Building a Green Connected Future: Smart (Internet of) Things for Smart Networks

A Thesis Presented

By

Georgia Koutsandria

to

The Department of Computer Science

in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

Sapienza University of Rome
Rome, Italy

October, 2018

Thesis Committee

Professor Chiara Petrioli (advisor)
Department of Computer Science
Sapienza University of Rome
Rome, Italy

Professor Gaia Maselli
Department of Computer Science
Sapienza University of Rome
Rome, Italy

Professor Enrico Tronci
Department of Computer Science
Sapienza University of Rome
Rome, Italy

External Reviewers

Professor Kaushik R. Chowdhury
Department of Electrical and Computer Engineering
Northeastern University
Boston, MA, USA

Professor Swades De
Department of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi, India

Abstract

The vision of Internet of Things (IoT) promises to reshape society by creating a future where we will be surrounded by a smart environment that is constantly aware of the users and has the ability to adapt to any changes. In the IoT, a huge variety of smart devices is interconnected to form a network of distributed agents that continuously share and process information. This communication paradigm has been recognized as one of the key enablers of the rapidly emerging applications that make up the fabric of the IoT. These networks, often called wireless sensor networks (WSNs), are characterized by the low cost of their components, their pervasive connectivity, and their self-organization features, which allow them to cooperate with other IoT elements to create large-scale heterogeneous information systems. However, a number of considerable challenges is arising when considering the design of large-scale WSNs. In particular, these networks are made up by embedded devices that suffer from severe power constraints and limited resources.

The advent of low-power sensor nodes coupled with intelligent software and hardware technologies has led to the era of *green wireless networks*. From the hardware perspective, green sensor nodes are endowed with energy scavenging capabilities to overcome energy-related limitations. They are also endowed with low-power triggering techniques, i.e., wake-up radios, to eliminate idle listening-induced communication costs. Green wireless networks are considered a fundamental vehicle for enabling all those critical IoT applications where devices, for different reasons, do not carry batteries, and that therefore only harvest energy and store it for future use. These networks are considered to have the potential of infinite lifetime since they do not depend on batteries, or on any other limited power sources. Wake-up radios, coupled with energy provisioning techniques, further assist on overcoming the physical constraints of traditional WSNs. In addition, they are particularly important in green WSNs scenarios in which it is difficult to achieve energy neutrality due to limited harvesting rates.

In this PhD thesis we set to investigate how different data forwarding mechanisms can make the most of these green wireless networks-enabling technologies, namely,

energy harvesting and wake-up radios. Specifically, we present a number of cross-layer routing approaches with different forwarding design choices and study their consequences on network performance. Among the most promising protocol design techniques, the past decade has shown the increasingly intensive adoption of techniques based on various forms of machine learning to increase and optimize the performance of WSNs. However, learning techniques can suffer from high computational costs as nodes drain a considerable percentage of their energy budget to run sophisticated software procedures, predict accurate information and determine optimal decision. This thesis addresses also the problem of local computational requirements of learning-based data forwarding strategies by investigating their impact on the performance of the network. Results indicate that local computation can be a major source of energy consumption; it's impact on network performance should not be neglected.

Στον αδερφό μου.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my advisor, professor Chiara Petrioli, for all her support and guidance that she has given me over the past years. She undoubtedly set an example of excellence as a researcher and a mentor that motivated me to achieve my goals. Her sharp insights on every research topic have influenced the evolution of my academic career and have allowed me to develop my critical thinking skills. I have tremendously benefited from my discussions with her, shorter or longer. Her continuous passion and energy to dive into a research problem made me admire her more: From the very first beginning I realized that a couple of figures were more than enough for her to provide me with a concrete feedback and with the answers to problems that seemed impossible to solve. She was a key enabler on making this ride a learning experience that academically shaped me as a person, and for that I thank her.

I would like to thank my thesis committee members, professor Gaia Maselli and professor Enrico Tronci, as well the external reviewers of my thesis professor Kaushik Chowdhury and professor Swades De, for generously offering their time out of their busy schedules, their guidance, and their invaluable feedback during this process.

I am very fortunate to have had the opportunity to collaborate with professor Stefano Basagni, who hosted me at the Northeastern University during my period abroad. He contributed to this thesis with genuine enthusiasm and he was always more than happy to provide me with honest advice, suggestions for improvements, and encouraging words whenever I needed them. He and his ingenious ideas definitely crafted my way of thinking and contributed not only to this thesis but also, and most importantly, to my intellectual growth. I would especially like to thank him for making this path enjoyable with numerous fun discussions during our meetings. Thank you Prof.!

During these past years, I was extremely lucky to have collaborated with more amazingly talented people. I would like to thank Dora Spenza, who acted as my role model in the first years of this journey. I owe to her a great deal of my research toolbox. I would especially like to thank Valerio Di Valerio for the support and the constant encouragement he has given me during the last years. Even though you were always listed before me as an author, alphabetically, I thank you for leading me towards a research path full of curiosity, patience and knowledge.

Many thanks go to Angelo Capossele, not only for being a precious lab-mate, but also for being a special friend, for passing to me at least a small fraction of his crazy ideas and for being a great supporter in tough moments. I would like to thank all the “underwater guys” of the Senses lab: Petrika Gjanci, Luigi Picari, and Daniele Spaccini, as well as the people of “terrestrial" side: Federico Ceccarelli, Fabrizio Gattuso, and Nupur Thakker for assisting on loading this journey with beautiful and funny moments.

Last but not least, I would like to thank my family: my brother, Panos, and my parents, Giannis and Thoula, for their continuous support and for their unconditional love. This thesis would not have been made possible without your struggle, love, and support. Feelings remain the same similar to “scripta manent:” I cannot find any appropriate words to express how grateful I am to all of you for every sacrifice that you have made on my behalf!

Contents

1 Introduction

The enormous growth of applications using wireless devices that has been observed in recent years has redefined our interaction with many physical aspects of our life in a way that has never been possible before. The multi-hop and ad hoc networking of the large amount of devices, called sensor nodes, is a communication paradigm that has been recognized as one of the key enablers of rapidly emerging applications, including those that make up the fabric of the Internet of Things (IoT) and Smart Cities. The networks formed by the sensor nodes, often called wireless sensor networks (WSNs), are characterized by the low cost of their components, pervasive connectivity, and self-organization features, which allow them to cooperate with other IoT elements to create large-scale heterogeneous information systems. Nodes gather sensory information and communicate wirelessly with other nodes to forward the sensed data towards a network data collector—the sink. They can be equipped with a variety of sensors, making them suitable for a wide range of applications with different characteristics and requirements. The widespread adoption of these networks for applications ranging from structural health monitoring of critical infrastructures [19] to healthcare systems for medical monitoring [2] witnesses the enhancement of the quality of everyday life.

The proliferation of applications and technologies for wireless networking, however, is hampered by limitations and challenges arising from the nature of many network devices (e.g., sensor nodes), including narrow energy availability and strict power requirements. The continuous process of data collection in wireless networks with battery-operated nodes establishes energy efficiency and optimization of utilized power as the most critical and essential requirements on their design. In fact, regular battery replacement can be cumbersome, even unfeasible in some cases, making their deployment difficult and expensive to maintain. For example, in healthcare monitoring systems, the main energy resource capacity of implanted or wearable sensors is limited by their small size, constituting the replacement of batteries their major performance bottleneck [34, 48]. Therefore, research efforts to obviate the joint problems of battery replacement and long lasting performance require determining the right combination of dedicated advanced techniques. To this end, the requirement of energy efficient wireless networks has driven research towards networks with energy harvesting capabilities. Networks deploying energy harvesting, often dubbed *green wireless networks*, are the leading force for expanding the capabilities of traditional wireless networks due to their capabilities of harvesting ambient energy and of the use of rechargeable batteries or supercapacitors to store the harvested energy for future use. It has been shown that energy harvesting can significantly extend the lifetime of the network [12]. Although nodes in energy harvesting-based WSNs have the potential of boundless lifetime, the uncertainty of the energy harvesting rates makes them susceptible to unpredictable operational "black outs:" When a node depletes its energy, it cannot longer function; once it harvests sufficient energy to perform its tasks, it becomes operational again. Clearly, the shorter the black outs, the better the network performance.

For this reason, all possible sources of useless energy consumption should be reduced to a minimum, which includes a node idly listening to the wireless channel awaiting for incoming transmissions. To obviate to idle listening, research has proposed low-power triggering techniques aimed at leaving nodes with their radio off until they need to receive data. Emerging low-power radio triggering techniques, such as wake-up radios [59, 44], are able to efficiently cope with the energy toll of communication. Network nodes in wake-up radio-enabled networks are equipped with two transceivers: A main transceiver (the *main radio*) that is used only to exchange packets, and a low-power wake-up transceiver (the *wake-up radio*) used to trigger nodes within wake-up communication range to turn on their main radio. It has been shown that by turning off the main transceiver when a node does not have to transmit or receive packets, the network energy consumption is reduced up to three orders of magnitude [59]. To further improve energy efficiency of wake-up radio-enabled green wireless networks, *semantic addressing* can be used to selectively wake-up a subset of neighboring nodes based on metrics such as distance from the destination and current energy status [59, 13, 51]. In this case, nodes are characterized by a set of wake-up addresses, each of them dynamically revised following the dynamics of node and network status. Semantic addressing capabilities are effectively used to enhance communication by allowing nodes to selectively wake-up a suitable subset of neighboring nodes. This subset is determined by the distance of nodes from the sink, and, greedily, by the residual energy along routes to the sink. Based on the above discussed challenges, issues and advances, more research efforts are required for making the vision of WSN-based IoT a reality. Particularly, routing protocols for green wireless networks need to be designed that draw benefits from the joint exploitation of wake-up radio technology and energy harvesting.

The remainder of this thesis is organized as follows. In Chapter 2 we give a general introduction on state-of-the-art solutions of data forwarding strategies in WSNs, including those specifically designed for wake-up radio-enabled networks. In Chapter 3 we present GreenRoutes, an energy-aware routing protocol for Energy Harvesting-based ("green") Wireless Sensor Networks that leverages self-powered technologies for eliminating the need of energy storage device replacement. GreenRoutes combines energy harvesting and wake-up radios with semantic addressing to enhance communication by allowing nodes to selectively wake-up a suitable subset of neighboring nodes. In GreenRoutes, this subset is determined by the distance of nodes from the sink, and, greedily, by the residual energy along routes to the sink. The performance of GreenRoutes has been compared to that of the Energy Harvest Wastage-Aware (EHWA) routing solution [40] in scenarios where all nodes harvest energy from the same source, either sun or wind. Results show that GreenRoutes achieves a packet delivery ratio significantly higher (up to 40%) than EHWA, while delivering packets faster and for less power.

In Chapter 4 we present a sophisticated learning-bases data forwarding strategy for green wireless networks that fully exploits the self-powered wake-up radio capabilities of the network nodes. The proposed strategy, named WHARP for Wake-up and HARvesting-based energy-Predictive forwarding, sends data to their destination by making decentralized and proactive decisions based on forecast energy and expected traffic. The performance of WHARP has been compared to that of the Energy Harvesting Wastage-Aware (EHWA) strategy through GreenCastalia-based simulations. Results show that our approach delivers up to 72% more packets, 1.6 times faster, and consuming 58% less energy than EHWA. This is obtained through a learned selection of forwarder relays allowing WHARP nodes to be operational 98% of the time: A 30% improvement over EHWA.

In Chapter 5 we study the performance of different data forwarding strategies for green wireless networks. In particular, we analyze and provide insights into the impact on performance of diverse forwarding design choices, ranging from traditional tree-based routing (CTP-WUR), to end-to-end energy-driven route selection (GreenRoutes), to the use of sophisticated learning models (WHARP). Results show that tree-based routing obtains lesser packet delivery ratio than WHARP, thus indicating that including energy harvesting awareness in route selection results in performance advantages. However, the proactive nature of route computation of CTP-WUR results in faster packet delivery and lower energy consumption, requesting further optimization of the cross-layer forwarding of GreenRoutes and WHARP.

The significant limitations in terms of memory, energy, and computational power, have prompted the design of protocols at all layers of the networking stack that are aware of these limitations, seeking to obtain performance that is adequate to support critical WSN applications. In fact, superior performance is being obtained by taking key protocol decisions based on the outcome of local *learning*-based computations, informing nodes on past and expected availability of resources, which allows optimized choices. However, learning techniques can suffer from high computational costs as nodes drain a considerable percentage of their energy budget to run sophisticated software procedures, predict accurate information and determine optimal decision. Our contribution to this topic is presented in Chapter 6, in which we investigate the impact on protocol performance of local computational requirements of learning techniques. We consider the recent routing solution, named WHARP, which is introduced in Chapter 4. WHARP makes decentralized and proactive decisions based on a Markov Decision Process (MDP) that takes into account key parameters of wireless green networks, including energy harvesting capabilities, and wake-up radio technology. We show that in these scenarios solving the MDP incurs energy expenditures by far superior to that required by wireless communication, even at very high data traffic. In order to maintain the performance advantages of the learning-based protocol machinery, we propose a heuristic solution that closely approximates the MDP trading off optimality for considerably lighter computational requirements. We compare the performance of WHARP using our new solution method (called W-HEU) to that of WHARP using the standard Backward Value Iteration solution methodology (W-BVI) through GreenCastalia-based simulations based on real computational energy measurements. Our results show that W-HEU outperforms W-BVI on key metrics such as energy consumption and packet delivery ratio, making up for the lost optimality of BVI through the remarkable energy savings of its lighter computational requirements.

The performance comparison presented in Chapter 5 provided us with useful insights about the different forwarding design choices and their consequences on network performance indicating that the sophisticated learning-based design of WHARP (Chapter 4) allows nodes to successfully select next-hop relays along routes without nodes that black out. In Chapter 7 we present a preview of an "on-going" work that takes under consideration the above mentioned insights by providing a brief description of WHARPNR-HEU (Wake-up and HARvesting-based energy-Predictive No-Rts with HEUristics). WHARPNR-HEU is built upon the WHARP forwarding strategy for green wireless networks. In particular, WHARPNR-HEU eliminates the RTS packet exchange to reduce energy consumption and end-to-end latencies. We further reduce consumptions due to unnecessary communication by including the MDP-based optimal forwarding decisions in the wake-up semantic addressing. Finally, we consider that the MDP is solved using a heuristics solution whose computational cost is taken under consideration in the performance evaluation.

List of publications

Papers accepted for publications and on-going works that constitute the basis of this thesis are listed in the following. For each work, the corresponding chapter is reported in parenthesis.

- S. Basagni, V. Di Valerio, G. Koutsandria, and C. Petrioli. "Wake-up radio-enabled routing for green wireless sensor networks". In: *Proceedings of the 86th IEEE Vehicular Technology Conference, IEEE VTC 2017-Fall*. Toronto, Ontario, Canada, September 24-27 2017.
  (Chapter 3)
- S. Basagni, V. Di Valerio, G. Koutsandria, C. Petrioli, and D. Spenza. "WHARP: a wake-up radio and harvesting-based forwarding strategy for green wireless networks". In: *Proceedings of the 14th IEEE International Conference on Mobile Ad hoc and Sensor Systems, IEEE MASS 2017*. Orlando, FL, USA, October 22-25 2017.
  (Chapter 4)
- S. Basagni, G. Koutsandria, and C. Petrioli. "A comparative performance evaluation of wake-up radio-based data forwarding for green wireless networks". In: *Proceedings of the 27th IEEE International Conference on Computer Communications and Networks, IEEE ICCCN 2018*. Hangzhou, China, July 30-August 2 2018.
  (Chapter 5. A journal version considering enhanced solutions and varied scenarios is in preparation.)
- S. Basagni, V. Di Valerio, G. Koutsandria, and C. Petrioli. "On the impact of local computation over routing performance in green wireless networks". In: *Proceedings of the 19th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, IEEE WoWMoM 2018*. Chania, Greece, June 12-15 2018.
  (Chapter 6)
- S. Basagni, V. Di Valerio, G. Koutsandria, C. Petrioli, and D. Spenza. *WHARPNR-HEU: Exploiting wake-up radios and energy harvesting for green wireless networks*. In preparation for journal submission. 2018
  (Chapter 7)

2 Review of Data Forwarding Strategies in Green Wireless Networks

This section aims at providing a brief recount of data forwarding strategies for WSNs, with an emphasis on networks with energy harvesting and on networks with wake-up radios, i.e., green wireless networks. Table 2.1 summarizes some of the forwarding strategies that we survey in this section. From the perspective of traditional multi-hop WSNs, a plethora of works has been proposed that focus on the design and the development of forwarding strategies in these networks, ranging from greedy geographic-based to energy-based forwarding schemes to allow efficient communication. Swades De [22] presented a greedy forwarding approach, namely Least Remaining Distance (LRD), which is based on the relationship of the hop count and the Euclidean distance between two nodes. LRD attempts to minimize the remaining distance to the destination node in each hop, by selecting as next-hop the node which yields the minimum distance to the destination node among the neighobring relays towards the destination. Multiple next-hop selection criteria can be considered to decide on the optimality of neighboring nodes, including link quality, delay, and minimum number of re-transmissions [6, 37, 23].

In addition, minimization of energy consumption in WSNs is of great importance to prolong their lifetime due to battery-related constraints. Energy efficient approaches have been discussed and evaluated in depth through surveys, reviews, and comparative studies [24, 63, 47]. Panigrahi et al. [46] proposed three variant forwarding strategies, namely, GMFP, LM-GMFP and VAR-GMFP, which are all energy-aware and take into account energy consumption and/or local information on the residual energy of nodes. In GMFP next-hop relays are selected based on the minimum energy per successful packet transmission and on the distance progress towards the destination. LM-GMFP extends the GMFP variant by considering also the residual energy of potential next-hop relays. Lastly, VAR-GMFR attempts to increase the lifetime of the network by maximizing the mean network energy consumption while minimizing the variance of residual energy on the nodes. Spachos et al. present an energy-aware opportunistic routing protocol for wireless sensor networks, namely EAOR, that uses the RTS/CTS handshake mechanism [58]. In EAOR the selection of the forwarder node is done based on the distance from the sink and the available

Tab. 2.1: Forwarding strategies in WSNs.

Name	Taxonomy	Power Management	Energy prediction/ harvesting
LRD [22]	Routing	Duty-cycled	No/No
CTP-WUR [13]	Tree-based	Wake-up Radio with semantic addressing	No/No
ALBA-WUR [59]	Geographic Cross-layer	Wake-up Radio with semantic addressing	No/No
T-ROME [36]	Geographic Cross-layer	Wake-up Radio	No/No
EHOR [27]	Geographic Routing	Wake-up Radio	No/Yes
FLOOD-WUP [51]	Topology-based Routing	Wake-up Radio	No/No
GREEN-WUP [51]	Energy-aware Cross-layer	Wake-up Radio with semantic addressing	No/Yes
WRTA [61]	Load-balancing Routing	Wake-up Radio	No/Yes
GreenRoutes [8]	Load-balancing Cross-layer	Wake-up Radio with semantic addressing	No/Yes
WHARP [9]	Energy-driven Cross-layer	Wake-up Radio with semantic addressing	Yes/Yes

energy level. Specifically, next hop selection is realized via a timer-based competition among neighboring nodes through which nodes with lower energy consumption and closer to the destination are more likely to be chosen as next hop. However, the vast majority of existing solutions is designed for battery-operated wireless networks and cannot be easily adapted to green networks.

By providing virtually unlimited energy to nodes, power-scavenging techniques tumble the generic fundamental hypothesis of limited energy resources that are depleted over time. Due to this unique characteristic of energy harvesting-based wireless networks, the design of harvesting-aware communication solutions requires a paradigm shift. In fact, while solutions for traditional wireless networks typically focus on minimizing the energy consumption, the additional goal of harvesting-based communication strategies is that of maximizing the sustainable workload [16]. Towards this direction, a variety of data forwarding strategies has been proposed, specifically targeting on energy harvesting wireless network [53]. Babayo et al. present a review paper where various energy management schemes are classified based on the energy-requirements of applications [5]. While the authors give several useful insights on how various existing works fall into different categories, this work provides little insights on protocol design. Similarly, the survey by Khan, Qureshi, and Iqbal provides a baseline discussion on developing energy efficient management schemes in WSNs [33]. They also investigate existing solutions from an energy perspective, where energy management can be handled on the basis of energy provisioning or energy consumption. A useful and quite comprehensive introduction to energy harvesting-based WSNs is provided by Basagni et al. [12] and by Mishra et al. [41]. These works mainly aim at exploring the opportunities and challenges of using networks with energy harvesting capabilities.

A first set of solutions considers networks solely powered by ambient energy harvesting, in which nodes have no long-term energy storage [14]. In the vast majority of applications nodes are equipped with large supercapacitors and/or rechargeable batteries to survive during periods of low energy intake [45]. In general, most of the available solutions focus only on the design of strategies at the network layer, failing to take into account the effect of realistic lower layers, e.g., non-ideal medium access control. Such approaches, which are far to be realistic, lead to significant over-estimation of achievable performance [30]. This aspect is further exacerbated by the use of over-simplified models of the harvesting process and of node energy consumption. Because of the uncertainty of the energy harvesting rates research has headed towards the design of protocols that consider these variations when routing packets. Some works make the assumption that the harvested energy is known for each node over a finite time horizon [39]. The authors in [40] present an energy wastage-aware route selection protocol for EH-WSNs, named the Energy Harvest Wastage Aware (EHWA) protocol. EHWA is an on-demand dynamic source routing (DSR) protocol that aims at minimizing the energy wastage due to battery overcharging. However, due to the high volume of broadcasting packets that is required by the mechanism of DSR to forward packets, such solutions could lead to high energy consumption.

A performance evaluation of routing protocols in device-to-device energy harvesting-based networks is presented in [56]. Two well-known routing protocols, namely Optimized Link State Routing (OLSR) and Ad hoc on Demand Distance Vector (AODV), are evaluated and their performance is compared. These two protocols, however, are not specifically designed for green wireless networks, they are not considered to be energy-aware, and their performance evaluation is limited only to two energy-related metrics and to packet delivery ratio. Eu et al. investigate the performance of different MAC schemes adapted to WSNs with energy harvesting capabilities under several metrics [26]. Through simulations, the authors discuss the behavior of the investigated approaches and how different parameters affect their performance. However, this work does not discuss any data forwarding strategies, which is the main concern of this thesis. A comparison of routing protocols for WSNs powered by ambient energy harvesting is provided by Hasenfratz et al. in [30]. Through a GreenCastalia-based simulation comparison, the authors analyze the performance of three routing approaches, namely, E-WME, R-MF, and R-MPRT. Their comparison is based on two performance metrics: Packet loss and energy consumption. Protocols are evaluated under several realistic scenarios, including usage of a low-power MAC protocol and models for lossy wireless channels. The protocols considered in this work have similar design, and do not consider cross-layer approaches. In particular, this thesis investigates the design of adaptive data forwarding strategies that are cross-layer, thus taking into account also lower-layers, and that are fully optimized for green wireless networks.

Lower energy consumption and longer lifetimes can be achieved by adopting techniques that allow nodes to switch their radio from on to off whenever nodes are in an "idle" state according to the preset duty cycle, thus drastically reducing power consumption. In [14], the authors present a solution specifically targeting on networks solely powered by ambient energy harvesting, in which nodes have no long-term energy storage. They propose OR-AHaD, a routing scheme with duty cycling that acts in an energy-aware adaptive manner, by taking into account the short-term estimated harvesting rate to adjust nodes duty cycle. Han. et al. [29], design a cross-layer optimized geographic routing that blends duty cycling and energy harvesting techniques to balance energy consumption. The development on energy harvesting technologies mitigates the energy scarcity issue by adopting duty-cycling techniques that allow the nodes to be active during a predefined amount of period and to be in a "sleep" mode in the rest of the time. In [27], the authors propose an energy harvesting opportunistic routing protocol (EHOR) specifically targeting on networks solely powered by energy harvesters. EHOR considers a grouping approach of potential nodes by taking into consideration the distance from the sink, as well as their residual energy in order to allocate transmission priorities. Even though adopting duty-cycling techniques slows down the depletion of the energy reservoir, nodes waste considerable amounts of energy during periods when they do not process data packets.

In the realm of green WSNs with wake-up radio capabilities, the works by Petrioli et al. [51], Spenza et al. [59], and Kumberg et al. [36] are worth citing. For details on wake-up radio technology suitable for green wireless networks see [52]. The first work is about the effectiveness of using wake-up radios for abating the latencies and energy consumption typical of solutions based on duty cycling [51]. This work also serves the purpose of introducing a new architecture for a wake-up radio. Two simple protocols, unicast and broadcast-based, namely GREEN-WUP and FLOOD-WUP, respectively, are described and compared to show that usage of wake-up radios allows remarkable performance improvements. A cross-layer approach for data gathering in wireless sensing systems, namely ALBA-WUR, was presented in [59]. ALBA-WUR, which is the version for wake-up radio of ALBA-R, is considered as an energy efficient data forwarding protocol for WSNs [50]. Besides showing once more the remarkable performance improvements achievable via wake-up radios, this work shows the flexibility of wake-up radio semantic addressing for re-designing complex data forwarding strategies. ALBA-WUR takes advantage of wake-up radio technologies with semantic addressing to selectively wake-up only those neighboring relays whose status makes them the best relays to process packets. Forwarder nodes are selected based on a pool of different policies that include the current traffic, channel conditions, and the geographic advancement towards the destination. Finally, the work by Kumberg et al. presents a new data forwarding protocol for WSNs with wake-up radios, named T-ROME, and compares it with CTP-WUR [36]. The main aim of the work is that of presenting a new protocol capable of taking advantage of wake-up radios. The relaying discovery follows a tree routing algorithm where nodes forward wake-up packets only to their parent nodes until the packet reaches the destination. During the transmission of data packets T-ROME makes use of different transmission ranges of wake-up and main radios to further reduce energy consumption. CTP-WUR is a cross-layer routing protocol for data gathering in wake-up radio based wireless networks described in [13]. In CTP-WUR wake-up packets contain the unique identifier (ID) of a node and a flag indicating that the packet should be further passed from the receiving node to its parent. A load-balancing routing protocol, named WRTA, proposed by Vodel et al. [61], aims at achieving reliable communication. WRTA takes into account the energy level of nodes with a large amount of data to select the shortest path.

However, most of these works do not consider nodes with energy harvesting capabilities, or it is explicitly concerned with extensive performance comparisons under multiple metrics. In this thesis, we focus on data forwarding strategies specifically designed for green wireless networks where nodes jointly exploit energy provisioning and wake-up radios to take optimal forwarding decisions. This has a significant impact on performance of the network including the overall energy consumption and the periods during which each relay is operational.

From a hardware perspective, Chen et al. [20] presented a low power wake-up radio receiver, namely REACH-Mote, which benefits from energy harvesting technologies to improve the wake-up range for passive wake-up radio sensor nodes. An improved version of REACH-Mote named REACH^2-Mote is described by Chen et al. [21] that furthers improves the wake-up range. by applying an improved energy harvesting module and a supply voltage regulator. In [32] an RF energy harvester wake-up radio receiver is proposed which can perform both range-based and directed-based wake-up. Paoli et al. [49] present Magonode++, a low-power wake-up radio mote which extends the capabilities of the Magonode [59] platform exploiting energy-harvesting and wake-up radio features. In addition, and despite the wide variety of solutions using learning techniques to increase and optimize the performance of WSNs [3, 35], the energy consumption needed for running the learning frameworks is not explicitly addressed in previous works or determined using real hardware. Contrary to widely recognized beliefs, this thesis clearly points out that local computation can be as impactful on protocol and overall network performance as communication. This was somewhat clear to those researchers who focused on a subclass of problems and applications of WSNs, such as those working on multi-media networking, or on implementing secure communications. For instance, when sensing involves images and video, and more generally, in the realm of multi-media WSNs, it is understood that the energy demanded by the sensory equipment is non-negligible. The paper by Tahir and Farrell provides an example of the importance of finding trade-offs between communication and computation energy consumption in wireless multi-media sensor networks [60]. In networks concerned with the computation and the transmission of large volumes of video frames, Arastouie et al. investigate the trade-off between computation and communication energy that allows the nodes to make the best decision on the level of compression of the frames [4]. It has always been clear that securing WSN communications is a particularly critical task because cryptographic methods are resource intensive, and the nodes of a WSN have limited resources. As a consequence, a significant number of works has investigated the cost of using security WSN protocols, and proposed solutions that trade off the level of security with energy consumption. Lee et al. analyze the computational cost of securing WSN communications by using different security techniques [38]. Their investigation, based on TelosB and MicaZ commercial motes, provide insights about which solution is suitable for which class of WSN applications, based on different energy costs and the specific application. Similarly, Wander et al. compare the energy expenditure and computation time of two public-key algorithms on wireless sensor nodes [62]. Based on their investigation, they show that in prevailing mote micro-controllers (such as the one of the Mica2dots mote) one method allows viable secure communication for its lighter computational requirements.

Whether for multi-media application for securing WSN communications, these works, and others of this nature, show that local computation can be a major source of energy consumption; it’s impact on network performance should not be neglected. This is also what we intend to investigate in this thesis for networks that use learning-based protocol design.

3

GreenRoutes: An End-to-End Energy-driven Route Selection for Green Wireless Networks

Forwarding strategies for green wireless networks need to be designed to draw benefits from the joint exploitation of wake-up radio technology and energy harvesting. In this chapter we describe GreenRoutes, a novel energy-aware cross-layer routing protocol for wake-up radio-based green networks. GreenRoutes is based on the idea of selecting as data forwarders (relays) the "best" neighboring nodes through the transmission of wake-up sequences. Each node is assigned two wake-up addresses. The first wake-up address is set to its distance (in wake-up radio hops) from the data collection point (the sink) and on a dynamically updated estimate of the energy available on the most recent route from that node to the sink. The second wake-up address corresponds to the node unique identifier (ID), according to some pre-set network naming.

We compare the performance of GreenRoutes with that of EHWA [40], a routing solution specifically designed for EH-WSNs (more details in Section 3.2.1.3), for varying parameters such as data traffic and energy harvesting sources, namely, solar and wind. Our GreenCastalia-based [15] simulation results show that, through the clever combination of energy harvesting and wake-up radio capabilities, GreenRoutes achieves a packet delivery ratio that is up to 40% more than that of EHWA, an end-to-end latency consistently lower than that of EHWA (up to 3.5 times lower), and an overall energy consumption that is a fraction of the energy spent by EHWA (up of over a half).

The remainder of this chapter is organized as follows. In Section 3.1 we describe the GreenRoutes protocol. The simulation tool, the chosen general parameters, and the baseline routing protocol, i.e., EHWA, used in our simulations, are introduced in Section 3.2.1. A comparative performance evaluation of GreenRoutes and EHWA is provided in Section 3.2. Finally, Section 3.3 concludes the work.

3.1 The GreenRoutes protocol

GreenRoutes is a cross-layer protocol, where each node that has a packet to transmit performs channel access and next-hop relay selection jointly. The selection of relay nodes is based on their distance (in hops) from the sink, and, greedily, on the energy available along routes to the sink. To describe the operations of our protocol in networks using wake-up radios with semantic addressing capabilities, we start by explaining how nodes determine their own wake-up addresses, and then we indicate the actions performed by a sender node to forward a data packet.

Wake-up address determination. The wake-up address w^i of a node i is obtained by juxtaposing the two binary sequences w_ℓ and $w_{\epsilon_\ell^i}$, representing the node hop distance from the sink $\ell \geq 1$, and an estimate of the energy ϵ_ℓ^i available on the most recently used route from node i to the sink. The hop distance ℓ is obtained through a sink-generated broadcast performed at the start of network operations. This hop count can be updated in time, depending on the dynamics of the network topology. The energy estimates ϵ_ℓ^i are computed by node i rounding the outcome of the following recursive equation.

$$\epsilon_\ell^i = \begin{cases} e^i & \text{if } \ell = 1 \\ \frac{e^i + \epsilon_{\ell-1}^j}{2} & \text{if } \ell > 1 \end{cases} \tag{3.1}$$

where $e^i \in \{0, \ldots, k\}$ is node i currently available energy, discretized into a set of $k+1$ values, and $\epsilon_{\ell-1}^j$ is the energy estimate from the node j used as relay for node i last forwarding. The estimate $\epsilon_{\ell-1}^j$ has been sent from node j to node i during the last data exchange (see details below). Computing energy estimates is computationally efficient, as Equation (3.1) unfolds into the sum of the terms of the following finite series:

$$\epsilon_\ell^i = \frac{e^i}{2} + \frac{e^{j_{\ell-1}}}{4} + \frac{e^{j_{\ell-2}}}{8} + \ldots + \frac{e^{j_2}}{2^{\ell-1}} + \frac{e^{j_1}}{2^{\ell-1}} \leq k. \tag{3.2}$$

We note that the contribution to ϵ_ℓ^i of nodes increasingly away from node i is exponentially decreasing. This aims at lowering the effects of possibly outdated energy information from far away nodes on node i current estimate. Overall, the wake-up address of node i so obtained, implements ways to choose nodes that are closer to the sink (the w_ℓ part of the address) jointly with a method to forward that packet through routes with the highest residual energy ($w_{\epsilon_\ell^i}$ component).

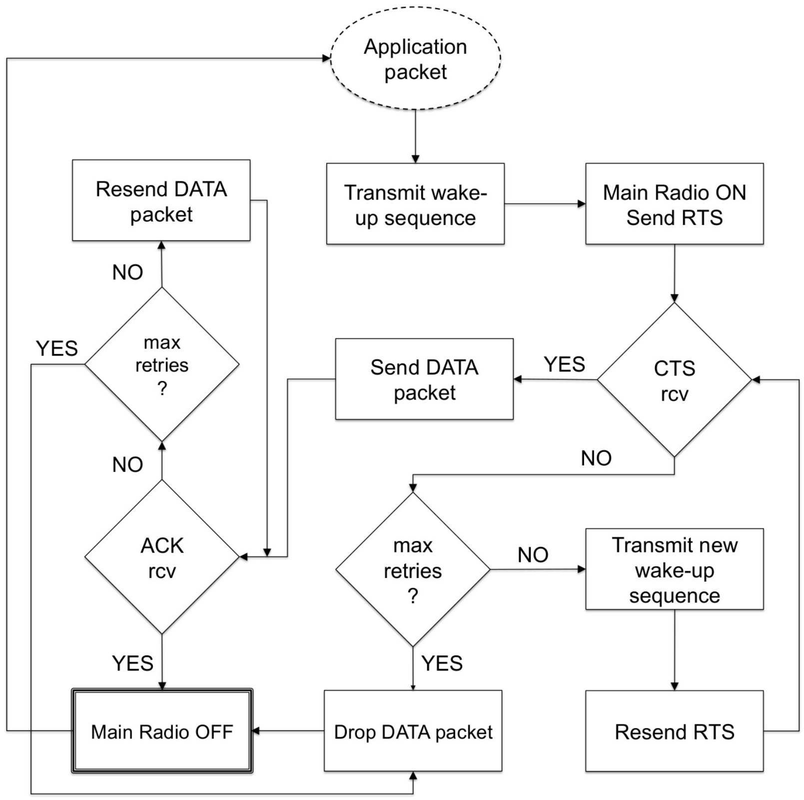


Fig. 3.1: Sender node i operations.

Data packet forwarding. When a node i that is ℓ hops away from the sink has to forward a packet, it transmits a wake-up sequence $w = w_{\ell-1}w_{\epsilon_{\ell-1}}$ aimed at waking up nodes whose hop count is $\ell - 1$ and that are part of a route with the highest possible energy $\epsilon_{\ell-1}$. To this aim, node i sets $w_{\epsilon_{\ell-1}} = k$. Then, it turns on its own main radio and transmits a request to send the packet among those nodes that it just woke up, if any. This is accomplished by transmitting an RTS packet. A newly woken up node j that has correctly received the RTS packet, awaits a certain amount of time δ_{e^j} and then transmits a clear-to-send (CTS) packet, declaring that it is available to forward the packet. This time is inversely proportional to node j current residual energy e^j, to allow node i to select the most energetic relay. Time δ_{e^j} is added with an extra small random delay for avoiding that nodes with the same residual energy send CTS packets at the same time. Node i forwards the data packet to the sender of the first CTS packet that it receives, ignoring subsequent CTS packets. Data packets that are not acknowledged by the intended receiver are re-transmitted up to a maximum number of times. If node i does not receive an acknowledgment for a data packet, the packet is discarded. Conversely, if no CTS packet has been received, node i broadcasts another wake-up sequence, this time trying to wake up those nodes whose energy level is $k - 1$. This process goes on until node i receives a CTS packet and a next hop relay j is found. After a predefined maximum number of retries is reached, if no relay is ever found, the packet is discarded. Diagrams illustrating the main operations of the senders and receivers are shown in Fig. 3.1 and Fig. 3.2.

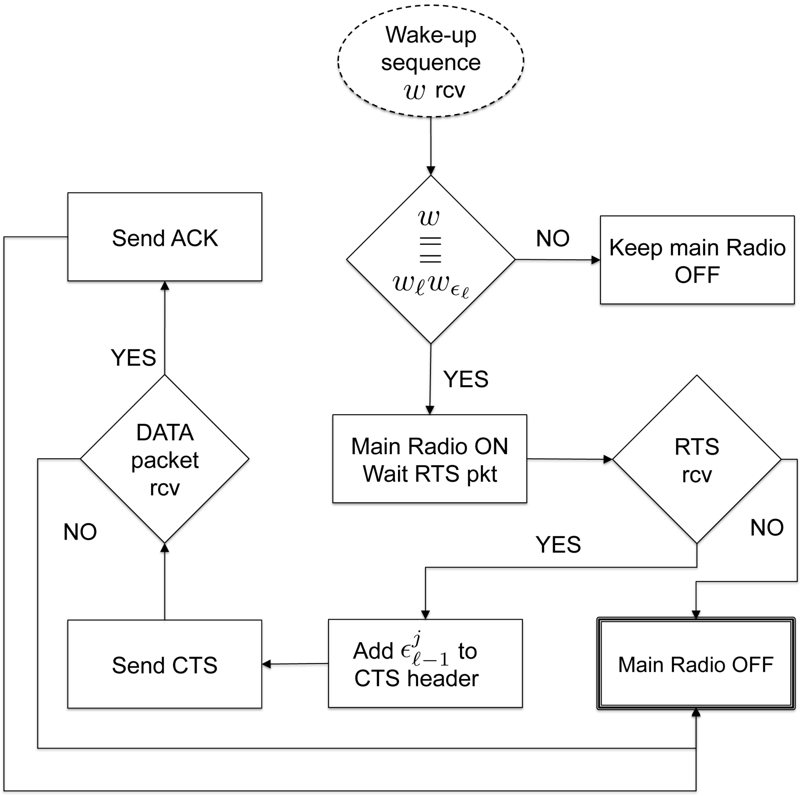


Fig. 3.2: Receiver node j operations.

We conclude the description of GreenRoutes with two notes. 1. The cross layer forwarding used by our protocol provides us with a practically costless way of updating neighboring nodes with energy estimates. Particularly, relay j includes in the CTS header the most updated value of $\epsilon^j_{\ell-1}$, which node i will use to update its one wake-up address. 2. The relay selection process can be time consuming because of the repeated RTS-CTS exchanges needed to find a relay j. Aiming to reduce this delay, and to further improve protocol efficiency, we stipulate that node i stores the ID of its last successful relay j for a predefined amount of time. All packets that node i needs to transmit within this time will be transmitted directly to j, without any new relay selection phase. In this case, node i will wake up node j directly, i.e., by using its ID as wake-up sequence.

3.2 Performance evaluation

In the following we start by describing the simulation scenario, the parameters, and the baseline protocol that are preliminaries to the simulation setup used in our performance evaluation. Investigated metrics are also introduced in this section. We conclude this section by introducing our simulation results. We note that preliminaries on simulation setup are used throughout the rest of this thesis, unless otherwise specified in subsequent chapters.

3.2.1 Simulation setup

All performance evaluation experiments were conducted using the open source GreenCastalia simulator [15]. GreenCastalia is an extension of the Castalia simulator [17] that is used on top of OMNeT++ to accurately model energy-related aspects of WSNs that benefit from energy harvesting technologies. We further extended its capabilities to model a prototype of wake-up radio [59].

3.2.1.1 Simulation scenarios and parameters

We consider connected networks where nodes are capable of sensing and of communicating wirelessly to each other. Node harvest energy from the environment using an external source, i.e., either via solar cells or via small wind turbines. The harvesting traces obtained from the National Renewable Laboratory at Oak Ridge [43]. The harvested energy is stored in a supercapacitor with a maximum operating voltage of 2.3V and a capacitance of 50F [25]. We decided for a batteryless network because of the beneficial features of supercapacitors, which offer long-lasting operation lifetime while retaining a high energy capacity level when compared to battery-operated networks [57]. Each node is equipped with on-board Sensirion SHT1x sensors to perform temperature measurements. The sensing power consumption is set to 3mW, and the completion time required by a measurement is set to 171ms [1]. Node-level sensing and processing is assumed to generate data packets following a Poisson process with inter-arrival time that varies depending on the considered scenario in each presented solution. Once a sensor measurement is taken, a data packet is generated that needs to be delivered to the sink. Among the sensor nodes a source node is randomly and uniformly chosen to generate a data packet. Data packets have a size of 58B, including the application payload (temperature measurements), and headers added by lower layers. The channel data rate is set to 250Kbps.

For the channel and radio models we use the default GreenCastalia settings. The transmission power of the main transceiver has been set to achieve energy conservation at -2dBm, leading to a transmission range R_m of 60m. The average path loss between two nodes is estimated using the log-normal shadowing model used in [55]. Packet collisions are determined using an additive interference model, by linearly summing-up at the receiver the effect of multiple signals simultaneously sent. We model the wake-up radio based on the specifications of the wake-up prototype and the experimental measurements presented in [59]. Each wake-up sequence is transmitted at 1Kbps and has a size of 1B. The energy model is that of the MagoNode++ mote, extended to comprise energy harvesting and wake-up radio capabilities [49]. This platform features the ultra-low-power CC1101 transceiver from the Texas Instruments [31], that allows transmission of the wake-up sequences at $+10$dBm. The wake-up receiver (WUR) features a maximum sensitivity of -55dBm with a wake-up range up to 45m. The power consumption of the WUR is set to 1.071μW. This model also considers the power consumption of the integrated ultra-low power microcontroller (MCU) used to perform wake-up addressing, which consumes 0.036μW and 54μW in idle and active states, respectively. The power consumption details used in our simulation scenarios are summarized in Table 3.1.

Tab. 3.1: Power consumption specifics.

	State	Value
Main Radio	Tx (-2dBm)	31.2mW
	Rx	33.6mW
Wake-up Radio	Wake-up Tx (10dBm)	90mW
	Wake-up Rx	1.071μW
MCU	Idle	0.036μW
	Active	54μW

3.2.1.2 Performance metrics

We evaluate the performance of our solution with respect to the following metrics. The following metrics are used in the rest of this thesis, unless otherwise specified.

1. The *packet delivery ratio* (PDR), i.e., the percentage of packets successfully delivered to the sink.
2. The *route length*, i.e., length of a route to the sink (in hop-count).
3. The *end-to-end latency*, defined as the time from packet generation to its correct delivery to the sink.
4. The *network energy consumption*, defined as the total amount of energy spent by all nodes to successfully deliver packets to the sink.

All results have been obtained by averaging the outcomes of a number simulation runs which obtains a 95% confidence with 5% precision. In order to evaluate steady-state performance, all metrics are collected after the initial network setup phase. In our results, the x-axis corresponds to the time between packets are generated over the simulation time.

3.2.1.3 A baseline routing protocol

We compare the performance of our solutions with the wake-up version of a routing solution specifically designed for EH-WSNs, namely, the Energy Harvest Wastage-

Aware (EHWA) protocol [40]. Specifically, EHWA is an on-demand dynamic source routing-based (DSR-based) protocol that implements a route selection scheme for wireless networks with energy harvesting. The aim of the strategy is that of minimizing the total energy wastage of the network. Wastage occurs when the capacity of the energy storage device reaches the maximum and further harvested energy cannot be stored. In EHWA each node is associated with its available energy, with a prediction of harvestable energy over a future period, and with an estimation of future energy consumption. A routing cost is assigned to each possible route between a source node i and the sink. The cost of a route is given by the sum of the energy consumed for transmission and of the energy wastage from both on-path and off-path nodes. On-path nodes are those that are part of the route from node i to the sink, while off-path nodes are nodes on other routes from node i to the sink. Once the sink has received information about all routes from node i, it selects the route that minimizes the energy wastage, and sends it back to node i. When node i, or any other node in the selected route, has a packet to forward it will send that packet through that route. Once a route is found, it is *cached* and it is used for a given period of time. In our simulations, EHWA has been extended to exploit wake-up radio capabilities where nodes are woken up based on their own ID.

3.2.2 Performance results

In this set of simulations, we consider connected networks where 64 nodes are randomly distributed as a 16×4 grid over an area of size $224 \times 56\text{m}^2$. The sink node is located at the bottom left corner of the deployment area. Network nodes harvest energy from the environment using the same external source, i.e., either via solar cells or via small wind turbines. We consider scenarios where only 70% of the nodes, randomly chosen, perform sensor readings and generate data packets with inter-arrival time in the range of $[1, 120]$ sec. Our simulation experiments are set to 4 days.

3.2.2.1 Packet delivery ratio

The packet delivery ratio of the two protocols for different harvesting sources is shown in Fig. 3.3a. In all scenarios, GreenRoutes clearly outperforms EHWA and consistently attains a packet delivery ratio higher than 92%. At the lowest traffic, GreenRoutes delivers approximately 1.4 times more packets than EHWA, regardless of the energy harvesting source. We notice that the performance of EHWA is decreasing with increasing traffic, while GreenRoutes exhibits a steady performance. The latter result depends on the clever selection of the next hop relay performed by GreenRoutes, which is lightweight and resource effective, waking up only those neighbors of a node that produce energy-efficient and shorter end-to-end

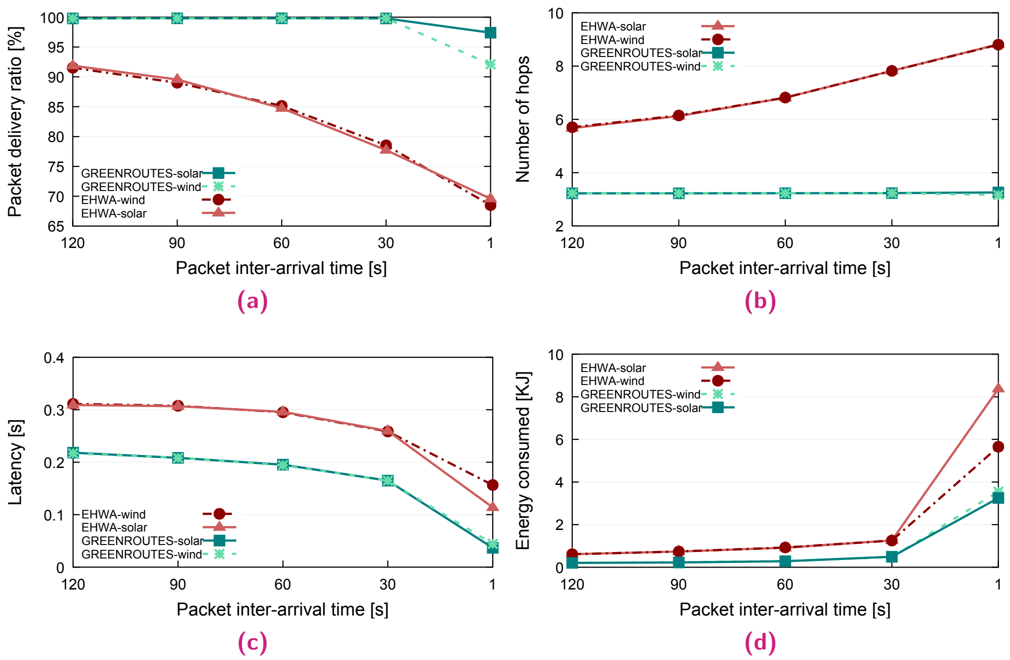


Fig. 3.3: Performance comparison of GreenRoutes and EHWA for increasing traffic.

routes. At the highest traffic, GreenRoutes achieves 28% higher packet delivery ratio than that of EHWA in the case of solar energy harvesting. This is because EHWA suffers from a high amount of both control and data packet transmissions, which results in higher interference. In particular, EHWA requires up to 10 transmissions to successfully deliver a data packet to the sink, while it takes an average of less than 4 times to deliver a packet using GreenRoutes. This clearly suggests that EHWA is not as "light" as GreenRoutes when it comes to packet transmissions, and its route selection mechanism often results in longer routes.

3.2.2.2 Route length

Fig. 3.3b shows the average number of hops that a data packet traverses to reach the sink. Both protocols obtain similar performance for both energy harvesting sources. GreenRoutes delivers a data packet using an average number of three hops. EHWA requires at least 1.7 times more nodes to successfully deliver data packets to the sink, depending on the traffic. This is because GreenRoutes chooses the next hop relay by considering nodes with smaller hop counts (i.e., closer to the sink), taking jointly into account the total available energy along the most recently used route. In EHWA the sink node chooses a route solely based on the total energy wastage. As a result, longer routes are often preferred to shorter ones.

3.2.2.3 End-to-end latency

The average end-to-end packet latency is shown in Fig. 3.3c. Independently of traffic

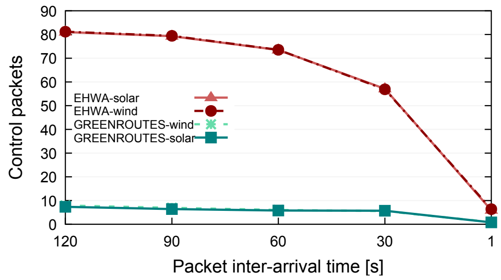
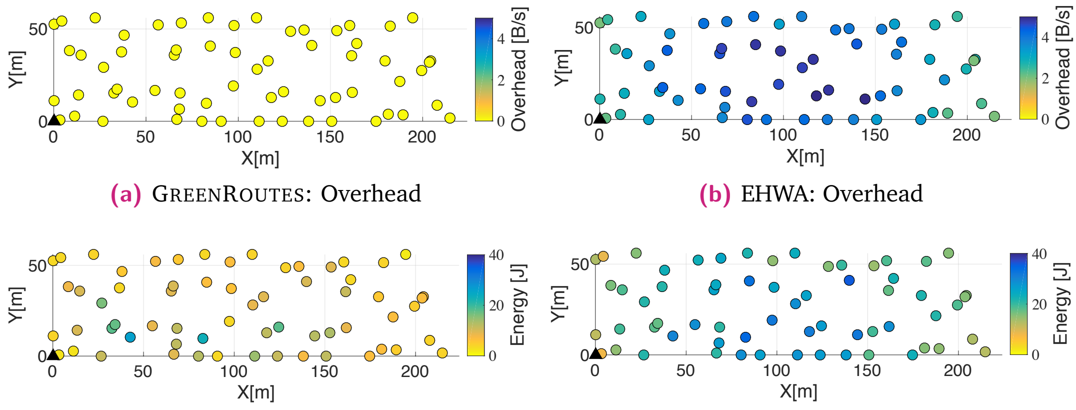


Fig. 3.4: Number of control packet transmissions normalized by the number of received packets.

and energy harvesting source, EHWA experiences higher latency, which can be up to 3.5 times higher than those experienced by GreenRoutes. This is because of the longer routes that packets travel to the sink and also because of the route selection mechanism of EHWA, which the sink performs prior to data packet transmission. Particularly, the sink needs to wait for a predefined time to gather information from the nodes; it then needs some time to compute suitable routes, and it finally needs further time to send the route information back to the nodes. We observe that end-to-end latency decreases with increasing traffic for both protocols. This is because they both make use of cached information, namely, a next-hop relay in the case of GreenRoutes, and full routes for EHWA. This eliminates a considerable amount of control packets, with beneficial effects on end-to-end latency, especially at high traffic (Fig. 3.4). Specifically, upon a successful packet transmission sender nodes store the information of the recently chosen next-hop relay and they directly transmit subsequent data packets to the cached relay without performing the next-hop relay selection procedure. When cached information is used, a sender node firstly transmits a wake-up sequence that includes the ID of the targeting relay, instead of broadcasting a wake-up sequence to wake-up nodes satisfying the criteria described earlier following the design of each forwarding strategy. We note that cached information is used for a given period of time and is updated once a new successful packet transmission occurs to adapt to any changes due to the dynamics of node and network status. As a result, sender nodes “take advantage" of cached information more frequently with increasing traffic as more data packets are transmitted using cached relays and without transmitting control packets leading to lower end-to-end latency.

3.2.2.4 Total energy consumption

Fig. 3.3d depicts the average total energy consumption incurred by the two protocols. Despite its higher packet delivery ratio, GreenRoutes always consumes less energy than EHWA. This is mainly due to the higher number of control packets that EHWA sends. The performance gap is more noticeable at the highest traffic, where EHWA consumes 61% more energy than GreenRoutes (solar harvesting case). We observe that EHWA consumes higher levels of energy in the case of solar harvesting vs.



(a) GREENROUTES: Overhead

(b) EHWA: Overhead

(c) GREENROUTES: Energy consumption

(d) EHWA: Energy consumption

Fig. 3.5: Per node overhead and energy consumption in networks with inter-arrival time of 30s and wind energy harvesting.

wind harvesting. This is consistent with the fact that packets take longer routes in scenarios with solar harvesting (Fig. 3.3b) due to higher energy wastage of routes. The performance of GREENROUTES is instead independent of the energy source, as its packets take shortest routes, and a limited number of control packets is needed to determine these routes. In general, GREENROUTES is more energy efficient in routing packets by jointly considering the distance of nodes from the sink, and, greedily, the residual energy along routes.

We further demonstrate the effectiveness of our approach by providing a quantitative assessment of the per node overhead and energy consumption of a sample topology (Fig. 3.5). The sink node is placed at the bottom left corner, depicted as a black triangle. Network nodes are depicted as circles. Our results concern nodes harvesting energy through a small wind turbine. Results about the overhead per time unit for GREENROUTES and EHWA are depicted through different colors in Fig. 3.5a and Fig. 3.5b. The darker the color the higher the overhead. We observe that EHWA nodes are colored in the shades of the darkest color, indicating higher levels of overhead, especially towards the center of the deployment area (Fig. 3.5b). This pattern is consistent with the behavior of a DSR-based protocol where nodes with a higher number of neighbors tend to receive and transmit a higher number of packets. Higher overhead leads to a higher energy consumption, as shown in Fig. 3.5c and Fig. 3.5d, where darker colors correspond to higher consumption. This affects the number of nodes that remain inactive for lack of energy, especially with high traffic. Specifically, at the highest traffic EHWA nodes remain inactive for a total of 52% of the simulation time. Nodes running GREENROUTES, instead, are inactive only for 7% of the time. In addition, GREENROUTES nodes are colored in lighter hues, showing lower overhead (Fig. 3.5a) and higher energy efficiency (Fig. 3.5c) throughout the network.

3.3 Conclusions

We presented GreenRoutes, a routing protocol for green networks that uses energy harvesting capabilities and wake-up radios with semantic addressing to efficiently select relays and routes with the best amount of residual energy. Through GreenCastalia-based simulations we compared the performance of GreenRoutes with that of EHWA, an energy wastage-aware routing protocol previously proposed. Results clearly show that GreenRoutes always outperforms EHWA with respect to every performance metric that we considered, regardless of traffic and of energy source considered, either sun or wind. In particular, GreenRoutes is able to deliver up to 40% more packets than EHWA to the sink, while consuming considerably less energy and delivering packets faster.

4 WHARP: A Wake-up Radio and Harvesting-based Forwarding Strategy for Green Wireless Networks

In this chapter we set to investigate how ambient energy can be judiciously managed to provide a data forwarding strategy that achieves high communication performance while maintaining nodes operative for the longest period of time. Our strategy, named Wake-up and HARvesting-based energy-Predictive forwarding (WHARP), leverages the combination of prediction-based techniques and Markov Decision Processes (MDP) to allow each node in the network to take pro-active forwarding and energy allocation decisions. Particularly, nodes take advantage of semantic addressing to wake up only those neighbors that can provide positive advances towards the the sink. Eventually, relay selection depends on the current and forecast energy at neighboring nodes, and on expected traffic.

The effectiveness of WHARP in providing energy efficient forwarding and long lasting node operations is demonstrated via simulation-based experimentation. We compare the performance of our solution to that of EHWA forwarding strategy [40], which was briefly introduced in Section 3.2.1.3. Performance results in scenarios with increasing traffic show impressive performance gains of WHARP over EHWA. In particular, network nodes running WHARP are able to deliver up to 72% more data packets than nodes running EHWA. Despite the remarkably higher packet delivery performance, WHARP consumes an average of 58% less energy than that consumed by EHWA. This makes nodes running WHARP operational for longer times (30%) than those running EHWA. We also observed that the smart selection of forwarder nodes makes WHARP effective in reducing data packet travel time, allowing packet delivery up to 1.6 times faster than EHWA.

The rest of the chapter is organized as follows. In Section 4.1 we describe notation and the networking scenario considered in this work. Section 4.2 describes WHARP in details. Performance evaluation results of WHARP and EHWA are shown in Section 4.3. Finally, Section 4.4 concludes the chapter.

4.1 Scenario and notation

This section introduces scenario and notation that are preliminary to the description of WHARP. We also provide background information on Markov Decision Processes (MDPs), a core component of our strategy.

Scenario. We consider a multi-hop wireless network made up of nodes statically deployed. Nodes are generically indicated as i and j. Each node is equipped with two wireless transceivers: (1) The main radio, which is used to transmit data and control packets. This radio consumes energy in the order of mWatts for receiving and transmitting information, and it is turned off unless needed. When off (*sleep mode*), the main radio consumes some three orders of magnitude less than when it is on (μW instead of mW). Main radios have a range which is usually in the tens of meters, e.g., 70m or up, as per prevailing technologies for wireless sensor motes. (2) A wake-up radio, which is used to wake-up (i.e., turn on) the main radio of selected neighboring nodes. This radio consumes energy in the order of mWatts for transmitting, and μWatts for receiving and in idle mode. It is usually always on. Wake-up radio transmitters send a wake-up sequence (or address) that is received by all nodes in (the wake-up radio) range. Only nodes that have that sequence as one of their wake-up addresses may decide to wake up; all other nodes remain with the main radio in sleep mode. For their operations nodes harvest energy from the surrounding environment (e.g., solar or wind energy) and store it in an energy storing device, e.g., a supercapacitor. There might be times when a node has not enough energy left for its operations (e.g., sensing, computation, communications, etc.). In this case the node turns off all its circuitry, and it is called an *all-off* node. It will restart its functions as soon as enough new energy has been harvested. Finally, nodes mount one or more sensors. The sensors produce data that is crafted into packets to be delivered to the network collector node, called *sink*. The architecture of a wake-up radio-enabled green node is depicted in Fig. 4.1.

Wake-up addresses. Each network node i takes two wake-up addresses. The first wake-up address is a binary sequence representing its distance $\ell_i \geq 1$ (in hops) from the sink. Hops are measured with respect to the wake-up radio range. For each node i, its hop distance ℓ_i is obtained through a broadcast started from the sink at the start of network operations. This hop count can be updated in time, depending on the dynamics of the network topology. The second wake-up address corresponds to the node unique identifier (ID), according to some set network naming.

A brief primer on MDPs. Markov Decision Processes (MDPs) provide a framework for modeling decisions that an *agent* can make in presence of system dynamics. Decisions lead to *actions* that are taken towards maximizing some notions of cumulative

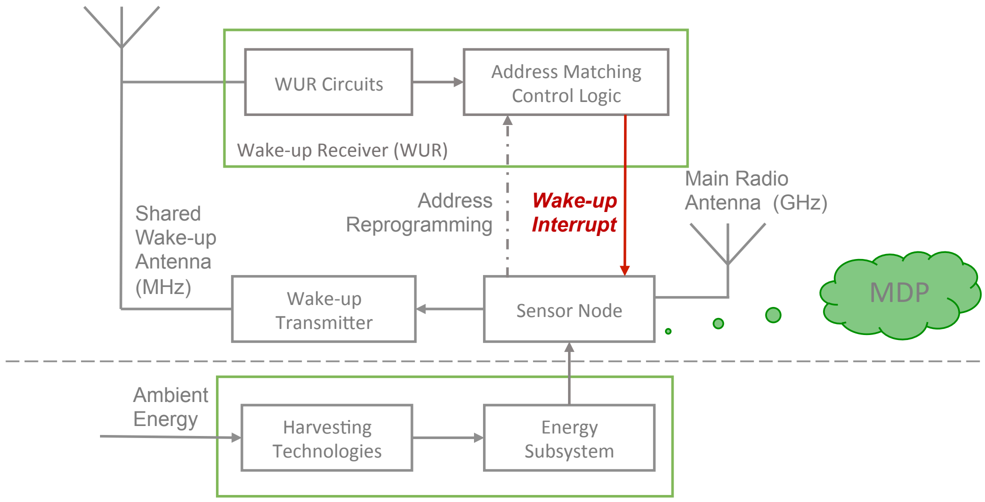


Fig. 4.1: The architecture of a green node.

reward. Given the set $\mathcal{S}$ of possible states of an agent, and the set $A(s)$ of the actions available at each state, a *policy* is a function π that associates to each state $s \in \mathcal{S}$ an action $a \in A(s)$, which is the action the agent should take to maximize the reward. We consider agents that follow a *discrete-time* model and make a decision every t_e time units. The time between two consecutive decisions is called *decision epoch*. The agent reward maximization problem over a finite horizon of N decision epochs, can be formalized as the following optimization problem, also known as a *Finite Horizon* MDP:

$$\max_{\pi} \; V_0^{\pi}(s) = E_s^{\pi}\left\{\sum_{n=0}^{N} \gamma^n r(s_n, a_n) \middle| s_0 = s\right\} \; \forall s \in \mathcal{S}, \tag{4.1}$$

where s_n and a_n are the system state and the action taken at the nth decision epoch, respectively, $r(s_n, a_n)$ is the expected reward associated to state s_n and decision a_n, and γ is the discount factor. The discount factor $0 \leq \gamma \leq 1$ models the uncertainty about the future: The farther the reward is in time, the least important it is. The $V_n^{\pi}(s)$ function is commonly known as the *value function*. It establishes how good it is for the agent to be in a given state at the nth decision epoch. Value functions are the means to solve the optimization problem of Equation (4.1) since, for each state $s \in \mathcal{S}$, the optimal policy π^* maximizing the value functions satisfies the Bellman optimality equations:

$$\begin{aligned} V_n^{\pi^*}(s) = \max_{a \in A(s)} \Bigg\{ & r(s_n, a_n) \\ & + \gamma \sum_{s' \in \mathcal{S}} P_{s_n \to s_{n+1}}^{a_n} V_{n+1}^{\pi^*}(s_{n+1}) \Bigg\}, \end{aligned} \tag{4.2}$$

where $P_{s_n \to s_{n+1}}^{a_n}$ is the transition probability from state s_n to state s_{n+1} after taking action a_n. Equations (4.2) state that the policy π^* that maximizes the reward depends on the immediate reward of taking action a_n from state s_n and on the expected discounted reward from the next state s_{n+1} onward. This is the power of MDPs: Optimal actions are taken depending also on future system states, and not only on the current configuration. We can solve the Bellman optimality equations using the Backward Value Iteration algorithm [54]. Its time complexity is $\mathcal{O}(N|A||\mathcal{S}|^2)$, which is linear in the number of decision epochs and actions, and quadratic in the cardinality of the state space.

4.2 WHARP forwarding

WHARP is a cross-layer strategy, where each node that has a packet to forward performs channel access and next-hop relay selection jointly. The selection of neighboring nodes is based on their distance (in "wake-up radio" hops) from the sink, and on their available energy.

When a node i with hop count ℓ_i has a packet to transmit, it broadcasts a wake-up sequence aimed at waking up its neighboring nodes with hop count $\ell_i - 1$. The wake-up sequence is followed by a Request-To-Send (RTS) packet transmitted using the main radio. On the receiving side, when a node j with hop count $\ell_j = \ell_i - 1$ receives a wake-up sequence from node i a decision is made about whether to turn on the main radio and start listening for an RTS, or to keep sleeping. This decision is based on a Markov Decision Process-based policy, whose details are provided in Section 4.2.1. If node j elects not to participate to the relay selection process, it simply keeps its main radio off. If instead the decision is that of turning on the main radio, node j starts waiting for an RTS packet. Upon receiving the RTS node j performs the following actions: (a) it computes a delay δ, (b) after that delay has passed, it sends a Clear-To-Send (CTS) packet to node i, and (c) turns its main radio to reception and awaits to receive the data packet. The delay δ is key to the efficient operation of WHARP as it provides an indication to node i of how suitable node j is to effectively forward packets towards the sink: The better a node is to be a relay, the shorter the delay. Details on the computation of δ are provided in Section 4.2.2. The sender i picks as relay the first node j from which it has received a CTS packet. Particularly, node i transmits the packet to node j directly, using its main radio. All nodes k, $k \neq j$, that sent a CTS but that do not receive a data packet within a set time period, or that overhear that the packet is being sent to node j, go back to sleep. After reception of the data packet, node j transmits an acknowledgment packet (ACK) to node i and goes back to sleep. Upon reception of the ACK packet node i also goes back to sleep. If node i does not receive an ACK from node j within a predefined time, it retransmits the data packet to node j till success, for at most K times. The data packet is dropped if all retransmission attempts fail.

4.2.1 An MDP-based model for relay selection

Every t_e time units (a *decision epoch*), or as soon as it restarts from an all-off state, each node i performs a computation whose output, either *green* or *red*, is used to decide whether node i should participate to a relay selection process or not. Particularly, for every wake-up sequence received in the current decision epoch, if the result of the computation is *green*, node i turns on its main radio and awaits for an RTS from the sender of the wake-up sequence. If the result is *red*, node i keeps its main radio in sleep mode, electing not to candidate itself as a forwarder. Clearly, we want the result of the computation to depend on the forecast available energy and on the energy that node i expects to consume for all its activities in the future. For this reason, this decision problem is modeled as a Markov Decision Process (MDP), which provides us with a framework to make decisions based on future system states (Section 4.1). In the context of our work an agent corresponds to a node, states represent energy, actions concern whether to forward packets or not, and the reward to be maximized concerns the time a node is on (i.e., capable of sensing) and able to participate to the network activities (i.e., forwards packets). The optimal policy to be determined is whether or not the node should be considered to be a WHARP forwarder (*green*) or not (*red*). In the following we provide details about all the ingredients of our MDP, and a way to compute the decision needed every time that node i receives a wake-up sequence.

States and actions. The state $s = b$ of each node is represented by its current energy level $b \in \{0, \dots, B_{max}\}$. State $s = 0$ denotes an all-off node. Actions concern the availability of a node to forward packets. Particularly, a_f indicates that the node is available to forward packets, and a_d indicates that the node will keep sleeping.

Transitions. We denote by h_n the energy harvested by a node in decision epoch n, $0 \leq n \leq N$. By b_n we indicate the energy level of the node in the nth decision epoch. We denote with e_n^x the t for sensing and for transmitting the corresponding data.[1] The overall energy available for packet forwarding in the nth epoch is thus

[1] Energy storing device leakage can be included in the computation of e_n^x.

$e_n = b_n + h_n - e_n^x$. Taking action a_n in state s_n transitions the node in epoch $n+1$ to the following state:

$$s_{n+1} = \begin{cases} e_n & \text{if } a_n = a_d \wedge b_n + h_n > e_n^x \\ e_n - e_n^{tx} & \text{if } a_n = a_f \wedge b_n + h_n > e_n^{tx} + e_n^x \\ 0 & \text{otherwise,} \end{cases} \tag{4.3}$$

where e_n^{tx} represents the energy spent to forward packets from other nodes. Specifically, a node transitions to a state where the energy is e_n if it chooses not to relay packets from neighboring nodes, but has enough energy for sensing operations and for transmitting its own packets. A node transitions to the state with energy $e_n - e_n^{tx}$ if it chooses to relay packets from neighboring nodes, and has enough energy for sensing operations, for transmitting its own packets and also packets from neighboring nodes. As expected the node dies (all-off) if the amount of energy is not sufficient to support sensing and/or transmission tasks: In this case the next state is $s_{n+1} = 0$.

We assume e_n^{tx} to follow some probability distribution $p^{e_{tx}}$, independently of the decision epoch. Conversely, we assume e_n^x to be equal to a constant value, i.e., $e_n^x = e^x$. Both probability distribution and value are constantly estimated by each node during its operation. The expected harvestable energy h_n is assumed to be known by means of some form of energy predictors, e.g., ProEnergy [18] or AEWMA [42].

When a node chooses action a_f in state s_n it transits to state s_{n+1} according to the probability law $P^{a_f}_{s_n \to s_{n+1}}$ defined as follows:

$$P^{a_f}_{s_n \to s_{n+1}} = \begin{cases} p^{e_{tx}}(e_n^{tx}) & \text{if } b_{n+1} > 0 \\ \sum\limits_{e_n^{tx} = e_n}^{\infty} p^{e_{tx}}(e_n^{tx}) & \text{if } b_{n+1} = 0. \end{cases} \tag{4.4}$$

If a node is not all-off (i.e., $b_{n+1} > 0$), then the transition probability coincides with the probability $p^{e_{tx}}(e_n^{tx})$ of consuming energy for forwarding packets from other nodes. Otherwise, the transition probability corresponds to energy consumption exceeding the node capability of forwarding packets. When the node chooses a_d, the next state is uniquely identified by h_n and e_n^x, and $P^{a_d}_{s_n \to s_{n+1}} = 1$.

Reward function. In an MDP approach, the behavior of the agent resides in the structure of the reward function $r(s_n, a_n)$. In the context of our work, a node should be available to forward packets, and should also remain awake as much as possible to keep sensing: Two contrasting goals. Therefore, our model should reward the node each time it chooses $a_n = a_f$, but should also penalize it when it dies. Specifically, when $a_n = a_f$ the reward function is defined as:

$$r(s_n, a_f) = r \cdot \sum_{e_n^{tx}=0}^{e_n} p^{e_{tx}}(e_n^{tx}) - c \cdot \sum_{e_n^{tx}=e_n}^{\infty} p^{e_{tx}}(e_n^{tx}), \tag{4.5}$$

where r is the positive reward that node i receives if it does not run out of energy in the current decision epoch, and c is the cost it incurs instead if it dies. In Equation (4.5), parameters r and c are weighted by the probability that the energy consumption in a decision epoch is respectively lower and higher than the available energy to forward packets. If $a_n = a_d$, the agent will get the reward $r(s_n, a_d) = 0$. We do not penalize a node if it dies transmitting its own data.

Solution method. The definitions above allow us to finally formulate the MDP Bellman Equations (4.2) for computing the optimal value functions $V_n^{\pi^*}$ and, in turn, the optimal policy π^*, i.e., either *green* or *red*. The solution of the Bellman equations is performed by using the Backward Value Iteration algorithm [54], a standard solution method for MDPs. By judiciously keeping the model simple and by choosing suitable time horizons and state space size, we can make the MDP efficiently solvable in practically any device.

4.2.2 Calculation of the CTS delay δ

Whenever node i sends an RTS, each neighboring node j that has elected to participate to the relay selection process replies with a CTS after a delay δ computed as follows:

$$\delta = (1 - \frac{b_j}{B_{max}}) \cdot \delta_{\text{MAX}} + \delta_{\text{RAND}}, \tag{4.6}$$

where b_j is node j current energy, δ_{MAX} is the local maximum possible delay, and $\delta_{\text{RAND}} < \delta_{\text{MAX}}$ is an extra small random delay used to avoid collisions of CTS packets at the sender. Considering the randomness of the delay δ_{RAND} and to ensure that nodes with higher energy always transmit a CTS packet before those with a lower level of energy, we consider that the maximum possible delay δ_{MAX} is computed based on a global maximum possible delay δ_{GM} and the local delay δ_{RAND} as follows:

$$\delta_{\text{MAX}} = \delta_{\text{GM}} - \delta_{\text{RAND}}. \tag{4.7}$$

In other words, the higher the energy at a node, the lower its delay in replying to the sender, and therefore the higher its chances to be selected as a relay.

We conclude the description of WHARP with an implementation note aimed at improving performance. The relay selection process can be time consuming because of the repeated RTS–CTS exchanges needed to find a relay j. Aiming to reduce this delay, we stipulate that node i stores the ID of its last successful relay j for a predefined amount of time. All packets that node i needs to transmit within this time will be transmitted to j, without any new relay selection phase. In this case, node i will wake up node j directly, i.e., by using its ID as wake-up sequence. We showcase an example of the WHARP forwarding strategy in Fig. 4.2.

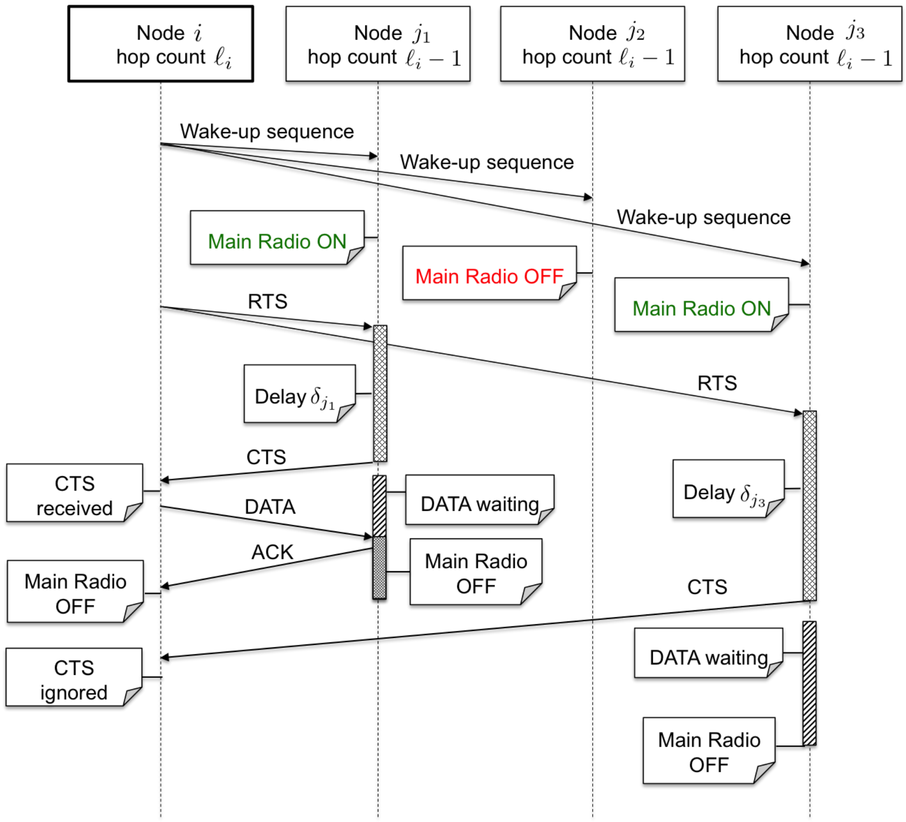


Fig. 4.2: WHARP forwarding: An example.

Node i, with hop count ℓ_i, has a packet to transmit. Nodes j_1, j_2 and j_3 are within its wake-up radio range, with hop count $\ell_i - 1$. Node i broadcasts a wake-up sequence to wake up its neighboring nodes j_1, j_2 and j_3. Nodes j_1 and j_3 get a *green* as a result of running the MDP, and turn their main radio on, awaiting the RTS packet. Node j_2 decides not to participate to the relay selection process and keeps its main radio off. Upon reception of the RTS nodes j_1 and j_3 compute the CTS delays δ_{j_1} and δ_{j_3}, respectively. Once the CTS delay has passed, both nodes reply with a CTS packet to sender i and activate the data packet waiting timer. Node i transmits the data packet to the node that transmitted the CTS first, i.e., node j_1 in our example. After reception of the data packet, node j_1 replies with an ACK packet and turns off its main radio. Node i ignores the subsequent CTS from node j_3, and goes back to sleep after receiving the ACK from node j_1. As node j_3 does not receive the data packet within the set waiting time it goes back to sleep.

4.3 Performance evaluation

We evaluate the effectiveness of the WHARP forwarding strategy by simulation-based experiments. We also compare its performance to the EHWA [40] baseline strategy for data forwarding introduced in Section 3.2.1.3. The main simulation parameters are described in Section 3.2.1. We consider that 119 sensor nodes are randomly and uniformly distributed over a square area of size $200 \times 200\text{m}^2$. The sink is statically placed at the top right corner of the deployment area. We consider sensor nodes with different harvesting capabilities. Particularly, half of the sensor nodes harvest energy using solar cells; the remaining 60 sensor nodes use micro wind turbines. Once a sensor measurement is taken, a data packet is generated that needs to be delivered to the sink. In our simulation results, we make use of the inter-arrival time between packets which ranges from 1 to 150 seconds. All results have been obtained by averaging the outcomes of 100 simulation runs, each of duration T_s of 3 days. In scenarios with varying data traffic, we compare the two data forwarding strategies with respect to the following key performance metrics: Packet delivery ratio, end-to-end latency, and total energy consumption, as defined in Section 3.2.1.2. The simulation parameters are summarized in Table 4.1, which also shows the values chosen for WHARP-specific parameters.

4.3.1 Simulation results

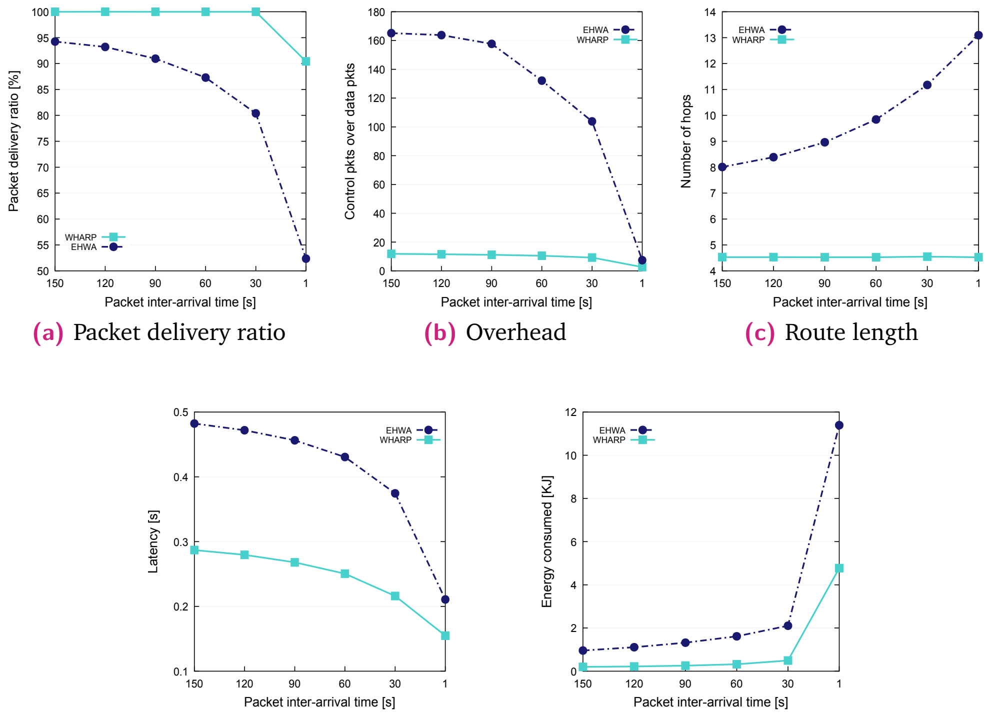
4.3.1.1 Packet delivery ratio

Fig. 4.3a shows the average PDR for increasing traffic. WHARP clearly outperforms EHWA as it always delivers more than 90% of packets, regardless of the traffic load. Conversely, the PDR performance of EHWA decreases abruptly as the traffic increases. At the highest traffic, WHARP delivers approximately 70% more packets than EHWA. The performance improvement depends on the smarter forwarding strategy enacted by WHARP: Senders only awake those neighbors that are closer to the sink and, among these, they select relays based on forecast energy and expected traffic. Other reasons that explain the superior performance of WHARP include the following. (i) Lower overhead. Fig. 4.3b shows the average number of control packets generated

Tab. 4.1: Simulation parameters.

	Definition	Value
T_s	Simulation duration	3d
M	Number of nodes	120
-	Deployment area size	200×200m^2
-	Capacitance of supercapacitor	50F
-	Supercapacitor max operating voltage	2.3V
$iaTime$	Inter-arrival time	$[1, 150]$ s
R_m	Main radio range	60m
r_c	Channel data rate	250Kbps
R_w	Wake-up radio range	45m
r_w	WUR sequences rate	1Kbps
-	Sensing power consumption	3mW
-	WUR power consumption	1.071μW
-	MCU power consumption (idle)	0.036μW
-	MCU power consumption (active)	54μW
T_c	Expiration of cached routes	200s
δ_{MAX}	Maximum CTS delay	75ms
δ_{RAND}	Extra random CTS delay	[0, 10ms]
N	Number of decision epochs	10
t_e	Decision epoch length	720s
γ	Discount factor	0.9
K	Max data packet retransmissions	10
-	Energy predictor	AEWMA [42]

by WHARP and EHWA, normalized to the total number of generated data packets. We observe that WHARP generates up to 14 times less control packets than EHWA, except at higher traffic, when EHWA reaps the advantages of route caching. The lower number of control packets generated by WHARP imposes a lower number of interference among packets, and a lower number of re-transmissions (up to 1.4 times less), and therefore a higher PDR. (ii) Lower route lengths. Fig. 4.3c depicts the average lengths of routes found by WHARP and EHWA. We observe that being based on hop distance, the average route length of WHARP routes is independent of traffic. Instead, EHWA nodes can send packets to nodes away from the sink, where less wastage occurs, thus finding longer routes. In fact, EHWA routes are almost two to three times longer than WHARP routes, especially at higher traffic, where nodes tend to be all-off more frequently. Shorter routes mean a lower number of packet transmissions, thus lower interference, and therefore a higher PDR. (iii) Lower number of all-off nodes. We observed that, on average, WHARP nodes are operational for 98% of the time, i.e., for 30% more time than nodes running EHWA (see also Fig. 4.4). A higher number of active nodes results in a higher number of available relays and, ultimately, in higher packet delivery ratio.



(a) Packet delivery ratio (b) Overhead (c) Route length

(d) End-to-end latency (e) Network energy consumption

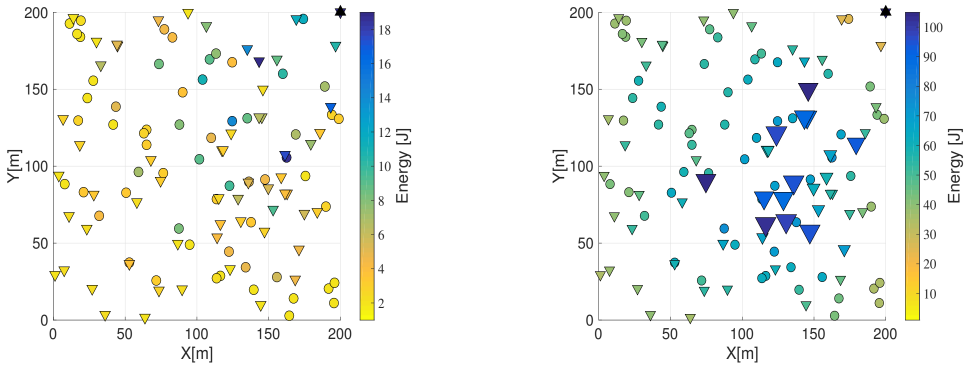
Fig. 4.3: Performance comparison of WHARP and EHWA for increasing traffic loads.

4.3.1.2 End-to-end latency

The average packet end-to-end latency is shown in Fig. 4.3d. Despite WHARP delivers significantly more packets than EHWA, it achieves a per packet latency that is up to 1.6 times lower than that incurred by EHWA. This is because, as noticed while discussing the PDR performance, EHWA packets travel significantly longer routes (see Fig. 4.3c). More important, the higher latency is also due to the sink-centered nature of EHWA, for which the sink must collect information from all nodes, compute routes, and distribute route back to the nodes before packet transmission. The relay selection strategy of WHARP is instead on-the-fly and hop-by-hop, eliminating the time needed for establishing a whole source-to-sink route. The performance of both protocols gets better with increasing traffic because a higher number of packets takes advantage of the caching of next hop relays (WHARP) and of routes (EHWA), which expedites packet forwarding (see Chapter 3.2.2.3).

4.3.1.3 Network energy consumption

Fig. 4.3e shows the average energy consumed by the network. Independently of traffic WHARP always outperforms EHWA, despite its significantly higher packet delivery ratio. The reasons are the same we highlighted for the previous metrics: EHWA has a higher overhead (Fig. 4.3b), which imposes a longer use of the main radio, and uses longer routes (Fig. 4.3c), which requires a higher number of packet transmissions on the main radio. The higher energy consumption of EHWA is also consistent with the fact that its nodes are all-off for up to 11 times more than WHARP nodes (Fig. 4.4). The performance gap increases with traffic. At the highest traffic, WHARP consumes 58% less energy than EHWA. In order to further demonstrate the



(a) WHARP (b) EHWA

Fig. 4.4: A joint snapshot of the all-off time and energy consumption per node: WHARP vs. EHWA.

effectiveness of WHARP in managing smartly the harvested energy, we show the all-off time and energy consumption of each node of a sample topology. Fig. 4.4 shows a network of 119 nodes plus the sink, depicted as a black star at the top right corner of the square deployment area. Nodes that harvest energy using solar cells are depicted as circles, while triangles correspond to nodes that use wind as their harvesting source. The size of each node is proportional to the total time the node run out of energy (all-off). Nodes that are operational for a longer period are displayed with smaller sizes. The color of each node indicates its energy consumption. The darker the color, the higher the energy consumed. The remarkable difference of range of the bar at the right of Fig. 4.4a (from 0 to 18) and Fig. 4.4b (from 0 to 100) reflects the remarkable difference in energy consumption between WHARP and EHWA, respectively. We observe that WHARP forwards packets in a "funnel" fashion as packets travel only to nodes that are closer to the sink. As a result, nodes closer to the sink consume more energy than other nodes further away (Fig. 4.4a, upper right). EHWA nodes forward packets to every neighboring node. As a result, EHWA nodes that are placed at the center of the square receive more packets that those toward the perimeter of the deployment area. This explain the higher levels of energy consumption for central nodes (Fig. 4.4b, center). Fig. 4.4 also highlights that using EHWA nodes are operational for less time. In fact, the color of nodes at the center is on the darker level, and their size bigger, which indicates the node is all-off for longer. We observe that the less operational nodes are those that use wind turbines. This is because of the lower amount of energy that a node can harvest using wind turbines compared to solar cells.

4.4 Conclusions

This chapter presented WHARP, a forwarding strategy for green wireless networks enabled by wake-up radio and energy harvesting capabilities. WHARP forwards data to the destination by making MDP-aided forwarding decisions based on forecast energy and expected traffic load, optimizing system performance over time. We compared the performance of WHARP with that of EHWA, a state-of-art energy wastage-based forwarding strategy through GreenCastalia simulations. Results show that the proposed strategy widely outperforms EHWA with respect to all considered performance metrics. Particularly, it consumes up to 58% less energy than EHWA while delivering significantly more packets (up to 72%) with an end-to-end latency up to 1.6 times lower. This allows nodes using WHARP to be operational for 98% of the time: A 30% improvement over EHWA.

5 A Comparative Performance Evaluation of Wake-up Radio-based Data Forwarding for Green Wireless Networks

This chapter concerns the comparative investigation of the performance of three data forwarding strategies in green wireless networks. We are concerned with recently proposed solutions that have been designed to take full advantage of both technologies of green networking, namely, energy harvesting and wake-up radios with semantic awakenings. Our investigation is aimed at providing insights about different forwarding design choices and their consequences on network performance.

The first forwarding strategy, named CTP-WUR [13], is built upon the Collection Tree Protocol (CTP) [28], re-designed to take advantage of wake-up receivers (WURs). Forwarding decisions are made by following a pre-built minimum-cost tree rooted at the sink, a somewhat classical solution for routing in WSNs. CTP-WUR takes advantage of wake-up radios by allowing the relay of wake-up requests, thus saving on main radio usage, especially for sending control packets. The other two data forwarding strategies, namely GreenRoutes and WHARP, are the two cross-layer approaches presented in Chapter 3 and Chapter 4, respectively. In GreenRoutes next-hop selection depends on the distance of a node from the sink and on the residual energy available along recently used routes to the sink (end-to-end energy awareness) [8]. In WHARP energy prediction techniques are used jointly with a Markov Decision Process (MDP) to allow each node to decide whether to make itself available for data forwarding or not [9].[1]

All three data forwarding solutions have been implemented in GreenCastalia [15]. Our results show that in general cross-layer approaches relying on contention-based mechanisms for relay selection suffer from higher delays because of the per packet RTS/CTS-like handshakes, and incur high energy consumption as the contention itself involves waking up multiple potential next-hop relays. Proactive approaches

[1] To the best of our knowledge, there are no other solutions explicitly designed for green wireless networks as defined in this work, i.e., exploiting both energy harvesting and wake-up radios. Performance comparisons with data forwarding for WSNs with no energy harvesting and/or no wake-up radio capabilities would be scarcely informative [8, 9].

like CTP-WUR are faster and lighter, because the next-hop relay is pre-determined. However, they show their limitation when nodes black out, which results in packet loss due to the lack of timely topology updates. Approaches that employ sophisticated learning models, like WHARP, successfully deal with nodes that run out of energy by implementing forwarding policies that penalize selection of relays on routes with nodes blacked out. We conclude also that the use of mechanisms to directly forward data packets to a known and already used next-hop relay, as in GREENROUTES and WHARP, can extremely decrease end-to-end latency. In fact, this latency is observed to be comparable to that incurred by strategies like CTP-WUR that do not require intensive use of control packets to gain channel access for data packets.

The remainder of this chapter is organized as follows. In Section 5.1 we provide a summary of the operations of the data forwarding strategies compared in this chapter. Section 5.2 presents and discusses the comparative performance evaluation results. Finally, Section 5.3 concludes the chapter.

5.1 Data forwarding strategies for green wireless networks

This section provides a summary of the three forwarding strategies considered in this chapter. The three solutions are described for green wireless networks made up of sensor nodes that are statically deployed and that can communicate wirelessly. Data packets may reach their final destination (the sink) through several nodes, i.e., routes can be multi-hop. Each sensor node is equipped with a pair of transceivers: 1) The main radio used for control and data packets, and 2) the wake-up radio used for waking up neighboring nodes to eventually select the next-hop relay for the data packet. Nodes are provided with at least one wake-up address, which is a binary sequence whose meaning depends on the design specifications of the forwarding strategy. Wake-up addresses can be updated in time, depending on nodes status and network dynamics.

5.1.1 CTP-WUR

CTP-WUR [13] is a wake-up radio-based data forwarding strategy built upon the widely used Collection Tree Protocol (CTP) [28]. The latter, as the name suggests, delivers data to the sink by building a tree structure that provides pre-determined routes for the data packets. CTP-WUR takes advantage of the tree-based routes for relaying wake-up sequences through intermediate nodes, so to wake up the grandparent of the sender node. The intermediate node, i.e., the parent of the

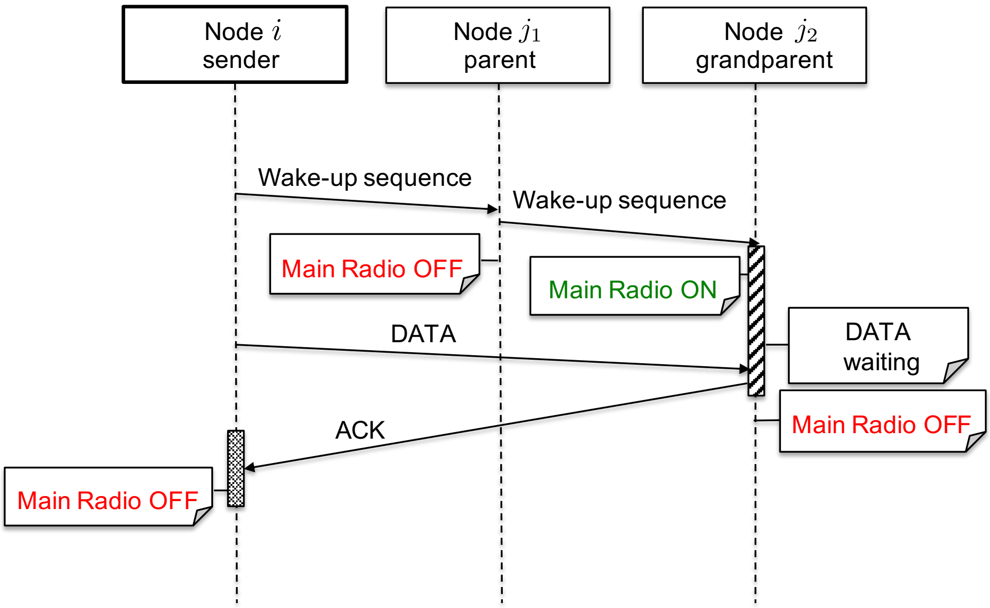


Fig. 5.1: CTP-WUR forwarding.

sender, is only responsible of relaying to the next-hop (the grandparent) the wake-up sequence without activating its main radio. Then, a sender node can transmit the data packet directly to the grandparent, saving considerable energy. The wake-up addressing mechanism of CTP-WUR requires each node to be given two addresses: 1) A broadcast wake-up address for activating its main radio when a control packet needs to be transmitted (this is used to broadcast beacons for updating the network topology tree in time), and 2) a wake-up address that consists of its unique identifier (for unicast routing). An example of the CTP-WUR forwarding strategy is shown in Fig. 5.1. When a sender node i has a data packet to transmit, it sends a wake-up sequence aiming to notify its parent, i.e., node j_1, that it has to wake up its own parent, node j_2 (which is node i grandparent). Node j_1 receives the wake-up sequence and understands that it is a relaying node by checking a flag bit. If the flag bit is activated, then node j_1 understands that it is an intermediate node and keeps its main radio in sleep mode. Then node j_1 sends a wake-up sequence to its parent node j_2 on its wake-up radio. Upon reception of the wake-up sequence node j_2 activates its main radio and awaits for data reception from node i. Node i transmits the data packet to node j_2 and awaits an acknowledgment (ACK) before going back to sleep. After receiving the data packet node j_2 sends an ACK back to node i and after that it turns off its main radio.

5.1.2 GreenRoutes

GreenRoutes is an end-to-end energy-aware and wake-up radio-based routing protocol specifically designed for green wireless networks [8]. Nodes select the next-hop relay for their packets based on the energy available along routes recently used to forward packets to the sink. An exchange of RTS/CTS packets à la IEEE 802.11 is used for channel access prior to the transmission of each data packet. At the start of the network operations, the sink node initiates a broadcast through which each node in the network acquires its distance from the sink (in wake-up hops). GreenRoutes takes advantage of semantic wake-up addressing to wake up only those neighboring nodes which are closer to the sink and that are the "best" in terms of available energy. Specifically, each node creates a wake-up address by juxtaposing two binary sequences: 1) One representing its hop distance, and 2) one representing an estimate of the available energy on the most recently used route to the sink. An example of the data forwarding operations in GreenRoutes is shown in Fig. 5.2. The figure refers to a sender node i that is ℓ_i hops away from the sink.

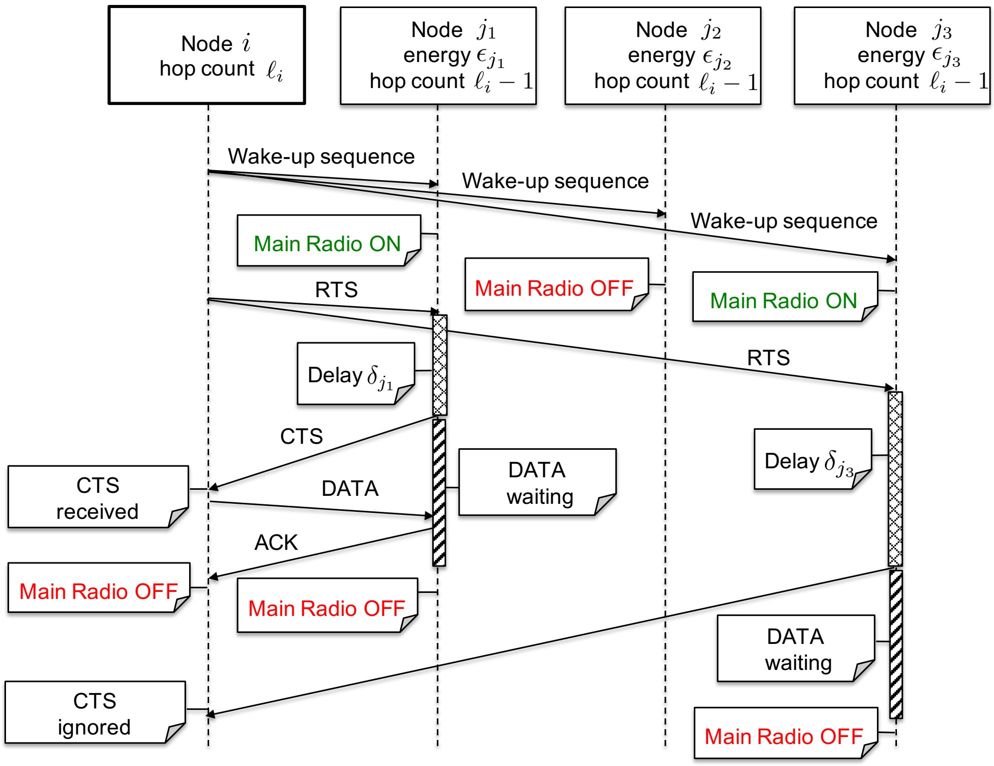


Fig. 5.2: GreenRoutes forwarding.

Node i has three neighbors j_1, j_2 and j_3 that are one hop closer to the sink. At the time when node i has a packet to transmit these three neighbors provide packet advancement on routes with available energy ϵ_{j_1}, ϵ_{j_2} and ϵ_{j_3}, respectively. Node i broadcasts a wake-up sequence to wake up those among neighbors j_1, j_2 and j_3

that are part of a route with available energy $\geq \epsilon_{\ell_i-1}$. Here we assume that only ϵ_{j_1} and ϵ_{j_3} are both $\geq \epsilon_{\ell_i-1}$. Therefore, only nodes j_1 and j_3 turn on their main radio. Node i transmits an RTS packet to j_1 and j_3 (on the main radio). Both nodes reply with a CTS packet after a time that is inversely proportional to ϵ_{j_1} and ϵ_{j_3}.

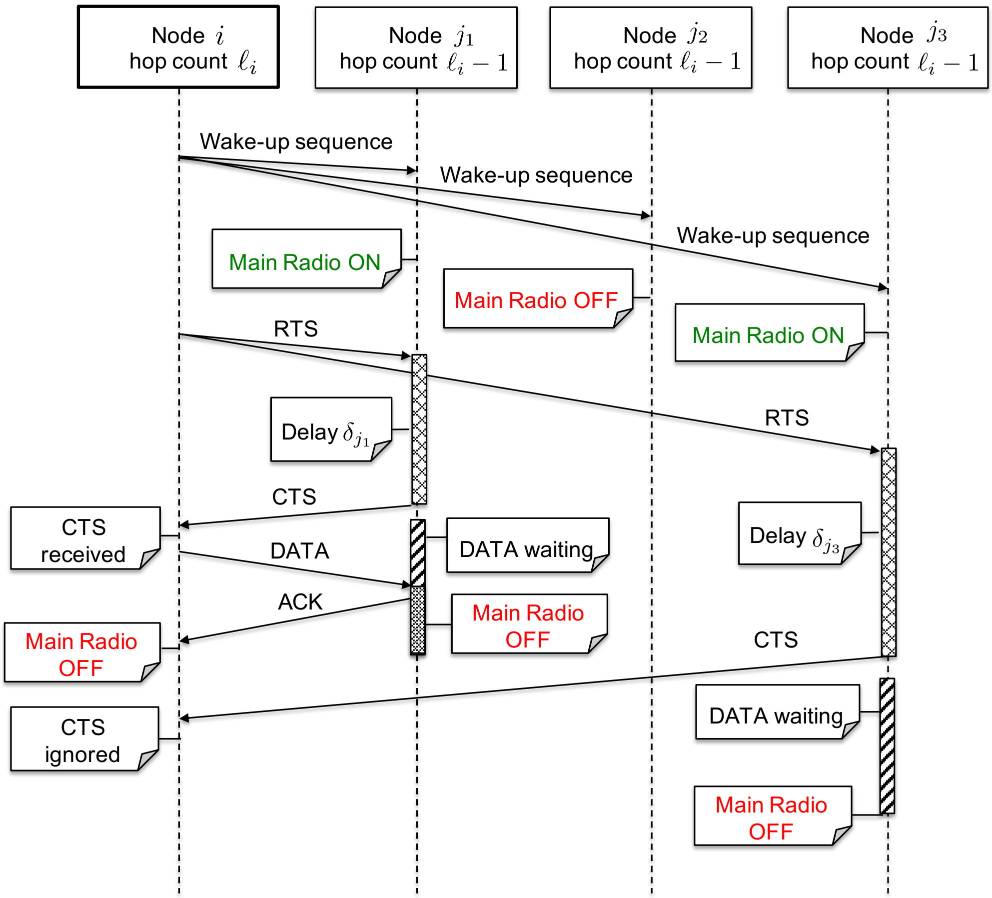


Fig. 5.3: WHARP forwarding.

The node with the highest available energy transmits the CTS packet earlier (node j_1 in the figure). Node i sends the data packet to node j_1, awaits an ACK and then turns its main radio off. Upon reception of the data, node j_1 acknowledges the packet and goes back to sleep. All other awaken nodes, i.e., node j_3, which do not receive the data packet, turn off their main radio after a predefined amount of time.

5.1.3 WHARP

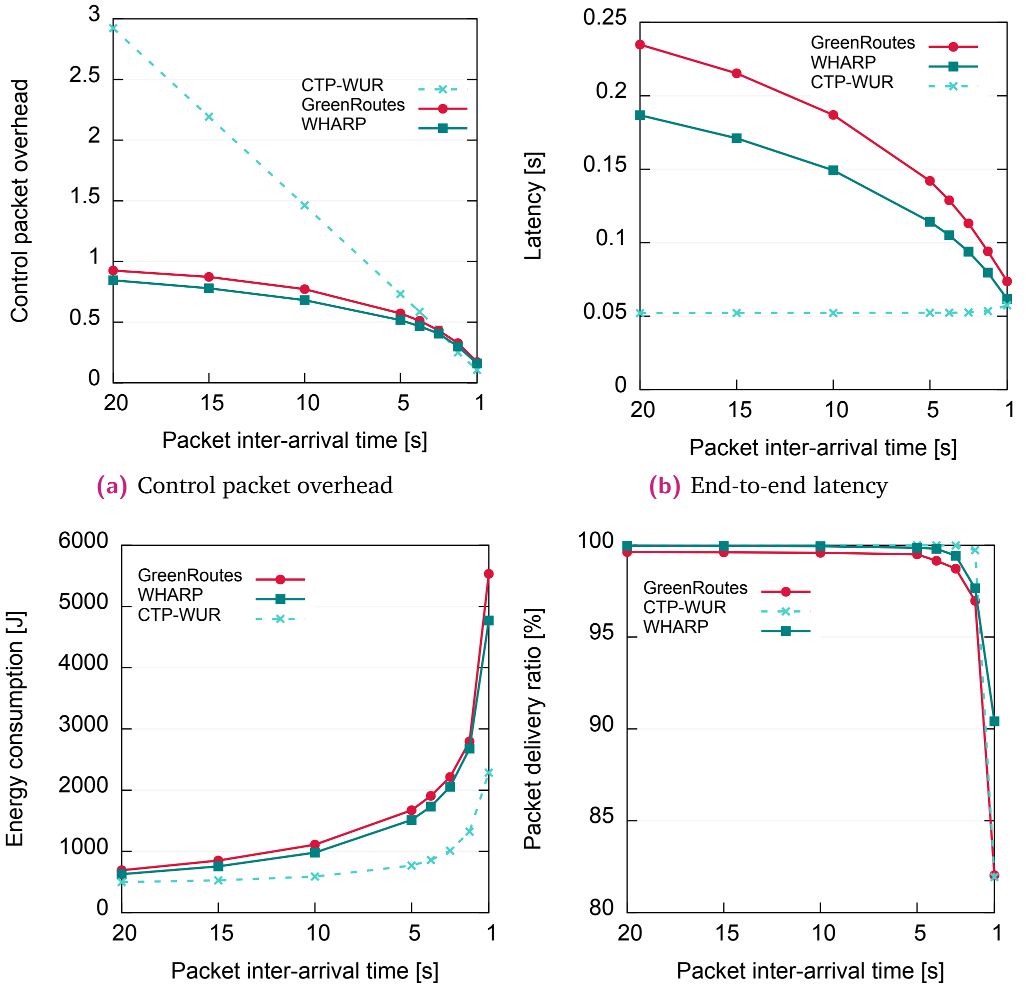
WHARP is a cross-layer forwarding strategy where channel access (MAC layer) and selection of the next-hop relay (network layer) are performed jointly [9]. A node decides whether to participate in the relay selection process based on its current capability to forward packets. Particularly, the decision to participate (or not) is based on a proactively computed Markov Decision Process-based policy. At the start of the network operations, the sink node initiates a broadcasting procedure through which each node in the network acquires its distance from the sink (in wake-up hops). This distance becomes the node wake-up address. Fig. 5.3 showcases an example of the WHARP forwarding strategy. In this example, node i, which is ℓ_i hops away from the sink, has a packet to transmit. Nodes j_1, j_2, and j_3 are neighboring nodes of node i that are one hop closer to the sink, and therefore, they are potential next-hop relay candidates. Node i broadcasts a wake-up sequence to wake up those among them that could provide "good" forwarding to the sink. The recipients of this wake-up sequence will wake up according to the result of running an MDP that takes into account the node residual energy and a measure of the harvestable energy that will be available in the near future. In particular, the process is capable of discouraging the selection of nodes that could black out shortly. In this example, we stipulate that nodes j_1 and j_3 decide to wake up and to participate to the relay selection process. Node j_2 instead decides not to participate, ignoring the wake-up sequence. Upon reception of the wake-up sequence nodes j_1 and j_3 activate their main radio and await for an RTS packet from node i. They then compute an energy-dependent delay, and after this delay they transmit a CTS packet to node i. In this example, node j_1 transmits the CTS packet before node j_3, thus winning the competition: It will be the recipient of the data packet. Once node j_1 receives the data packet, it replies with an ACK and switches back to sleep mode. Node i ignores any subsequent CTS packets and turns off its main radio. Node j_3, which did not receive a packet, eventually turns off its main radio after a predefined amount of time.

5.2 Performance evaluation

In this section we compare the performance of CTP-WUR, GreenRoutes and WHARP. We consider networks with 119 nodes that are randomly and uniformly positioned in a 200m by 200m area. The sink is located at the upper right corner of the area with a packet inter-arrival time ranging in the set $\{20, 15, 10, 5, 4, 3, 2, 1\}$, corresponding to traffic from low (inter-arrival time of 20s) to medium/high (1s). The size of the payload of each data packet is 36B. The total size of packets sent by GreenRoutes and WHARP is 58B, which adds to the payload the bytes of the headers added at different layers. CTP-WUR transmits packets whose total size is 70B. (The 12 extra bytes are needed for MAC and network layer functions.) Half of the nodes are equipped with solar cells; the remaining nodes harvest energy using micro wind turbines. For the rest of the simulation parameters we refer to Section 3.2.1.

5.2.1 Simulation results

The performance of the three data forwarding strategies is compared with respect to the following key performance metrics: Packet overhead, measuring the fraction of control traffic generated by each protocol, end-to-end latency of all packets successfully delivered to the sink, the total energy consumption spent by the network, and the packet delivery ratio.



(a) Control packet overhead

(b) End-to-end latency

(c) Energy consumption

(d) Packet delivery ratio

Fig. 5.4: Performance comparison of CTP-WUR, GreenRoutes and WHARP for increasing traffic.

5.2.1.1 Packet overhead

The average number of bytes of control packets generated by each forwarding strategy, normalized to the total number of bytes of data packets is shown in Fig. 5.4a. At the lowest traffic, CTP-WUR has a packet overhead which is approximately 3.1

and 3.5 times higher than that of GreenRoutes and WHARP, respectively. This mostly depends on the difference in size of both control and data packets of each forwarding strategies. Specifically, CTP-WUR sends control packets whose size is 25B, while the total size of control packets in GreenRoutes and WHARP is 14B and 12B, respectively. Both GreenRoutes and WHARP send control packets every time nodes need to forward a data packet (RTS/CTS handshake). We observe that the number of a sender neighbors that wakes up and participates in the handshake on the main radio is slightly inferior for WHARP, because of the optimized policy from the MDP. This justifies the slightly lower packet overhead of WHARP over GreenRoutes. Independently of the forwarding strategy, packet overhead decreases with increasing traffic. In CTP-WUR nodes transmit control packets independently of the traffic load. As a consequence, packet overhead decreases with more data packets being generated. GreenRoutes and WHARP send control packets for route selection (RTS/CTS packets). Therefore, packet overhead would be expected to increase with traffic. However, both protocol take advantage of an "ID cashing" mechanism implemented to reduce RTS/CTS-induced overhead (and delays). This mechanism allows a sender node i to store the ID of its last successful relay j for a predefined amount of time τ. All packets that node i needs to transmit within τ seconds will be transmitted directly to node j, without any new relay selection phase. (In this case, node i will wake up node j directly, i.e., by using its ID as wake-up sequence.) We notice that the higher the traffic, the higher the number of packets to be transmitted within τ seconds, and therefore the lower the number of handshakes for relay selection. As a result, at the highest traffic all three mechanisms have almost the same packet overhead.

5.2.1.2 End-to-end latency

The average end-to-end latency for delivering a data packet to the sink is shown in Fig. 5.4b. Independently of traffic, CTP-WUR consistently delivers packets with lower latency. More specifically, GreenRoutes and WHARP experience latency up to 4.5 and 3.6 times higher than those incurred by CTP-WUR, respectively. This is due to the cross-layer nature of both GreenRoutes and WHARP, requiring nodes to engage in a time consuming RTS/CTS handshake before sending a data packet. Latency remains largely independent of traffic for CTP-WUR, because of the simple tree-based mechanism for determining routes, and the relay of wake up sequences which further reduces route lengths. However, we observe slightly higher end-to-end latency with increasing traffic for CTP-WUR. This is because the number of nodes that switch to an all-off state is increasing leading to "invalid" paths based on the current tree-based paths, and CTP-WURhas to re-construct its paths based on the available nodes. Latency instead decreases with increasing traffic for both GreenRoutes and WHARP because of the ID cashing mechanism for reducing RTS/CTS-induced delays (see Chapter 3.2.2.3). We notice that the higher the traffic, the higher the number of packets to be transmitted within τ seconds, and therefore the lower the number of time-consuming handshakes for relay selection (Section 5.2.1.1).

5.2.1.3 Total energy consumption

Fig. 5.4c shows the average network energy consumption. Independently of traffic, CTP-WUR always outperforms all other approaches, spending up to 2 and 2.4 times less energy than GreenRoutes and WHARP, respectively. This depends on the relay selection strategy of the latter protocols, which possibly wakes up multiple nodes, whose main radio stays on until the contention for selecting a relay is completed. This does not happen with CTP-WUR, where a node only wakes up one node (its grandparent). As expected, the performance gap among the different protocols increases with traffic. This is due to the higher number of interference among packets, resulting in a higher number of contentions for relay selection and in a higher number of re-transmissions. We notice that GreenRoutes spends slightly more energy than WHARP because its control packets are slightly bigger in size, and also because nodes stay with their main radio on for a longer time.

5.2.1.4 Packet delivery ratio

Fig. 5.4d depicts the average packet delivery ratio of all strategies for increasing traffic. CTP-WUR successfully delivers almost all data packets to the sink for traffic with inter-arrival times higher than 2s. Its performance, however, drops to 82% at the highest considered traffic. This is because of the tree-based topology of CTP-WUR, which provides a node with only one possible relay (its grandparent). As medium/high traffic imposes higher energy consumption, nodes may be non operational for temporary lack of energy for longer times (blacked out nodes). If the grandparent of a sender is blacked out it cannot receive the packet, which will be discarded by the sender after a predetermined number of transmission attempts. GreenRoutes shows similar performance and similarly suffers from nodes that are blacked out. WHARP instead consistently achieves a packet delivery ratio higher than 90%, irrespective of traffic. This is due to the optimized relay selection policy provided by the MDP, which takes energy and harvested energy explicitly into account, makes nodes avoid waking up if they are not a good fit, and explicitly penalizes choosing relays to routes with blacked out nodes.

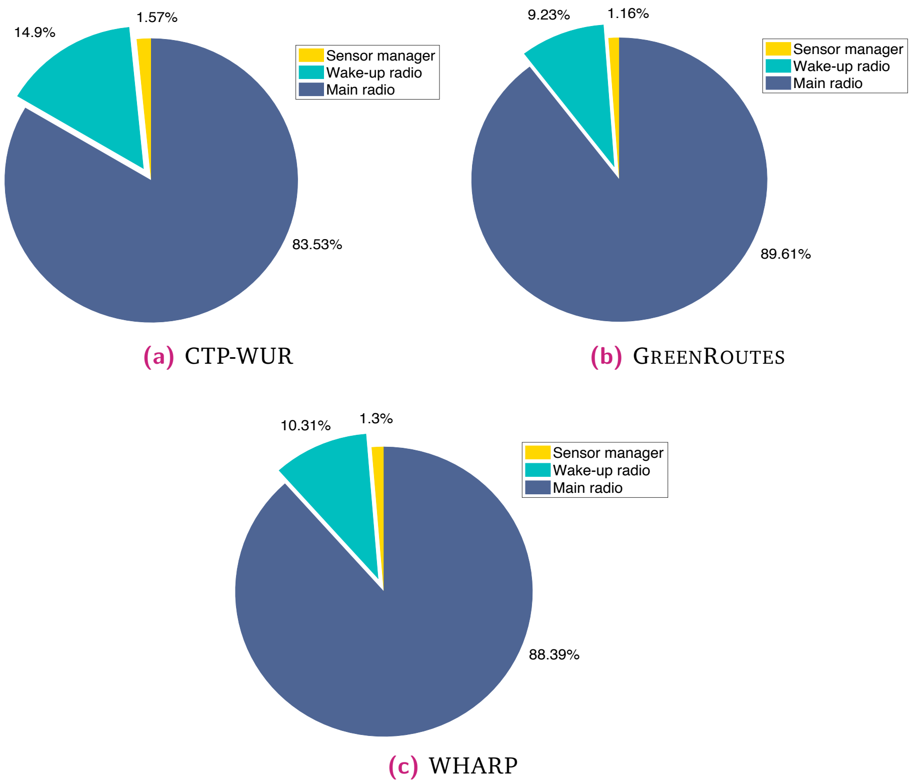
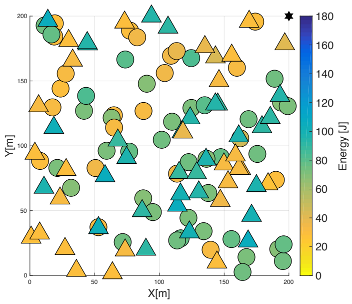
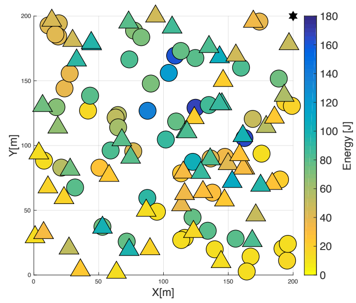


Fig. 5.5: Energy consumption breakdown in networks with packet inter-arrival time of 10s.

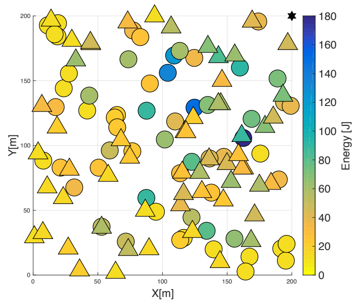
Fig. 5.5 and Fig. 5.6 depict results that allow us to delve deeper into the differences of the three strategies concerning their use of energy. Particularly, Fig. 5.5 shows the energy consumed by the sensor manager, the main radio, and the wake-up radio when running CTP-WUR (Fig. 5.5a), GREENROUTES (Fig. 5.5b), and WHARP (Fig. 5.5c) in networks with moderate traffic (the packet inter-arrival time is 10s). The energy consumed is expressed as the percentage of the total energy consumed by each forwarding strategy. The main radio drains most of the available energy, independently of the forwarding strategy. We observe that the energy consumed by the wake-up radio for transmitting a wake-up sequence is around three times higher than the energy consumed by the main radio for transmitting a data packet. Despite data packets are longer, the time needed to transmit the 8 bits of a wake-up sequence at 1kbps is higher, hence the higher amount of energy drained. The dominant component of the overall energy consumption is however due to the reception of packets on the main radio. For instance, in CTP-WUR, due to data packets and control packets reception, the main radio stays in receiving mode 16 times longer than in transmission. This number grows to 64 and 56 times more in GREENROUTES and WHARP, respectively, as their relay selection strategy wakes up multiple nodes, and makes them stay in receiving mode for significant amounts of time. The energy consumed by the wake-up radio for receiving a wake-up sequence is negligible, as our receiver consumes in the μW, and stays receiving for short



(a) CTP-WUR



(b) GreenRoutes



(c) WHARP

Fig. 5.6: Per node energy consumption in networks with medium/high traffic and heterogeneous energy harvesting sources.

periods of time. This motivates why the energy consumed by the main radio is dominant, with consumption from 5.6 to 9.7 times higher than those incurred by the wake-up radio.

Fig. 5.6 shows the snapshot of a selected, exemplary topology, with the sink placed at the upper right corner (depicted as a black star). This scenario refers to a network with medium/high traffic (the packet inter-arrival time is 1s). Sensor nodes are depicted as circles or triangles, depending on their energy source, sun or wind, respectively. The color of a node indicates the energy it consumed throughout the simulation time: The darker the color, the higher the energy consumed. No node running CTP-WUR is colored in the darker shades (Fig. 5.6a), which is indicative of the fact that, overall, it is the most energy-efficient solution (see also Fig. 5.4c and Fig. 5.5a). Some of the nodes running GreenRoutes and WHARP instead sport darker colors (Fig. 5.6b and Fig. 5.6c, respectively). This indicates that, especially at the highest traffic considered, they stay with their main radio on for longer periods of time. We observe that among the three forwarding strategies, the only one that is capable of effectively avoid draining energy from the most energy challenged nodes (typically those powered by wind energy) is WHARP. This is because the WHARP relay selection strategy is driven by a Markov Decision Process that explicitly penalizes selecting nodes that could black out in the near future. As a result, aside from the nodes closer to the sink ("funneling effect"), most of the nodes running WHARP show low to moderate energy consumption.

5.3 Conclusions

This chapter presents a comparative performance evaluation of three data forwarding strategies for green wireless networks, where protocol design takes explicitly into account usage of a wake-up radio, and where nodes are capable of energy harvesting. All three approaches, namely, CTP-WUR, GreenRoutes and WHARP, achieve exemplary performance under a variety of performance metrics. The results of our GreenCastalia-based simulations show that approaches like WHARP and GreenRoutes that rely on contention-based mechanisms for relay selection incur high latency and energy consumption as the contention itself is time-consuming and involves multiple potential next-hop relays. Approaches like CTP-WUR obtain instead faster and lighter performance, because of the more traditional, more proactive way of determining routes. However, in energy harvesting-based networks where nodes can temporarily black out these approaches incur packet loss due to the lack of timely topology updates. It takes the sophistication of machine learning to allow design à la WHARP to succeed in selecting next-hop relays along routes without nodes that black out. We finally observe that the use of techniques to directly forward data packets to a known and already used next-hop relay ("ID caching"), as in GreenRoutes and WHARP, can decrease end-to-end latency to values similar to that of strategies which do not require the transmission of control packets for channel reservation, such as CTP-WUR.

6 On the Impact of Local Computation Over Routing Performance in Green Wireless Networks

The energy consumption challenge has hastened the design of protocols at all layers of the networking stack that are energy aware and aim to ensure energy sustainability. Among the most promising protocol design techniques, the past decade has shown the increasingly intensive adoption of techniques based on various forms of machine learning [3]. In Chapter 4 we discussed how superior performance is being obtained by taking key protocol decisions based on the outcome of local learning computations that are based on past and expected availability of resources. However, learning techniques can suffer from high computational costs as nodes drain a considerable percentage of their energy budget to run sophisticated software procedures, predict accurate information and determine optimal decision. In this chapter we investigate the impact of the energy consumption of local computations, especially those induced by machine learning-based techniques, on the overall network performance.

In fact, while learning-based solutions have flourished for a wide variety of WSN applications, the demonstration of their performance effectiveness does not consider the cost of running the learning methodologies on which they are based. We consider the recent routing solution, named WHARP, presented in Chapter 4, that uses a standard method for the solution of the MDP, namely, the Backward Value Iteration (BVI) algorithm [54]. In this work, we set to study the impact of running BVI periodically for guiding WHARP operations. As our experimental investigation shows that BVI imposes heavy energy consumption, we set out to design a heuristic solution, named W-HEU, for solving the MDP that is computationally lighter than BVI. Our method trades off the optimality of methods like BVI for greater energy savings. Due to the lower computational energy requirements, W-HEU allows nodes to stay operational for a longer time, independently of the specific scenario considered. Particularly, every node can perform sensing and forwarding duties for *at least* 72% of the simulation time. This is remarkably better than when nodes run W-BVI, where they are instead operational for *at most* 72% of the simulation time.

The remainder of the chapter is so organized. Section 6.1 presents the details of the proposed heuristic solution method. Performance comparison results and discussion are introduced in Section 6.2. Finally, we conclude this chapter in Section 6.3.

6.1 Solving the MDP

An MDP can be solved via standard techniques such as Backward Value Iteration or any other method for solving the Bellman equations (Equation 4.2) [54]. Backward Value Iteration is, for instance, the method used to solve the MDP in [9].

In this section, we present a heuristic method that, differently from known standard techniques, does not impose high computational energy consumption for solving the MDP. We formulate a simple *threshold policy* that output either *green* or *red* only based on the value of the reward function r corresponding to the decision of being available for forwarding the packet (a_f). Particularly:

> In each decision epoch n a node computes the reward function $r(s_n, a_f)$. If $r(s_n, a_f) > 0$, then output *green*, otherwise, output *red*.

Our solution is heuristic in nature, in that it does not always provide the *optimal decision* that other solution methods would provide. However, since computing the reward function is much simpler than solving the Bellman equations through value iteration, linear programming or other standard methodologies, we expect lower computational requirements, better energy consumption and superior network performance.

In the remainder of this section we provide the rationale for our heuristic method. Particularly, we prove that when $r(s_n, a_f) \leq 0$ then our heuristic provides the same optimal decision that other optimal solution method would provide. We then show that when $r(s_n, a_f) > 0$, we output the optimal solution only in case the node does not harvest any energy in the nth decision epoch ($h_n = 0$). In case $r(s_n, a_f) > 0$ and $h_n > 0$ the solution we output, namely, *green*, may be sub-optimal. We start by proving the following:

Lemma 1 *For each $n = 1, \ldots, N$ the value function $V_n^{\pi^*}(s)$ is non decreasing in s.*

Let us define $q_n(k|s_n, a_n)$ as the probability that state s_{n+1} in decision epoch $n+1$ exceeds $k-1$, i.e., that the energy level of a node will be greater than or equal to a value k. Formally:

$$q_n(k|s_n, a_n) = \sum_{s_{n+1}=k}^{\infty} P^a_{s \to s_{n+1}}. \tag{6.1}$$

We claim that $q_n(k|s_n, a_n)$ is non decreasing in s_n, for all $k \in \mathcal{S}$, $a_n \in \mathcal{A}$, and $n = 1, \ldots, N-1$. We can prove this by contradiction. Let us assume that $q_n(k|s_n, a_n)$ is a decreasing function of s_n, for all k, a_n, and decision epoch n. We now consider two consecutive states s_n^+ and s_n^-, with $s_n^+ > s_n^-$. It follows that

$$q_n(k|s_n^+, a_n) - q_n(k|s_n^-, a_n) < 0. \tag{6.2}$$

We denote by s_{n+1}^+ and s_{n+1}^- the states in decision epoch $n+1$ in which the system transits from s_n^+ and s_n^-, respectively. We consider two cases, depending on the action taken.

1) $a_n = a_d$. In this case we know that state transitions are deterministic and uniquely identified by h_n and e_n^x, leading to $s_{n+1}^+ > s_{n+1}^-$ with probability 1. We can define $q_n(k|s_n^+, a_d)$ as:

$$q_n(k|s_n^+, a_d) = \begin{cases} 1 & \text{if } s_{n+1}^+ \geq k, \\ 0 & \text{otherwise.} \end{cases} \tag{6.3}$$

(We can do similarly for $q_n(k|s_n^-, a_d)$.) Since $s_{n+1}^+ > s_{n+1}^-$, it follows that $q_n(k|s_n^+, a_d) \geq q_n(k|s_n^-, a_d)$, contradicting our assumption.

2) $a_n = a_f$. In this case state transitions are probabilistic and we can define $q_n(k|s_n^+, a_f)$ (and, similarly, $q_n(k|s_n^-, a_f)$) as:

$$q_n(k|s_n^+, a_f) = \begin{cases} \displaystyle\sum_{e^{tx}=0}^{e_n^+ - k} p^{e^{tx}}(e^{tx}) & \text{if } k < e_n^+, \\ 0 & \text{otherwise.} \end{cases} \tag{6.4}$$

The intuition is that states higher than k can be reached only if they are lower than the overall energy e_n available for packet forwarding. Since $e_n^+ > e_n^-$, it follows that

$$\sum_{e^{tx}=0}^{e_n^+ - k} p^{e^{tx}}(e^{tx}) > \sum_{e^{tx}=0}^{e_n^- - k} p^{e^{tx}}(e^{tx}) \tag{6.5}$$

and, consequently, $q_n(k|s_n^+, a_f) \geq q_n(k|s_n^-, a_f)$, contradicting again our assumption.

We can conclude that $q_n(k|s_n, a_n)$ is non decreasing in s_n, for all $k \in \mathcal{S}$, $a_n \in \mathcal{A}$, and $n = 1, \ldots, N-1$. The lemma claim follows from plugging this result into Proposition 4.7.3 of [54], which also uses that for each action a_n and epoch n the reward function $r(s_n, a_n)$ is non decreasing in s_n. This is true by construction.$\diamond$ The above lemma is key to prove the following result.

Theorem 1 *For each decision epoch n, $n = 1, \ldots, N$, and state $s_n \in \mathcal{S}$ such that $r(s_n, a_f) < 0$, action a_d is optimal.*

Let us assume that when the reward is negative it would be better to transmit. From equations (4.2) and (4.3) it follows that:

$$r(s_n, a_f) + \gamma \sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V^{\pi^*}_{n+1}(e_n - e^{tx}) > \gamma V^{\pi^*}_{n+1}(e_n), \tag{6.6}$$

i.e., the value function associated to a_f is higher than that associated to a_d. Since $r(s_n, a_f)$ is negative, we can write:

$$\sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V^{\pi^*}_{n+1}(e_n - e^{tx}) > V^{\pi^*}_{n+1}(e_n). \tag{6.7}$$

However, V^{π^*} is non-decreasing by Lemma 1 and $V^{\pi^*}_{n+1}(e_n)$ cannot be lower than the weighted sum of values lower than or equal to $V^{\pi^*}(e_n)$ itself. This contradicts our assumption and ends our proof. $\diamond$

Theorem 1 allows our solution method to output *red* as the optimal decision, i.e., as if it was computed by standard, yet computationally more expensive techniques for solving MDPs. In the following we show that our method outputs optimal *green* decisions provided that no harvesting happens in epoch n. We start with the following lemma.

Lemma 2 *If there is no harvesting, i.e., $h_n = 0$ for each $n = 1, \ldots, N$, if $r(s_n, a_f) < 0$ then $V^{\pi^*}_n(s_n) = 0$.*

We proceed by backward induction on the number of epochs. In the last decision epoch N, $V^{\pi^*}_N(s) = \max\{r(s_N, a_f), r(s_N, a_d)\}$. Since $r(s_N, a_d) = 0$, it follows that whenever $r(s_N, a_f) < 0$ it is better to drop packets, and $V^{\pi^*}_N(s) = 0$. We now consider a generic decision epoch n so that $r(s_n, a_f) < 0$. From equations (4.2) and (4.3) we can write:

$$V^{\pi^*}_n(s) = \max\{r(s_n, a_f) + \gamma \sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V^{\pi^*}_{n+1}(e_n - e^{tx}), \gamma V^{\pi^*}_{n+1}(e_n)\}. \tag{6.8}$$

Since there is no harvesting, the energy level in decision epoch $n + 1$ has to be lower than the current energy, independently of the chosen action. As a result, the reward function in the next state s_{n+1} will be negative as well and, by the induction hypothesis and Lemma 1, we can rewrite Equation (6.8) as:

$$V_n^{\pi^*}(s) = \max\{r(s_n, a_f), 0\}. \tag{6.9}$$

Since $r(s_n, a_f) < 0$, we have that $V_n^{\pi^*}(s) = 0$. $\diamond$

We can finally prove the following:

Theorem 2 *If there is no harvesting, i.e.,* $h_n = 0$*, for each decision epoch* n*,* $n = 1, \ldots, N$*, and state* $s_n \in \mathcal{S}$ *such that* $r(s_n, a_f) > 0$*, action* a_f *is optimal.*

We proceed by backward induction on the number of epochs. In the last decision epoch if $r(s_N, a_f) > 0$ it is better to transmit packets, otherwise the reward would be 0. We now assume that, in a generic decision epoch n, $r(s_n, a_f)$ is positive but the optimal action is to drop packets. Our assumption can be expressed by the following equation (a straightforward application of the value function definition):

$$r(s_n, a_f) + \gamma \sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V_{n+1}^{\pi^*}(e_n - e^{tx}) < \gamma V_{n+1}^{\pi^*}(e_n). \tag{6.10}$$

Let us define $s_{n+1} = e_n$, and let us evaluate Equation (6.10) depending on the value of $r(s_{n+1}, a_f)$. *a)* $r(s_{n+1}, a_f) < 0$. In this case we know by the induction hypothesis that $V_{n+1}^{\pi^*}(s_{n+1}) = 0$. Thanks to Lemma 1, Equation 6.10 becomes $r(s_n, a_f) < 0$ which contradicts the assumption that $r(s_n, a_f)$ is positive.

b) $r(s_{n+1}, a_f) > 0$. In this case we can expand the second term in Equation (6.10) by exploiting the induction hypothesis as:

$$\begin{aligned} r(s_n, a_f) + \gamma \sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V_{n+1}^{\pi^*}(e_n - e^{tx}) < \\ \gamma \left(r(s_{n+1}, a_f) + \gamma \sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V_{n+2}^{\pi^*}(e_{n+1} - e^{tx}) \right). \end{aligned} \tag{6.11}$$

Since there is no harvesting, the energy available in state s_n is greater than or equal to that in state s_{n+1}, which implies that $r(s_n, a_f) \geq r(s_{n+1}, a_f)$. We can simplify the above equation as:

$$\sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V_{n+1}^{\pi^*}(e_n - e^{tx}) < \sum_{e^{tx}=0}^{\infty} p^{e^{tx}}(e^{tx}) V_{n+2}^{\pi^*}(e_{n+1} - e^{tx}). \tag{6.12}$$

If we focus on each couple of terms $V^{\pi^*}_{n+1}(e_n - e^{tx})$ and $V^{\pi^*}_{n+2}(e_{n+1} - e^{tx})$, we know that they are both zero if $r(e_n - e^{tx}, a_f)$ is negative (Lemma 2). When it is positive, we can expand $V^{\pi^*}_{n+1}(e_n - e^{tx})$ exploiting the induction hypothesis as:

$$\begin{aligned} V^{\pi^*}_{n+1}(e_n - e^{tx}) = r(e_n - e^{tx}, a_f) + \\ \gamma \sum_{e^{tx'}=0}^{\infty} p^{e^{tx}}(e^{tx'}) V^{\pi^*}_{n+2}(e_{n+1} - e^{tx} - e^{tx'}) \\ > V^{\pi^*}_{n+2}(e_{n+1} - e^{tx}). \end{aligned} \tag{6.13}$$

This is simply the value function equation computed in state $e_n - e^{tx}$ and decision epoch $n+1$. The last inequality holds because we know by induction that if the reward is positive it is better to choose action a_f than to drop packets. This proves that Equation (6.12) cannot hold because we have:

$$V^{\pi^*}_{n+1}(e_n - e^{tx}) > V^{\pi^*}_{n+2}(e_{n+1} - e^{tx}). \tag{6.14}$$

This contradicts our original assumption, ending our proof. $\diamond$

Theorem 1 and Theorem 2 provide proof that our heuristic method obtains the same optimal solutions that standard methods for solving MDPs would provide, with the exception of epochs with positive rewards and energy harvesting intake. In this latter case, our *green* decision may be sub-optimal.

6.2 Performance evaluation

We demonstrate the benefits of the heuristic solution method presented in Section 6.1 through an extensive and diverse set of simulation-based experiments. Particularly, we compare the performance of WHARP where the MDP is solved by the Backward Value Iteration method (W-BVI), with that of when the MDP is solved by our heuristic (W-HEU). In the following we start by describing the way we determine the energy cost of computing each solution method. We then introduce our simulation results.

6.2.1 Computational energy cost

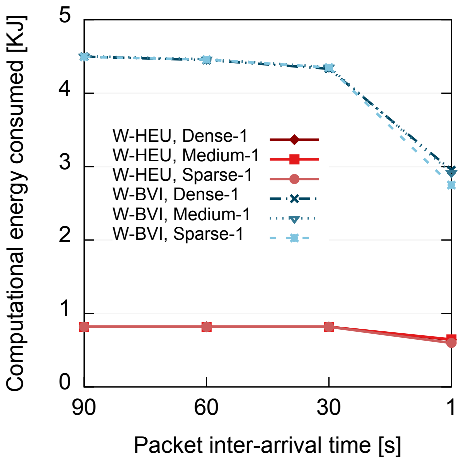
Commonly with most simulators used in WSN research, GreenCastalia does not take into account the energy consumption needed for computational purposes. This is platform and algorithm-depended. In order the establish the energy consumption of computing both MDP solution methods, we implemented them in TinyOS (the operative system used by the MagoNode++), and measured their execution time. The MagoNode++ mote [49], which has energy harvesting and wake-up radio capabilities, is shown in Fig. 6.1. Using these execution times we computed the corresponding computational energy consumption according to the specifications of the MagoNode++, which uses the ATmega256RFR2 microcontroller, consuming 4.1mA at 3V when in active mode. We then extended GreenCastalia to take into account the computational energy consumption measured on the MagoNode++ by extending the simulator resource manager to take into account the measured computational energy consumption.

Fig. 6.1: A MagoNode++ mote.

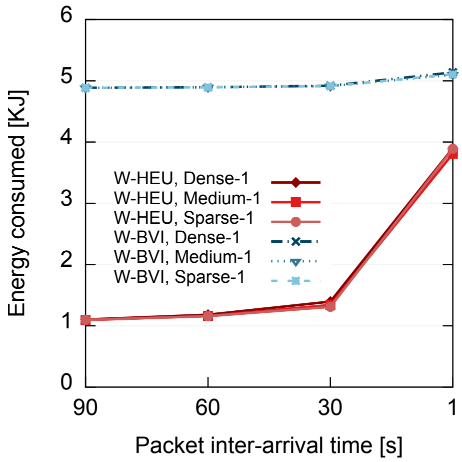
6.2.2 Simulation scenario and parameters

We consider WSNs made up of 64 sensor nodes (modeled as MagoNode++) which are positioned according to a randomized grid deployment, i.e., they are laid down as a 16×4 grid where the actual location of each node is randomly displaced from the precise grid point by 10%. The size of the grid deployment area differs depending on the considered spatial node density. In this work we consider three different sizes: 1) $324 \times 80\text{m}^2$ (sparse scenario); 2) $224 \times 56\text{m}^2$ (medium density scenario), and 3) $160 \times 40\text{m}^2$ (dense scenario). The sink node is placed at the bottom left corner of the deployed area. We stipulate that the per node inter-arrival time between packets ranges in the set of four values $\{90, 60, 30, 1\}$ seconds, corresponding to low, medium, high and very high traffic. Each sensor node acts as both a relay and a source.

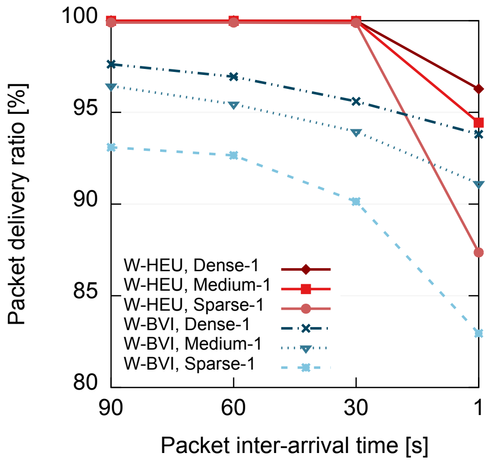
We consider two scenarios for harvesting sources: 1) Homogeneous, i.e., all nodes harvest energy using micro wind turbines; 2) Heterogeneous, i.e., half of the nodes are equipped with solar cells and the remaining harvest energy using micro wind turbines. In this simulation setup wind harvesting traces are real traces collected in Rome, Italy, for a period of one month during summer The rest of the simulation metrics are as described in Section 3.2.1. The performance of the two solution methods is evaluated with respect to the following key performance metrics: The network computational energy consumption, defined as the total amount of computational energy spent by all nodes, the network energy consumption, the packet delivery ratio, and the all-off ratio defined as the average percentage of simulation time a node was all-off. We do not show results for metrics such as route lengths and end-to-end data packet latency as we observed that they are not noticeably affected by changing the solution method. All results have been obtained by averaging the outcomes of 100 simulation runs, each of duration T_s of 4 days.



(a) Computational energy consumption



(b) Network energy consumption



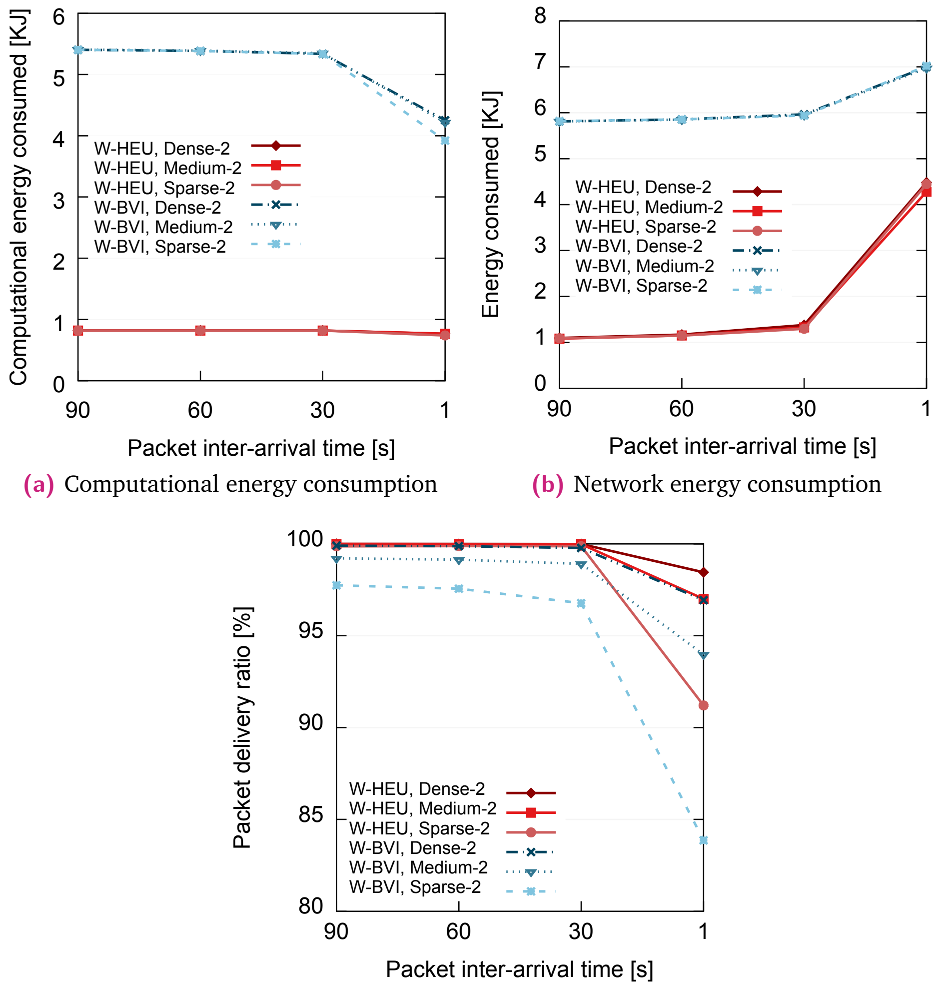
(c) Packet delivery ratio

Fig. 6.2: W-HEU vs. W-BVI for homogeneous harvesting sources in sparse, medium, and dense networks.

6.2.3 Simulation results

6.2.3.1 Network computational energy consumption

Fig. 6.2a and Fig. 6.3a show the total computational energy spent by the network for increasing traffic and for homogeneous and heterogeneous energy harvesting

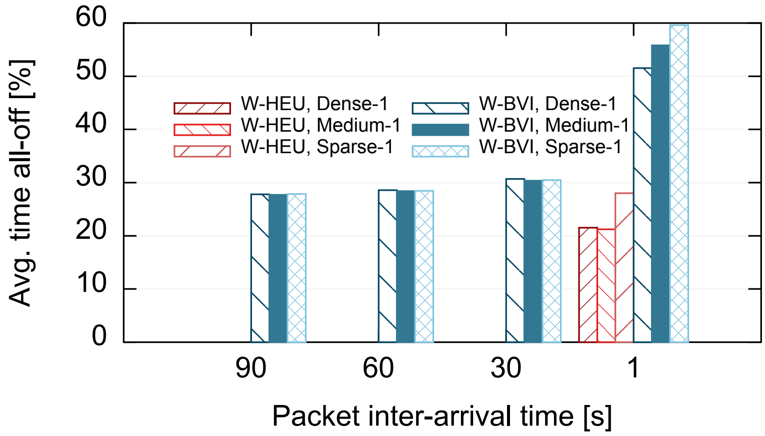


(a) Computational energy consumption (b) Network energy consumption

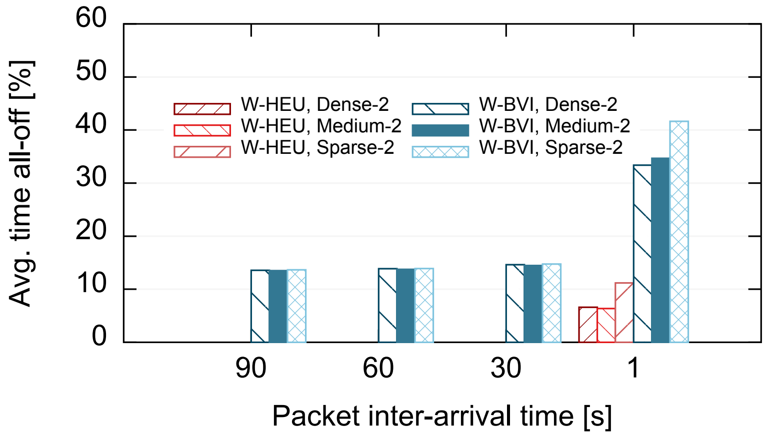
(c) Packet delivery ratio

Fig. 6.3: W-HEU vs. W-BVI for heterogeneous harvesting sources in sparse, medium, and dense networks.

sources, respectively. Independently of node density and harvesting source, W-BVI requires at least 5.5 times more energy to compute the solution of the MDP compared to W-HEU. Overall, independently of the network density, the consumption trends of both solution methods are similar. Their actual performance, however, depends on the energy harvesting source. This is mainly because of the higher energy harvesting rates in the case of heterogeneous harvesting sources. The total energy spent for computational purposes decreases for increasing traffic. This is more noticeable for W-BVI than for W-HEU because by using the former method nodes consume considerably more energy, which sends a higher number of nodes to the all-off state (see also Fig. 6.4a). As a consequence, with so many nodes all-off, the computational energy consumption is more limited. We notice that at the highest traffic, W-HEU also shows a drop in computational energy consumption. However, even in this case, nodes running W-HEU consume an average of 82% less energy that those using W-BVI (heterogeneous harvesting sources, Fig. 6.3a).



(a) All-off ratio: Homogeneous harvesting sources



(b) All-off ratio: Heterogeneous harvesting sources

Fig. 6.4: W-HEU vs. W-BVI: Per node average all-off time for increasing traffic (% of the simulation time).

6.2.3.2 Network energy consumption

The total energy consumption incurred by the two solution methods is depicted in Fig. 6.2b and Fig. 6.3b. W-HEU always consumes less energy than W-BVI. This is mainly due to the higher computational energy consumption that W-BVI requires to solve the MDP. The performance gap is more noticeable at the lowest traffic, where W-BVI consumes 81% more energy that W-HEU (heterogeneous harvesting sources, Fig. 6.3b). At higher traffic, and despite lower computational energy consumption, both solution methods consume more energy. This is because of the higher energy toll imposed by dealing with a (considerably) higher number of packets to transmit/receive. However, because of its lower computational energy expenditure, W-HEU manages to consume approximately up to 1.6 times less energy than W-BVI (highest traffic, Fig. 6.3b). These results show rather clearly the importance of considering the computational energy costs, as their impact on the energy spent by the network is quite remarkable (see also Fig. 6.5 and discussion below).

6.2.3.3 Packet delivery ratio

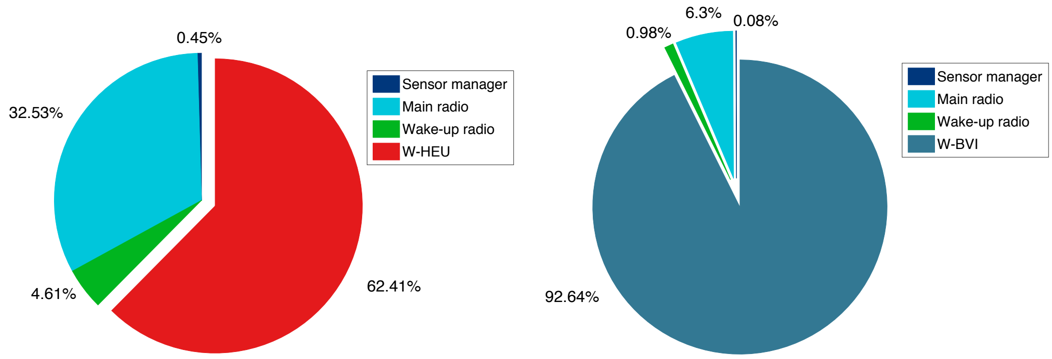
The packet delivery ratio for the two harvesting source scenarios is shown in Fig. 6.2c and Fig. 6.3c. In both scenarios, W-HEU clearly outperforms W-BVI and consistently attains a packet delivery ratio higher than 87%. This is because nodes running W-BVI spend more energy and they tend to go all-off more regularly, as shown in Fig. 6.4. Nodes running W-HEU remain active in the network for at least 72% of the time allowing higher packet generation rates. As shown in both figures, and especially clearly in Fig. 6.2c, the PDR of both solutions decreases with increasing traffic. This is because WHARP has to process more packets, deal with more interference and retransmissions, etc., which results in higher energy consumption. The superior performance of W-HEU depends on the fact that due to the lower energy consumption of the heuristic solution method, more nodes are active, and therefore less packets are dropped for lack of finding potential relays.

We notice that the performance of both W-HEU and W-BVI also depends on the density of the network. Specifically, the higher the density, the higher the PDR. This is explained by the higher number of active nodes, which means that a higher number of potential relays are available to the sender to forward its packet. Finally, we notice a difference in the performance of WHARP in scenarios with different energy harvesting sources. In general, the PDR of both W-HEU and W-BVI is higher in the heterogeneous source scenarios. This is because harvesting energy from solar panels provides more energy to the nodes, that are in active state considerably more time than when they only use wind turbines (see also Fig. 6.4).

6.2.3.4 All-off ratio

The average per node all-off ratio is shown in Fig. 6.4. Independently of traffic and energy harvesting sources, W-BVI experiences higher all-off ratios, which can be up to 3.7 time more than those experienced by W-HEU. This is because of the higher computational cost of solving the MDP using W-BVI. At the lower traffic, nodes running W-BVI go to an all-off state approximately 28 and 14 times more than W-HEU, for homogeneous and heterogeneous energy harvesting sources, respectively. In the heterogeneous case, both solutions perform better than in the case where nodes harvest energy only through wind. This pattern is consistent with the fact that nodes harvest more energy in scenarios with solar harvesting.

To further demonstrate the higher computational energy consumption incurred by W-BVI, we investigate the energy consumption breakdown of different node activities. Fig. 6.5a and Fig. 6.5b show the energy consumed by the main radio, the wake-



(a) W-HEU (b) W-BVI

Fig. 6.5: W-HEU vs. W-BVI: Energy consumption breakdown in networks with packet inter-arrival time of 30s, heterogeneous energy harvesting sources, in a network of medium density (%).

up radio, the sensor manager, and by the solution method for W-HEU and W-BVI, respectively. The figures refer to networks of medium density, with packet inter-arrival time of 30s and heterogeneous energy harvesting sources. The breakdown is expressed as the percentage of the total energy consumption incurred by both solutions (as shown in Fig. 6.3b). We observe that in both cases the computational energy expenditure is prevailing over all other forms of energy consumption. The ratio among the energy consumed for local computation vs. that for communication (say, on the main radio) is however visibly different for the two solutions. W-HEU local consumption is almost twice the energy spent for data communications. In case of W-BVI, instead, nodes spend most of their energy for local computational purposes (specifically, 14.7 times more than for data communication). As a result, we note that nodes using W-HEU have more available energy to spend for communication purposes, remain active 100% of the time, and successfully deliver more packets (Fig. 6.4b and Fig. 6.3c). The remarkable difference of energy spent for solving the MDP using W-BVI provides further justification of its reduced packet delivery ratio.

6.3 Conclusions

In this chapter we investigate the impact of computational requirements of learning techniques on the performance of protocols for data forwarding in green wireless networks. We introduced a heuristic solution method that approximates the Backward Value Iteration (BVI) algorithm used to solve the MDP in the high-performance routing protocol WHARP. We compared the performance of WHARP using our heuristic (W-HEU) and that of WHARP using BVI (W-BVI) through GreenCastalia-based simulations, where we extended the capabilities of the simulator to reflect the energy consumption due to local computations. Results show that W-HEU outperforms W-BVI because of its remarkably lower computational cost: When nodes run W-HEU the network consumes up to 5.35 times less energy than when they run W-BVI. This results in higher operational times and allows nodes running W-HEU to be active for *at least* 72% of the simulation time, while nodes using W-BVI remain active for *at most* 72% of the simulation time.

7 WHARPNR-HEU: A Preview

In Chapter 5 we presented a performance comparison of different forwarding strategies for green wireless networks. The acquired insights about the different forwarding design choices and their consequences on network performance indicated that the sophisticated learning-based design of WHARP (Chapter 4) allows nodes to successfully select next-hop relays along routes without nodes that black out. However, the proactive nature of route computation of CTP-WUR [13] results in faster packet delivery and lower energy consumption, requesting further optimization of the cross-layer forwarding strategies presented in this thesis.

In this chapter we present a preview of an on-going work by introducing WHARPNR-HEU, which stands for Wake-up and HARvesting-based energy-Predictive No-Rts with HEUristics, a forwarding strategy inspired by the WHARP protocol presented in Chapter 4. Similar to WHARP, forwarding decisions in WHARPNR-HEU are the outcome of a Markov Decision Process (MDP) that takes into account important aspects of the network, including the local energy availability, harvested, and consumed. The additional steps being taken in WHARPNR-HEU are: i) We eliminate the RTS phase to reduce power consumption and end-to-end latency; ii) Optimal forwarding decisions are included in the wake-up semantic addressing to eliminate power consumption due to unnecessary communication; iii) We consider the heuristics solution for solving the MDP (see Chapter 6); iv) The computational cost of the heuristics solution is taken under consideration in the simulation experiments.

7.1 Description of WHARPNR-HEU

WHARPNR-HEU is a cross-layer forwarding strategy, where interaction between the MAC and the network layers is enabled to allow joint channel access and next-hop selection. Next-selection is dictated by the distance from the sink of the neighboring nodes and by their capability to act as forwarders. Source nodes that have a packet to forward, initially broadcast a wake-up sequence aimed at waking up neighboring nodes that are one-hop closer to the sink and that are capable of forwarding packets, in terms of energy. Only neighboring nodes that have a positive forwarding capability and satisfy the distance condition receive the wake-up sequence from the sender node. The rest of the neighboring nodes ignore the transmitted wake-up sequence and remain in a “sleep" mode. Nodes evaluate their capability of participating in the forwarding procedure by computing MDP-based optimal forwarding decisions, as in WHARP.

The main procedure followed by each node to compute optimal forwarding decisions is summarized in Algorithm ComputeAction. Details about the formulation of the MDP and its solution method(s) can be found in Chapter 4 and in Chapter 6.

Algorithm 1 ComputeAction

for all $n \in \{0, \cdots, N\}$ **do** *#For each decision epoch*
  for all $b \in \{0, \cdots, B_{max}\}$ **do***#For each battery level*
    $e_n = b_n + h_n - e_n^x$ *#Compute the overall available energy for packet forwarding*
    if $a_n = a_d \wedge b_n + h_n > e_n^x$ **then** *#Compute next state* s_{n+1} *based on action taken*
      $s_{n+1} = e_n$
    else if $a_n = a_f \wedge b_n + h_n > e_n^{tx} + e_n^x$ **then**
      $s_{n+1} = e_n - e_n^{tx}$
    else
      $s_{n+1} = 0$ *#Node goes all-off if the amount of energy is not sufficient*
    end if
    if $a_n = a_f$ **then** *#Compute the reward function*
      $r(s_n, a_f) = r \cdot \sum_{e_n^{tx}=0}^{e_n} p^{e_{tx}}(e_n^{tx}) - c \cdot \sum_{e_n^{tx}=e_n}^{\infty} p^{e_{tx}}(e_n^{tx})$
    else if $a_n = a_d$ **then**
      $r(s_n, a_d) = 0$
    end if
    $V_n^{\pi^*}(s) = \max_{a_n \in \mathcal{A}} \left\{ r(s_n, a_n) \gamma \sum_{s_{n+1} \in \mathcal{S}} P_{s_n \to s_{n+1}}^{a_n} V_{n+1}^{\pi^*}(s_{n+1}) \right\}$ *#Compute forwarding decisions*
  end for
end for
return $\pi^* \in \{$*green*, *red*$\}$ *#Forwarding participation decisions*

We conclude the preview of WHARPNR-HEU by showcasing of the data forwarding procedure (Fig. 7.1). Node i, with hop count ℓ_i, has a packet to transmit. Nodes j_1, j_2, j_3 and j_4 are within its wake-up radio range, with hop count $\ell_i - 1$. Node i broadcasts a wake-up sequence to wake up these nodes among its neighboring nodes that elected to participate in the relay selection process in the current decision epoch, i.e., action is *green*. Nodes j_1 and j_4 got a *green* as a result of running the MDP, updated their second wake-up address, and can receive wake-up sequences. Nodes j_2 and j_3 decided not to participate to the relay selection process, i.e., action is *red*. Upon reception of the wake-up sequence nodes j_1 and j_4 compute the CTS delays δ_{j_1} and δ_{j_4}, respectively. Once the CTS delay has passed, both nodes activate their main radio, reply with a CTS packet to sender i and activate the data packet waiting timer. Node i transmits the data packet to the node that transmitted the CTS first, i.e., node j_1 in our example. After reception of the data packet, node j_1 replies with an ACK packet and turns off its main radio. Node i goes back to sleep after receiving the ACK from node j_1, and any subsequent CTS packets are ignored. As node j_4 does not receive the data packet within the set waiting time it goes back to sleep.

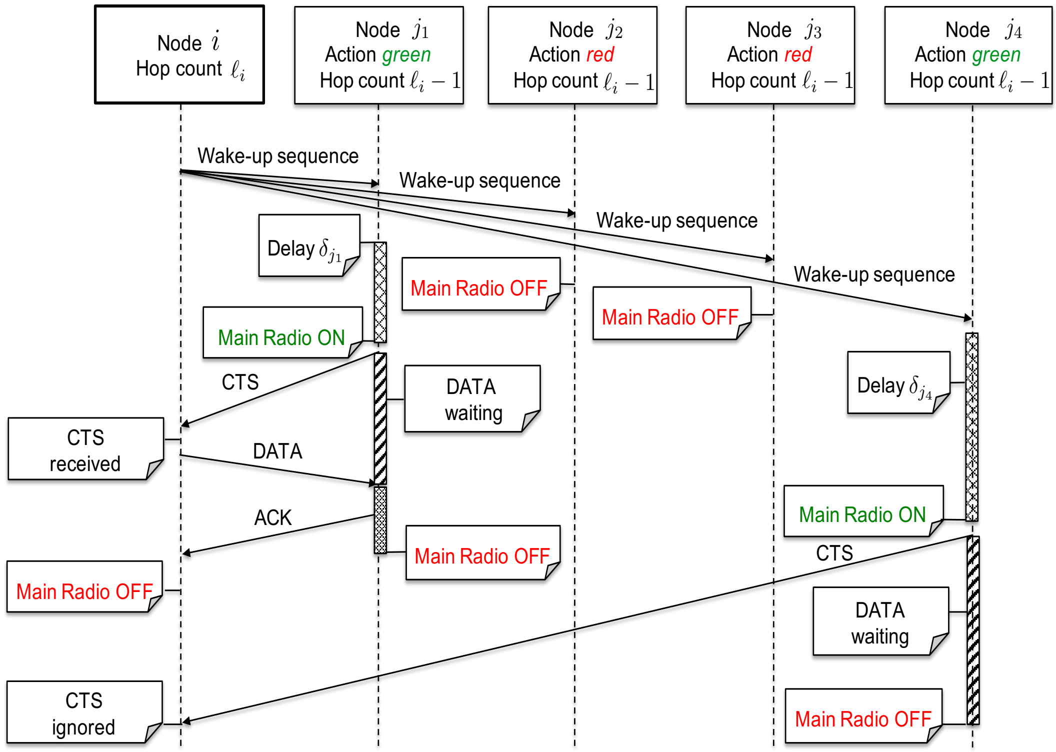


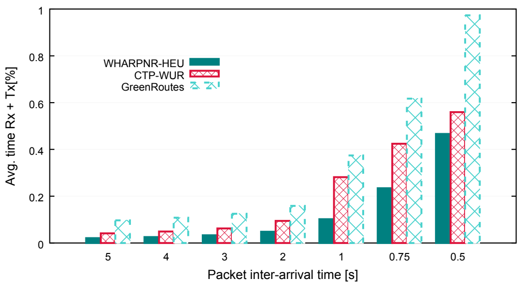
Fig. 7.1: WHARPNR-HEU forwarding: An example.

7.2 Performance evaluation

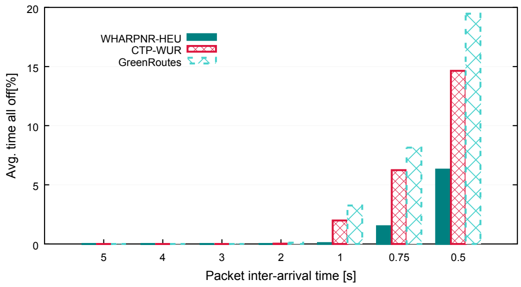
In this section we discuss a preliminary set of results of a simulation-based performance evaluation of WHARPNR-HEU. Its performance is compared to that of GreenRoutes [8], the forwarding strategy introduced in Chapter 3, and to that of CTP-WUR [36].

7.2.1 Simulation setup

All investigated protocols have been implemented in the GreenCastalia [15] simulator. We consider connected networks with 63 sensor nodes and one sink node. Sensor nodes are randomly and uniformly distributed over a rectangular area of size $224 \times 56\text{m}^2$, whereas the sink node has coordinates (0,0). The transmission range of the nodes on the main and wake-up radio is set to 60m and 25m, respectively. Therefore, the average degree of a node is 7 nodes on the wake-up radio, which corresponds to a medium/dense network scenario. We consider that the packet



(a) Tx and Rx on main radio



(b) All-off ratio

Fig. 7.2: Left: Per node average time spent on the main radio Rx and Tx for increasing traffic (% of active time); Right: Per node average all-off time for increasing traffic (% of the simulation time).

inter-arrival time of the network ranges in {5,4,3,2,1,0.75,0.5}s. Sensor nodes are equipped with harvesting capabilities, where half of them harvest energy through small wind turbines and the rest using solar panels. The rest of the parameters used in this performance evaluation, including the considered energy models, remain as in Chapter 3.2.1.1. We consider that the size of data and control packets are the same as in Chapter 5.2. The channel data rate is set to 250Kbps, while the rate for transmitting wake-up sequences is set to 5Kbps.

7.2.2 Simulation results

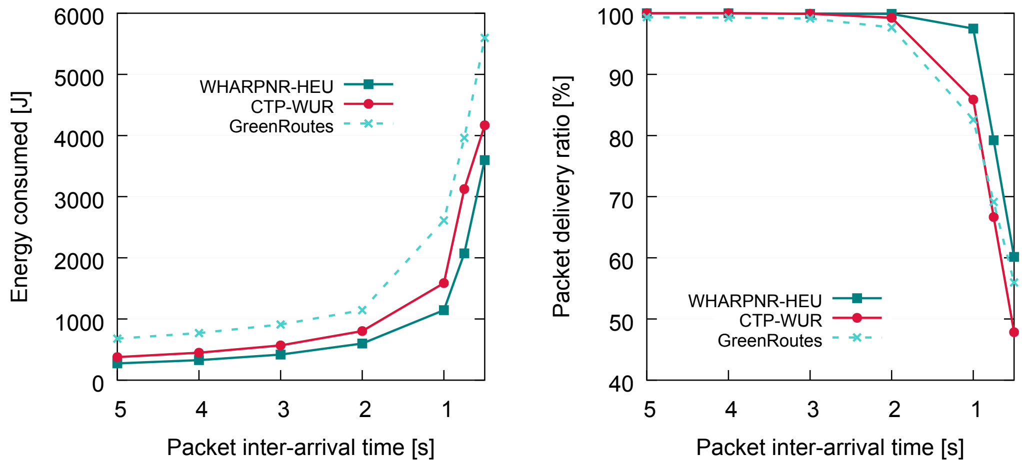
The performance of WHARPNR-HEU is assessed through the investigation of the following metrics.

1. *The time spent on main radio Rx and Tx*, defined as the average percentage of the active time of a node spent on transmitting and on receiving using the main radio.
2. *The total energy consumption*, defined as the total energy consumed by the network to successfully deliver packets to the sink.
3. *The packet delivery ratio*, i.e., the percentage of packets successfully delivered to the sink

All results are obtained by averaging the data of a number of simulation runs, which achieves a statistical confidence of 95% within a 5% precision.

7.2.2.1 Time spent on main radio Rx and Tx

Fig. 7.2a concerns results on the average time spent by a node with its main radio Rx and Tx activated. At the lowest traffic, WHARPNR-HEU uses the main radio



(a) Total energy consumption **(b)** Packet delivery ratio

Fig. 7.3: Performance comparison of CTP-WUR, GREENROUTES and WHARPNR-HEU for increasing traffic.

approximately 4.6 and 2 times less than GREENROUTES and CTP-WUR, respectively. Nodes running GREENROUTES transmit RTS/CTS packets every time they have to forward a data packet. In CTP-WUR nodes periodically broadcast control packets for building or maintaining the tree topology. In WHARPNR-HEU, only one type of control packet exists: Neighboring nodes which are next-hop relay candidates, show their availability by broadcasting a CTS packet (source nodes do not transmit RTS packets). This allows nodes running WHARPNR-HEU to activate their main radio less than GREENROUTES and CTP-WUR. At the highest traffic, nodes in WHARPNR-HEU activate their main radio for approximately 16% and 52% less time than GREENROUTES and CTP-WUR, respectively. In addition to the lower packet overhead that WHARPNR-HEU exhibits, this is also because of the higher number of interference among packets, and of a higher number of packet re-transmissions with increasing traffic.

7.2.2.2 Total energy consumption

The percentage of the total energy consumed by each forwarding strategy is shown in Fig. 7.3a. Independently of traffic, WHARPNR-HEU always outperforms all other approaches despite the computational cost of solving the MDP using the heuristics solution to take forwarding actions. In particular, WHARPNR-HEU spends up to 36% and 14% less energy than GREENROUTES and CTP-WUR, respectively. The main radio, as expected, drains most of the available energy, independently of the forwarding strategy. This is due to the fact that nodes spend longer time with their main radio activated for routing activities. In fact, even though nodes wake up only when required, control and data packets are transmitted using the main radio which compels nodes to keep their main radio on for longer periods. As shown in Fig. 7.3a CTP-WUR and GREENROUTES keep their main radio active for longer timer periods resulting to higher energy consumption.

7.2.2.3 Packet delivery ratio

Fig. 7.3b depicts the average packet delivery ratio of CTP-WUR, GREENROUTES and WHARPNR-HEU for increasing traffic. The best performance is shown by WHARPNR-HEU, which always delivers more packets to the sink while consuming less energy. While all the three protocols perform well for inter-arrival times higher than 1s, as expected, the PDR decreases with increasing traffic. This is due to a higher level of interference, energy consumption and more nodes that go all-off resulting to a higher number of contentions for relay selection and in a higher number of packet re-transmissions(Fig. 7.2b). At the highest traffic, WHARPNR-HEU delivers approximately 9% and 26% more packets to the sink than GREENROUTES and CTP-WUR, respectively. The tree-based topology used in CTP-WUR dictates that a node has only one possible relay to forward a data packet. When nodes are operational for shorter time periods, sender nodes experience difficulties on delivering data packets to the predefined grandparent. In particular, when the grandparent of a sender node is not reachable, in terms of packet delivery, the sender instead attempts to forward the packet to its parent after a number of unsuccessful re-transmissions. If this transmission also fails, the packet is dropped. On the other hand, WHARPNR-HEU and GREENROUTES have more candidate relays to forward the data packet depending on the forwarding mechanism in each case allowing them to deliver more packets even at the highest traffic where more nodes run out of energy and remain in an all-off state. However, we observed that on average, WHARPNR-HEU nodes are operational for at least 94% of the time (Fig. 7.2b). A higher number of active nodes results in a higher number of available relays and, ultimately, in higher packet delivery ratio.

7.3 Conclusions

In this chapter we presented a preview of a forwarding strategy that focuses on the design of an optimized version of the work introduced in Chapter 4, named WHARPNR-HEU. By eliminating the transmission of control packets we reduce the latency incurred by the forwarding strategy as well as the energy consumed by the nodes in the network, while retaining the exemplary performances of WHARP such as high packet delivery ratio. In this set of simulation experiments, we consider the use of higher data rates to transmit the wake-up sequences, i.e., 5Kbps.

In addition, the GreenCastalia implementation of WHARPNR-HEU included the computational energy cost incurred by solving the MDP using the W-HEU solution method introduced in Chapter 6. In the final version of WHARPNR-HEU we plan to include of full set of simulation results under a diverse range of realistic scenarios while performing more intensive performance comparisons with existing wake-up radio-based data forwarding strategies. Our ultimate goal is to implement the data forwarding strategy on a real-life deployment in order to evaluate its performance, using the Magonode++ motes [49].

8 Concluding Remarks

Wireless sensor networks face significant limitations in terms of memory, energy, and computational power. This has prompted the design of protocols at all layers of the networking stack that are aware of these limitations, seeking to obtain performance that is adequate to support critical WSN applications.

This thesis shows a clear trend towards networks whose nodes and their protocol stack are fully aware of both energy harvesting and wake-up radio capabilities: Green wireless networks. Studies on protocol design, on performance comparisons, and on solution testing are still largely uncharted territory. This thesis aims at starting the exploration of data forwarding solutions for green wireless networks, and to provide insights about which design choices are best for achieving the performance needed by critical applications of WSNs.

In Chapter 3 we presented GREENROUTES, a cross-layer routing protocol for green networks that efficiently selects next-hop relays based on their distance from the sink, and, greedily, on the available along routes to the sink. Results clearly show that GREENROUTES outperforms existing solutions with respect to every performance metric that we considered, regardless of traffic and of energy source considered, either sun or wind.

In Chapter 4 we presented WHARP, for Wake-up and HARvesting-based energy-Predictive forwarding. WHARPis a cross-layer data forwarding strategy for green wireless networks that exploits the use of learning-based techniques to take optimum forwarding decisions based on forecast energy and expected traffic. Our simulation results indicate that the proactive nature of WHARP allows nodes in the network to remain active for long periods by taking optimal actions.

In Chapter 5 we focused on a comparative performance evaluation of three different data forwarding strategies for green wireless networks, namely, CTP-WUR, GREEN-ROUTES, and WHARP, which have been shown to outperform previous state-of-art solutions. The performance evaluation provided us with a set of insights into the impact on the network performance of different design choices used in each routing strategy. Specifically, results show that tree-based solutions obtain lower packet delivery ratio than that obtained by solutions that include energy awareness in route decisions. However, the relaying feature of CTP-WURallows faster packet delivery and lower energy consumptions, calling for further improvements and optimization on cross-layer approaches such as that of GREENROUTES and WHARP.

Chapter 6 investigated the impact on protocol performance of local computational requirements of learning techniques. While superior performance is being obtained by taking key protocol decisions based on the outcome of local learning-based computations, the computational requirements of such approaches should not be neglected. We presented a heuristic solution that closely approximates the MDP trading off optimality of the WHARP solution method, which was presented in Chapter 4, for considerably lighter computational requirements. Results show that solving the MDP using standard methods incurs energy expenditures by far superior to that required by wireless communication. The performance comparison of the two solution methods (W-BVI vs. W-HEU) included the computational cost measured in each case using real hardware. Our heuristics solution, i.e., W-HEU, outperforms W-BVI on key metrics such as energy consumption and packet delivery ratio, making up for the lost optimality of BVI through the remarkable energy savings of its lighter computational requirements

Finally, in Chapter 7 we briefly introduced WHARPNR-HEU, an optimized version of WHARP. WHARPNR-HEU takes advantage of the learning-based forwarding procedure presented in WHARP while achieving better performance with reduced end-to-end latency and energy consumption by eliminating the RTS phase and by including optimal forwarding decisions in the wake-up semantic addressing. Through a first set of simulation results we showcase that WHARPNR-HEU outerperforms existing routing solutions for green wireless networks despite the computational cost of solving the MDP to take forwarding actions.

Bibliography

[1] Sensirion AG. *SHT1x datasheet: Humidity and temperature sensor IC*. 2011 (cit. on p. 17).

[2] H. Alemdar and C Ersoy. “Wireless sensor networks for healthcare: a survey”. In: *Computer Networks* 54.15 (2010), pp. 2688–2710 (cit. on p. 1).

[3] M. A. Alsheikh, S. Lin, D. Niyato, and H. P. Tan. “Machine learning in wireless sensor networks: algorithms, strategies, and applications”. In: *IEEE Communications Surveys Tutorials* 16.4 (2014), pp. 1996–2018 (cit. on pp. 11, 51).

[4] N. Arastouie, M. Sabaei, and H. S. Shahreza. “A novel approach for trade-off between computation and communication cost in wireless sensor networks”. In: *Proceedings of 12th International Conference on Advanced Communication Technology, ICACT 2010*. Feb. 11-14 2010. (Cit. on p. 12).

[5] A. A. Babayo, M. H. Anisi, and I. Ali. “A Review on energy management schemes in energy harvesting wireless sensor networks”. In: *Renewable and Sustainable Energy Reviews* 76 (2017), pp. 1176–1184 (cit. on p. 8).

[6] Suman Banerjee and Archan Misra. “Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks”. In: *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking &Amp; Computing*. MobiHoc ’02. Lausanne, Switzerland: ACM, 2002, pp. 146–156 (cit. on p. 7).

[7] S. Basagni, V. Di Valerio, G. Koutsandria, and C. Petrioli. “On the impact of local computation over routing performance in green wireless networks”. In: *Proceedings of the 19th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, IEEE WoWMoM 2018*. Chania, Greece, June 12-15 2018. (Cit. on p. 5).

[8] S. Basagni, V. Di Valerio, G. Koutsandria, and C. Petrioli. “Wake-up radio-enabled routing for green wireless sensor networks”. In: *Proceedings of the 86th IEEE Vehicular Technology Conference, IEEE VTC 2017-Fall*. Toronto, Ontario, Canada, September 24-27 2017. (Cit. on pp. 5, 8, 39, 42, 67).

[9] S. Basagni, V. Di Valerio, G. Koutsandria, C. Petrioli, and D. Spenza. "WHARP: a wake-up radio and harvesting-based forwarding strategy for green wireless networks". In: *Proceedings of the 14th IEEE International Conference on Mobile Ad hoc and Sensor Systems, IEEE MASS 2017*. Orlando, FL, USA, October 22-25 2017. (Cit. on pp. 5, 8, 39, 43, 52).

[10] S. Basagni, V. Di Valerio, G. Koutsandria, C. Petrioli, and D. Spenza. *WHARPNR-HEU: Exploiting wake-up radios and energy harvesting for green wireless networks*. In preparation for journal submission. 2018 (cit. on p. 5).

[11] S. Basagni, G. Koutsandria, and C. Petrioli. "A comparative performance evaluation of wake-up radio-based data forwarding for green wireless networks". In: *Proceedings of the 27th IEEE International Conference on Computer Communications and Networks, IEEE ICCCN 2018*. Hangzhou, China, July 30-August 2 2018. (Cit. on p. 5).

[12] S. Basagni, M. Y. Naderi, Chiara Petrioli, and Dora Spenza. "Wireless sensor networks with energy harvesting". In: *Mobile Ad Hoc Networking: The Cutting Edge Directions*. Hoboken, NJ: John Wiley and Sons, Inc., 2013. Chap. 20, pp. 701–736 (cit. on pp. 2, 9).

[13] S. Basagni, C. Petrioli, and D. Spenza. "CTP-WUR: the collection tree protocol in wake-up radio WSNs for critical applications". In: *Proceedings of IEEE International Conference on Computing, Networking and Communications, ICNC 2016*. Kauai, HI, February 15–18 2016. (Cit. on pp. 2, 8, 11, 39, 40, 65).

[14] S. S. Beheshtiha, H. P. Tan, and M. Sabaei. "Opportunistic routing with adaptive harvesting-aware duty cycling in energy harvesting WSN". In: *Proceedings of the 15th IEEE International Symposium on Wireless Personal Multimedia Communications, IEEE WPMC 2012*. September 24–27 2012. (Cit. on pp. 9, 10).

[15] D. Benedetti, C. Petrioli, and D. Spenza. "GreenCastalia: an energy-harvesting-enabled framework for the castalia simulator". In: *Proceedings of 11th ACM Conference on Embedded Networked Sensor Systems, ACM SenSys 2013*. Rome, Italy, November 11-14 2013. (Cit. on pp. 13, 17, 39, 67).

[16] A. Bogliolo, S. Delpriori, E. Lattanzi, and A. Seraghiti. "Self-adapting maximum flow routing for autonomous wireless sensor networks". In: *Cluster Computing* 14.1 (2011), pp. 1–14 (cit. on p. 8).

[17] A. Boulis. "Castalia: revealing pitfalls in designing distributed algorithms in WSN". In: *Proceedings of 5th ACM Conference on Embedded Networked Sensor Systems, ACM SenSys 2007*. Sydney, Australia, November 6-9 2007. (Cit. on p. 17).

[18] A. Cammarano, C. Petrioli, and D. Spenza. "Online energy harvesting prediction in environmentally powered wireless sensor networks". In: *IEEE Sensors Journal* 16.17 (2016), pp. 6793–6804 (cit. on p. 30).

[19] A. Cammarano, D. Spenza, and C. Petrioli. “Energy-harvesting WSNs for structural health monitoring of underground train tunnels”. In: *Proceedings of 32nd IEEE International Conference on Computer Communications, IEEE INFOCOM WKSHPS 2013*. Turin, Italy, April 14-19 2013 (cit. on p. 1).

[20] L. Chen, S. Cool, H. Ba, W. Heinzelman, I. Demirkol, U. Muncuk, K. Chowdhury, and S. Basagni. “Range extension of passive wake-up radio systems through energy harvesting”. In: *IEEE International Conference on Communications (ICC) 2013*. Budapest, Hungary, June 2013. (Cit. on p. 11).

[21] L. Chen, J. Warner, P. L. Yung, D. Zhou, W. Heinzelman, I. Demirkol, U. Muncuk, K. Chowdhury, and S. Basagni. “REACH2-Mote: A Range-Extending Passive Wake-Up Wireless Sensor Node”. In: *ACM Trans. Sen. Netw.* 11.4 (Dec. 2015), 64:1–64:33 (cit. on p. 11).

[22] S. De. “On hop count and euclidean distance in greedy forwarding in wireless ad hoc networks”. In: *IEEE Communications Letters* 9.11 (2005), pp. 1000–1002 (cit. on pp. 7, 8).

[23] K. Egoh and S. De. “A Multi-Criteria Receiver-Side Relay Election Approach in Wireless Ad Hoc Networks”. In: *IEEE Military Communications Conference (MILCOM) 2006*. 2006, pp. 1–7 (cit. on p. 7).

[24] S. Ehsan and B. Hamdaoui. “A survey on energy-efficient routing techniques with QoS assurances for wireless multimedia sensor networks”. In: *IEEE Communications Surveys & Tutorials* 14.2 (2012), pp. 265–278 (cit. on p. 7).

[25] *Electric double layer capacitors (gold capacitor)*. URL: `http://www.farnell.com/datasheets/2015895.pdf` (cit. on p. 17).

[26] Z. A. Eu, H.-P. Tan, and W. K.G. Seah. “Design and performance analysis of MAC schemes for wireless sensor networks powered by ambient energy harvesting”. In: *Ad Hoc Networks* 9.3 (2011), pp. 300–323 (cit. on p. 9).

[27] Zhi Ang Eu, Hwee-Pink Tan, and Winston KG Seah. “Opportunistic Routing in Wireless Sensor Networks Powered by Ambient Energy Harvesting”. In: *Computer Networks* 54.17 (2010), pp. 2943–2966 (cit. on pp. 8, 10).

[28] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. “Collection tree protocol”. In: *Proceedings of 7th ACM Conference on Embedded Networked Sensor Systems, ACM SenSys 2009*. Berkeley, CA, November 4–6 2009. (Cit. on pp. 39, 40).

[29] G. Han, Y. Dong, H. Guo, L. Shu, and D. Wu. “Cross-layer optimized routing in wireless sensor networks with duty cycle and energy harvesting”. In: *Wireless Communications and Mobile Computing* 15.16 (2015), pp. 1957–1981 (cit. on p. 10).

[30] D. Hasenfratz, A. Meier, C. Moser, J. J. Chen, and L. Thiele. “Analysis, comparison, and optimization of routing protocols for energy harvesting wireless sensor networks”. In: *Proceedings of IEEE ISUTC 2010*. 2010 (cit. on p. 9).

[31] Texas Instruments. *CC1101 datasheet*. rev. SWRS061I. 2013 (cit. on p. 18).

[32] K. Kaushik, D. Mishra, S. De, K. R. Chowdhury, and W. Heinzelman. “Low-Cost Wake-Up Receiver for RF Energy Harvesting Wireless Sensor Networks”. In: *IEEE Sensors Journal* 16.16 (2016), pp. 6270–6278 (cit. on p. 11).

[33] j. A. Khan, H. K. Qureshi, and A. Iqbal. “Energy management in wireless sensor networks”. In: *Computers and Electrical Engineering* 41 (2015), pp. 159–176 (cit. on p. 8).

[34] Yvonne Klöpping. “Active Implantable Medical Devices: winning the Power Struggle”. In: *European Medical Device Technology* 3.2 (2012), pp. 38–41 (cit. on p. 1).

[35] R. V. Kulkarni, A. Forster, and G. K. Venayagamoorthy. “Computational Intelligence in Wireless Sensor Networks: A Survey”. In: *IEEE Communications Surveys Tutorials* 13.1 (2011), pp. 68–96 (cit. on p. 11).

[36] T. Kumberg, M. Schink, L. M. Reindl, and C. Schindelhauer. “T-ROME: A simple and energy efficient tree routing protocol for low-power wake-up receivers”. In: *Ad Hoc Networks* 59 (2017), pp. 97–115 (cit. on pp. 8, 10, 11, 67).

[37] J. Kuruvila, A. Nayak, and I. Stojmenovic. “Hop count optimal position based packet routing algorithms for ad hoc wireless networks with a realistic physical layer”. In: *2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE Cat. No.04EX975)*. 2004, pp. 398–405 (cit. on p. 7).

[38] J. Lee, K. Kapitanova, and S. H. Son. “The price of security in wireless sensor networks”. In: *Computer Networks* 54.17 (Dec. 2010), pp. 2967–2978 (cit. on p. 12).

[39] J. Marašević, C. Stein, and G. Zussman. “Max-min fair rate allocation and routing in energy harvesting networks: algorithmic analysis”. In: *Proc. of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM MobiHoc 2014*. Philadelphia, PA, USA, August 11 - 14 2014. (Cit. on p. 9).

[40] Gina Martinez, Shufang Li, and Chi Zhou. “Wastage-aware routing in energy-harvesting wireless sensor networks”. In: *IEEE Sensors Journal*. Vol. 14. 9. 2014, pp. 2967–2974 (cit. on pp. 3, 9, 13, 19, 25, 33).

[41] D. Mishra, S. De, S. Jana, S. Basagni, K. R. Chowdhury, and W. Heinzelman. “Smart RF energy harvesting communications: Challenges and opportunities”. In: *IEEE Communications Magazine* 53.4 (2015), pp. 70–78 (cit. on p. 9).

[42] D. K. Noh and K. Kang. "Balanced energy allocation scheme for a solar-powered sensor system and its effects on network-wide performance". In: *Journal of Computer and System Sciences* 77.5 (2011), pp. 917 –932 (cit. on pp. 30, 34).

[43] *NREL: measurement and instrumentation data center*. URL: `http://www.nrel.gov/midc` (cit. on p. 17).

[44] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl. "Performance evaluation and comparative analysis of subcarrier modulation wake-up radio systems for energy-efficient wireless sensor networks". In: *Sensors* 14.1 (2013), pp. 22–51 (cit. on p. 2).

[45] N. Pais, B.K. Cetin, N. Pratas, F.J. Velez, N. Prasad, and R. Prasad. "Cost-benefit aware routing protocol for wireless sensor networks with hybrid energy storage system". In: *Journal of Green Engineering* 1.2 (2011), pp. 189–208 (cit. on p. 9).

[46] B. Panigrahi, S. De, B. S. Panda, and J. . Lan Sun Luk. "Network lifetime maximising distributed forwarding strategies in ad hoc wireless sensor networks". In: *IET Communications* 6.14 (2012), pp. 2138–2148 (cit. on p. 7).

[47] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados. "Energy-efficient routing protocols in wireless sensor networks: a survey". In: *IEEE Communications Surveys Tutorials* 15.2 (2013), pp. 551–591 (cit. on p. 7).

[48] A. Pantelopoulos and N. G. Bourbakis. "A survey on wearable sensor-based systems for health monitoring and prognosis". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 40.1 (2010), pp. 1–12 (cit. on p. 1).

[49] M. Paoli, Dora Spenza, Chiara Petrioli, M. Magno, and L. Benini. "MagoNode++: A Wake-up Radio-enabled Wireless Sensor Mote for Energy-neutral Applications". In: *Proceedings of ACM/IEEE IPSN 2016*. 2016, pp. 1–2 (cit. on pp. 11, 18, 56, 71).

[50] C. Petrioli, M. Nati, P. Casari, M. Zorzi, and S. Basagni. "ALBA-R: Load-balancing geographic routing around connectivity holes in wireless sensor networks". In: *IEEE Transactions on Parallel and Distributed Systems* 24.3 (2014), pp. 529–539 (cit. on p. 10).

[51] C. Petrioli, D. Spenza, P. Tommasino, and A. Trifiletti. "A novel wake-up receiver with addressing capability for wireless sensor nodes". In: *Proceedings of the 10th IEEE International Conference on Distributed Computing in Sensor Systems, IEEE DCOSS 2014*. Marina Del Rey, California, May 26-28 2014 (cit. on pp. 2, 8, 10).

[52] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli. “Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey”. In: *IEEE Communications Surveys Tutorials* 19.4 (2017), pp. 2117–2157 (cit. on p. 10).

[53] R. V. Prasad, S. Devasenapathy, V. S. Rao, and J. Vazifehdan. “Reincarnation in the ambiance: Devices and networks with energy harvesting”. In: *IEEE Communications Surveys & Tutorials* 16.1 (2014), pp. 195–213 (cit. on p. 8).

[54] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014 (cit. on pp. 28, 31, 51–53).

[55] T. S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall Communication Engineering and Emerging Technologies. Upper Saddle River, NJ: Prentice Hall PTR, 2002 (cit. on p. 17).

[56] S. Riaz, H. K. Qureshi, and M. Saleem. “Performance evaluation of routing protocols in energy harvesting D2D network”. In: *Proceedings of ICE Cube 2016*. Quetta, Pakistan, 2016, pp. 251–255 (cit. on p. 9).

[57] Farhan I Simjee and Pai H Chou. “Efficient charging of supercapacitors for extended lifetime of wireless sensor nodes”. In: *IEEE Transactions on Power Electronics* 23.3 (2008), pp. 1526–1536 (cit. on p. 17).

[58] P. Spachos, P. Chatzimisios, and D. Hatzinakos. “Energy aware opportunistic routing in wireless sensor networks”. In: *IEEE Globecom Workshops 2012*. Dec. 2012 (cit. on p. 7).

[59] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli. “Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems”. In: *Proceedings of IEEE International Conference on Computer Communications, IEEE INFOCOM 2015*. Toronto, Ontario, Canada, April 2015 (cit. on pp. 2, 8, 10, 11, 17).

[60] M. Tahir and R. Farrell. “Optimal communication-computation tradeoff for wireless multimedia sensor network lifetime maximization”. In: *Proceedings of IEEE Wireless Communications and Networking Conference, IEEE WCNC 2009*. Budapest, Hungary (cit. on p. 12).

[61] M. Vodel, M. Lippmann, and W. Hardt. “WRTA - Wake-Up-Receiver optimised routing and topology optimisation approach”. In: *IEEE International Conference on ITS Telecommunications 2012*. 2012, pp. 329–334 (cit. on pp. 8, 11).

[62] A. S. Wander, N. Gura, H. Eberle, V. Gupta, and S. C. Shantz. “Energy analysis of public-key cryptography for wireless sensor networks”. In: *Proceedings of IEEE International Conference on Pervasive Computing and Communications 2005, IEEE PerCom 2015*. Kauai Island, HI, USA (cit. on p. 12).

[63] C. Wei, J. Yang, Y. Gao, and Z. Zhang. "Cluster-based routing protocols in wireless sensor networks: A survey". In: *Proceedings of the International Conference on Computer Science and Network Technology, ICCSNT 2011*. Harbin, China, 2011 (cit. on p. 7).

List of Figures