Energy-efficient routing in wireless sensor networks using power-source type identification

Konstantin Mikhaylov* and Jouni Tervonen

RFMedia Laboratory,
Oulu Southern Institute,
University of Oulu,
Vierimaantie 5, Ylivieska 84100, Finland
E-mail: konstantin.mikhaylov@oulu.fi
E-mail: jouni.tervonen@oulu.fi
*Corresponding author

Abstract: The real-life wireless sensor networks (WSNs) nowadays often include nodes that are powered by various power sources: mains; primary or secondary batteries; or energy harvesting systems, which generate power from the environment. Although information about the parameters of the available power sources for the WSN node is crucial for optimising the operation of the whole network, the contemporary WSN nodes can only estimate the level of their supply voltage at the current time. Therefore, in this paper we are suggesting a simple mechanism that identifies the type of WSN node's power source based on the measurements of the supply voltage. Based on the suggested power-source type identification (PSTID) mechanism, we introduce and propose the special routing protocol that is intended for WSNs containing nodes that have different power supply sources. The suggested routing protocol allows a significant increase in the lifetime of the whole network compared to the scenarios when no PSTID data is available. The proposed routing protocol is more universal then the existing routing protocols that take into account only the value of the supply voltage.

Keywords: energy-efficiency; power source; identification; wireless sensor networks; WSNs; heterogeneous network; routing; embedded systems.

Reference to this paper should be made as follows: Mikhaylov, K. and Tervonen, J. (2012) 'Energy-efficient routing in wireless sensor networks using power-source type identification', *Int. J. Space-Based and Situated Computing*, Vol. 2, No. 4, pp.253–266.

Biographical notes: Konstantin Mikhaylov received his BSc (2006) and MSc (2008) in Electrical Engineering, with the focus on wireless systems, from St. Petersburg State Polytechnical University, St. Petersburg, Russian Federation. Since 2008, he has been working as a Researcher for Wireless and Embedded Systems in the RFMedia Laboratory of Oulu Southern Institute, University of Oulu, Ylivieska, Finland. His main research interests include embedded systems, non-specific short range wireless communication technologies and wireless sensor networks, especially the problems of energy efficiency, available resources estimation and hardware-resources aware operation adaptation for those systems.

Jouni Tervonen received his Dipl.Eng. (MSc) and DSc (Tech.) in Electrical Engineering from the Helsinki University of Technology, Espoo, Finland in 1992 and 1997, respectively. Since 2004, he has been working as the Research Manager with Oulu Southern Institute, the University of Oulu. He is in charge of the Wireless Data Transmission Group of Oulu Southern Institute at the RFMedia Laboratory in Ylivieska, Finland. Currently, his main research interests are wireless sensor networks, embedded systems and applications of emerging 4G telecommunication standards.

1 Introduction

During recent years, multiple novel electronic devices that utilise various low-power microcontrollers have been introduced. The appearance of these has boosted the development and utilisation of various environmental energy harvesting methods for the power supply of contemporary electronic devices. This has added a third alternative in the portfolio of possible power supply options for electronic devices, which previously included only two options: mains and primary or secondary batteries. In some applications, the devices that have a similar structure and functionality but are powered by various power supply sources can coexist within the same system. This anticipates advantages from the methods for identifying *the type of the power supply source used by a device* and for *optimising the operation of a device based on its power supply source*. One example of such applications is wireless sensor networks (WSNs).

The development and standardisation of communication protocols for WSN over recent years, i.e., the introduction of IEEE 802.15.4 (IEEE Std. 802.15.4, 2006), 6LowPAN (IETF Std. RFC4944, 2007) and Bluetooth low-energy standards (Bluetooth SIG, 2010), provide the means for unifying communications within WSNs and between WSNs and external systems. This has allowed the introduction of huge WSNs that will include numerous nodes with various architectures, and which will sense different parameters for various applications but utilise common radio channels and the same protocol for communication. An issue that is evidently important for handling these complex systems is the availability of methods for estimating resources on each node. The most important of these resources are:

- available services (i.e., available sensors and other peripherals)
- available memory and computational power
- available energy.

In recent years, several methods have been introduced, which enable identification and which automatically use the peripheral components available on a WSN node [e.g., the IEEE 1451 set of standards for smart transducer interfaces (IEEE Std. 21450, 2010) and the Sensor Web Enablement suite of specifications from the Open Geospatial Consortium (OGC) (OGC 07–165, 2007)]. Most of the operating systems that are used on WSN nodes nowadays can also provide some means to track the available memory or processing power (Farooq and Kunz, 2010). Nonetheless, to the best of our knowledge, currently there are no standard mechanisms that can be used by existing WSNs to identify the source of power of a WSN node on the run. Instead, for the existing WSN nodes the information on their source of power is either hardcoded in the node's memory during manufacturing or is provided by special proprietary circuitry solutions. The drawbacks to this are the absence of a power source changing possibility for the first scenario and the additional node's cost, size, weight and manufacturing complicity and low portability for the second.

Therefore, in this paper we suggest a novel power-source type identification (PSTID) mechanism to be used by the WSN nodes. This utilises the commonly available supply voltage measurement mechanism to define whether the node is powered by mains, batteries or energy harvesting systems. In the second part of the paper, we introduce a novel routing protocol that is intended for heterogeneous WSNs consisting of nodes with different types of power sources. The suggested routing protocol allows us to significantly increase the probability for mains-powered nodes being selected as routers in WSNs, thus increasing the lifetime of the network. The presented evaluation results reveal that the suggested PSTID-aware routing significantly outperforms, in terms of the network lifetime, the existing routing protocols that rely only on the supply voltage as the measure of the available energy.

The remainder of the article is organised as follows: Section 2 describes the current state-of-the-art process for power source identification methods. Section 3 focuses on the details of the proposed PSTID mechanism. Section 4 discusses the benefits of having information on the power source for routing and introduces the novel routing protocol that utilises the PSTID data. Section 5 concludes the paper and summarises the results.

2 Related work

The possibility of getting information about the power supply source that is used by an electronic device provides various advantages. This information is especially important for various battery-powered devices, for which it reveals the available battery energy and thus the remaining operation time of the device (Dias, 1994; Lahiri et al., 2002; Stolitzka and Dawson, 1994). Besides, in the case of a secondary battery, the possibility of identifying the battery can be used to select the most efficient battery charging mode or to estimate the time required to completely charge the battery (Dias, 1994). For devices that are powered by mains or DC power sources, information about power source characteristics reveals whether the power source is capable of supplying adequate power for an additional load. As has been shown in multiple works (e.g., Jongerden et al., 2010; Lahiri et al., 2002; Mikhaylov and Tervonen, 2010; Mikhaylov et al., 2012a), in order to truly maximise the energy efficiency of a device it is essential to consider as a whole the source of the energy and the system that consumes it.

Up until now, power-source identification was mainly addressing the recognition of batteries in portable electronic devices. Already, in the early 1990s, the concept of the smart battery had been developed – a battery with methods for its identification, charge control, fuel gauging and battery protection (Stolitzka and Dawson, 1994). Besides the completely smart batteries that would support all of these features, different methods for supporting particular smart features for already existing batteries have been suggested. E.g., Dias (1994) proposed to embed in each battery pack a limited amount of memory – a label – that would contain the parameters of this particular pack (chemistry, capacity, identification data). This information could be accessed by an electronic device through the standardised 1-wire interface.

In the late 1990s, the manufacturers of batteries and electronic devices developed the smart battery specification (SBS) (SBS Implementers Forum, 1998). The SBS defined the SMBus interface, data format and data flow between the smart battery, SMBus host, smart battery charger and other devices. The SBS mechanisms are independent of the battery chemistry and are capable of providing portable electronic devices with valuable information for power management and charge control. Nowadays, SBS is widely used in batteries installed in various portable electronics such as cell phones, digital cameras and laptop computers (Atmel Corporation, 2006).

During recent years, the IntellBatt concept has been suggested (Mandal et al., 2010) – the novel battery that features an intelligent battery cell array with a manager that actively schedules cells to optimise capacity and charge delivery as well as managing battery safety to ensure robustness and reliability. As reported by Mandal et al. (2010), the integration of the manager and the battery enabled an increase in the lifetime of the test-bed system by 22%.

Although the above-mentioned smart batteries can provide multiple advantages for WSN systems and their identification methods can be further extended to support other types of power sources (i.e., mains or various recently developed energy harvesting methods), they have some disadvantages. The most serious disadvantage which affects applications with restricted resources, including many WSNs (Kuorilehto et al., 2007), is the cost of a smart battery and its interface to the device (Stolitzka and Dawson, 1994). Indeed, as reported by Ahmed et al. (2006), nowadays, the production cost of a single WSN node can be less than one US dollar which is comparable to the price for basic batteries. Meanwhile, the use of smart batteries in WSN nodes would significantly increase the overall price of the application.

To partially overcome this problem, in the next section we introduce a new method that provides some valuable information about the power source of a WSN node. The specifics of the suggested method are that it *does not require any additional hardware components* for its implementation, while providing the capability to estimate the type of power source used by a WSN node.

3 Proposed method for PSTID

3.1 *Existing power supply options for WSN nodes and their parameters*

WSN nodes can nowadays use one of three (or a combination of) the following power supply options:

- electrical grid (through AC/DC converter) or external stabilised power source
- batteries or accumulators
- energy from environmental harvesting systems with capacitive storage.

WSN nodes with a power supply from the electrical grid usually do not have any strict limitations on the energy they consume. This is the reason why this power supply option is used for devices with a high duty cycle, e.g., routers and access points (AP) in WSN [e.g., Wisepro (2011)]. For the majority of WSN node platforms, because they require a supply voltage in the order of 2–5 V DC, an external AC/DC power adapter is required for converting AC voltage to DC. This causes additional loss of energy and makes the use of low-power modes for such nodes ineffective (Mikhaylov et al., 2012a). Another option that is often used for supplying power to WSN APs is the available supply line of USB or other data exchange interfaces. Often (e.g., Atmel Corporation, 2008; Texas Instruments, 2009a) 5 V of USB supply voltage is converted on the node to a lower voltage of 3.0–3.6 V.

The second power supply option, which is often used for WSN end devices (ED), is either rechargeable or not rechargeable batteries. These power sources have a limited energy capacitance that also depends on the parameters of the used battery, environmental parameters and the node's current consumption. The influence of these parameters on the node's available energy and on methods for lifetime maximisation for such systems is a complicated problem that is beyond the scope of this article. Some of the corresponding data can be found e.g., in Chulsung et al. (2005), Mikhaylov et al. (2012a), Penella et al. (2009) and Pomerantz (1990).

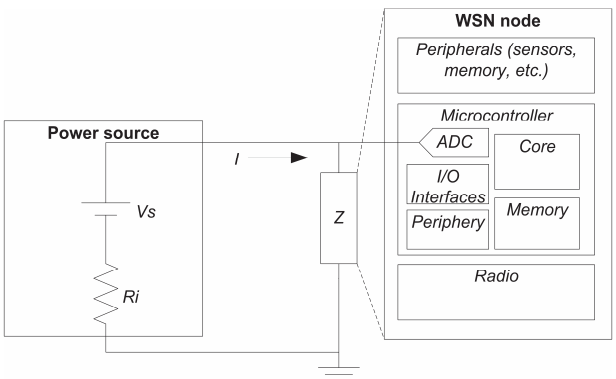
The third possible option for WSN node operation support is the usage of energy harvested from the environment. Currently, use of the following energy sources has been demonstrated (Mikhaylov and Tervonen, 2011): light; temperature difference; vibration and movement; water, air or gas flow; electrical or magnetic fields and chemical reactions [e.g., Thomson (2008)]. For all methods, energy is initially harvested from the environment in a storage capacitor that usually has very limited capacity, e.g., see AdaptivEnergy RLP (2010), Micropelt GmbH (2010) and Texas Instruments (2010). It is later consumed by the WSN node. Usually, the amount of energy that can be collected from the environment in any period of time is rather small. Therefore, energy harvesting systems have to accumulate energy over a relatively long period of time, before the WSN node would get enough to make measurements and send packets.

Nowadays, there are two solutions that allow WSN nodes to obtain information about their power source. The first option which can be used if the WSN node's power source is not subject to future change, is to hardcode information about the power source as part of the embedded software of the WSN node. The obvious drawbacks to this option are the impossibility to change the power source in future plus the additional complexity in manufacturing and service processes. For example, during the manufacturing of each node, it is necessary to hardcode the power source data; during wireless reprogramming, as the embedded software differs, it can become necessary to separately reprogramme the nodes with different power source types.

The second solution, which overcomes these problems, is to identify the type of WSN node power source on the run. One of the obvious solutions for this approach is to use an external signal (e.g., a special switch/jumper or special circuits/chips) to specify the power source type. The drawbacks to it are the increased price, size and weight of the WSN node and more complex development and manufacturing procedures for such nodes, due to the utilisation of additional components.

The other option is to use the already existing WSN node mechanisms for getting information on their power source. Unfortunately, the majority of WSN nodes have only one such mechanism – for measuring the supply voltage. As this mechanism cannot reliably identify the type of power source on its own, it is usually used for estimating the available energy of battery-powered nodes (the lower the battery voltage is, the less energy is available). Or, it can be used for checking the possibility of launching operations which require a certain level of supply voltage [e.g., writing to electrically erasable programmable read-only memory (EEPROM) or running the microcontroller at higher clock frequency]. The supply voltage measurement in WSN nodes is usually implemented by connecting the supply voltage line to one of the lines of the WSN node microcontroller's built-in Analogue-to-digital converter (ADC) (see Figure 1).

Figure 1 Equivalent circuit for power source connection to a WSN node



As it has already been noted, numerous WSNs have very limited resources due to the 'low cost and low power' nature of the systems (Kuorilehto et al., 2007). Therefore, identification of the power source for WSNs must be as simple as possible and should be done, whenever possible, with minimal use of additional hardware components.

3.2 Suggested method for identifying power supply source type for WSN nodes

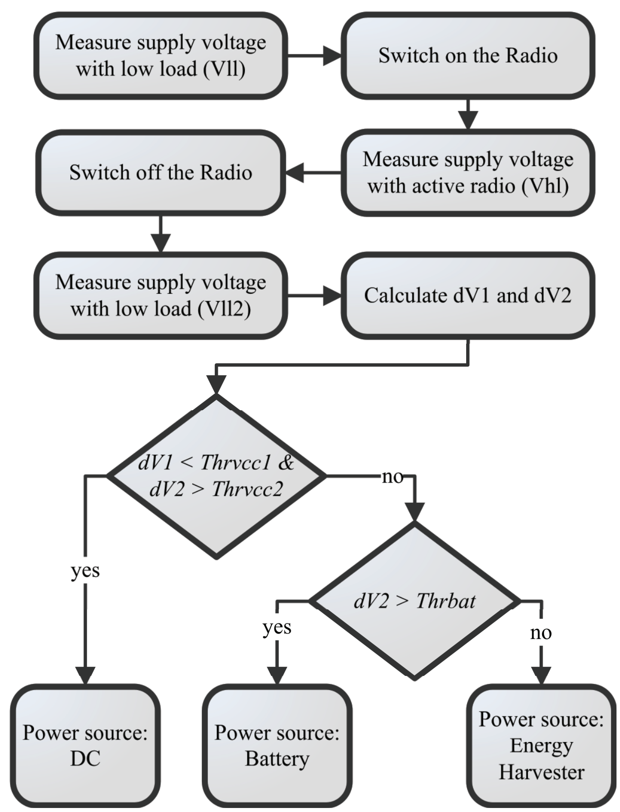
This method was first introduced by Mikhaylov and Tervonen (2011). As revealed in Figure 1, the source of power of WSN nodes can be represented by the source of voltage V_s in series with the internal resistive impedance of power source R_i. The generalised impedance Z represents the whole WSN node and varies for different working modes of the WSN node. Therefore, the voltage that is measured by the ADC (V_{AD}) of the WSN node depends on V_s, R_i and the current I (1), consumed by the WSN node.

$$V_{AD} = V_S - \frac{I}{R_i} \tag{1}$$

The resistance R_i depends on the WSN node's source of power and provides valuable information for identifying the type of power source. For batteries, R_i is usually below one Ohm; e.g., for AA batteries it is in the order of 0.1–0.5 Ohm (Pomerantz, 1990) and when the node is powered by mains, R_i is usually significantly higher. The current I, that is consumed by the WSN node depends on its working mode, and for contemporary platforms, ranges from several μA during sleep mode and up to 15–30 mA when the radio transmits or receives (Mikhaylov and Tervonen, 2010; Texas Instruments, 2009a, 2009b). Therefore, as it is easy to see using (1), such levels of current consumption would cause a voltage drop over the R_i of around 0.52 mV if I is 2 mA and 5.2 mV if I is 20 mA, with the corresponding change of $V_{AD,}$ assuming that the node is powered by two AA batteries; thus, R_i is 0.28 Ohm and V_s is 3 V (Pomerantz, 1990). Although this change is rather small, it can be easily detected by the ADC in contemporary microcontrollers.

To implement identification of the power source type, we suggest a very simple algorithm (see Figure 2) that is based on three measurements of the supply voltage made by the ADC of the WSN node (see Figure 1). The first measurement (V_{ll}) is done when the WSN node is running in low current consumption mode (all systems except the ADC are switched off, I < 1 mA). The second measurement (V_{hl}) is made when the node is in high current consuming mode (e.g., radio is in transmit or receive mode, I = 10–20 mA). The third measurement (V_{ll2}) is done after the second one, once the node switches back to low current consuming mode (I < 1 mA). Based on the results of those measurements are calculated dV_1 and dV_2 using (2) and (3) respectively. The first of the calculated parameters (dV_1) is significantly affected by the internal resistance of the power source, while the second parameter (dV_2) also allows account to be taken of the changes in supply source voltage over the period with high current consumption that is typical when the node is powered by energy harvesting.

Figure 2 Suggested algorithm for WSN node PSTID

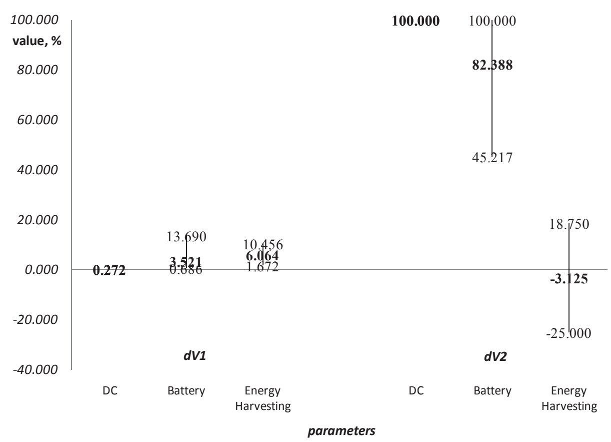


To enable a decision on the power source, the calculated values of dV_1 and dV_2 should be compared to the threshold values, Thr_{vcc1}, Thr_{vcc2} and Thr_{bat}, that are defined experimentally for each platform. The measured values for V_{ll}, V_{hl} and V_{ll2} and the corresponding values for dV_1 and dV_2 for two real-life WSN node platforms (Texas Instruments, 2009a, 2009b) powered by various power sources are presented in Table 1 and Figures 3(a) and 3(b).

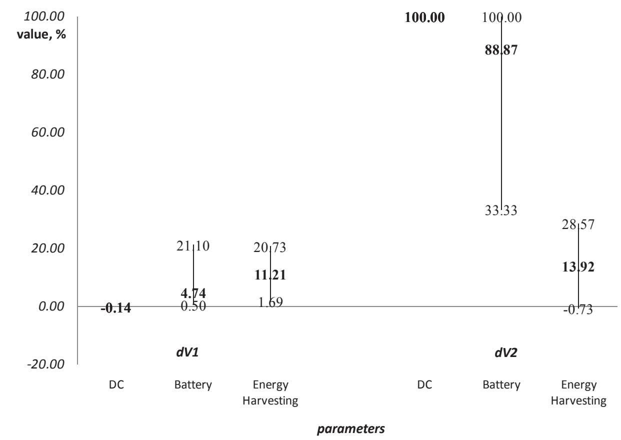
$$dV_1 = \frac{V_{ll} - dV_{hl}}{dV_{ll}} \tag{2}$$

$$dV_2 = \frac{V_{ll2} - dV_{hl}}{dV_{ll} - dV_{hl}} \tag{3}$$

Figure 3 Values of dV_1 and dV_2 parameters for real-life WSN platforms with various power supply sources, (a) dispersion and average values (bold) of dV_1 and dV_2 for eZ430-RF2500 node with different power supply options (b) dispersion and average values (bold) of dV_1 and dV_2 for CC2510MINI-DK node with different power supply options



(a)



(b)

Table 1 V_{ll}, V_{hl} and V_{ll2} values for real-life WSN node platforms with various power supply sources

Power source	*eZ430-RF2500*			*CC2500-Mini*		
	Vll, mV	*Vhl, mV*	*Vll2, mV*	*Vll, mV*	*Vhl, mV*	*Vll2, mV*
DC power adapter from grid	3.579	3.584	3.579	NA	NA	NA
DC power adapter from laptop	3.579	3.584	3.579	3.367	3.358	3.367
2xAAA batteries (alkaline, new)	2.962	2.933	2.962	3.056	3.025	3.054
2xAAA batteries (alkaline, used)	2.499	2.420	2.493	2.582	2.494	2.574
2xAAA accumulators (Ni-MH, new)	2.555	2.525	2.555	2.646	2.618	2.646
2xAA batteries (alkaline, new)	3.091	3.066	3.086	3.193	3.171	3.190
2xAA accumulators (Ni-MH, new)	2.587	2.528	2.587	2.673	2.622	2.673
2xAA accumulators (Ni-MH, used)	1.758	1.749	1.752	1.870	1.857	1.864
CR2032 battery (lithium, new)	2.971	2.344	2.921	3.076	2.655	2.845
2xAG4 batteries (alkaline, new)	2.977	2.725	2.953	3.073	2.856	3.016
2xAG10 batteries (alkaline, new)	2.909	2.742	2.909	3.047	2.964	3.034
2xAG13 batteries (alkaline new)	3.027	2.930	3.027	3.137	3.041	3.131
Solar energy harvesting (Texas Instruments, 2010)	4.033	3.965	3.984	3.505	3.446	3.457
Vibration energy harvesting (AdaptivEnergy RLP, 2010)	3.228	2.559	2.554	3.292	2.948	2.862

As can be seen from the presented data, for both tested platforms the threshold values can be set to: Thr_{vcc1} = 0.5%, Thr_{vcc2} = 99.5% and Thr_{bat} = 30%. Further details concerning the implementation and evaluation of the suggested PSTID mechanism can be found in Mikhaylov and Tervonen (2011).

The suggested PSTID mechanism is the first, to the best of our knowledge, which allows WSN nodes to identify the type of the available power source. Besides, the suggested PSTID mechanism does not require any additional hardware components and consumes very moderate WSN node resources for its implementation (consider Mikhaylov and Tervonen, 2011). In the next section we will discuss how the information provided by this mechanism can be used to improve the energy efficiency of WSNs.

4 The use of PSTID data for routing in WSNs

Real-life WSNs nowadays often include nodes that have different power sources (Mikhaylov and Tervonen, 2011; Singh et al., 2010). As discussed in Section 3, with WSNs it is sensible to use mains-powered nodes as routers or the most heavily used sensor nodes and battery-powered nodes as the sensor nodes with a low duty cycle (consider e.g., Mikhaylov et al., 2012b; Wisepro, 2011).

In the past few years, the problem of energy efficient routing in WSNs was a field of intensive research for many researchers and multiple novel routing protocols have been proposed. Nowadays, among the most popular routing protocols for WSNs are (Padmanabhan and Kamalakkannan, 2011; Singh et al., 2010; Villalba et al., 2009): CHR (Chu et al., 2002); directed diffusion (Intanagonwiwat et al., 2000); EAD (Boukerche et al., 2003); GAF (Xu et al., 2001); LEACH and its numerous extensions (Heinzelman et al., 2000, 2002); MECN (Rodoplu and Meng, 1999); PEGASIS (Lindsey and Raghavendra, 2002); SPIN (Kulik et al., 2002); TEEN and its modifications (Manjeshwar and Agrawal, 2001). In this paper, we will not discuss all of these protocols in detail – we will just note that most of these protocols have been developed with the assumption that *all the nodes in the WSN have a homogeneous structure and the same resources* (usually assuming that nodes are powered by batteries with known remaining energy at any moment in time, which is far removed from real life). Therefore, many of those protocols require substantial modifications to support WSNs where nodes can have different power supply sources.

To evaluate the benefits one can obtain from having data about the WSN node's power sources, we have chosen a scenario which we have faced many times in real-life WSN applications and which is typical for habitat monitoring (Intanagonwiwat et al., 2000; Madden et al., 2002), industrial monitoring (Mikhaylov et al., 2012b); contamination transport monitoring (Estrin et al., 2002), forest fire pre-warning (Ye et al., 2002) and some other environmental monitoring applications (Boukerche et al., 2003). During the evaluation, we assume that WSNs consist of:

- *one* sink node
- *M sensor nodes* powered by mains
- *N sensor nodes* powered by batteries.

For the sake of generality we assume that all these nodes are randomly deployed within the target area of (x, y) metres, which exceeds the maximum communication range of a single WSN node. Also we assume that the nodes already have unique identifiers that can be used to address them. The main task of the routing protocol is to organise the network that will ensure the reliable delivery of messages generated by the sensor nodes (messages are generated at random) *to the sink* taking into account the available information about the node power sources. Therefore, the target WSN should contain three types of nodes:

- one *sink* node
- *leaf nodes* – the nodes that stay in low-power mode and switch on the radio only when they have some data to send to the sink
- *repeater* nodes ('non-leaf nodes' in Boukerche et al., 2003) that both take measurements and relay traffic from the other nodes to the sink or other *repeaters*.

In the next subsection, we will discuss in detail the suggested route building protocol and the results of its evaluation for different scenarios.

4.1 Suggested PSTID data aware routing protocol

For building the route within the WSN, we developed a simple routing protocol (see Figure 4) that is based on the energy-aware data-centric (EAD) routing protocol by Boukerche et al. (2003), but which has some modifications that were introduced to support nodes with different power source options within the same WSN.

We assume that after network initialisation, each sensor has its radio transceiver on and is sensing the common channel. The *sink* node initiates route building by broadcasting the *route_build* packet (see Figure 5). Besides the packet type identifier that distinguishes the *route_build* traffic from normal *data* traffic, the *route_build* packet includes (see Figure 4) unique identifiers (IDs) both for the transmitter node (transmitter ID) and its parent node (host ID). When a WSN node receives a *route_build* packet for the first time, it sets the packet source node as its parent node and backs off for T_{WAIT} before retransmitting the *route_build* packet further. The back-off period, T_{WAIT}, depends on the available information about the node's power source and is discussed in detail below. The host ID in the retransmitted *route_build* packet is used to acknowledge to the parent node that it has at least one child (thus making it a *repeater*) and to enable the other nodes to connect to the WSN. Once the *route_build* packet has been retransmitted, the node starts the waiting timer. The waiting timer counts up to T_{ACKmax} – the period of time that exceeds the maximum possible back-off delay. If, by the time that the timer expires, the node has not received the *route_build* packet retransmission where it has been marked as the other node's parent, it starts acting as a *leaf-node*. At this point, the *sensor node* is already considered to be part of the network and is able to start sending data to the *sink* node.

Figure 4 State diagram for the heuristic run of the proposed routing protocol by any node other than sink and *route_build* packet format (see online version for colours)

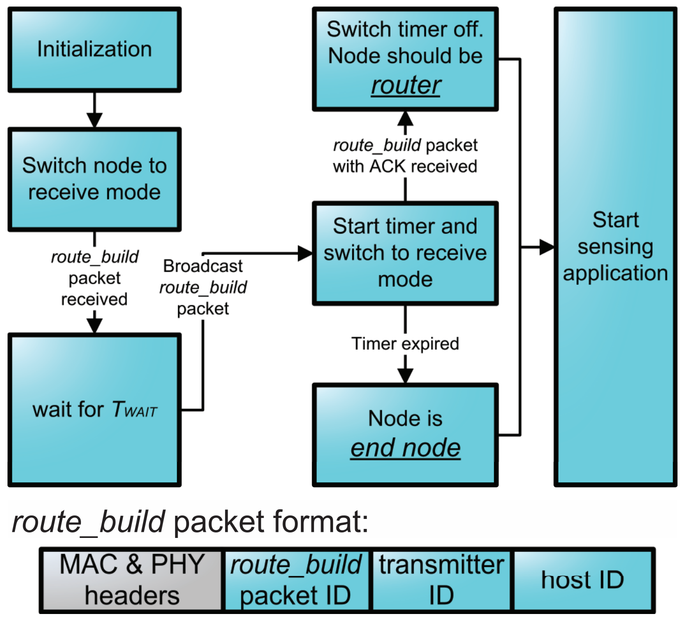
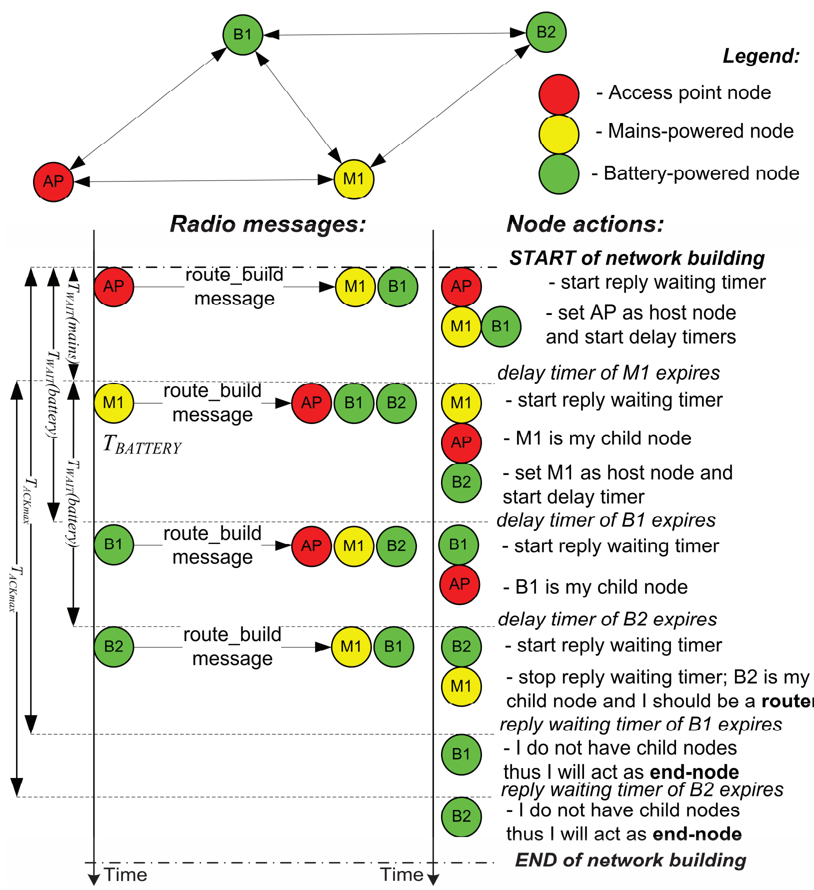


Figure 5 Illustration for network building by the suggested routing algorithm (see online version for colours)



The major differences of the suggested protocol compared with the EAD are that: the delay that is introduced before retransmitting the *route_build* packet on each node is dependent on the available power source data; and that the received retransmissions of the *route_build* packets are used as confirmation that a node will need to act as a router. Overall, we have evaluated the protocol for four different power source data availability scenarios:

- no information about the power source of a node is available (4)

$$T_{WAIT} = T_0 + random\left(0; T_{rand}\right); T_{rand} << T_0 \tag{4}$$

- only the result of supply voltage measurement (similar to the classical EAD) is available [see (5), where V_{cc} is the result of supply voltage measurement by ADC];

$$\begin{aligned} T_{WAIT} &= T_1 + T_{VCC}\left(V_{cc}\right) + random\left(0; T_{rand}\right); \\ &\quad T_{rand} << T_1 + T_{VCC}\left(V_{cc}\right) \end{aligned} \tag{5}$$

- the PSTID data for all nodes is available and a constant back-off is used (PSTID-CBO) [see (6) – depending on whether the node's power source is battery or mains either T_{MAINS} or $T_{BATTERY}$ is used, $T_{MAINS} < T_{BATTERY}$]

$$T_{WAIT} = \begin{cases} T_{MAINS} + random\left(0; T_{rand}\right); T_{rand} << T_{MAINS} \\ T_{BATTERY} + random\left(0; T_{rand}\right); T_{rand} << T_{BATTERY} \end{cases} \tag{6}$$

- the PSTID data for all nodes is available and the back-off includes constant and voltage-based components (PSTID-VBO) [see (7)].

$${}_{WAIT} = \begin{cases} \qquad T_{MAINS} + random\left(0; T_{rand}\right); \\ \qquad T_{rand} << T_{MAINS} \\ T_{BATTERY} + T_{VCC}\left(V_{cc}\right) + random\left(0; T_{rand}\right); \\ T_{rand} << T_{BATTERY} + T_{VCC}\left(V_{cc}\right) \end{cases} \tag{7}$$

The small random delay *random*(0; T_{rand}) has been introduced to decrease the probability of retransmitted packet collisions while using a carrier sense multiple access (CSMA) scheme for media access.

4.2 Evaluation of the PSTID data-aware routing protocol

To evaluate the suggested routing protocol, we carried out the two-stage simulation using the MiXiM (2012) extension for the OMNeT++ (2012) framework. During the simulations, we used the physical layer and the CSMA implementation from MiXiM with the required changes for enabling low-power sleep mode for the *leaf-nodes* and the simple battery model developed by Feeney and Willkomm (2010) for the battery-powered nodes [assuming also that the voltage supplied by the battery depends linearly on battery energy – see (8)].

$$V_{BAT} = 3.3\text{ V} - E*_{BAT} \cdot 1.5\text{ V} \tag{8}$$

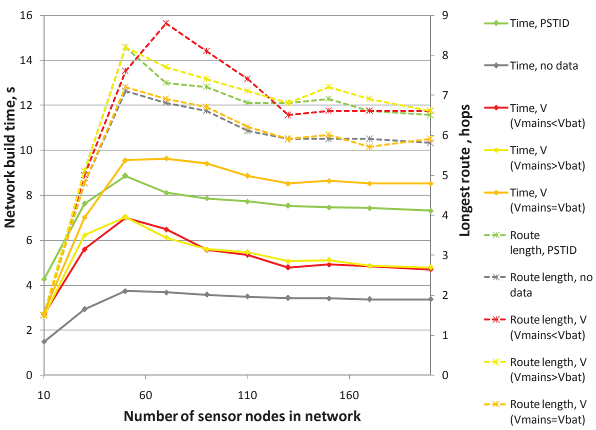
During the first stage, we evaluated the effect of the available power-source data on the network structure. The simulations were carried out for 10 to 200 *sensor nodes* (half of them were powered by batteries and the other half from mains) plus one *sink* node randomly placed within the test area of 500 m by 500 m. During the simulations for all the *sensor nodes* we used the hardware model of the Texas Instruments CC2430 system-on-chip, which has the same microcontroller core as the CC2510 that was used for the PSTID algorithm hardware evaluation in Section 2. The maximum communication distance for the nodes in the simulated network was around one hundred metres.

The simulations were carried out for 500 networks (ten simulations for each point on the chart using various node distributions within the test area) for five scenarios. The first two scenarios represented cases when no data about the power source was available [see (4)] and when the PSTID data was available [see (6) and (7)]. The three other scenarios represented the cases when the supply voltage level was used as the measure of the available node's energy [i.e., the classical implementation of EAD protocol, see (5)] and the supply voltage of mains-powered nodes was much higher (V_{BAT} = 2 V, Vcc = 3.3 V), much lower (V_{BAT} = 3.3 V, Vcc = 2 V) or equal (V_{BAT} = Vcc = 2.4 V) to the supply voltage of the battery-powered nodes. The back-off times used in the simulated network were: T_0 = 0.2 s; T_1 = 0.2 s; T_{MAINS} = 0.2 s; $T_{BATTERY}$ = 1 s and T_{rand} = 0.1 s. T_{VCC} was specified as: T_{VCC} (V_{CC}) = 3.3 V – V_{cc} thus giving the delay from 0 s for 3.3 V supply voltage (maximum supply voltage) to 1.5 s for 1.8 V supply voltage (minimum supply voltage). The maximum length for the *route_build* packet acknowledge waiting time was set to T_{ACKmax} = 2 s.

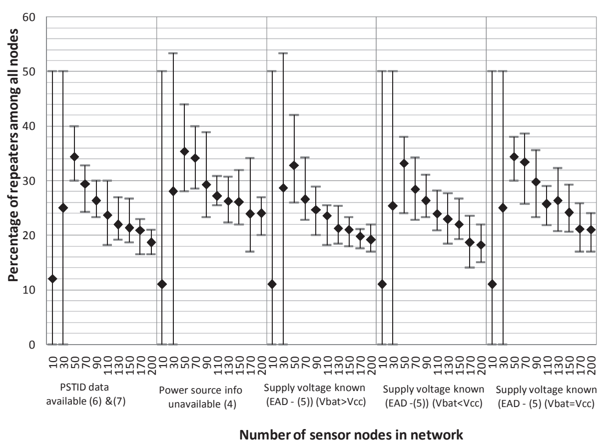
The results of the simulation, that reveal the network building time (estimated as the time from the start of network building up to the last *leaf-node* has stopped its wait timer) and the length of the longest route between the *leaf-node* and the *sink* in the resulting network, the percentage of the *repeaters* among all *sensor nodes* and the percentage of mains-powered *repeaters* among all *repeaters*, are presented in Figures 6(a) to 6(c) respectively.

As revealed in Figure 6(a), both the network building time and the maximum route length in the build network for all scenarios rapidly increased with the increase of the number of the nodes within the network and they reached their maximums when the network composed 50–70 nodes and then started to slowly decrease. A similar effect can be seen also in Figure 6(b) for the overall amount of routers in the network. This is the effect of node density in the WSN. So, the networks of less than 50 nodes commonly had unconnected nodes (e.g., the network with ten nodes had on average 73% of the nodes unconnected, the network with 30 nodes –37%, with 50 nodes –2%) and also the number of routes in those networks was somewhat limited and often the existing routes had to include many hops. Meanwhile, with the increase in node density, more routes between *sensor nodes* and the *sink* appeared and it became possible to select the shorter ones.

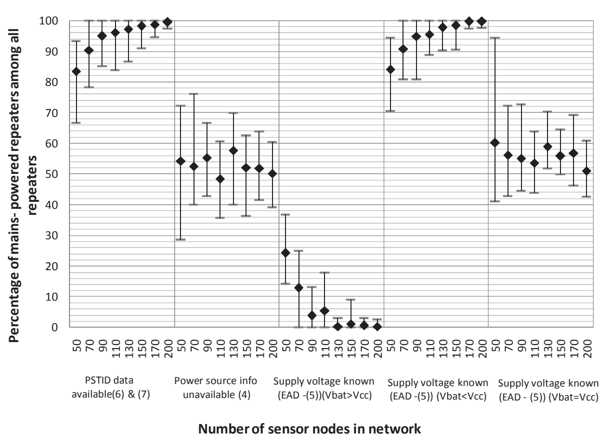
Figure 6 Results of network construction simulation for different power source data availability scenarios, (a) effect of sensor node number on average network build time and length of the longest route in WSN (b) effect of sensor node number on the amount of repeaters in the WSN (black diamond = average, grey segment = dispersion of values) (c) effect of sensor node number on the amount of mains powered repeaters in the WSN (black diamond = average, grey segment = dispersion of values) (see online version for colours)



(a)



(b)



(c)

As revealed in Figure 6(c), for the scenario when no power source data is available, the number of the battery and mains-powered *repeaters* was around the same. The availability of the precise information on the node's power sources data and the usage of suggested back-off mechanisms [see (6) and (7)] allowed a significant increase in the number of mains-powered *sensor nodes* that were selected as *repeaters*, thus increasing the lifetime of the whole network. Similar to EAD, the suggested protocol was tested for different supply voltage relations, but the results were the same. Meanwhile, as can be seen in Figure 6(c), the results of the classical EAD protocol, based on the supply voltage measurement, heavily depend on the relationship between supply voltage values for the battery and mains-powered nodes. As revealed in Figure 6(c), (5) is effective only when the battery voltage is much lower than the supply voltage of the mains-powered nodes. In the opposite case, the use of this method resulted in selecting mostly battery-powered *sensor nodes* as *repeaters*.

During the second stage of the evaluation, we tested the suggested protocols for long-term operation and compared these with the existing routing protocols having a similar purpose. For comparison, we used the gossip-based sleep protocol (GSP) by Hou and Tipper (2004); the maximum capacity path (MCP) protocol by Huang and Jan (2004); the EAD protocol by Boukerche et al. (2003); and the Energy-Aware Minimum Connected Dominating Set (EAMCDS) constructing algorithm by Raei et al. (2011). To adapt the protocols to the tested scenario, to equalise the test conditions and to enable the use of the energy-saving features of the protocols, the following changes have been made.

For the GSP, the probability of each sensor node remaining awake during each network rebuild was set to 0.7 and it was independent of the power source characteristics. After the network rebuild, the minimum hop routing protocol was used to define actual routes from the sensor nodes to the sink (the awake nodes were acting as *repeaters*, the sleeping nodes could only be *leaf-nodes*).

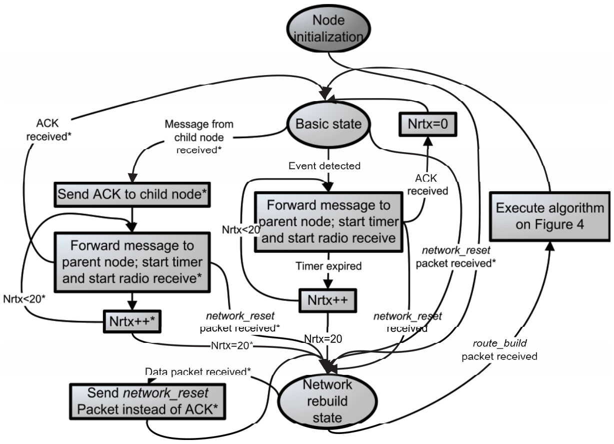
For the MCP protocol, during the network rebuild all the intermediate nodes for the paths constructed by the protocol were assigned to be *repeaters*. All remaining nodes acted as *leaf-nodes*.

For the EAMCDS, the maximum length of the first phase of protocol operation (maximum independent set definition) was set to ten seconds. After the second phase (MCDS definition) that continued for another ten seconds, nodes within MCDS ('blue' nodes according to the protocol description) were assigned as *repeaters* and the other nodes ('grey') became the *leaf-nodes*. Finally, the minimum hop routing protocol was used to define the actual routes between the sensor nodes and the sink.

For all the protocols discussed above, we have used the supply voltage level as the measure of the remaining energy. During the simulation, we assumed that for mains-powered nodes, V_{MAINS} is constant and for battery-powered nodes, the supply voltage level V_{BAT} decreases linearly from

3.3 V up to 1.8 V with the decrease of the relative available energy $0 \leq E^*_{BAT} \leq 1$ (8).

Figure 7 State diagram for the application extending the suggested routing mechanism that is executed by any node other than sink



Note: *Additional operations executed by *repeaters*.

The state diagram of the operation for the suggested algorithm during the second phase of the evaluation is presented in Figure 7. As it is possible to see, after the initialisation, a *sensor node* switches to the *network rebuild state* and waits for the reception of the *route_build* packet and uses the above-suggested (see Figure 4) routing protocol to define whether it should act as a *repeater* or as a *leaf-node*. Once finished, the node switches either to sleep mode (if it is a *leaf-node*) or to receive mode (if it is a *repeater*). When an event happens (i.e., a new event is detected or a new data packet is received), the node forwards a message to its parent node. To improve communication reliability, the acknowledgement (ACK) mechanism is used. Additionally, the developed application allows *sink node* to rebuild the network from scratch when required. The periodic network rebuilds can be used to add new *sensor nodes* to the network, to provide the support of node mobility or to further increase energy efficiency by rotating battery-powered routers (see below). During the second phase of the evaluation we have used the same node models and values of T_0, T_1, T_{MAINS}, $T_{BATTERY}$, T_{rand} and T_{VCC} as during the first phase. The maximum period for awaiting the ACK for the data packet was defined as $T_{DATA\_ACKmax} = 0.2$ s. The messages were generated by each *sensor node* at a random moment with a random delay of 1 s to 30 s between two messages. Every single hour the network was rebuilt, the *sink* node was issuing the *network_reset* packet, waiting one minute for the network reset and issuing the *route_build* packet to rebuild the network. The nominal capacity of the battery was specified as 1,000 mAh (corresponding to common AAA alkaline batteries). The simulation for each scenario was stopped when the first battery-powered node depleted its battery (this moment was considered as the end of the network lifetime).

The tests were done for two network layouts – the ‘sparse’ and the ‘dense’ networks, that are illustrated in Figure 8(a). The resulting networks’ layouts for different routing protocols for the cases when $V_{BAT} > V_{MAINS}$ and $V_{BAT} < V_{MAINS}$ are presented in Figures 8(b) to 8(f). Tables 2 and 3 present the results of the long-term simulation for the cases when $V_{MAINS} = 3$ V and $V_{MAINS} = 2$ V and V_{BAT} is defined by (8) for sparse and dense networks respectively. As can be seen from the presented data, the suggested protocols utilising PSTID data significantly outperform the other tested protocols in terms of the resulting network lifetime, especially for the dense networks scenario. As revealed in Table 2, the routing using PSTID-VBO for the sparse network increased its lifetime by two to three times compared with the existing protocols. As can be seen from Table 3, for the dense network with multiple mains-powered nodes, the advantage of the suggested PSTID-based routing protocols over existing protocols is six to ten times. The PSTID-CBO worked quite well for the dense network scenario, but for the sparse network, where it was necessary to use battery-powered nodes as routers, the performance of PSTID-CBO was worse than that of EAD or PSTID-VBO. The major reason for this was the low level of rotation for the battery-powered nodes which were acting as *repeaters*. The routes that were constructed according to the PSTID-VBO protocol have more hops than the routes built by the other protocols which slightly increased the time for packet delivery and network traffic (i.e., the number of packets transmitted/received in the network). Nonetheless, for the tested scenario, this is not critical and cannot influence the resulting network lifetime as the *repeaters* cannot in any case use sleep modes due to the random data generation scenario. The obtained simulation results reveal that for the network built using GSP, unconnected nodes exist. This happens due to the random selection of sleeping nodes, which can result in some parts of the network being ‘cut-off’ by the sleeping nodes [consider Figure 8(b) for example]. For the EAMCDS, the unconnected nodes appeared every now and then when the nodes were dropping the routing packets during network build due to interference or noise. For the other protocols, this had not happened as often as the amount of packets sent during network construction was significantly lower (see Tables 2 and 3).

Figure 8 Example layouts of sparse ($N_{mains} = 4$, $N_{bat} = 9$ in 300 m × 300 m area) and dense network ($N_{mains} = 9$, $N_{bat} = 18$ in 300 m × 300 m area) and resulting networks for different routing protocols for $V_{bat} < V_{mains}$ and $V_{mains} > V_{bat}$ scenarios (a common legend is presented above the subfigures), (a) ‘sparse’ and ‘dense’ layouts before network build (b) networks built by GSP ($P_{awake} = 0.7$) (c) networks built by MCP (d) networks built by EAMCDS (e) networks built by EAD (f) networks built by suggested PSTID protocols (see online version for colours)

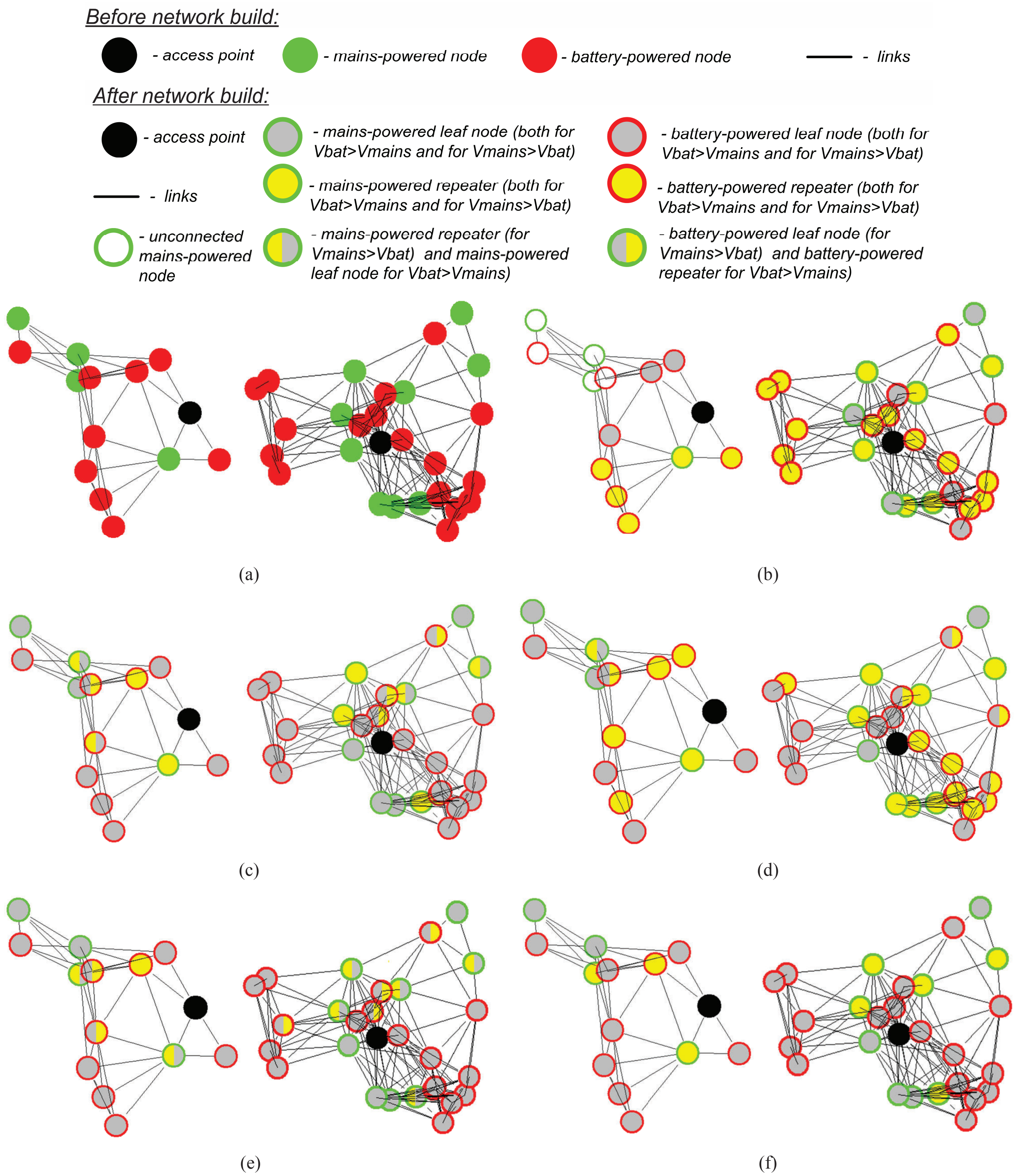


Table 2 Results of long-term simulations for 'sparse' network scenario for V_{MAINS} = 3 V and V_{MAINS} = 2 V

Parameters	*'Sparse' network [see Figure 8(a)] V_{MAINS} = 3 V*						*'Sparse' network [see Figure 8(a)] V_{MAINS} = 2 V*					
	GSP	*MCP*	*EAMCDS*	*EAD*	*PSTID-CBO*	*PSTID-VBO*	*GSP*	*MCP*	*EAMCDS*	*EAD*	*PSTID-CBO*	*PSTID-VBO*
Lifetime, hours	49.1	58.1	37.2	86.1	63.2	101.0	49.1	42.2	37.2	43.2	63.2	101.0
Average longest route, hops	3.1	3.1	3.2	3.4	3.0	3.5	3.1	3.1	3.1	3.1	3.0	3.5
Average number of *repeaters*	8.6	3.8	7.4	3.5	3.1	3.4	8.6	4.2	7.1	3.9	3.1	3.4
Average number of *leaf-nodes*	3.9	9.2	5.5	9.5	9.9	9.6	3.9	8.8	5.8	9.1	9.9	9.6
Average number of unconnected nodes	0.54	0.00	0.06	0.00	0.00	0.00	0.54	0.00	0.08	0.00	0.00	0.00
Average number of mains-powered *repeaters*	2.8	1.8	2.3	1.9	2.0	2.0	2.8	1.6	1.8	0.9	2.0	2.0
Average number of battery-powered *repeaters*	5.8	2.0	5.1	1.6	1.1	1.4	5.8	2.6	5.3	3.0	1.1	1.4
Average traffic through mains-powered *repeaters**, thousands of packets	4.67	4.94	4.89	7.06	5.69	7.64	4.67	3.98	3.22	2.35	5.69	7.64
Average traffic through battery-powered *repeaters**, thousands of packets	6.33	6.41	5.80	5.54	4.84	4.80	6.33	7.12	7.35	9.58	4.84	4.80
Average traffic for network building, packets	9.1	13.0	31.5	13.0	13.0	13.0	9.1	13.0	30.7	13.0	13.0	13.0

Note: *During single network rebuild period (i.e., one hour).

Table 3 Results of long-term simulations for 'dense' network scenario for V_{MAINS} = 3 V and V_{MAINS} = 2 V

Parameters	*'Dense' network [see Figure 8(b)] V_{MAINS} = 3 V*						*'Dense' network [see Figure 8(b)] V_{MAINS} = 2 V*					
	GSP	*MCP*	*EAMCDS*	*EAD*	*PSTID-CBO*	*PSTID-VBO*	*GSP*	*MCP*	*EAMCDS*	*EAD*	*PSTID-CBO*	*PSTID-VBO*
Lifetime, hours	44.1	72.1	37.2	786	939	858	44.1	52.1	37.2	119	939	858
Average longest route, hops	3.1	3.1	3.1	3.0	3.0	3.0	3.1	3.1	3.1	3.0	3.0	3.0
Average number of *repeaters*	18.8	7.5	17.4	5.7	5.4	5.6	18.8	8.5	19.2	6.1	5.4	5.6
Average number of *leaf-nodes*	8.0	19.5	9.3	21.3	21.5	21.4	8.0	18.5	7.7	20.9	21.5	21.4
Average number of unconnected nodes	0.16	0.00	0.22	0.00	0.03	0.00	0.16	0.00	0.00	0.00	0.03	0.00
Average number of mains-powered *repeaters*	6.2	5.0	7.4	5.5	5.4	5.5	6.2	3.7	7.7	3.3	5.4	5.5
Average number of battery-powered *repeaters*	12.6	2.4	10.1	0.2	0.1	0.1	12.6	4.7	11.6	2.7	0.1	0.1
Average traffic through mains-powered *repeaters**, thousands of packets	9.7	10.1	10.8	13.2	13.5	13.5	9.7	5.1	6.3	7.2	13.5	13.5
Average traffic through battery-powered *repeaters**, thousands of packets	4.32	3.97	2.65	0.36	0.06	0.08	4.32	8.50	7.33	7.20	0.06	0.08
Average traffic for network building, packets	20.6	27.0	66.4	27.0	27.0	27.0	20.6	27.0	71.0	27.0	27.0	27.0

Note: *During single network rebuild period (i.e., one hour).

5 Conclusions

Real-life WSNs nowadays often include nodes that are powered by various power sources: mains; primary or secondary batteries; or energy from environmental harvesting systems. In the paper, we have suggested and evaluated the PSTID – the simple mechanism that allows identification of the type of power source based on several measurements of the supply voltage. The major advantage of the suggested PSTID mechanism is its simplicity – it does not require any external hardware components and it can be implemented by any contemporary microcontroller on a WSN node or on another device with minimum resource consumption.

Using the data that became available from the suggested PSTID mechanism, we introduced and proposed a novel routing protocol for heterogeneous WSNs where nodes are powered by various power sources. As has been proven by the simulation results, the usage of PSTID data and the corresponding routing protocol (i.e., PSTID-VBO) provides a significant increase in the lifetime of the WSN and is effective both for cases when there are multiple mains-powered nodes and when there are only several mains-powered nodes within the network. Unlike implementation for traditional routing protocols that have to use the supply voltage level as the measure of available energy, the suggested PSTID-based routing is effective both for cases when the supply voltages on the mains-powered nodes are above and when the voltages are below the supply voltages of battery-powered nodes.

Usage of the suggested PSTID mechanism and the corresponding routing protocol in a real-life WSN allows one to manufacture, without any differentiation, all nodes that will be supplied by different power sources. Indeed, all WSN nodes, regardless of their power sources, can contain the same software that will include the PSTID mechanism implementation to obtain the required information about the power source type and the suggested routing protocol that will use PSTID data to ensure energy-efficient network building.

References

AdaptivEnergy RLP (2010) *Random Vibration Joule-Thief Datasheet*, AdaptivEnergy RLP, Hampton VA, USA.

Ahmed, A., Ali, J., Raza, A. and Abbas, G. (2006) `Wired vs. wireless deployment support for wireless sensor networks', *Proceedings of TENCON '06*, pp.1–3.

Atmel Corporation (2006) *AVR453: Smart Battery Reference Design*, Atmel Corporation, San Jose, California, USA.

Atmel Corporation (2008) *AVR2016: RZRAVEN Hardware User's Guide*, Atmel Corporation, San Jose, California, USA.

Bluetooth SIG (2010) *Bluetooth Specification*, Bluetooth SIG Std. 4.0.

Boukerche, A., Cheng, X. and Linus, J. (2003) 'Energy-aware data-centric routing in microsensor networks', *Proceedings of the ACM MSWiM '03*, pp.42–49.

Chu, M., Haussecker, H. and Zhao, F. (2002) 'Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks', *International Journal of High Performance Computing Applications*, Vol. 16, No. 3, pp.293–313.

Chulsung, P., Lahiri, K. and Raghunathan, A. (2005) 'Battery discharge characteristics of wireless sensor nodes: an experimental analysis', *Proceedings of IEEE SECON '05*, pp.430–440.

Dias, D. (1994) 'A cost effective approach to intelligent battery packs', *Proceedings of WESCON '94*, September, pp.321–323.

Estrin, D., Sayeed, A. and Srivastava, M. (2002) 'Wireless sensor networks', available at http://nesl.ee.ucla.edu/tutorials/mobicom02 (accessed on 20 July 2012).

Farooq, M.O. and Kunz, T. (2010) `Operating systems for wireless sensor networks: a survey', *Sensors*, Vol. 11, No. 6, pp.5900–5930.

Feeney, L. and Willkomm, D. (2010) 'Energy framework: an extensible framework for simulating battery consumption in wireless networks', *Proceedings of the SIMUTools '10*, pp.1–4.

Heinzelman, W.R., Chandrakasan, A. and Balakrishnan, H. (2000) 'Energy-efficient communication protocol for wireless microsensor networks', *Proceedings of the HICSS '00*, Vol. 8, pp.8020.

Heinzelman, W.R., Chandrakasan, A. and Balakrishnan, H. (2002) 'An application-specific protocol architecture for wireless microsensor networks', *IEEE Transactions on Wireless Communications*, Vol. 1, No. 4, pp.660–670.

Hou, X. and Tipper, D. (2004) 'Gossip-based sleep protocol (GSP) for energy efficient routing in wireless ad hoc networks', *Proceedings of IEEE WCNC '04*, Vol. 3, pp.21–25.

Huang, S.C. and Jan, R.H. (2004) 'Energy-aware, load balanced routing schemes for sensor networks', *Proceedings of ICPAD '04*, pp.419–425.

IEEE Std. 21450 (2010) 'Information technology Smart transducer interface for sensors and actuators common functions, communication protocols, and transducer electronic data sheet (TEDS) formats', IEEE Std. ISO/IEC/IEEE 21450.

IEEE Std. 802.15.4 (2006) 'Part 15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs)', IEEE Std. 802.15.4.

IETF Std. RFC4944 (2007) 'Transmission of IPv6 packets over IEEE 802.15.4 networks', IETF Std. RFC4944.

Intanagonwiwat, C., Govindan, R. and Estrin, D. (2000) 'Directed diffusion: A scalable and robust communication paradigm for sensor networks', *Proceedings of the ACM MobiCom '00*, pp.56–67.

Jongerden, M., Mereacre, A., Bohnenkamp, H., Haverkort, B. and Katoen, J. (2010) 'Computing optimal schedules of battery usage in embedded systems', *IEEE Transactions on Industrial Informatics*, Vol.6, No.3, pp.276–286.

Kulik, J., Heinzelman, W. and Balakrishnan, H. (2002) 'Negotiation-based protocols for disseminating information in wireless sensor networks', *Wireless Networks*, Vol. 8, Nos. 2/3, pp.169–185.

Kuorilehto, M., Kohvakka, M., Suhonen, J., Hamalainen, P., Hannikainen, M. and Hamalainen, T. (2007) *Ultra-Low Energy Wireless Sensor Networks in Practice: Theory, Realization and Deployment*, John Wiley & Sons, Hoboken, NJ.

Lahiri, K., Raghunathan, A., Dey, S. and Panigrahi, D. (2002) 'Battery-driven system design: a new frontier in low power design', *Proceedings of ASP-DAC '02*, pp.261–267.

Lindsey, S. and Raghavendra, C.S. (2002) 'PEGASIS: power-efficient gathering in sensor information system', *Proceedings of the IEEE Aerospace Conference*, pp.1125–1130.

Madden, S., Franklin, M., Hellerstein, J. and Hong, W. (2002) 'TAG: a tiny aggregation service for ad-hoc sensor networks', *Proceedings of the OSDI '02*, pp.131–146.

Mandal, S.K., Mahapatra, R.N., Bhojwani, P.S. and Mohanty, S.P. (2010) 'IntellBatt: toward a smarter battery', *Computer*, Vol. 43, No. 3, pp.67–71.

Manjeshwar, A. and Agrawal, D.P. (2001) 'TEEN: a protocol for enhanced efficiency in wireless sensor networks', *Proceedings of IPDPS '09*, pp.2009–2015.

Micropelt GmbH (2010) *TE-Power NODE Datasheet*, Micropelt GmbH, Freiburg, Germany.

Mikhaylov, K. and Tervonen, J. (2010) 'Optimization of microcontroller hardware parameters for wireless sensor network node power consumption and lifetime improvement', *Proceedings of ICUMT '10*, pp. 1150–1156.

Mikhaylov, K. and Tervonen, J., (2011) 'Node's power source type identification in wireless sensor networks', *Proceedings of the BWCCA '11*, pp.521–525.

Mikhaylov, K., Tervonen, J. and Fadeev, D. (2012a) 'Development of energy efficiency aware applications using commercial low power embedded systems', in K. Tanaka (Ed.): *Embedded Systems – Theory and Design Methodology*, InTech, Rijeka, Croatia, pp.407–430.

Mikhaylov, K., Tervonen, J., Heikkilä, J. and Känsäkoski, J. (2012b) 'Wireless sensor networks in industrial environment: real-life evaluation results', *Proceedings of the BCFIC '12*, pp.1–7.

MiXiM (2012) 'MiXiM project', available at http://mixim.sourceforge.net/ (accessed on 20 July 2012).

OGC 07-165 (2007) 'OGC sensor web enablement: overview and high level architecture', OpenGIS®, White Paper, OGC 07-165.

OMNeT++ (2012) 'OMNet++ network simulation framework', available at http://www.omnetpp.org/ (accessed on 20 July 2012).

Padmanabhan, K. and Kamalakkannan, P. (2011) 'A study on energy efficient routing protocols in wireless sensor networks', *European Journal of Scientific Research*, Vol. 60, No. 4, pp.499–511.

Penella, M., Albesa, J. and Gasulla, M. (2009) 'Powering wireless sensor nodes: primary batteries versus energy harvesting', *Proceedings of I2MTC '09*, pp.1–6.

Pomerantz, D. (1990) 'The characterization of high rate batteries', *IEEE Transactions on Consumer Electronics*, Vol. 36, No. 4, pp.954–958.

Raei, H., Sarram, M. and Ghasemzadeh, M. (2011) 'Energy-aware distributed algorithm for virtual backbone in wireless sensor networks with bidirectional links', *Scientific Research and Essays*, Vol. 6, No. 10, pp.2154–2163.

Rodoplu, V. and Meng, T.H. (1999) 'Minimum energy mobile wireless networks', *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 8, pp.1333–1344.

SBS Implementers Forum (1998) *Smart Battery Data Specification, Smart Battery System Specifications*, Revision 1.1.

Singh, S.S., Singh, M.P. and Singh, D.K. (2010) 'A survey of energy-efficient hierarchical cluster-based routing in wireless sensor networks', *Int. J. of Advanced Networking and Applications*, Vol. 2, No. 2, pp.570–580.

Stolitzka, D. and Dawson, W.S. (1994) 'When is it intelligent to use a smart battery?', *Proceedings of BCAA '94*, pp.205–222.

Texas Instruments (2009a) *slau227e: eZ430-RF2500 Development Tool User's Guide*, Texas Instruments, Dallas, Texas, USA.

Texas Instruments (2009b) *swru236a: CC1110 & CC2510 Mini Development Kit User's Guide*, Texas Instruments, Dallas, Texas, USA.

Texas Instruments (2010) *slau273: eZ430-RF2500-SEH Solar Energy Harvesting Development Tool*, Texas Instruments, Dallas, Texas, USA.

Thomson, E. (2008) 'Preventing forest fires with tree power', *MIT Tech Talk*, September 24.

Villalba, L., Orozco, A., Cabrera, A. and Abbas, C. (2009) 'Routing protocols in wireless sensor networks', *Sensors*, Vol. 9, No. 11, pp.8399–8421.

Wisepro (2011) *WISE Wireless Sensor Network*, Wisepro OY, Alavieska, Finland.

Xu, Y., Heidemann, J. and Estrin, D. (2001) 'Geography-informed energy conservation for ad hoc routing', *Proceedings of the ACM/IEEE MobiCom '01*, pp.70–84.

Ye, W., Heidemann, J. and Estrin, D. (2002) 'An energy-efficient MAC protocol for wireless sensor networks', *Proceedings of the INFOCOM '02*, pp.1567–157.