2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems

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

Stefano Basagni,* Valerio Di Valerio,† Georgia Koutsandria,† Chiara Petrioli,† Dora Spenza†
$\star$ ECE Dept., Northeastern University, Boston, MA
Email: basagni@ece.neu.edu
† Computer Science Department, University of Rome "La Sapienza," Rome, Italy
Email: {divalerio, koutsandria, petrioli, spenza}@di.uniroma1.it

***Abstract*—Green wireless networks are characterized by devices that are pervasively deployed and that harvest energy from the surrounding environment. Devices are also endowed with low-power triggering techniques (e.g., wake-up radios) to obviate costly idle communication times. In this paper, we present a novel 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.**

I. 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. Wireless networking enhances our ability to observe and interact with the physical environment by enabling many applications such as remote tracking, monitoring, control, data collection, and surveillance. The widespread adoption of these networks for applications ranging from structural health monitoring of critical infrastructures [1] 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., sensors), including narrow energy availability and rigorous 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 [3], [4]. Therefore, research efforts to obviate the joint problems of battery replacement and achieve 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 the blend of two promising technologies: Energy harvesting and low-power radio triggering. 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 [5]. However, the stochastic nature of harvestable energy sources exposes the need of further technological advancements and energy efficient techniques. Emerging low-power radio triggering techniques, such as wake-up radios [6], [7], 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 [6]. 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 [6], [8], [9]. 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. While research on energy harvesting-based wireless networks has been flourishing [5], [10], [11], [12], and solutions for networks employing wake-up radio technology are being proposed and tested [6], [8], [9], [13], exploring the benefits of combining these technologies is still uncharted territory.

In this paper 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.

2155-6814/17 $31.00 © 2017 IEEE
DOI 10.1109/MASS.2017.18

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 network data collector node (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 a previously proposed strategy for data forwarding in wireless networks with energy harvesting, namely, the Energy Harvesting Wastage-Aware (EHWA) forwarding strategy [14]. We implemented WHARP and a version of EHWA extended to use wake-up radios in the open-source simulator GreenCastalia [15]. 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 paper is organized as follows. In Section II we describe notation and the networking scenario considered in this paper. Section III describes WHARP in details. Performance evaluation results of WHARP and EHWA are shown in Section IV. Section V presents a literature review. Finally, Section VI concludes the paper.

II. 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. 1.

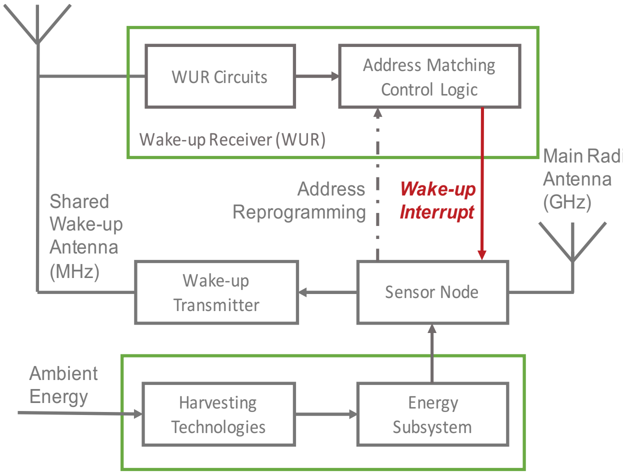


Fig. 1: The architecture of a green node.

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 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:

$$\max_{\pi} \; V_0^{\pi}(s) = E_s^{\pi}\left\{\sum_{n=0}^{N} \gamma^n r(s_n, a_n) \middle| s_0 = s\right\} \; \forall s \in \mathcal{S}, \quad (1)$$

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 (1) since, for each state $s \in \mathcal{S}$, the optimal policy π^* maximizing the value functions satisfies the Bellman optimality equations:

$$\begin{aligned} V_n^{\pi^*}(s) = \max_{a \in A(s)} \Bigg\{ & r(s_n, a_n) \\ & + \gamma \sum_{s' \in \mathcal{S}} P_{s_n \rightarrow s_{n+1}}^{a_n} V_{n+1}^{\pi^*}(s_{n+1}) \Bigg\}, \end{aligned} \quad (2)$$

where $P_{s_n \rightarrow s_{n+1}}^{a_n}$ is the transition probability from state s_n to state s_{n+1} after taking action a_n. Equations (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 [16]. 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.

III. 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 III-A. 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 III-B. 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.

A. 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 II). 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 $b \in \{0, \ldots, B_{max}\}$. 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 energy spent for sensing and for transmitting the corresponding data.[1] The overall energy available for packet forwarding in the nth epoch is thus $e_n = b_n + h_n - e_n^x$. Taking action a_n in state s_n transitions the node in epoch $n+1$ to the following state:

$$s_{n+1} = \begin{cases} e_n & \text{if } a_n = a_d \wedge b_n + h_n > e_n^x \\ e_n - e_n^{tx} & \text{if } a_n = a_f \wedge b_n + h_n > e_n^{tx} + e_n^x \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

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 [17] or AEWMA [18].

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

$$P_{s_n \to s_{n+1}}^{a_f} = \begin{cases} p^{e_{tx}}(e_n^{tx}) & \text{if } b_{n+1} > 0 \\ \displaystyle\sum_{e_n^{tx}=e_n}^{\infty} p^{e_{tx}}(e_n^{tx}) & \text{if } b_{n+1} = 0. \end{cases} \quad (4)$$

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 $P_{s_n \to s_{n+1}}^{a_d} = 1$.

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:

$$r(s_n, a_f) = r \cdot \sum_{e_n^{tx}=0}^{e_n} p^{e_{tx}}(e_n^{tx}) - c \cdot \sum_{e_n^{tx}=e_n}^{\infty} p^{e_{tx}}(e_n^{tx}), \quad (5)$$

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 (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 (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 [16], 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.

B. 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:

$$\delta = \left(1 - \frac{b_j}{B_{max}}\right) \cdot \delta_{\text{MAX}} + \delta_{\text{RAND}}, \quad (6)$$

where b_j is node j current energy, δ_{MAX} is the 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. 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.

---

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

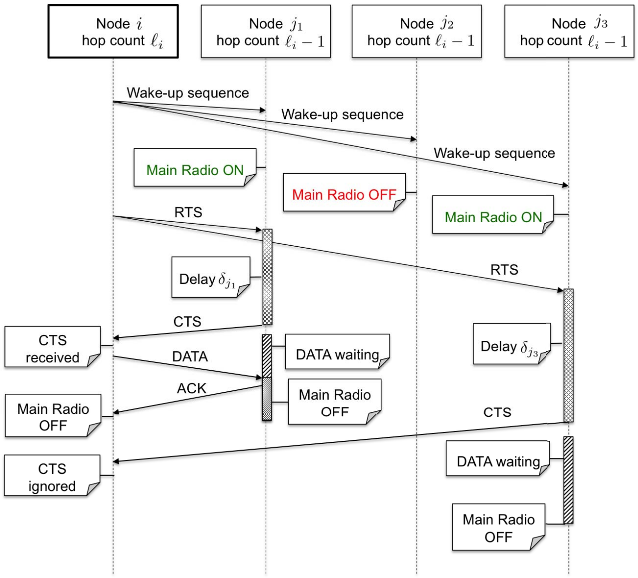


Fig. 2: WHARP forwarding: An example.

We showcase an example of the WHARP forwarding strategy in Fig. 2. 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.

IV. PERFORMANCE EVALUATION

We evaluate the effectiveness of the WHARP forwarding strategy by simulation-based experiments. We also compare its performance to a baseline strategy for data forwarding previously proposed for networks with energy harvesting nodes, namely, the Energy Harvesting Wastage-Aware protocol (EHWA) [14]. We implemented both protocols in the open-source simulator GreenCastalia [15], an extension of the Castalia simulator [19]. GreenCastalia features modules for energy harvesting from heterogeneous sources, and accurately models energy-related aspects of wireless networks. We start by providing a summary of the EHWA strategy. We then describe the considered simulation scenarios, the metrics that we investigated, and the simulation results.

A. The EHWA strategy

EHWA is an on-demand dynamic source routing-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. EHWA has been extended to exploit wake-up radio capabilities.

B. Simulation scenarios and parameters

We consider connected green networks made up of a sink and 119 sensor nodes, each capable of sensing and of communicating wirelessly to each other. 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. Power is supplied to each node via a supercapacitor with maximum operating voltage of 2.3V and capacitance of 50F [20]. We decided for a battery-less 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 [21]. 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. Harvesting traces for both sources are obtained from the National Renewable Laboratory at Oak Ridge [22]. Nodes act as relay and source nodes. 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 [23]. Once a sensor measurement is taken, a data packet is generated that needs to be delivered to the sink. This data generation occurs according to a Poisson process of intensity λ packets per second. In our simulation results, we make use of the inter-arrival time between packets, which is defined as $iaTime = 1/\lambda$, and ranges from 1 to 150 seconds. Once a packet is generated, a source node is randomly and uniformly chosen among the nodes. Data packets have a size of 58B, including the application payload (temperature measurements), and headers added by lower layers. The channel data rate r_c is set to 250Kbps.

TABLE I: 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 [18]

The energy model is that of the MagoNode++ mote, extended to comprise energy harvesting and wake-up radio capabilities [24]. Channel and radio models are set based on 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 lognormal shadowing model is used to estimate the average path loss between nodes [25]. Packet collisions are determined using an additive interference model, by linearly summing-up at the receiver the effect of multiple signals simultaneously sent. The wake-up radio is modeled based on the specifications of a wake-up radio receiver prototype of our design that we tested and characterized [6]. Wake-up sequences are sent at $+10$dBm using the low-power CC1101 transceiver from the Texas Instruments [26]. They are 1B long, and are transmitted at 1Kbps. The power consumption of the receiver is 1.071μW. Its sensitivity is -55dBm, leading to a maximum wake-up range R_w of 45m. 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 simulation parameters are summarized in Table I, which also shows the values chosen for WHARP-specific parameters.

C. Investigated metrics

The performance of WHARP and EHWA is compared with respect to the following metrics.

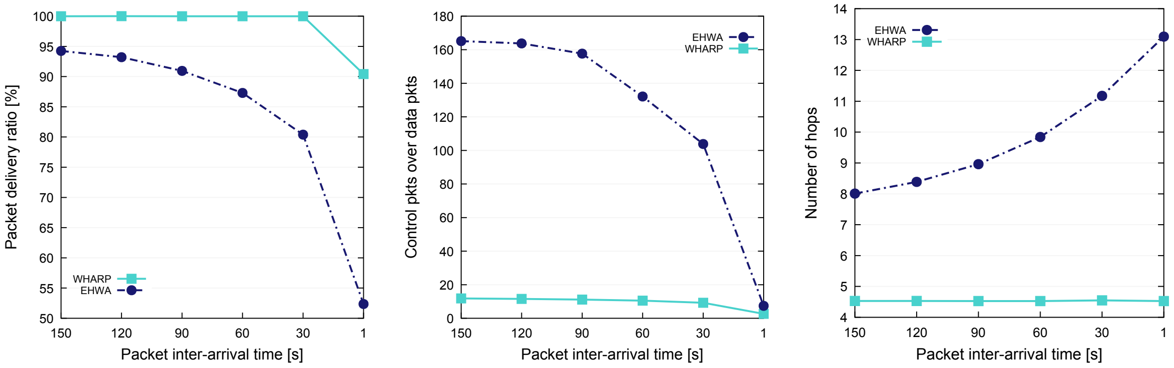
1) The *packet delivery ratio* (PDR), i.e., the percentage of packets successfully delivered to the sink.
2) The *end-to-end latency*, defined as the time from packet generation to its correct delivery to the sink.
3) The *network energy consumption*, defined as the total amount of energy spent by all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

All results have been obtained by averaging the outcomes of 100 simulation runs, each of duration T_s of 3 days. This number of runs obtains a 95% confidence with 5% precision. In order to evaluate steady-state performance, all metrics are collected after the initial network setup phase.

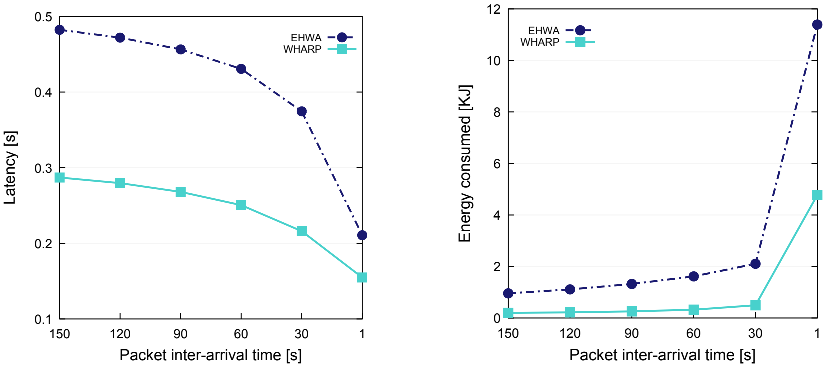
D. Simulation results

1) Packet delivery ratio: Fig. 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. 3b shows the average number of control packets generated 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. 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). A higher number of active nodes results in a higher number of available relays and, ultimately, in higher packet delivery ratio.

2) End-to-end latency: The average packet end-to-end latency is shown in Fig. 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. 3c). More important, the higher latency is also due



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



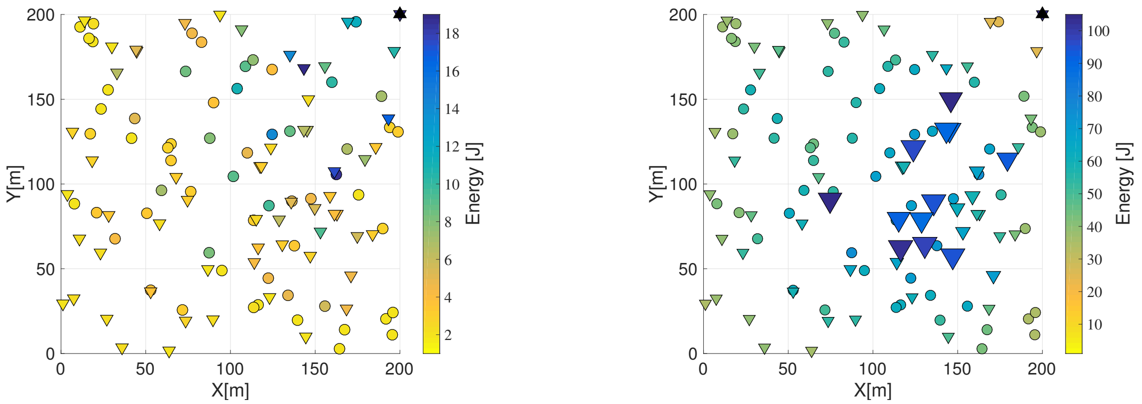
(d) End-to-end latency (e) Network energy consumption

Fig. 3: Performance comparison of WHARP and EHWA for increasing traffic loads.

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.

3) Network energy consumption: Fig. 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. 3b), which imposes a longer use of the main radio, and uses longer routes (Fig. 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). The performance gap increases with traffic. At the highest traffic, WHARP consumes 58% less energy than EHWA.

In order to further demonstrate the 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 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. 4a (from 0 to 18) and Fig. 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. 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. 4b, center). Fig. 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



(a) WHARP

(b) EHWA

Fig. 4: A joint snapshot of the all-off time and energy consumption per node: WHARP vs. EHWA.

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.

V. RELATED WORK

Recent research in traditional wireless networks has been primarily focused on routing schemes where energy efficiency is an essential consideration [27]. Spachos et al. present an energy-aware opportunistic routing protocol for wireless sensor networks, namely EAOR, that uses the RTS/CTS handshake mechanism [28]. In EAOR the selection of the forwarder node is done based on the distance from the sink and the available energy level. However, the vast majority of existing solutions is designed for battery-operated wireless networks and cannot be easily adapted to green networks. In fact, 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 green wireless networks, the design of harvesting-aware communication solutions requires a paradigm shift.

In particular, while solutions for traditional wireless networks focus on minimizing the energy consumption, the additional goal of harvesting-based communication strategies is that of maximizing the sustainable workload [29]. Towards this direction, a variety of data forwarding strategies have been proposed, specifically targeting on energy harvesting wireless networks [5],[10]. A first set of solutions considers networks solely powered by ambient energy harvesting, in which nodes have no long-term energy storage [11]. However, in the vast majority of applications nodes are equipped with large supercapacitors and/or rechargeable batteries to survive during periods of low energy intake [30]. 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 [31]. This aspect is further exacerbated by the use of over-simplified models of the harvesting process and of node energy consumption. To handle the problem of uncertain energy availability, some works make the assumption that the harvested energy is known for each node over a finite time horizon [32]. In this work, our goal is to investigate the design of an adaptive data forwarding strategy that is cross-layer, thus taking into account also lower-layers, and that is fully optimized for green wireless networks.

Recent 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 [12], 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.

Despite the numerous approaches tailored to wireless networks, as of now only a handful of solutions have been proposed that adopt wake-up radio techniques to overcome the barriers to energy efficiency. A cross-layer approach for data gathering in wireless sensing systems, namely ALBA-WUR, was presented in [6]. 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. CTP-WUR is a cross-layer routing protocol for data gathering in wake-up radio based wireless networks described in [8]. 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. Kumberg et al. in [33] proposed T-ROME, an energy efficient and wake-up radio enabled cross-layer routing protocol for wireless networks. 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. In this work, we propose a data forwarding strategy that selects a relay node by taking under consideration energy-related information and by considering that relays have energy scavenging capabilities. This has a significant impact on the overall energy consumption and on the periods during which each relay is operational.

VI. CONCLUSIONS

This paper 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.

REFERENCES

[1] A. Cammarano, D. Spenza, and C. Petrioli, "Energy-harvesting WSNs for structural health monitoring of underground train tunnels," in *Proceedings of IEEE INFOCOM WKSHPS 2013*, 2013, pp. 75–76.

[2] H. Alemdar and C. Ersoy, "Wireless sensor networks for healthcare: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2688–2710, 2010.

[3] Y. Klöpping, "Active implantable medical devices: winning the power struggle," *European Medical Device Technology*, vol. 3, no. 2, pp. 38–41, March/April 2012.

[4] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 1, pp. 1–12, Jan. 2010.

[5] S. Basagni, M. Y. Naderi, C. Petrioli, and D. Spenza, "Wireless sensor networks with energy harvesting," in *Mobile Ad Hoc Networking: The Cutting Edge Directions*. Hoboken, NJ: John Wiley and Sons, Inc., 2013, ch. 20, pp. 701–736.

[6] 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 INFOCOM 2015*, April 2015, pp. 522–530.

[7] 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," *Sensors*, vol. 14, no. 1, pp. 22–51, 2013.

[8] 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 ICNC 2016*, February 15-18 2016.

[9] C. Petrioli, D. Spenza, P. Tommasino, and A. Trifiletti, "A novel wake-up receiver with addressing capability for wireless sensor nodes," in *Proceedings of IEEE DCOSS 2014*, 2014, pp. 18–25.

[10] R. V. Prasad, S. Devasenapathy, V. S. Rao, and J. Vazifehdan, "Reincarnation in the ambiance: Devices and networks with energy harvesting," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 195–213, 2014.

[11] S. S. Beheshtiha, H. P. Tan, and M. Sabaei, "Opportunistic routing with adaptive harvesting-aware duty cycling in energy harvesting WSN," in *Proceedings of IEEE WPMC 2012*, September 24–27 2012, pp. 90–94.

[12] Z. A. Eu, H.-P. Tan, and W. K. Seah, "Opportunistic routing in wireless sensor networks powered by ambient energy harvesting," *Computer Networks*, vol. 54, no. 17, pp. 2943–2966, 2010.

[13] S. Basagni, F. Ceccarelli, F. Gattuso, and C. Petrioli, "Demo abstract: Abating LPL-induced latency with wake-up radio technology," in *Proceedings of ACM IoT DI 2017*, Pittsburg, PA, April 18–21 2017, pp. 1–2.

[14] G. Martinez, S. Li, and C. Zhou, "Wastage-aware routing in energy-harvesting wireless sensor networks," *IEEE Sensors Journal*, vol. 14, no. 9, pp. 2967–2974, 2014.

[15] D. Benedetti, C. Petrioli, and D. Spenza, "GreenCastalia: An energy-harvesting-enabled framework for the Castalia simulator," in *Proceedings of ACM SenSys 2013*, November 11-13 2013.

[16] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2014.

[17] A. Cammarano, C. Petrioli, and D. Spenza, "Online energy harvesting prediction in environmentally powered wireless sensor networks," *IEEE Sensors Journal*, vol. 16, no. 17, pp. 6793–6804, 2016.

[18] D. K. Noh and K. Kang, "Balanced energy allocation scheme for a solar-powered sensor system and its effects on network-wide performance," *Journal of Computer and System Sciences*, vol. 77, no. 5, pp. 917 – 932, 2011.

[19] A. Boulis, "Castalia: Revealing pitfalls in designing distributed algorithms in WSN," in *Proceedings of ACM SenSys 2007*, November 6-9 2007, pp. 407–408.

[20] Electric double layer capacitors (gold capacitor). [Online]. Available: http://www.farnell.com/datasheets/2015895.pdf

[21] F. I. Simjee and P. H. Chou, "Efficient charging of supercapacitors for extended lifetime of wireless sensor nodes," *IEEE Transactions on Power Electronics*, vol. 23, no. 3, pp. 1526–1536, May 2 2008.

[22] NREL: Measurement and instrumentation data center. [Online]. Available: http: //www.nrel.gov/midc

[23] S. AG, "SHT1x datasheet: Humidity and temperature sensor IC," 2011.

[24] M. Paoli, D. Spenza, C. 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*, April 11-14 2016, pp. 1–2.

[25] T. S. Rappaport, *Wireless Communications: Principles and Practice*, ser. Prentice Hall Communication Engineering and Emerging Technologies. Upper Saddle River, NJ: Prentice Hall PTR, 2002.

[26] T. Instruments, "CC1101 datasheet," rev. SWRS061I, 2013.

[27] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[28] P. Spachos, P. Chatzimisios, and D. Hatzinakos, "Energy aware opportunistic routing in wireless sensor networks," in *IEEE Globecom Workshops 2012*, Dec 2012, pp. 405–409.

[29] A. Bogliolo, S. Delpriori, E. Lattanzi, and A. Seraghiti, "Self-adapting maximum flow routing for autonomous wireless sensor networks," *Cluster Computing*, vol. 14, no. 1, pp. 1–14, 2011.

[30] N. Pais, B. Cetin, N. Pratas, F. Velez, N. Prasad, and R. Prasad, "Cost-benefit aware routing protocol for wireless sensor networks with hybrid energy storage system," *Journal of Green Engineering*, vol. 1, no. 2, pp. 189–208, 2011.

[31] 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*, June 2010, pp. 19–26.

[32] J. Marašević, C. Stein, and G. Zussman, "Max-min fair rate allocation and routing in energy harvesting networks: Algorithmic analysis," in *Proceedings of ACM MobiHoc 2014*, August 11-14 2014, pp. 367–376.

[33] T. Kumberg, M. Schink, L. Reindl, and C. Schindelhauer, "T-ROME: A simple and energy efficient tree routing protocol for low-power wake-up receivers," *Ad Hoc Networks*, February 2017.