherwise, first they contact their neighbors to cross-check readings. If the check leads to a suspicious problem, the surrounding densely deployed μ-nodes are activated to get fine-grained information and eventually report the damage.

A similar mechanism exploits a coarse-grained description of a monitored area to identify places which need a more accurate observation. When these areas are discovered, there are two main approaches to get more accurate measurements: (i) sleeping high-resolution sensors residing in the area are activated to sample the quantity of interest; (ii) a robotic sensor is instructed to reach the place which needs more accuracy. Approaches of this kind are known also as *multi-scale sampling*. For example, in [131] a multi-scale approach is applied to a fire emergency scenario. The sensor field is instrumented with static sensors which monitor the environment. When a given area presents an anomaly – i.e. the sampled temperature is over a given threshold – static nodes ask the sink for a deeper investigation. Then the sink dispatches a mobile sensor to visit the emergency location, collect data from the static sensors and take a snapshot of the event scene. After observing the event, the mobile sensor gets back to the sink and reports collected data. A similar solution, applied to environmental monitoring, is proposed in [122].

5.2.3. Model-based active sampling

Model-based active sampling takes an approach similar to data prediction (see Section 5.1). However data prediction keeps the sampling frequency fixed, and uses the periodical acquisition to tune the model. Although this approach reduces the number of communications, it does not impact on the power consumption due to data acquisition. On the other hand, model-based active sampling reduces the number of data samples by using a computed model.

The Barbie-Q (BBQ) query system [30] is a nice example of this approach. The core components of the query system are a probabilistic model and a planner, both residing at the sink. The model is probabilistic, i.e., starting from a given number of samples, a probability density function (pdf) over a set of attributes is derived. The resulting pdf is flexible enough to get both spatial or temporal correlations. The time varying evolution of a set of attributes is modeled as a Markovian process. From the obtained pdfs it is possible to derive the accuracy (i.e. in terms of probability) that a value is included within a user-specified interval. Moreover the model is updated by combining the pdfs with the observed samples, so that future values can be effectively forecasted. The model is built by the sink after an initial learning phase in which nodes transmit sampled data to get a first instance of the pdfs. The stored model is then updated along with received answers to queries. It's up to the planner to decide in which way to collect data. To this end, the planner builds a query plan including a list of sensors to be queried and the most relevant quantities to get. For example, when a user is interested in the temperature sensed in a given area, the planner not only chooses which sensors will be contacted, but also which quantities are to be sampled. In fact, the temperature can be measured directly with the dedicated transducer, but can also be derived from the voltage measured at the destination node (this is an example of correlation between different attributes). As a voltage measurement is much cheaper than a temperature measurement, the planner may choose to get the voltage at some nodes in order to reduce the overall power consumption associated to the query. Upon receiving a query, the planner computes the associated observation cost by considering both sampling and communication costs. As computing the optimal solution has exponential complexity, the authors propose a polynomial-time heuristic which is effective to find practical solutions.

A similar approach is used in [45], where an Adaptive Sampling Approach to Data Collection (ASAP) is proposed. In contrast with BBQ, ASAP splits the network into clusters. To this end, a cluster formation phase is performed to elect cluster heads and select which nodes belong to a given cluster. The metrics used to group nodes within the same cluster include the similarity of sensor readings and the hop count. Therefore, clusters are further divided into subclusters. In fact, not all nodes in a cluster are required to sample the environment. Within a subcluster, only a single node (*sampler*) can acquire data from the environment and send them to the sink. By using an initial set of samples provided by all nodes in the cluster, the cluster head constructs the subclusters and elect samplers (in such a way that there is at least one sampler per subcluster). The probabilistic models – which exploit both spatial and temporal correlations – are built in-network for each subcluster and are sent to the sink. Then the sink can derive sensed data by using the actual data received from samplers or predict them through the model for the other nodes. Both clusters and subclusters are periodically recomputed. Note that the model update requires only to exchange data within a subcluster, so that the communication overhead is reduced as well.

A different approach is taken by [98], where a Utility-based Sensing and Communication (USAC) protocol is presented in the context of glacial environment monitoring. In this case, a limited-window linear regression model is used to forecast samples. The algorithm for updating the sampling frequency is fully distributed, i.e. it is evaluated at each sensor node, and works as follows. If the predicted value falls outside the confidence interval, then the sampling frequency is increased to a pre-defined maximum value f_{max}. This improves the accuracy during the model update, which follows sudden changes in the observed data. On the other side – i.e., if the prediction lies within the confidence interval – then the sampling frequency is decreased by a factor $\alpha \in [0,1]$, unless a minimum pre-defined frequency f_{min} is reached. In addition to the sensing model, the authors also define a routing protocol which accounts for the energy spent due to both sensing and communication. In particular, an opportunity metric is derived, so that sensors which are not relaying data can perform additional sampling. In addition, routes where data is sampled with lower frequency can be preferred to routes where nodes spend more energy for sampling.

5.2.4. Discussion

Adaptive sampling techniques are very promising, because they are quite general and efficient. However, most of the proposed solutions are limited only to a single characterization (i.e. adaptive in time or in space). A more energy-efficient approach would combine both time and space in a single solution, so that multiple directions of information redundancy could be exploited at the same time, as in [110]. In addition, adaptive sampling techniques are often implemented in a centralized fashion, e.g. because they require rather huge computations. To this end, additional work has to be done for reducing the complexity of these solutions, so that viable distributed approaches can be afforded as well. This direction has been recently addressed by [74], where a decentralized adaptive sampling technique is introduced in the context of the FloodNet system. In this scenario, sensor nodes are powered by solar cells. As a consequence, nodes have a limited energy budget available for sampling the environment (i.e. they can take a maximum number of samples per day). The proposed solution is able to adapt the sampling rate on the basis of the available energy, while, at the same time, minimizing the total uncertainty error.

On the other hand, hierarchical sampling techniques are actually feasible when the sensed phenomena can be characterized exploiting different features (i.e., quantities which can be sensed by different transceivers), and the most important is also the most energy hungry. So, this approach is very energy-efficient, but also very tight to the specific application. In fact, it may not always applicable, depending on the specific requirements. Furthermore, the cost associated with the extra transceiver on the sensors may be relevant.

Model-based active sampling solutions share almost the same strengths and weaknesses of the techniques presented in Section 5.1, although in this case the goal of the prediction is to save energy due to data acquisition. These techniques are promising, but they should be improved in the direction of deriving distributed protocols for model computation and diffusion in the network.

6. Mobility-based energy conservation schemes

In this Section we complete the survey by introducing the last energy conservation scheme, which is represented by mobility (Fig. 22). Actually, most of the literature about wireless sensor networks, especially in the early stages of the research in this field, has assumed the reference architecture of Fig. 1 (see Section 1 for details). In this scenario, nodes are assumed to be static, and their density is expected to be large enough to allow communication between any two nodes, eventually by using a multi-hop path. More recently, however, mobility has been considered as an alternative solution for energy-efficient data collection in wireless sensor networks.

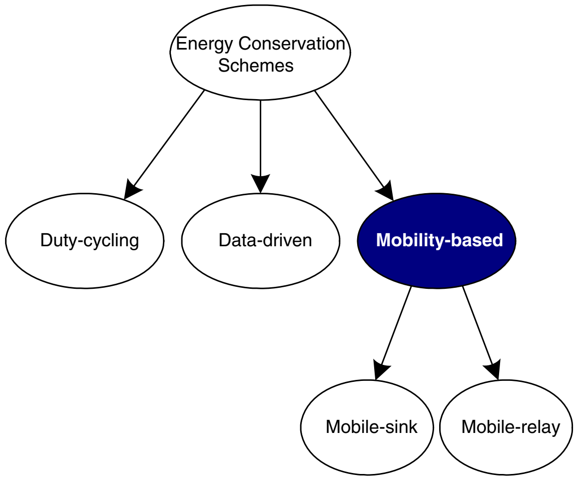

Fig. 22. Classification of mobility-based techniques to energy conservation.

Mobility of sensor nodes is actually feasible, and it can be accomplished in different ways [2]. For example, sensors can be equipped with mobilizers for changing their location. As mobilizers are generally quite expensive from the energy consumption standpoint, adding mobility to sensor nodes may be not convenient. In fact, the resulting energy consumption may be greater than the energy gain due to mobility itself. So, instead of making each sensor node mobile, mobility can be limited to special nodes which are less energy constrained than the ordinary ones. In this case, mobility is strictly tied to the heterogeneity of sensor nodes. On the other side, instead of providing mobilizers, sensors can be placed on elements which are mobile of their own (e.g. animals, cars and so on). There are two different options in this case. First, all sensors are put onto mobile elements, so that all nodes in the network are mobile. Alternatively, only a limited number of special nodes can be placed on mobile elements, while the other sensors are stationary. Anyway, in both cases there is no additional energy consumption overhead due to mobility, but the mobility pattern of mobile elements has to be taken into account during the network design phase (more details are provided below).

By introducing mobility in wireless sensor networks, several issues regarding *connectivity* can be afforded. First, during sensor network design, a *sparse architecture* may be considered as an option, when the application requirements may be satisfied all the same. In this case, it is not required to deploy a large number of nodes, as the constraint of connectivity is relaxed because mobile elements can reach eventual isolated nodes in the network. A different situation happens when a network, assumed to be dense by design, actually turns out to be sparse after the deployment. For example, nodes involved in a random deployment might be not sufficient to cover a given area as expected, due to physical obstacles or damages during placement. In this context, solutions exploiting Unmanned Aircrafts as mobile collectors [66,136] can be successfully used. In addition, an initially connected network can turn into a set of disconnected subnetworks due to hardware failures or energy depletion. In these cases, nodes can exploit mobility in order to remove partitions and re-organize the network so that all nodes are connected again [33]. In this case, the sensor network lifetime can be extended as well.

Mobility is also useful for reducing *energy consumption*. Packets coming from sensor nodes traverse the network towards the sink by following a multi-hop path. When the sink is static, a few paths can be more loaded than others, depending on the network topology and packet generation rates at sources. Generally, nodes closer to the sink also have to relay more packets so that they are subject to premature energy depletion, even when techniques for energy conservation are applied [83]. On the other hand, the traffic flow can be altered if a designated mobile device makes itself responsible for data collection (*mobile data collector*). Ordinary nodes wait for the passage of the mobile device and route messages towards it, so that the communication with mobile data collector takes place in proximity (directly or at most with a limited multi-hop traversal). As a consequence, ordinary nodes can save energy thanks to reduced link errors, contention overhead and forwarding. In addition, the mobile device can visit the network in order to spread more uniformly the energy consumption due to data communication.

As mentioned in Section 3 (and illustrated in Fig. 22), mobility-based energy conservation schemes can be classified depending on the nature of the mobile element, i.e. a mobile sink (MS) or a mobile relay (MR).

6.1. Mobile-sink-based approaches

Many approaches proposed in the literature about sensor networks with mobile sinks (MSs) rely on a Linear Programming (LP) formulation which is exploited in order to optimize parameters such the network lifetime and so on. For example, in [139] the authors propose a model consisting of a MS which can move to a limited number of locations (sink sites) to visit a given sensor and communicate with it (sensors are supposed to be arranged in a square grid within the sensing area). During visits to nodes, the sink stays at the node location for a period of time (sojourn). Nodes not in the coverage area of the sink can send messages along multi-hop paths ending at the MS and obtained using shortest path routing. The authors derive a LP formulation in order to obtain the optimal sojourn times at each sink site. The provided solution maximizes the network lifetime while enforcing balanced energy expenditure, but do not consider the costs due to sink relocation. A similar approach, exploiting multiple MSs, is proposed in [40]. Simulation results show that the multiple sink approach of [139 and 40] can achieve a network lifetime which is five/ten times longer than with the static sink approach.

The model of [139] has been extended in [100], where no specific assumption is made on the way sensors are arranged in the sensing area. In addition, [100] also considers the residual energy at sensors and the routing policy, so that it obtains a network lifetime two times longer than the one achieved with [139]. While both [139 and 100] consider centralized approaches, [16] propose a distributed protocol to approximate the optimal scheme. To this end, the Greedy Maximum Residual Energy (GMRE) scheme is introduced. According to GRME, the MS selects as the new location (among feasible sites) the one which is surrounded by nodes with the higher residual energy. In order to obtain information about the residual energy, a special sentinel node is selected around each feasible site. Sentinels get the energy information from the surrounding nodes and answer the query coming from the MS. The MS uses this information to decide whether or not it should move. Another heuristic-based relocation scheme is considered in [1], where the MS selects its new location in proximity to the nodes with the higher traffic generation rates.

A different class of solutions jointly consider mobility and routing. For example, in [87], an analytical model is developed in order to characterize the network lifetime. Sensors are assumed to be distributed in a circular area. For simplicity, the mobility problem is addressed first by assuming a shortest path routing protocol. Then, the routing strategy is revisited so as to obtain a better outcome. As a result, the optimal mobility strategy is obtained when the MS moves along the periphery of the sensed area (i.e. the perimeter of the circle). With the optimal strategy, nodes near to the border are less loaded than the nodes at the center of the sensed area. The energy surplus of these nodes can be thus exploited to improve the routing strategy. The MS can move on a circle with a radius less then the sensed area's. The circle described by the MS splits the sensed area in two portions: an inner circle and an outer annulus. Nodes in the annulus perform round routing, while nodes in the inner circle apply shortest path routing.

Finally, some researchers have focused on the definition of a data collection/dissemination scheme suitable to sensor networks with MSs. For instance, in [25] the authors evaluate by simulations the joint impact of MS mobility and data-collection strategy. Several scenarios are analyzed by varying the mobility pattern (various kinds of random and deterministic walks) and the different data collection paradigm (MS-initiated, partial and complete multi-hop). In addition to this study, a number of data dissemination schemes targeted to MSs derive from the well-known Directed Diffusion [61]. For example, Two-Tier Data Dissemination (TTDD) [88] is a low-power protocol for efficient data delivery to multiple MSs. Instead of passively waiting for queries coming from sinks, sensor nodes can proactively build a structure to set up forwarding. To this end, the sensing field is represented as a set of grid points. The nodes closest to the grid points (dissemination nodes) are in charge of acquiring forwarding information. Dissemination nodes are the higher tier of the forwarding structure. The lower tier is composed by sensor nodes within the local grid square of the MS current location (cell). As soon as a node has data available, it builds the grid structure by recursive propagation of data announcement messages. As a result of this grid construction phase, dissemination nodes are elected. Then, the MS sends a query by flooding a message within its current cell. The dissemination node closest to the MS will propagate the query along the grid towards the data source. As a result of the query forwarding process, the path from the data source to the sink is obtained so that the requested data can traverse the network in the opposite direction. TTDD assumes that nodes locations are known and a geographic routing protocol is used. An improvement over TTDD is given in [76], where Scalable Energy-efficient Asynchronous Dissemination protocol (SEAD) is introduced. SEAD builds and maintains a dissemination tree (d-tree for short) in which stationary nodes are used as end-points on behalf of the MS. The adopted scheme caches sensed data in the d-tree in such a way to reduce the energy consumption due to data collection.

6.2. Mobile-relay-based approaches

The Mobile Relay (MR) model for data collection in multi-hop ad hoc networks has already been explored in the context of opportunistic networks [12]. One of the most well-known approaches is given by the message ferrying scheme [69, 149]. Message ferries are special mobile nodes which are introduced into a sparse mobile ad hoc network to offer the service of message relaying. Message ferries move around in the network area and collect data from source nodes. They carry stored data and forward them towards the destination node. Thus, message ferries can be seen as a moving communication infrastructure which accommodates data transfer in sparse wireless networks.

A similar scheme has also been proposed in the context of sparse wireless sensor networks through the data-MULE system [64, 119]. In detail, the data-MULE system consists of a three-tier architecture (Fig. 23).

(i) The lower level is occupied by the sensor nodes that periodically perform data sampling from and about the surrounding environment.
(ii) The middle level consists of mobile agents named Mobile Ubiquitous LAN Extensions, or MULEs for short. MULEs move around in the area covered by sensors to gather their data, which have previously been collected and temporarily stored in local buffers. Data MULEs can be for example people, animals, or vehicles too. Generally, they move independently from each other and from the sensor positions by following unpredictable routes. Whenever they get within reach of a sensor they gather information from it.
(iii) The upper level consists of a set of Access Points (APs) which receive information from the MULEs. They are connected to a sink node where the data received is synchronized and stored, multiple copies are identified, and acknowledgments are managed.

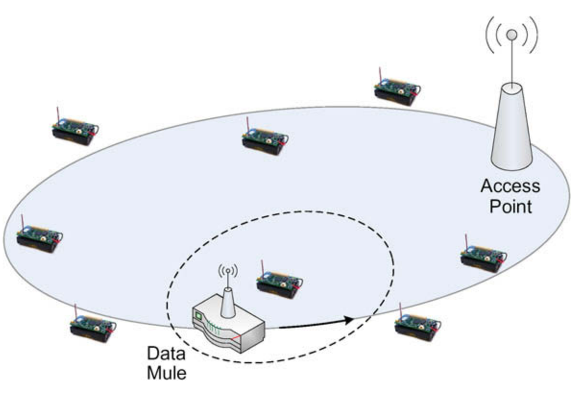

Fig. 23. System architecture of a wireless sensor network with mobile relays.

Sensor nodes – which are supposed to be static – wait for a MULE to pass by and send data to it. Sensor-to-MULE transmissions make use of short-range radio signals and hence energy consumption is low. While moving around, the MULE eventually passes by any AP and transmits the data collected from sensors to it. The authors of [71 and 126] assume that the MR moves along a pre-determined path which is fixed. In fact, changing the trajectory of the MR is not always possible in case of sensor networks because sensors may be deployed in places with obstacles, on rough terrain, or generally where unmanned vehicles can move only in certain directions. Sensor nodes which are located in proximity of the MR path send their data directly to the MR when passing by. Nodes which are far apart from the path followed by the MR send their data over a multi-hop path towards the MR when it passes by or alternatively to one of the nodes which are positioned near to the path of the MR. These nodes act as data caches until the MR passes and finally collects all stored data. Energy saving is addressed in that a large number of nodes is visited by the MR and can thus transmit data over a single hop connection using short range radio. The other nodes which are not in proximity of the path followed by the MR send their data over a multi-hop path which is however shorter, and thus cheaper, with respect to the path established towards a fixed sink node in a classical dense wireless sensor network. To manage this kind of data collection, nodes self-organize into clusters where cluster heads are the nodes which are nearer to the path of the MR whereas the other nodes of the cluster send their data to the cluster head for storage until the next visit of the MR. Data from the sensor nodes of the cluster travel towards the cluster heads according to the directed diffusion protocol. Election of the cluster heads is kept after the first traversal of the MR. During this traversal the MR does not collect any data. Transmissions from cluster heads to the MR occur only when the MR is in proximity so as not to waste energy in useless transmissions. As the trajectory of the MR is assumed to be fixed, it can be controlled only in time. The MR can move at a constant speed worked out, for example, depending on the buffer constraints of the cluster heads. Each cluster head is thus visited before its buffer runs out of space. However, better performance is experienced when the MR alternates between two states: moving at a certain constant speed or stopping. So MR moves fast in places with no, or only a few, sensors and stops in proximity of cluster heads where sensor deployment is denser. The determination of places where sensor deployment is denser (congested regions) is done at each traversal of the MR.

Thanks to the short-range radio communication, the Data MULEs architecture is an energy-efficient solution for data gathering in sparse sensor networks. It also guarantees scalability and flexibility against the network size. Unfortunately, this solution has a couple of limits, both depending on the randomness of the MULEs' motion. First, the latency for data arrival at the sink may be considerable, because (possibly) long time intervals elapse from the sampling instant to the moment the MULE takes the data, and then till the time the MULE actually reaches the AP and delivers the data to it. The second drawback is the fact that sensors have to continuously wait for any MULE to pass and cannot sleep. This leads to energy wastage. Finally, energy-efficient approaches based on a single data mule have limited scalability. To this end in [65] the previous work of [71] is extended by considering multiple mobile elements. An example application of this model in the context of underwater sensor networks [3] is given by [135], where Underwater Autonomous Vehicles are exploited to monitor and model the behavior of the underwater ecosystems.

The architecture of systems described so far assumes an heterogeneous network composed by MRs and static nodes. There are also examples of sensor networks where all nodes are placed on mobile elements. An example of this kind is Zebranet [67], a system for wildlife tracking focused on the monitoring of zebras. A system similar to Zebranet, SWIM, is presented in [124 and 50] in the context of a wildlife telemetry application for monitoring of whales. We present the more interesting aspects of Zebranet in detail below. The animals are equipped with special collars embedding sensor nodes, each including a GPS unit and a dual radio. One of the radio is used for short-range communication, e.g. it is used when zebras gather around water sources. The other radio is used to reach the access point and the animals which are far away from the others. The access point is a vehicle which sometimes traverses the monitored area to gather data. It is worth noting that in this kind of system all nodes are mobile, i.e. both the sink and the sensor nodes, and zebras act as MRs. Zebras act as peers, so that they exchange data during encounters. As zebras are mobile, it is likely that after some time the animals will find other contacts and exchange data again. When, a zebra reaches the area covered by the access point, it uploads the data it is carrying – i.e. its own data and data collected from the encountered peers. A possible solution for data exchange consists in a simple flooding protocol, so that data are pushed to neighbors as soon as they are discovered. Even though this approach can lead to a high success rate (in terms of the number of data collected by the access point), it has excessive bandwidth, capacity and energy demands. In order to save energy, a history-based data collection and dissemination protocol is proposed. Each node is assigned to a hierarchy level, where the level expresses the likelihood of a node being close to the access point. In detail, a level of a node depends on its ability to have successfully transmitted data to the access point in the past. In fact, nodes which have recently been in the range of the access point are likely to relay messages directly or, at most, through a limited number of other nodes. When a node encounters other peers, it first asks their hierarchy level, then it sends data to the one with the highest level. The hierarchy level of a node is increased when it comes in the range of the access point. Conversely, the level is decreased as nodes remain far from the access point. The history-based data dissemination protocol is proved to be efficient in terms of energy and success rate by simulation.

6.3. Discussion

The recent research activity about wireless sensor networks with mobile data collectors has well characterized the behavior of the network and has outlined possible solutions to achieve near-optimal data collection schemes or energy efficient sink movements. However, as mobility-based energy conservation schemes are relatively new in the field of wireless sensor networks, many aspects need to be studied with more attention.

An important aspect is related to the timely discovery of the mobile element by the stationary nodes. Energy-efficient discovery schemes are thus required that minimize energy consumption while keeping the probability of missing contacts with the mobile elements as low as possible. To this end most of the solutions proposed in the literature embed a simple periodic wakeup scheme with an active period [64,71,114,119]. However, the discovery scheme can be targeted to the mobility pattern of sinks/relays by exploiting its distinctive characterization.

Another problem is to define an efficient data transfer protocol specifically targeted to communications between a node and a mobile element. This issue has been analyzed in a number of papers, such as in [136 and 11]. Specifically, the authors of [136] present an opportunistic-ALOHA MAC specifically targeted to wireless sensor networks with aerial vehicles as mobile collectors. The authors of [11], instead, derive an upper bound for the performance of ARQ-based data-transfer protocols and show that commonly adopted schemes for communication between sensors and the mobile data collecting node have low performance, leading to unneeded energy wastage. To this end, an Adaptive Data Transfer (ADT) protocol is proposed for exploiting the past history and tune the communication parameters so as to reduce data transfer times. Actually, which is the best communication approach for data transfer from sensor nodes to the MR is still an open issue.

A problem close to the prior discussion is given by transmission scheduling. In the literature, this issue has been deeply analyzed from the mobile data collector point of view, i.e. the amount of time it has to stay in a given area to collect data coming from static nodes. Unfortunately, the same attention has not been devoted to the other side of communication, i.e. when sensor nodes should transmit gathered data to the mobile element. For example, when the mobile elements visit the same node more than once at different distances, from an energy consumption standpoint it may be convenient for static sensors to defer transmissions at the instant in which the mobile element is closer to the source node. Although some works, such as [11 and 18], deal with the problem of transmission scheduling at sensors, the topic still has to be characterized in depth.

Finally, the energy consumption has to be better characterized with reference to Quality of Service parameters, such as the fraction of reported data or the maximum latency. For example, it is of a limited help to consume a very little fraction of energy at static nodes, when only a small percent of sensed data could be successfully transferred to the mobile data collector. A number of papers including [36 and 127] address the problem of buffering, i.e. the chance data is lost at sources because the storage resources of nodes are scarce. However, most of these proposals give a little attention to the energy spent per transferred message, but focus on the way the mobile element should move to visit nodes in a timely fashion. As the data transfer efficiency strictly depends on the detection of the mobile elements, there is a need to jointly characterize protocols for discovery and data transfer, as in [8]. The next step, which is to be done, deals with design protocols which can adapt to different scenarios (e.g. the mobility pattern of the mobile elements) by automatically tuning the operating parameters to fit actual operating conditions.

7. Conclusions

In this paper we have surveyed the main approaches to energy conservation in wireless sensor networks. Special attention has been devoted to a systematic and comprehensive classification of the solutions proposed in the literature. We did not limit our discussion to topics that have received wide interest in the past, but we have also stressed the importance of different approaches such as data-driven and mobility-based schemes. It is worth noting that the considered approaches should not be considered as alternatives, they should rather be exploited together.

We can draw final observations about the different approaches to energy management. As far as "traditional" techniques to energy saving, an important aspect which has to be investigated more deeply is the integration of the different approaches into a single off-the-shelf workable solution. This involves characterizing the interactions between different protocols and exploiting cross-layer interactions.

Another interesting point is that most of the solutions proposed in the literature assume that the energy consumption of the radio is much higher than the energy consumption due to data sampling or data processing. Many real applications, however, have shown the power consumption of the sensor is comparable to, or even greater than, the power needed by the radio. In addition, the sampling phase may need a long time – especially if we compare it to the time needed for communications – so that the energy consumption of the sensor itself can be very high as well. We think that the field of energy conservation targeted to data acquisition has not been fully explored yet, so that there is room for developing convenient techniques to reduce the energy consumption of the sensors.

Finally, we observe an increasing interest towards a sparse sensor network architecture. In many practical applications such a network can be very efficient and robust if communication protocols can appropriately exploit the mobility of collector nodes. We are persuaded that this class of approaches will get an even greater importance and attention within the research community in the next years.

Acknowledgements

This work was carried out under the financial support of the Italian Ministry for Education and Scientific Research (MIUR) in the framework of the FIRB ArtDeCO (Adaptive InfRasTructures for DECentralized Organizations) project and the Information Society Technologies Program of the European Commission under the FP6-2005-NEST-PATH MEMORY project.

References

[1] K. Akkaya, M. Younis, "Energy-aware to mobile gateway in wireless sensor networks", in: Proc. IEEE Globecom 2004 Workshops, November 29–December 3, Dallas, United States, 2004, pp. 16–21.
[2] I.F. Akyildiz, I.H. Kasimoglu, Wireless sensor and actor networks: research challenges, Ad Hoc Networks Journal 2 (4) (2004) 351–367.
[3] I.F. Akyildiz, D. Pompili, T. Melodia, Underwater acoustic sensor networks: research challenges, Ad Hoc Networks 3 (3) (2005) 257–279.
[4] I.F. Akyildiz, T. Melodia, K.R. Chowdhury, A survey on wireless multimedia sensor networks, Computer Networks 51 (4) (2007) 921–960.
[5] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Computer Networks 38 (4) (2002).
[6] C. Alippi, G. Anastasi, C. Galperti, F. Mancini, M. Roveri, Adaptive sampling for energy conservation in wireless sensor networks for snow monitoring applications", in: Proc. IEEE International Workshop on Mobile Ad Hoc and Sensor Systems for Global and Homeland Security (MASS-GHS 2007), Pisa, Italy, October 8, 2007.
[7] G. Anastasi, E. Borgia, M. Conti, E. Gregori, A. Passarella, Understanding the real behavior of 802.11 and mote ad hoc networks, Pervasive and Mobile Computing 1 (2) (2005).
[8] G. Anastasi, M. Conti, M. Di Francesco, "Reliable and energy-efficient data collection in sensor networks with data mules: an integrated performance evaluation", in: Proc. 13th IEEE Symposium on Computers and Communications (ISCC 2008), Marrakech, Morocco, July 6–9, 2008.
[9] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, An adaptive and low-latency power management protocol for wireless sensor networks, in: Proc 4th ACM International Workshop on Mobility Management and Wireless Access (MobiWac 2006), Torremolinos, Spain, October 2, 2006.
[10] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, How to prolong the lifetime of wireless sensor networks", in: M. Denko, L. Yang (Eds.), Mobile Ad hoc and Pervasive Communications, American Scientific Publishers, in press (Chapter 6). http://info.iet.unipi.it/~anastasi/papers/Yang.pdf.
[11] G. Anastasi, M. Conti, E. Monaldi, A. Passarella, An adaptive data-transfer protocol for sensor networks with data mules, in: Proc. IEEE International Symposium on a World of Wireless, Mobile, and Multimedia Networks (WoWMoM 2007), Helsinki, Finland, June 18–21, 2007.
[12] G. Anastasi, M. Conti, A. Passarella, L. Pelusi, Mobile-relay forwarding in opportunistic networks, in: M. Ibnkahla, (Ed.), Adaptation and Cross Layer Design in Wireless Networks, (Chapter 13), Taylor and Francis, New York (USA), July 2008.
[13] J. Ansari, D. Pankin, P. Mähönen, Radio-triggered wake-ups with addressing capabilities for extremely low power sensor network applications, in: Proc. 5th European Conference on Wireless Sensor Networks (EWSN 2008), Bologna, Italy, January 30–February 1, 2008.
[14] K. Arisha, M. Youssef, M. Younis, Energy-aware TDMA-based MAC for sensor networks, in: Proc. IEEE Workshop on Integrated Management of Power Aware Communications, Computing and Networking (IMPACCT 2002), New York City, USA, May 2002.
[15] T. Armstrong, Wake-up based power management in multi-hop wireless networks. http://www.eecg.toronto.edu/~trevor/Wakeup/index.html.

[16] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, Z.M. Wang, Controlled sink mobility for prolonging wireless sensor networks lifetime, ACM/Elsevier Journal on Wireless Networks (2007).
[17] M. Basseville, I.V. Nikiforov, Detection of Abrupt Changes: Theory and Application, Prentice-Hall, Inc., 1993.
[18] L. Bölöni, D. Turgut, Should i send now or send later? a decision-theoretic approach to transmission scheduling in sensor networks with mobile sinks, Special Issue of Wiley's Wireless Communications and Mobile Computing Journal (WCMC) on Mobility Management and Wireless Access, 2007.
[19] P. Buonadonna, D. Gay, J. Hellerstein, W. Hong, S. Madden, TASK: sensor network in a box, in: Proc. European Workshop on Sensor Networks (EWSN 2005), January 2005.
[20] Q. Cao, T. Abdelzaher, T. He, J. Stankovic, Toward optimal sleep scheduling in sensor networks for rare event detection, in: Proc. International Conference on Information Processing in Sensor Networks (IPSN 2005), April 2005.
[21] P. Casari, A. Marcucci, M. Nati, C. Petrioli, M. Zorzi, A detailed simulation study of geographic random forwarding (GeRaF) in wireless sensor networks, in: Proc. Military Communications Conference 2005 (MILCOM 2005), October 17–20, vol. 1, 2005, pp. 59–68.
[22] A. Cerpa, D. Estrin, Ascent: adaptive self-configuring sensor network topologies, in: Proc. IEEE Infocom, 2002
[23] A. Chakrabarti, A. Sabharwal, B. Aazhang, Using predictable observer mobility for power efficient design of sensor networks, in: Proc. 2nd International Workshop on Information Processing in Sensor Networks (IPSN 2003), 2003, pp.129–145.
[24] S. Chatterjea, L. van Hoesel, P. Havinga, AI-LMAC: an adaptive, information-centric and lightweight MAC protocol for wireless sensor networks, in: Proc. 2nd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2004), Melbourne, Australia, December 2004.
[25] I. Chatzigiannakis, A. Kinalis, S. Nikoletseas, Sink mobility protocols for data collection in wireless sensor networks, in: Proc. 4th International Workshop on Mobility Management and Wireless Access (MobiWAC 2006), Terromolinos, Spain, October 2006, pp. 52–59.
[26] B. Chen, K. Jamieson, H. Balakrishnan, R. Morris, Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks, ACM Wireless Networks 8 (5) (2002).
[27] D. Chu, A. Deshpande, J.M. Hellerstein, W. Hong, Approximate data collection in sensor networks using probabilistic models, in: Proc. 22nd International Conference on Data Engineering (ICDE06), Atlanta, GA, April 3–8, 2006, p. 48.
[28] T.V. Dam and K. Langendoen, An adaptive energy-efficient MAC protocol for wireless sensor networks, in: The First ACM Conference on Embedded Networked Sensor Systems (Sensys'03), Los Angeles, CA, USA, November 2003.
[29] I. Demirkol, C. Ersoy, F. Alagoz, MAC protocols for wireless sensor networks: a survey, IEEE Communications Magazine (2006).
[30] A. Deshpande, C. Guestrin, S. Madden, J.M. Hellerstein, W. Hong, Model-driven data acquisition in sensor networks, in: Proc. 30th International Conference on Very Large Data Bases (VLDB 2004), Toronto, Canada, 29 August–3 September, 2004.
[31] A. Deshpande, S. Madden, MauveDB: supporting model-based user views in database systems, in: Proc. 25th ACM International Conference on Management of Data (SIGMOD'06), Chicago (IL), USA, June 27–29, 2006, pp. 73–84.
[32] D. Diamond, Energy consumption issues in chemo/biosensing using WSNs", Energy and Materials: Critical Issues for Wireless Sensor Networks Workshop, June 30, 2006.
[33] G. Dini, M. Pelagatti, I.M. Savino, An algorithm for reconnecting wireless sensor network partitions, in: Proc. 5th European Conference on Wireless Sensor Networks (EWSN 2008), Bologna, Italy, January 30–February 1, 2008, pp. 253–267.
[34] M. Ditzel, F.H. Elferink, "Low-power radar for wireless sensor networks, in: Proc. 3rd European Radar Conference (EuRAD), September 2006.
[35] O. Dousse, P. Mannersalo, P. Thiran, Latency of wireless sensor networks with uncoordinated power saving mechanisms, in: Proc. 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '04), Tokyo, Japan, May 24–26, 2004.
[36] E. Ekici, Y. Gu, D. Bozdag, Mobility-based communication in wireless sensor networks, IEEE Communications Magazine 44 (6) (2006) 56–62.
[37] A. Ephremides, O. Mowafi, Analysis of a hybrid access scheme for buffered user probabilistic time division, IEEE Transactions on Software Engineering 8 (1) (1982) 52–61.
[38] Y.R. Faizulkhakov, Time Synchronization Methods for Wireless Sensor Networks: A Survey, Programming and Computer Software, 33(4), Plenum Press, 2007.
[39] E. Fasolo, M. Rossi, J. Widmer, M. Zorzi, In-network aggregation techniques for wireless sensor networks: a survey, IEEE Wireless Communications 14 (2) (2007) 70–87.
[40] S.R. Gandham, M. Dawande, R. Prakash, S. Venkatesan, Energy efficient schemes for wireless sensor networks with multiple mobile base stations, in: Proc. IEEE Globecom 2003, San Francisco, CA December 1–5, vol. 1, 2003, pp. 377–381.
[41] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, D. Estrin, Networking issues in wireless sensor networks, Journal of Parallel and Distributed Computing 64 (2004) 799–814.
[42] P.B. Godfrey, D. Ratajczak, Naps: scalable, robust topology management in wireless ad hoc networks, in: Proc. 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04), Berkeley, California, USA, April 26–27, 2004.
[43] S. Goel, T. Imielinski, Prediction-based monitoring in sensor networks: taking lessons from MPEG, ACM Computer Communication Review 31 (5) (2001).
[44] S. Goel, A. Passarella, T. Imielinski, Using buddies to live longer in a boring world, in: Proc. IEEE International Workshop on Sensor Networks and Systems for Pervasive Computing (PerSeNS 2006), Pisa, Italy, March 13, 2006.
[45] B. Gedik, L. Liu, P.S. Yu, ASAP: an adaptive sampling approach to data collection in sensor networks, IEEE Transactions on Parallel Distributed Systems 18 (12) (2007).
[46] G. Grimmett, Percolation, second ed., Springer-Verlag, 1999.
[47] L. Gu, J. Stankovic, Radio-triggered wake-up for wireless sensor networks, Real-Time Systems Journal 29 (2005) 157–182.
[48] L. Guo, J. Rabaey, Low power distributed MAC for ad hoc sensor radio networks, in: Proc. IEEE Globecom, 2001
[49] J. Haartsen, The bluetooth radio system, IEEE Personal Communications 7 (1) (2000) 28–36.
[50] Z.J. Haas, T. Small, A new networking model for biological applications of ad hoc sensor networks, IEEE/ACM Transactions on Networking (TON) 14 (1) (2006) 27–40.
[51] G. Halkes, K. Langendoen, Crankshaft: an energy-efficient MAC-protocol for dense wireless sensor networks, in: Proc. 4th European Conference on Wireless Sensor Networks (EWSN'07), Delft, The Netherlands, January 2007.
[52] Q. Han, S. Mehrotra, N. Venkatasubramanian, Energy efficient data collection in distributed sensor environments, in: Proc. 24th IEEE International Conference on Distributed Computing Systems (ICDCS'04), March, 2004, pp. 590–597.
[53] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proc. Hawaii International Conference on System Sciences (HICSS-34), January, 2000.
[54] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D.E. Culler, K.S.J. Pister, System architecture directions for networked sensors, in: Proc. ASPLOS 2000, November, 2000
[55] B. Hohlt, E. Brewer, Network power scheduling for tiny OS applications", in: Proc. 2nd IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS'06), San Francisco, California, June 2006.
[56] B. Hohlt, L. Doherty, E. Brewer, Flexible power scheduling for sensor networks, in: Proc. ACM Workshop on Information Processing in Sensor Networks (ISPN 2004), Berkeley, USA, April 26–27, 2004.
[57] IEEE 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999.
[58] IEEE 802.15.4, Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), May 2003.
[59] IEEE Pervasive Computing, Energy Harvesting and Conservation, vol. 4, issue 1, January–March 2005.
[60] IEEE Pervasive Computing, RFID Technology, vol. 4, issue 1, January–March 2006.
[61] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, in: Proc. Sixth Annual International Conference on Mobile Computing and Networking (MobiCOM'00), Boston, Massachussetts, August 2000.
[62] A. Jain, J.Y. Chang, Adaptive sampling for sensor networks, in: Proc. 1st international workshop on Data management for sensor networks (DMSN 2004), Toronto, Canada, August 30th, 2004, pp. 10–16.
[63] A. Jain, E.Y. Chang, Y.-F. Wang, Adaptive stream resource management using Kalman filters, in: Proc. ACM International

Conference on Management of Data (SIGMOD2004), Paris (France), June 13–18, 2004, pp. 11–22.
[64] S. Jain, R. Shah, W. Brunette, G. Borriello, S. Roy, Exploiting mobility for energy efficient data collection in wireless sensor networks, ACM/Springer Mobile Networks and Applications 11 (2006) 327–339.
[65] D. Jea, A.A. Somasundara, M.B. Srivastava, Multiple controlled mobile elements (data mules) for data collection in sensor networks, in: Proc. 1st IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS'05), Marina del Rey, California, June 2005.
[66] A. Jenkins, D. Henkel, T. Brown, Sensor data collection through unmanned aircraft gateways, in: Proc. AIAA Infotech@Aerospace 2007 Conference and Exhibit, Rohnert Park, California, May 7–10, 2007.
[67] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, D. Rubenstein, Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with Zebranet, in: Proc. Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2002.
[68] H. Jun, M. Ammar, E. Zegura, Power management in delay tolerant networks: a framework and knowledge-based mechanisms, in: Proc. IEEE Conference on Sensor and Ad Hoc Communication and Networks (SeCon 2005), 2005.
[69] H. Jun, W. Zhao, M. Ammar, E. Zegura, C. Lee, Trading latency for energy in wireless ad hoc networks using message ferrying, in: Proc. IEEE PerCom Workshops, International Workshop on Pervasive Wireless Networking (PWN 2005), March 2005.
[70] B. Kanagal, A. Deshpande, Online filtering, smoothing and probabilistic modeling of streaming data, in: Proc. 24th International Conference on Data Engineering (ICDE 2008), Cancún, México, April 7–12, 2008.
[71] A. Kansal, A.A. Somasundara, D.D. Jea, M.B. Srivastava, D. Estrin, Intelligent fluid infrastructure for embedded networks, in: Proc. 2nd ACM/SIGMOBILE International Conference on Mobile Systems, Applications, and Services (MobySys 2004), Boston, MA, June 6–9, 2004, pp. 111–124.
[72] H. Karl, A. Willig, Protocols and Architectures for Wireless Sensor Networks, first ed., Topology Control, Wiley, 2005. June 24 (Chapter 10).
[73] A. Keshavarzian, H. Lee, L. Venkatraman, Wakeup scheduling in wireless sensor networks, in: Proc. ACM MobiHoc 2006, Florence Italy, May 2006, pp. 322–333.
[74] J. Kho, A. Rogers, N.R. Jennings, Decentralised adaptive sampling of wireless sensor networks, in: Proc. First International Workshop on Agent Technology for Sensor Networks (ATSN 07), Honolulu, Hawaii, USA, May 14, 2007.
[75] T. Kijewski-Correa, M. Haenggi, P. Antsaklis, Wireless sensor networks for structural health monitoring: a multi-scale approach, ASCE Structures 2006 Congress (2006).
[76] H.S. Kim, T.F. Abdelzaher, W.H. Kwon, Minimum energy asynchronous dissemination to mobile sinks in wireless sensor networks, in: Proc. First International Conference on Embedded Networked Sensor Systems (SenSys 2003), Los Angeles, November 5–7, 2003, pp. 193–204.
[77] Z. Kong, E.M. Yeh, Distributed energy management algorithm for large-scale wireless sensor networks, in: Proc. 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '07), Montreal, Quebec, Canada, September 09–14, 2007.
[78] F. Koushanfar, N. Taft, M. Potkonjak, Sleeping coordination for comprehensive sensing using isotonic regression and domatic partitions, in: Proc. Infocom 2006, Barcelona, Spain, April 2006.
[79] K. Langendoen, Medium access control in wireless sensor networks, in: H. Wu, Y. Pan (Eds.), Medium Access Control in Wireless Networks, Practice and Standards, vol. II, Nova Science Publishers, 2008 (Book Chapter).
[80] K. Langendoen, N. Reijers, Distributed localization in wireless sensor networks: a quantitative comparison, Computer Networks 43 (4) (2003).
[81] Y.-Ä. Le Borgne, S. Santini, G. Bontempi, Adaptive Model Selection for Time Series Prediction in Wireless Sensor Networks, Signal Processing 87 (12) (2007) 3010–3020.
[82] J. Li, G. Lazarou, A bit-map-assisted energy-efficient MAC scheme for wireless sensor networks, in: Proc. International Symposium on Information Processing in Sensor Networks (IPSN 2004), Berkeley USA, April 2004, pp. 56–60.
[83] J. Li, P. Mohapatra, Analytical modeling and mitigation techniques for the energy hole problem in sensor networks, Pervasive Mobile Computing 3 (3) (2007) 233–254.
[84] Y. Li, W. Ye, J. Heidemann, Energy and latency control, in low duty cycle MAC protocols, in: Proc. IEEE Wireless Communication and Networking Conference, New Orleans, USA, March 2005.
[85] G. Lu, B. Krishnamachari, C.S. Raghavendra, An adaptive energy-efficient and low-latency Mac for data gathering in wireless sensor networks, in: Proc. 18th International Parallel and Distributed Processing Symposium, April 2004, pp. 224, 26–30.
[86] G. Lu, N. Sadagopan, B. Krishnamachari, A. Goel, Delay efficient sleep scheduling in wireless sensor networks, in: Proc. IEEE Infocom 2005, March, 2005.
[87] J. Luo, J.P. Hubaux, Joint mobility and routing for lifetime elongation in wireless sensor networks, in: Proc. IEEE Infocom 2005, Miami, USA, March 13–17, vol. 3, 2005, pp. 1735–1746.
[88] H. Luo, F. Ye, J. Cheng, S. Lu, L. Zhang, TTDD: two-tier data dissemination in large-scale wireless sensor networks, Elsevier/ACM Wireless Networks 11 (1–2) (2005) 161–175.
[89] S. Madden, M. Franklin, J. Hellerstein, W. Hong, TAG: a tiny aggregation service for ad hoc sensor networks, in: Proc. Annual Symposium on Operating Systems Design and Implementation (OSDI), 2002.
[90] S. Madden, M.J. Franklin, J.M. Hellerstein, W. Hong, TinyDB: an acquisitional query processing system for sensor networks, ACM Transactions on Database Systems 30 (1) (2005) 122–173.
[91] U. Malesci, S. Madden, A measurement-based analysis of the interaction between network layers in TinyOS, in: Proc. 3rd European Conference on Wireless Sensor Networks (EWSN 2006), Zurich, Switzerland, February 13–15, 2006.
[92] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, J. Anderson, Wireless sensor networks for habitat monitoring, in: Proc. ACM Workshop on Wireless Sensor Networks and Applications, Atlanta, USA, September 2002, pp. 88–97.
[93] G. Mao, B. Fidan, B.D.O. Anderson, Wireless sensor network localization techniques, Elsevier/ACM Computer Networks 51 (10) (2007) 2529–2553.
[94] T. Melodia, D. Pompili, I.F. Akyildiz, On the interdependence of distributed topology control and geographical routing in ad hoc and sensor networks, IEEE Journal of Selected Areas in Communications 23 (3) (2005) 520–533.
[95] D. Mirza, M. Owrang, C. Schurgers, Energy-efficient wakeup scheduling for maximizing lifetime of IEEE 802.15.4 Networks, in: Proc. International Conference on Wireless Internet (WICON'05), Budapest, Hungary, July 2005, pp. 130–137.
[96] P. Naik, K. Sivalingam, A Survey of MAC Protocols for Wireless Sensor Networks, in: C. Raghavendra, K. Sivalingam, T. Znati (Eds.), Wireless Sensor Networks, Kluwer Academic Publishers., 2004 (Chapter 5).
[97] M. Nosovich, T. Todd, Low power rendezvous and RFID wakeup for embedded wireless networks, in: Proc. of Annual IEEE Computer Communications Workshop (CCW 2000), Captiva Island, USA, October 15–18, 2000.
[98] P. Padhy, R.K. Dash, K. Martinez, N.R. Jennings, A utility-based sensing and communication model for a glacial sensor network, in: Proc. Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '06), Hakodate, Japan, May 08–12, 2006.
[99] N.A. Pantazis, D.D. Vergados, A survey on power control issues in wireless sensor networks, IEEE Communications Surveys and Tutorials 9 (4) (2007) 86–107.
[100] I. Papadimitriou, L. Georgiadis, Energy-aware routing to maximize lifetime in wireless sensor networks with mobile sink, Journal of Communications Software and Systems 2 (2) (2006) 141–151.
[101] V. Paruchuri, S. Basavaraju, R. Kannan, S. Iyengar, Random asynchronous wakeup protocol for sensor networks, in: Proc. IEEE International Conference on Broadband Networks (BROADNETS 2004), 2004.
[102] J. Polastre, J. Hill, D. Culler, Versatile low power media access for sensor networks, in: Proc. Second ACM Conference on Embedded Networked Sensor Systems (SenSys), November 3–5, 2004.
[103] G. Pottie, W. Kaiser, Wireless integrated network sensors, Communication of ACM 43 (5) (2000) 51–58.
[104] A. Prati, R. Vezzani, L. Benini, E. Farella, P. Zappi, An integrated multi-modal sensor network for video surveillance, in: Proc. Third ACM International Workshop on Video Surveillance and Sensor Networks (VSSN '05), Hilton, Singapore, November 11, 2005.
[105] S.S. Pradhan, K. Ramchandran, Distributed source coding using syndromes (DISCUS): design and construction, IEEE Transactions on Information Theory 49 (3) (2003) 626–643.
[106] J. Rabaey, J. Ammer, T. Karalar, S. Li, B. Otis, M. Sheets, T. Tuan, Picoradios for wireless sensor networks: the next challenge in

ultra-low-power design, in: Proc. International Solid-State Circuits Conference, San Francisco, CA, February 3–7, 2002.
[107] V. Raghunathan, S. Ganeriwal, M. Srivastava, Emerging techniques for long lived wireless sensor networks, IEEE Communications Magazine 44 (4) (2006) 108–114.
[108] V. Raghunathan, C. Schurghers, S. Park, M. Srivastava, Energy-aware wireless microsensor networks, IEEE Signal Processing Magazine (2002) 40–50.
[109] M. Rahimi, R. Baer, O. Iroezi, J. Garcia, J. Warrior, D. Estrin, M.B. Srivastava, Cyclops: in situ image sensing and interpretation, in: Proc. 3rd International Conference on Embedded Networked Sensor Systems (SenSys 2005), San Diego, California, USA, November 2–4, 2005, pp. 192–204.
[110] M. Rahimi, M. Hansen, W.J. Kaiser, G.S. Sukhatme, D. Estrin, Adaptive sampling for environmental field estimation using robotic sensors, in: Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005), 2–6 August, 2005, pp.3692–3698.
[111] V. Rajendran, J. Garcia-Luna-Aceves, K. Obraczka, Efficient application-aware medium access for sensor networks, in: Proc. 2nd IEEE Conference on Mobile Ad-hoc and Sensor Systems (MASS 2005), Washington, DC, November 2005.
[112] V. Rajendran, K. Obracza, J.J. Garcia-Luna Aceves, Energy-efficient, collision-free medium access control for wireless sensor networks, in: Proc. ACM SenSys 2003, Los Angeles (USA), November 2003.
[113] I. Rhee, A. Warrier, M Aia, J. Min, Z-MAC: a hybrid MAC for wireless sensor networks, in: Proc. ACM SenSys 2005, S Diego, USA, November 2005.
[114] V.P. Sadaphal, B.N. Jain, Random and periodic sleep schedules for target detection in sensor networks, in: Proc. 4th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS 2007), Pisa, 8–11 October 2007.
[115] P. Santi, Topology control in wireless ad hoc and sensor networks, ACM Computing Survey 37 (2) (2005) 164–194.
[116] B. Schott, M. Bajura, J. Czarnaski, J. Flidr, T. Tho, L. Wang, A modular power-aware microsensor with >1000× dynamic power range, in: Proc. Fourth International Symposium on Information Processing in Sensor Networks (IPSN 2005), April 15, 2005, pp. 469–474.
[117] C. Schurgers, V. Tsiatsis, S. Ganeriwal, M.B. Srivastava, Optimizing sensor networks in the energy-latency-density design space, IEEE Transactions on Mobile Computing 1 (1) (2002) 70–80.
[118] C. Schurgers, V. Tsiatsis, M.B. Srivastava, STEM: topology management for energy efficient sensor networks, in: IEEE Aerospace Conference'02, Big Sky, MT, March 10–15, 2002.
[119] R.C. Shah, S. Roy, S. Jain, W. Brunette, Data MULEs: modeling a three-tier architecture for sparse sensor networks, in: Proc. IEEE International Workshop on Sensor Network Protocols and Applications (SNPA 2003), May 11, 2003, pp. 30–41.
[120] E. Shih, P. Bahl, M. Sinclair, Wake on wireless: an event driven energy saving strategy for battery operated devices, in: Proc. ACM MobiCom 2002, Atlanta, USA, September 2002.
[121] G. Simon, M. Maróti, Á. Lédeczi, G. Balogh, B. Kusy, A. Nádas, G. Pap, J. Sallai, K. Frampton, Sensor network-based countersniper system, in: Proc. 2nd International Conference on Embedded Networked Sensor Systems (SenSys'04), November 03–05, 2004.
[122] A. Singh, D. Budzik, W. Chen, M.A. Batalin, M. Stealey, H. Borgstrom, W.J. Kaiser, Multiscale sensing: a new paradigm for actuated sensing of high frequency dynamic phenomena, in: Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems 2006 (IROS06), Beijing (China), 9–15 October 2006, pp. 328–335.
[123] F. Sivrikaya, B. Yener, Time synchronization in sensor networks: a survey, IEEE Network 18 (4) (2004) 45–50.
[124] T. Small, Z. Haas, The shared wireless infostation model – a new ad hoc networking paradigm (or where there is a whale, there is a way), in: Proc. ACM MobiHoc 2003, 2003.
[125] I. Solis, K. Obraczka, The impact of timing in data aggregation for sensor networks, in: Proc. IEEE International Conference on Communications (ICC 2004), June, 2004.
[126] A.A. Somasundara, A. Kansal, D.D. Jea, D. Estrin, M.B. Srivastava, Controllably mobile infrastructure for low energy embedded networks, IEEE Transactions on Mobile Computing 5 (8) (2006) 958–973.
[127] A.A. Somasundara, A. Ramamoorthy, and M.B. Srivastava, Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines, in: Proc. 25th IEEE International Real-Time Systems Symposium (RTSS'04), 2004, pp. 296–305.
[128] D.R. Stinson, Combinatorial Designs: Construction and Analysis, Springer, 2003.
[129] C. Tang, C.S. Raghavendra, Compression Techniques for Wireless Sensor Networks, Book Wireless Sensor Networks, Kluwer Academic Publishers., 2004. pp. 207-231 (Chapter 10).
[130] Y. Tseng, C. Hsu, T. Hsieh, Power saving protocols for IEEE 802.11 ad hoc networks, in: Proc. IEEE Infocom 2002, New York, USA, June 2002.
[131] Y.-C. Tseng, Y.C. Wang, K.-Y. Cheng, Y.-Y. Hsieh, iMouse: an integrated mobile surveillance and wireless sensor system, IEEE Computer 40 (6) (2007) 60–66.
[132] D. Tulone, S. Madden, PAQ: time series forecasting for approximate query answering in sensor networks, in: Proc. 3rd European Conference on Wireless Sensor Networks (EWSN06), February 21–37, 2006.
[133] D. Tulone, S. Madden, An energy-efficient querying framework in sensor networks for detecting node similarities, in: Proc. 9th International ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM06), October 2006, pp. 291–300.
[134] L. van Hoesel, P. Havinga, A lightweight medium access protocol (LMAC) for wireless sensor networks, in: Proc. 1st International Workshop on Networked Sensing Systems (INSS 2004), Tokyo, Japan, June 2004.
[135] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, P. Corke, Data collection, storage, and retrieval with an underwater sensor network, Proc. 3rd ACM Conference on Embedded Networked Sensor Systems (SenSys 2005), San Diego, November 2–4, 2005.
[136] P. Venkitasubramaniam, S. Adireddy, L. Tong, Sensor networks with mobile access: optimal random access and coding, IEEE Journal on Selected Areas in Communications 22 (6) (2004) 1058–1068.
[137] M.C. Vuran, O.B. Akan, I.F. Akyildiz, Spatio-temporal correlation: theory and applications for wireless sensor networks, Computer Networks Journal 45 (3) (2004) 245–261.
[138] M.C. Vuran, I.F. Akyildiz, Spatial correlation-based collaborative medium access control in wireless sensor networks, IEEE/ACM Transactions on Networking 14 (2) (2006) 316–329.
[139] Z.M. Wang, S. Basagni, E. Melachrinoudis, C. Petrioli, Exploiting sink mobility for maximizing sensor networks lifetime, in: Proc. 38th Annual Hawaii International Conference on System Sciences (HICSS'05), Hawaii, January 03–06, 2005.
[140] A. Warrier, S.Park J. Mina, I. Rheea, How much energy saving does topology control offer for wireless sensor networks? – a practical study, Elsevier/ACM Computer Communications 30 (14-15) (2007) 2867–2879.
[141] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, M. Welsh, Deploying a wireless sensor network on an active volcano, in: IEEE Internet Computing, Special Issue on Data-Driven Applications in Sensor Networks, March/April 2006.
[142] R. Willett, A. Martin, R. Nowak, Backcasting: adaptive sampling for sensor networks, in: Proc. Third International Symposium on Information Processing in Sensor Networks (IPSN 2004), 26–27 April, 2004, pp. 124–133.
[143] M. Wu, C.W. Chen, Multiple Bit Stream Image Transmission over Wireless Sensor Networks, Book Sensor Network Operations, IEEE & Wiley Interscience, 2006. pp. 677-687 (Chapter 13).
[144] Z. Xiong; A.D. Liveris, S. Cheng, Distributed source coding for sensor networks, IEEE Signal Processing Magazine 21 (5) (2004) 80–94.
[145] Y. Xu, J. Heidemann, D. Estrin, Geography-informed energy conservation for ad hoc, in: Proc. ACM MobiCom, 2001, Rome, 2001, pp. 70–84.
[146] X. Yang, N. Vaidya, A wakeup scheme for sensor networks: achieving balance between energy saving and end-to-end delay, in: Proc. IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2004), 2004, pp. 19–26.
[147] W. Ye, J. Heidemann, Medium access control in wireless sensor networks, in: C. Raghavendra, K. Sivalingam, T. Znati (Eds.), Wireless Sensor Networks, Kluwer Academic Publishers., 2004 (Chapter 4).
[148] W. Ye, J. Heidemann, D. Estrin, Medium access control with coordinated adaptive sleeping for wireless sensor networks, IEEE/ACM Transactions on Networking 12 (3) (2004) 493–506.
[149] W. Zhao, M. Ammar, E. Zegura, A message ferrying approach for data delivery in sparse mobile ad hoc networks, in: Proc. ACM MobiHoc 2004, Tokyo, Japan, May 2004.
[150] R. Zheng, J. Hou, L. Sha, Asynchronous wakeup for ad hoc networks, in: Proc. ACM MobiHoc 2003, Annapolis, USA, June 1–3, 2003, pp. 35–45.
[151] T. Zheng, S. Radhakrishnan, V. Sarangan, Pmac: an adaptive energy-efficient MAC protocol for wireless sensor networks, in: Proc. 19th

IEEE International Parallel and Distributed Processing Symposium (IPDPS 2005), Denver, CO, April 2005.
[152] J. Zhou, D. De Roure, FloodNet: coupling adaptive sampling with energy aware routing in a flood warning system, Journal of Computer Science and Technology 22 (1) (2007) 121–130.
[153] M. Zorzi, R.R. Rao, Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance, IEEE Transactions Mobile Computing 2 (4) (2003) 337–348.
[154] M. Zorzi, R.R. Rao, Geographic Random Forwarding (GeRaF) for Ad Hoc and Sensor Networks: Energy and Latency Performance, IEEE Transactions Mobile Computing 2 (4) (2003) 349–365.

Giuseppe Anastasi is an associate professor of Computer Engineering at the Department of Information Engineering of the University of Pisa, Italy. He received the M.S. degree in Electrical Engineering, and the Ph.D. degree in Computer Engineering, both from the University of Pisa, in 1990 and 1995, respectively. His current research interests include mobile and pervasive computing, ad hoc and sensor networks. He is a co-editor of the book Advanced Lectures in Networking (LNCS 2497, Springer, 2002), and has published more than 70 papers in the area of computer networking and pervasive computing. He is an area editor of the Elsevier journal of Pervasive and Mobile Computing (PMC), and of the ASP Journal of Ubiquitous Computing and Intelligence (JUCI). He has served as Program Co-chair of IEEE WoWMoM 2008, Vice Program Chair of IEEE MASS 2007, General Co-chair of IEEE WoWMoM 2005, Workshops Chair of IEEE PerCom 2006, IEEE WoWMoM 2006, and IEEE ICCCN 2007. He is a member of the IEEE Computer Society.

Marco Conti is a research director at IIT, an institute of the Italian National Research Council (CNR). He co-authored the book "Metropolitan Area Networks" (1997) and is co-editor of the books "Mobile Ad Hoc Networking" (2004) and "Mobile Ad Hoc Networks: From Theory to Reality" (2007). He published in journals and conference proceedings more than 200 research papers related to design, modeling, and performance evaluation of computer-network architectures and protocols. He served as general chair of *IEEE MASS 2007* and *ACM REALMAN 2006*, and as general co-chair of IEEE *WoWMoM 2006* and of *ACM MobiOpp 2007*. He has been TPC chair of *IEEE PerCom 2006*, and of the *IFIP-TC6* Conferences "Networking2002" and "PWC2003". He served as TPC co-chair of *ACM WoWMoM 2002*, *WiOpt '04*, *IEEE WoWMoM 2005*, and *ACM MobiHoc 2006*. He is Associate Editor-in-Chief of Pervasive and Mobile Computing Journal, and he is on the editorial board of: IEEE Transactions on Mobile Computing, Ad Hoc Networks journal and Wireless Ad Hoc and Sensor Networks: An International Journal.

Mario Di Francesco is a Ph.D. student at the Department of Information Engineering of the University of Pisa, Italy. He received the B.S. and M.S. degrees in Computer Engineering, both from the University of Pisa, in 2002 and 2005, respectively. His current research interests include pervasive computing and wireless sensor networks. He has served as publication co-chair of *IEEE WoWMoM 2006* and as a reviewer for many international conferences and journals.

Andrea Passarella is a Researcher at the IIT Institute of the National Research Council (CNR), Italy. Before joining IIT, he was a Research Associate at the Computer Laboratory of the University of Cambridge, UK. He received the Ph.D. and M.S. Degrees in Computer Engineering, both from the University of Pisa, Italy, in 2005 and 2001, respectively. His current research is mostly on opportunistic and delay-tolerant networking. More in general, he works on mobile ad hoc networks, specifically on p2p systems, multicasting, transport protocols, and energy-efficient protocols. He was TPC Co-Chair of *ACM MobiOpp 2007*, Vice-Chair of *IEEE REALMAN 2005*, *ACM REALMAN 2006*, and *IEEE MDC 2006*. He was Demo Co-Chair of *IEEE MASS 2007*, and is serving as Demo Chair of*IEEE PerCom 2009*. He served and is currently serving in the TPC of several international conferences, including IEEE PerCom, IEEE WoWMoM, IEEE MASS, and workshops. He is an Associate Technical Editor for IEEE Communications Magazine, and in the Editorial Board of the Inderscience Int. J. Autonomous and Adaptive Communications Systems.