Kai Geissdoerfer *Networked Embedded Systems Lab, TU Dresden, Germany* **Mikołaj Chwalisz** *Telecommunication Networks Group, TU Berlin, Germany* **Marco Zimmerling** *Networked Embedded Systems Lab, TU Dresden, Germany*

Editor: Steven Ko

TAKING A DEEP DIVE INTO THE BATTERYLESS INTERNET OF THINGS WITH SHEPHERD

Collaboration of batteryless devices is essential to their success in replacing traditional battery-based systems. Without significant energy storage, spatio-temporal fluctuations of ambient energy availability become critical for the correct functioning of these systems. We present *Shepherd*, a testbed for the batteryless Internet of Things (IoT) that can record and reproduce spatio-temporal characteristics of real energy environments to obtain insights into the challenges and opportunities of operating groups of batteryless sensor nodes.

Batteryless devices promise to overcome the drawbacks of batteries, such as bulkiness, wear-out, and toxicity. The limited energy capacity of capacitors, however, requires intermittently executing the software, which may harm reliability [2]. Focusing on *multiple* rather than individual batteryless devices not only enables exciting new applications (e.g., swarms of nanosatellites [4]), but may also offer a new perspective on the reliability issue: Is it possible to operate a distributed collection of batteryless devices so that the group is more reliable than each device alone? How would efficient and reliable programming models, wireless communication protocols, and runtimes for collections of batteryless devices look like? Can the cooperation of distributed batteryless devices bring about benefits in terms of overall efficiency and effectiveness, similar to cooperation in multi-agent systems [5]?

Unfortunately, the research community lacks a tool to investigate these questions. Such a tool must be able to synchronously record rapidly changing energy conditions at different points in space. Even having such energy traces, it is hard to accurately model and predict the performance and behavior of a real batteryless system because of the complex behavior of circuits exposed to an intermittent power supply. To develop and compare novel designs, it is thus necessary to experiment under time-varying energy availability, so the tool must also be able to faithfully reproduce energy environments from recorded or generated traces.

Recording and replaying harvested energy is hard, and *few solutions exist for individual devices*. For example, to overcome high costs of specialized lab equipment, Ekho [1] uses custom-designed, affordable hardware to record and emulate an energy source with limited accuracy and resolution. Established wireless sensor network testbeds support synchronous recording of current draw [3] or energy consumption [6], but cannot profile the complex behavior of an energy-harvesting system by reproducing an energy environment.

To fill this gap, we present Shepherd, a portable testbed for the batteryless IoT. Shepherd's main novelty is the combined capability of accurately recording and

Photo, istockphoto.com

replaying high-resolution voltage and current traces synchronously across spatially distributed batteryless devices. Specifically, Shepherd can dynamically record and emulate current from 1.4μA to 25mA and voltage from 44μV to 3V at a sampling rate of 100kSps with a synchronization error of no more than 2.4μs among the devices. This provides an unprecedented visibility into energy environments and faithfully reproduces those real-world conditions for the systematic development and evaluation of distributed batteryless applications and services. Shepherd is affordable (about 200 USD per device) and portable (i.e., supports mobile and outdoor scenarios), while offering all amenities of state-of-the-art testbeds.

DESIGN AND IMPLEMENTATION

We envision the following workflow for the development and evaluation of distributed batteryless systems. A number of testbed nodes equipped with the desired energy-harvesting technology are deployed in the target environment. The testbed records the harvested energy at each node for a user-defined period of time. The user retrieves the data and analyzes them to gain an understanding of the characteristics of the energy environment. With the help of the testbed, the user can then develop, test, and validate ideas involving batteryless devices by repeatedly replaying the recorded energy traces. This enables repeatable, experiment-driven research into open problems, such as time synchronization, wireless networking, or distributed sensing and actuation using collections of batteryless nodes.

Motivated by this workflow, we design Shepherd as a network of Shepherd nodes that are synchronized and operate in one of two modes (see Figure 1): During *recording*, energy flows from the harvester to a dummy load, which is part of a powerful *observer* platform. The observer measures current and voltage, and timestamps the data with respect to the testbed-wide timeline before shipping them to a remote database. During *emulation*, data are sent from the database to the Shepherd nodes from where they are fed into a harvesting emulator that outputs the corresponding voltage and current to an attached node. Shepherd can replay data from previous recordings or emulate data that were generated using, for example, a spatio-temporal energy model. While replaying, the observer also monitors the node's power draw, samples the GPIO pins, and records serial messages in order to provide insight into the execution of the distributed application or protocol under the emulated energy environment.

Hardware Architecture

A Shepherd node (see Figure 2) consists of an *observer* and three types of *capelets*. The *harvesting capelet* hosts the harvesting transducer and all components required to operate it; the *storage capelet* hosts the energy buffer, typically a capacitor; and the *target capelet* hosts the sensor node. The capelets are connected to the observer through well-defined interfaces, which makes for a modular design that users can easily customize to their needs in terms of harvesting modality, energy storage, and node platform. The observer further includes a custom-designed analog frontend, the *Shepherd cape*, and a *BeagleBone* single-board computer. The Shepherd cape hosts components and circuitry required to record and replay energy-harvesting traces. The on-board DC-DC converter supports a converter-based energy harvesting architecture with maximum power point (MPP) tracking. During recording, the current into the converter is amplified with a precision instrumentation amplifier and sampled with an 18-bit analog-to-digital converter (ADC). During emulation, a two-channel digital-to-analog converter provides the DC-DC converter with a reference voltage and sets the corresponding input current with a precision voltage to current converter circuit. Multiple observers connect via their BeagleBones' Ethernet ports with each other and to a host that stores the data and runs a tool we provide for orchestrating a collection of distributed Shepherd nodes.

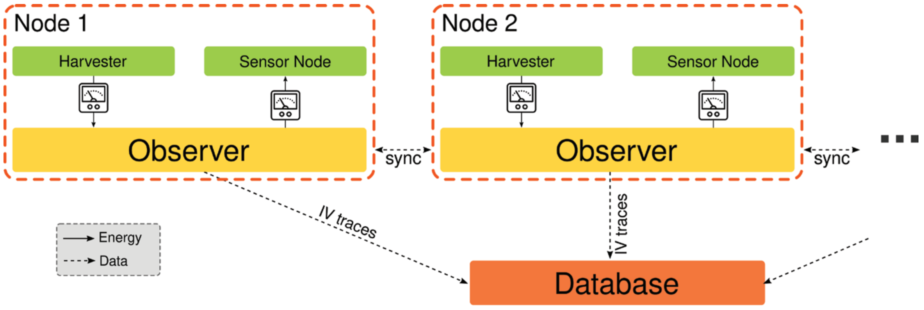

FIGURE 1. Shepherd consists of a network of time-synchronized Shepherd nodes. During recording, each observer measures current and voltage from the harvester and sends the timestamped data to a database. During emulation, data are sent from the database to the Shepherd nodes, where the harvesting emulator outputs the corresponding voltage and current to an attached sensor node.

Time Synchronization

Shepherd supports two mechanisms for synchronizing a group of Shepherd nodes that can be flexibly combined (see Figure 3), via GPS or the Precision Time Protocol (PTP). Many GPS receivers output a highly accurate pulse per second signal. Together with information about the global time of that second, we synchronize a Shepherd node to the globally accurate GPS time. The advantage of GPS is that the nodes do not need to be physically connected. For example, Shepherd can be used to explore long-range wireless networks of solar energy harvesting devices, where physical connections are hardly feasible. However, GPS receivers always require line of sight to the satellites, which limits deployment locations.

For this reason, Shepherd also supports PTP, a time synchronization protocol over Ethernet links. A number of PTP slaves synchronizes to a common master elected automatically based on clock quality estimation.

Performance

We characterized Shepherd's performance in an extensive evaluation campaign. Table 1 lists the key performance indicators. In summary, Shepherd provides novel functionality and unprecedented performance for a reasonable price (about 200 USD per node).

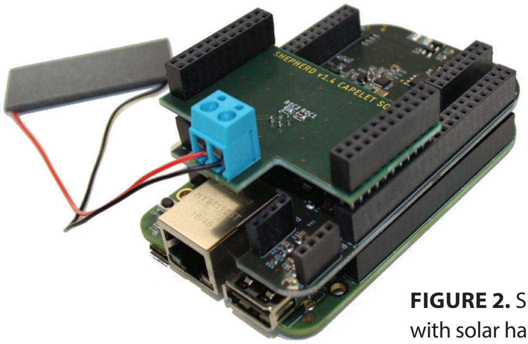

FIGURE 2. Shepherd node equipped with solar harvesting capelet.

TABLE 1. Shepherd's key performance characteristics

	Current	Voltage
Resolution	381nA	76.3µV
Dynamic Range	1.4µA-25mA	44µV-3V
Sampling Rate	100kHz	
Synchronization Error	(69±602)ns	

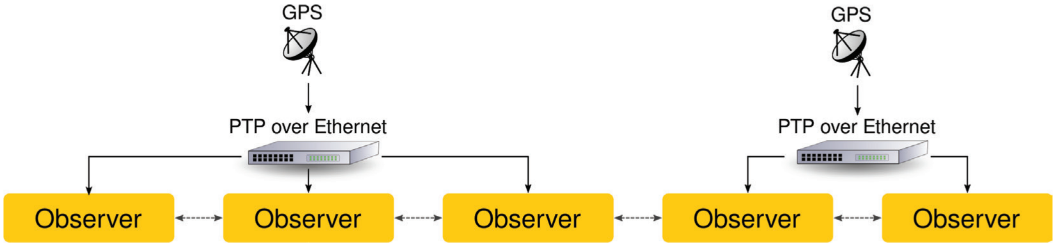

FIGURE 3. Flexibly combining GPS and PTP for time synchronization allows large-scale indoor and outdoor Shepherd deployments to operate.

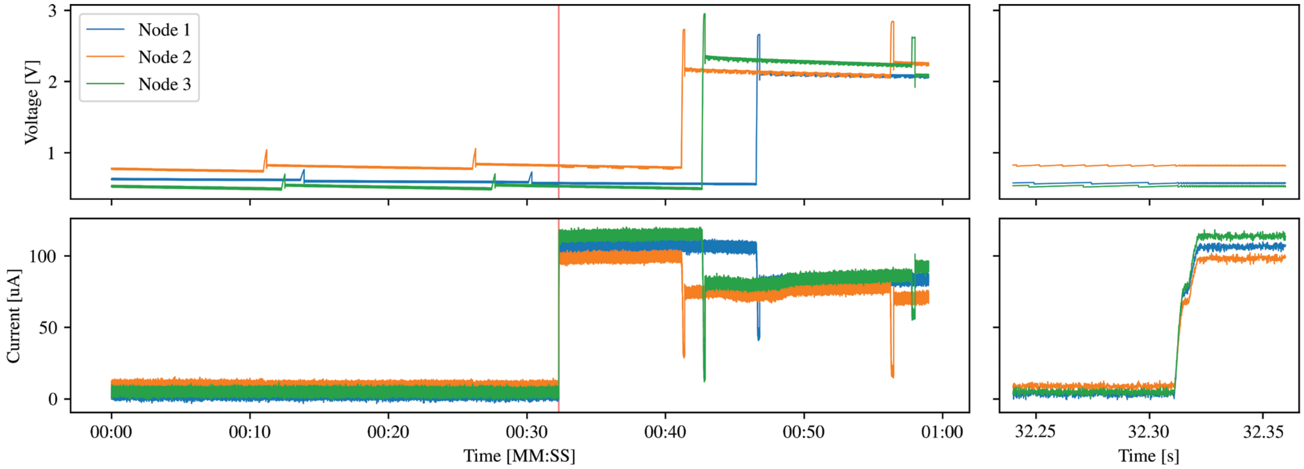

FIGURE 4. Voltage and current traces recorded by three Shepherd nodes in an indoor solar energy-harvesting scenario. The plots on the right zoom into the moment the room lights are switched on (marked in red on the left).

SHEPHERD IN ACTION

We demonstrate the utility of Shepherd by testing a distributed algorithm for batteryless nodes in a real energy environment.

Setting: We consider an indoor solar energy harvesting scenario. The testbed comprises three Shepherd nodes equipped with a 7cm x 2cm solar panel, a nRF52840 capelet, and a storage capelet with a 150µF ceramic capacitor. The three nodes are placed on tables inside a room without windows, receiving only little light through a glass door. Node 2 is slightly tilted, facing that glass door, while nodes 1 and 3 are oriented horizontally. We develop an example application for the nRF52840 that wakes up from system-off mode when the capacitor voltage exceeds 2.8V. After initialization, the application senses the capacitor voltage every 125ms. When the capacitor voltage reaches 3V, it executes a task that samples the temperature sensor and transmits the reading to a base station by embedding it into Bluetooth Low Energy advertisement packets sent on the three advertisement channels. The application continues to sample the capacitor voltage until the 3V threshold is exceeded again or until it falls below 2.3V. The application indicates state changes (system-off, sampling supply voltage, etc.) through GPIO pins, allowing us to track the logical state of each node with high resolution.

Recording: We record harvesting current and voltage for 60 seconds synchronously on the three Shepherd nodes. Initially, the room lights are switched off. After 30 seconds, we turn the room lights on and keep recording for another 30 seconds. Figure 4 shows voltage and current recorded by the three nodes. Initially, the voltages are

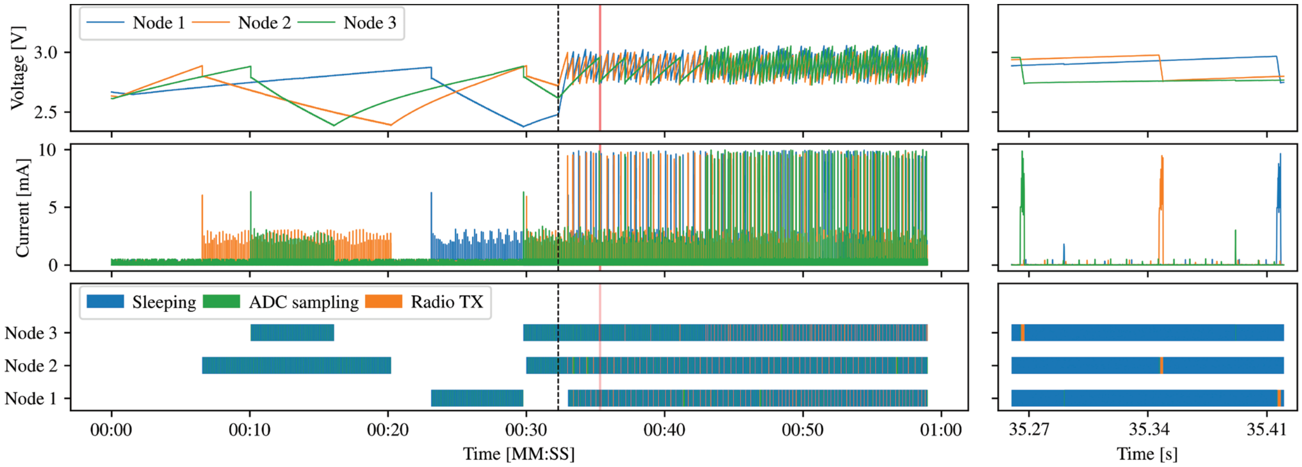

FIGURE 5. Capacitor voltage, current, and logical states when replaying the previously recorded traces to three sensor nodes with Shepherd. The black dashed line indicates when the lights are switched on. The plots on the right zoom into the time marked with a red solid line.

low. The spikes at around 12, 30, 45, and 60 seconds are due to the MPP tracking of the BQ25504 DC-DC converter. To this end, the BQ25504 shortly disconnects the harvester and samples the solar panel's open-circuit voltage. As soon as the light is switched on, the current rises sharply on all three nodes, as shown in the zoomed-in plots of Figure 4.

Emulation: We replay the recorded traces to the nRF52840 devices running our example application, while recording capacitor voltage, current, and GPIO traces. Figure 5 shows that all three nodes start in system-off mode. The small amount of energy extracted from the solar panels is sufficient to slowly charge the capacitors while the nodes are still powered off. Following the recorded harvesting traces, node 2 receives most energy and is thus the first to reach the 2.8V power-on threshold. However, as long as the room lights are switched off, the periodic sampling of the supply voltage is not sustainable, leading to a decreasing capacitor voltage and eventually power-off. After switching on the room lights, the nodes power up quickly and accumulate enough energy to read out the temperature sensors and send the readings to the base station. In the zoomed-in plots of Figure 5, we see that the nodes execute the task at different times and that the execution drains the capacitor voltage much faster than it rises during charging.

Analysis of the recorded energy harvesting traces shows that some phenomena, like MPP tracking, affect nodes independently, whereas changes in the energy environment affect all nodes at the same time. Emulation allows us to relate spatio-temporal energy availability with the system-wide behavior and performance of distributed batteryless devices. Overall, the observations and insights presented here would be very difficult, if at all possible, to attain without a tool like Shepherd. ■

Excerpted from "Shepherd: A Portable Testbed for the Batteryless IoT," from *SenSys '19: Proceedings of the 17th Conference on Embedded Networked Sensor Systems*, with permission. https://dl.acm.org/doi/10.1145/3356250.3360042 © ACM 2019

Kai Geissdoerfer received his BSc and MSc in Electrical Engineering from Technische Universität Berlin, Gemany in 2015 and 2017 respectively. He is a PhD student at the Networked Embedded Systems Lab at Technische Universität Dresden, Germany and a visiting research student at the Distributed Sensing Systems group at CSIRO, Australia. His research focuses on energy harvesting and autonomous batteryless sensing systems. kai.geissdoerfer@tu-dresden.de

Mikołaj Chwalisz received his MSc in electrical and computer engineering from the Warsaw University of Technology and in computer engineering from the Technische Universität Berlin in 2011. His focus has been on coexistence and cooperation of heterogeneous wireless networks working on experimentally driven solutions and testbed orchestration. Currently, he is an Embedded Systems Developer at R3 Communications GmbH, focusing on developing and testing ultra-reliable wireless industrial communications network systems. chwalisz@tkn.tu-berlin.de

Marco Zimmerling holds a Diploma degree in computer science (2009) from TU Dresden, Germany, and a PhD in computer engineering (2015) from ETH Zurich, Switzerland. Currently, he leads the Networked Embedded Systems Lab within the Center for Advancing Electronics at TU Dresden. His interests are cyber-physical systems, with a special focus on dependable and sustainable wireless embedded systems. marco.zimmerling@tu-dresden.de

REFERENCES

[1] Josiah Hester, Lanny Sitanayah, Timothy Scott, and Jacob Sorber. 2017. Realistic and repeatable emulation of energy harvesting environments. *ACM Transactions on Sensor Networks (TOSN)* 13, 2.

[2] Neal Jackson, Joshua Adkins, and Prabal Dutta. 2019. Capacity over capacitance for reliable energy harvesting sensors. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks (IPSN).*

[3] Roman Lim, Federico Ferrari, Marco Zimmerling, Christoph Walser, Philipp Sommer, and Jan Beutel. 2013. FlockLab: A testbed for distributed, synchronized tracing and profiling of wireless embedded systems. In *Proceedings of the 12th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN).*

[4] Sujay Narayana, R. Muralishankar, R. Venkatesha Prasad, and Vijay S. Rao. 2019. Recovering bits from thin air: Demodulation of bandpass sampled noisy signals for space IoT. In *Proceedings of the 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN).*

[5] Rui Rocha, Jorge Dias, and Adriano Carvalho. 2005. Cooperative multi-robot systems: A study of vision-based 3-D mapping using information theory. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA).*

[6] Markus Schuß, Carlo Alberto Boano, Manuel Weber, Matthias Schulz, Matthias Hollick, and Kay Römer. 2019. JamLab-NG: Benchmarking low-power wireless protocols under controlable and repeatable Wi-Fi interference. In *Proceedings of the 16th International Conference on Embedded Wireless Systems and Networks (EWSN).*