Christian Renner, Florian Meier, and Volker Turau. Holistic Online Energy Assessment: Feasibility and Practical Application. In Proceedings of the 9th IEEE International Conference on Networked Sensing Systems, INSS'12, Antwerp, Belgium, June 2012.

Holistic Online Energy Assessment: Feasibility and Practical Application

Christian Renner, Florian Meier, and Volker Turau

***Abstract*—Combining energy harvesting with energy-aware scheduling enables perpetually operating sensor networks. The practical realization of this goal yet requires reliable and precise holistic online energy assessment. While the building blocks—assessing residual energy, predicting energy intake, and tracing energy consumption—have been studied in detail, the analysis of their interaction on a real platform has been neglected. This paper answers the question, whether these techniques can be easily joined to give a precise and correct picture of a sensor node's energetic state and behavior. For this purpose, we model the energy flow of a prototype energy-harvesting sensor node and evaluate the joint performance of state-of-the-art energy assessment based on a field test. We verify the system model and show the feasibility of holistic energy assessment, which tolerates small configuration errors, achievable with a combination of generic configuration and online calibration. We also analyze the feasibility of forecasting a node's future energetic state, and find that the presented method gives sufficient results for uniformly distributed consumption profiles.**

***Index Terms*—Energy harvesting, Energy management, Energy measurement, Supercapacitors, Solar energy**

I. Introduction

Energy-harvesting sensor nodes [8], [17] open the door to unlimited and uninterrupted operation. It seems that gaps in collected data—reducing the expressiveness of measurements—and manual intervention for battery replacement—inferring large costs, logistic problems, and severe intrusion [19], [11]—are things of the past. Yet, non-intrusive monitoring of phenomena demands devices of tiny size, which come at a non-negligible cost: the amount of harvested energy is decreased while it must still satisfy the average power consumed by applications and algorithms executed on the node. Unfortunately, the actual extent of neither harvested nor consumed energy is known in advance. Energy consumption depends on difficult to foresee network load, e.g., routing services provided by the node. Energy harvest often depends on the exact positioning of the harvester, since local and seasonal effects dominate energy production.

Thus, a sensor node must adapt its consumption to the available energy resources. Several researchers have tackled this problem [5], [18], [20], [10], [2]. In brief, nodes have to adjust their schedule or duty cycle to achieve energy-neutral operation [9], i.e., consumption must not exceed harvest. To maintain this principle in times of low harvest, energy must be buffered in energy-abundant periods. Supercaps have been frequently used, as they combine small size and cheap prices with a capacity to back up operation for several days. Reliable while facile energy reserve estimation is possible [8], [13].

To adapt a node's duty cycle or find an energy-compatible schedule, tracking the consumption of a sensor node is mandatory. Methods for hardware- [4], [6] and software-based [3], [7] tracking have been presented recently. Solar cells are used as harvesters on many platforms [8], [17]. They deliver sufficient energy with a diurnal pattern, enabling energy intake forecasts [12], [1], [15], [9] for depletion-safe and smooth operation with infrequently adapted duty cycles [10], [2], [20].

The tools for holistic online energy assessment are ready, but existing research has mainly focused on the individual components and their optimization. However, their interaction has been neglected, so that three essential and closely linked questions have not been answered so far:

- How well does joining these techniques work w.r.t. to giving a precise and correct picture of a sensor node's energetic situation and behavior?
- Is it possible to forecast a node's future energetic state?
- How precise a calibration of hardware components, e.g., sensors and consumption traces, is required; or, is individual node configuration needless?

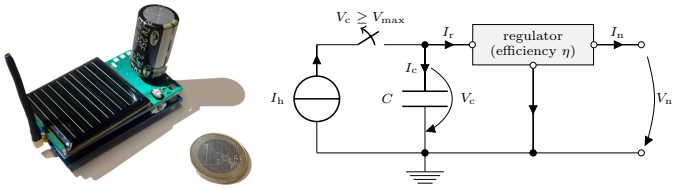
This paper makes the following contributions to answer these questions. We introduce an energy-harvesting supply for the Iris platform and model its energy flow. We implement an energy-assessment software layer in TinyOS running in a five-node deployment with our hardware for three weeks. A detailed evaluation finally verifies the model and proves the utility of existing energy-assessment techniques in practice. We identify the main influencing factors of energy misjudging and those with low impact on system preciseness. We finally show under which conditions our model can forecast future energetic states of a sensor node, which is the fundament of more advanced algorithms for energy-aware scheduling.

II. Energy-Harvesting Power Supply

To evaluate holistic energy management in sensor networks, we built a customized energy-harvesting power supply for the Iris sensor node. We will refer to this power supply as *harvester*. It consists of three building blocks: a solar cell as energy-harvesting source, a supercap for buffering energy, and a switching regulator to supply the sensor node with a stable, constant voltage. The harvester is equipped with a light sensor

The authors are with the Institute of Telematics at Hamburg University of Technology, Schwarzenbergstrasse 95, D-21073 Hamburg, Germany (e-mail: christian.renner@tu-harburg.de). This work has been partially funded by the German Research Foundation under contract number TU 221/4-1. Thanks to Janos Sallai for his valuable input.

978-1-4673-1786-3/12/$31.00



(a) Energy harvester (b) Simplified equivalent circuit

Fig. 1: Energy harvester hardware and equivalent circuit

and a temperature sensor. A picture of the harvester mounted on an Iris node with removed battery pack is shown in Fig. 1a, accompanied by a simplified equivalent circuit in Fig. 1b.

A. Harvesting Source

A solar cell with a maximum current of 35 mA and a size of $39 \times 39\,\mathrm{mm}^2$ serves as harvesting source. We chose a direct charging circuit instead of a maximum power-point tracker as in [17]. The advantages are a simple, low-cost circuit and a charging current only depending on the lighting conditions. The main disadvantage is that the harvested energy depends on the supercap voltage. The harvester provides a sensor for measuring the current produced by the solar cell. This is achieved by measuring and amplifying the voltage induced by the solar current across a $1\,\Omega$ shunt resistor (1% precision). The result is fed to one of the ADC ports of the Iris node. The conversion factor of the ADC reading is $15\,\mathrm{mA/V}$. Unfortunately, the operational amplifier introduces a bias and a non-linearity for small voltages; both vary from board to board. A compensation model is discussed in Sect. III-C.

B. Energy Buffer

The harvester is designed for supercaps with a maximum voltage of $V_{\max} = 2.7\,\mathrm{V}$. Results from the literature [13], [8] show that supercaps with 25 to 100 F give a good trade-off between size, capacity, and price; e.g., a 50 F supercap can operate an Iris node at a 1% radio duty cycle for more than two days without harvesting energy. The supercap is connected to one of the node's ADC ports, so that its voltage and thus state-of-charge can be read by the sensor node. To protect the supercap from overcharging, the harvester automatically disconnects the supercap from the solar cell, if its voltage exceeds $V_{\max}$. Note that it is still possible to measure the current produced by the solar cell with the sensor discussed in Sect. II-A.

C. Switching Regulator

The sensor node is supplied by a Texas Instruments TPS 61220 switching regulator with a constant voltage of $V_n = 2.7\,\mathrm{V}$. We measured a switching efficiency η of 75% to 95% and a cut-off voltage of $V_{\mathrm{cut}} = 0.5\,\mathrm{V}$ for our platform.

III. Energy Assessment for Sensor Nodes

This section discusses methods to (i) assess the supercap's energy reserves, (ii) track the node's consumption, and (iii) forecast the harvester's energy intake.

A. Energy Reserve

There have been attempts to assess the energy reserves in a supercap, e.g., in [13]. The authors compare the model of an ideal capacitor with an extended model trying to encompass leakage effects. The latter introduces an extra amount of computational complexity paired with additional parameters, while the benefit reveals itself to be rather small: The model yields increased precision only for a high supercap voltage. Jiang et al. evaluate leakage behavior of supercaps from 10 F to 50 F and conclude that leakage varies from one supercap to another [8], so that manual configuration would be necessary.

Besides the influence of self-discharge (leakage) on supercap voltage, we found no model that incorporates the charging history of the supercap nor the influence of ambient temperature. Both influence supercap voltage according to the data sheets; yet, the exact impact is unknown due to the lack of research. A method for determining the actual capacity of a supercap online is presented in [14]. Knowing the actual capacity is required, as it influences the energy reserve linearly.

B. Energy Consumption

Tracing the consumption of sensor nodes offline, i.e., prior to deployment, is generally infeasible due to the dynamic nature of sensor network protocols. Online estimation has been widely accepted as the standard approach [7], [4], [3].

Hardware-based solutions, e.g., iCount [4], measure the switching frequency of the regulator many sensor node platforms are shipped with. These approaches exploit the fact that there is a nearly linear relationship between switching frequency and input current for a constant input voltage. This method is not applicable in our case, because we supply the regulator with a supercap. The regulator's input voltage thus ranges from 0.5 V to 2.7 V, so that a more complex model is required due to the non-linear impact of input voltage (cf.[4]).

Software-based energy estimation is presented and evaluated in, e.g., [7] and [3]. Each hardware component has a typical load, which depends on the its state. In a simple model, components are on or off. More refined models encompass additional states, e.g., the radio can be off, listening, receiving, or sending. In order to track the consumption of a sensor node, three steps must be taken. Firstly, the hardware states have to be identified and the corresponding load has to be measured. Secondly, hooks have to be added to the hardware drivers to track all state changes. Thirdly, timing with micro-second precision is required to retrieve the node's consumption by summing up all load-time products.

Regardless of the tool for assessing consumption, its purpose is to give a more fine-grained view on the energy consumption of the individual jobs executed by a sensor node. This knowledge is subsequently used to enable duty-cycle adaptation and online energy management, e.g., cf. [20], [2].

C. Energy Intake

Obtaining the current energy intake is achieved with a sensor. Despite possible imponderables regarding sensor calibration, there is no particular challenge involved. To allow

for sustainable and efficient operation of sensor nodes, past energy intake recordings can be turned into a forecast [9], [2], [1], [12], which enables a sensor node to determine an energy-neutral duty-cycle or job schedule.

The general approach for prediction is to exploit the cyclic behavior of harvesting sources, e.g., the diurnal pattern of solar cells. Each cycle is divided into time slots, usually of equal length. Energy intake in each of these slots is averaged and smoothed with past values. The resulting values constitute the forecast. Recent approaches (e.g., [12]) aim at refining the forecast by weighting future slot values with the relative course of the current cycle w.r.t. the smoothed course.

IV. System Model

The simplified circuit of our harvester in Fig. 1b yields a mathematical model encompassing the three building blocks (energy reserve, intake, consumption):

$$I_\mathrm{c} = I_\mathrm{h} - I_\mathrm{r} \quad \text{with} \quad I_\mathrm{r} = \frac{I_\mathrm{n} \cdot V_\mathrm{n}}{\eta \cdot V_\mathrm{c}} \,. \tag{1}$$

To maintain the general simplicity of this equation and based on the findings in Sect. III-A, we model the relationship of supercap voltage V_c and current I_c as an ideal capacitor, i.e., $I_\mathrm{c} = C \cdot \dot{V}_\mathrm{c}$. The resulting system model is

$$C \cdot \dot{V}_\mathrm{c} = I_\mathrm{h} - \frac{I_\mathrm{n} \cdot V_\mathrm{n}}{\eta \cdot V_\mathrm{c}} \,. \tag{2}$$

A. Predicting the Future Energetic State

If all parameters in (2) are known, predicting the future energetic system state is possible. This in turn enables efficient while perpetual operation. However, an exact solution of (2) requires the precise courses of I_h and I_n. When it comes to predicting the future state of the system, this is not the case, particularly because an exact estimate I_h does not exist in real deployments. We thus make the following assumptions and simplifications for practical reasons: (i) The harvester current I_h and the sensor node load current I_n are piecewise constant functions, and (ii) the regulator efficiency η is constant. The first assumption complies with state-of-the-art energy intake prediction (cf. Sect. III-C) and online consumption assessment (cf. Sect. III-B). It may be possible to describe regulator efficiency by a step-wise constant function of V_c, but we did not explore this relaxation.

For $I_\mathrm{h} = \mathrm{const.}$, $\eta = \mathrm{const.}$, $I_\mathrm{n} = \mathrm{const.}$, (2) reduces to an ordinary, first-order differential equation (ODE) of the form

$$\dot{y} = b - \frac{a}{y} \quad \left(y = V_\mathrm{c} \,,\; a = \frac{V_\mathrm{n} \cdot I_\mathrm{n}}{\eta \cdot C} \geq 0 \,,\; b = \frac{I_\mathrm{h}}{C} \geq 0 \right) . \tag{3}$$

In the following, we assume $y = y(t)$ and $y_0 = y(t_0)$, where $t \geq t_0$ are points in time and $\Delta t = t - t_0$. To solve this equation, two special cases have to be considered:

1) For $y = \frac{a}{b}$, we find $\dot{y} = 0 \Rightarrow y = y_0$
2) If $b = 0$, the solution is $y = \sqrt{y_0^2 - 2a \cdot \Delta t}$

In any other case, this equation has the implicit solution

$$0 = y - y_0 + \frac{a}{b} \cdot \log\left(\frac{a - b \cdot y}{a - b \cdot y_0} \right) - b \cdot \Delta t = f(y, y_0, \Delta t) \,, \tag{4}$$

which can be solved, e.g., with Newton's Method

$$y_{n+1} = y_n - \frac{f(y_n, y_0)}{f'(y_n, y_0)} \,, \tag{5}$$

for a given number of iteration steps or until an absolute or relative error is achieved. The overcharging protection (cf. Sect. II-B) requires manual enforcement of $y = V_\mathrm{c} \leq V_\mathrm{max}$.

B. Enabled Technologies: Practical Application

According to Sect. III-C, an energy intake forecast consists of a set of time slots and mean values. Energy intake I_h does hence not change within a slot. Assuming a constant average load I_n in each slot reduces the overhead of predicting the future energetic state by solving (4) with (5) drastically, because only one solution has to be calculated for each end of a time slot, and updates have to be calculated only if a slot elapses. The results presented in [1] and [15] suggest that 12 to 24 slots per day are sufficient, so that recalculations are required at most hourly. This approach can be used to predict the future voltage course of the supercap for a given average load, but becomes meaningful, only if the following two questions are answered: How to obtain the average node load, and how to capitalize on the voltage course?

The first question can be answered with a survey of existing research, e.g., adapting the (radio) duty cycle [10] is a common technique to achieve perpetual operation. The duty cycle directly yields (a lower bound on) the average power consumption of a node. Task scheduling is another option, if the power consumption of tasks is determined as in [20]. Moreover, a system with discrete performance levels is currently investigated.

Answering the second question is also possible by a look at existing research. Kansal motivates the principle of energy-neutral operation [9], i.e., a node must not consume more energy than it has harvested. Translated to the predicted voltage course, a node has to make sure that its (predicted) voltage never falls below a critical threshold and that the present voltage is conserved w.r.t. to the last voltage prediction.

V. Implementation

To evaluate the system model from Sect. IV in a field test, we implemented an energy-aware layer and an evaluation application for our sensor node platform for TinyOS 2.1. This section introduces its components and their capabilities of online configuration.

A. Energy Consumption Tracker

We implemented an annotation tool for TinyOS, which adds function hooks at compile time to the hardware abstraction layer to keep track of hardware power-state changes. Currently, the microcontroller (MCU), radio, and LEDs are supported. The tool traces the consumption of each hardware component separately with a resolution of µA s. The current draw of each hardware state is defined in a common configuration file. We produced a generic current trace by averaging over a selection of nodes. Table I supports this approach. There are notable

node	MCU sleep	MCU adc nr	MCU idle	MCU active	radio listen	LEDs red	LEDs yellow	LEDs green
1	0.0285	1.195	2.854	6.877	14.435	2.062	1.677	1.827
2	0.0274	1.357	3.002	7.144	14.398	2.047	1.677	1.808
3	0.0285	1.162	2.975	7.325	14.555	2.052	1.684	1.836
4	0.0281	1.211	2.897	8.384	14.795	2.011	1.641	1.795
5	0.0299	1.195	2.862	7.215	14.489	2.058	1.686	1.830
6	0.0298	1.260	2.740	6.810	14.850	2.091	1.654	1.860
7	0.0301	1.229	2.760	6.940	14.780	2.118	1.653	1.853
mean	0.029	1.230	2.870	7.242	14.615	2.063	1.667	1.830
std.-dev.	0.001	0.064	0.099	0.537	0.189	0.034	0.018	0.023

TABLE I: Iris node consumption (mA) in different operation states. The MCU states cover the consumption of the radio-off state, transmitting is not considered separately

deviations only for an active MCU. Note that we add an extra $10\,\mu A$ to the MCU sleep consumption to account for self-consumption of the harvester, see Fig 2a.

The time spent in a state is obtained by running one of the 16-bit Iris counters with a resolution of $8\,\mu s$. This counter is synchronized with the asynchronous milli-second timer via hardware counter capture. This concept only works with the internal $8\,MHz$ RC-oscillator instead of the crystal, which requires a few ms to wake the node and thus makes synchronization impossible. Note that synchronization is required, because only the asynchronous timer is running when the node is in sleep mode. Our approach is inspired by [16].

The regulator input load I_r is obtained by using (1). This conversion is not performed per state update, since this would increase overhead due to another multiplication of 32 bit values. In contrast, the software uses a constant regulator efficiency of $\eta = 0.85$ and performs the conversion only upon application request. Figure 2b shows a representative efficiency trace for our harvester. The chosen efficiency constitutes the approximate mean value for different input voltages and load profiles. It is generally possible to improve this approach by performing the conversion with a periodicity of a few seconds or minutes and using interpolated values of η from a look-up table w.r.t. the current V_c: V_c changes slowly, if the node is sleeping most of the time. However, this approach requires a precise and per-node configuration of the look-up table.

B. Energy Intake Assessment

Energy intake of the solar cell is measured with the sensor described in Sect. II-A. The relationship between solar current I_h and the amplified voltage across the shunt resistor V_s exhibits a notable variance among individual boards. To prevent an individual configuration of each board, we devised a self-configuring method using an empirically derived model:

$$I_\mathrm{h} = \alpha \cdot (V_\mathrm{s} - V_\mathrm{b}) - (\alpha \cdot (V_\mathrm{s} - V_\mathrm{b}) + \beta \cdot V_\mathrm{b})^{-1} + (\beta \cdot V_\mathrm{b})^{-1} \quad (6)$$

The parameters α and β have been determined empirically and are the generic part of the model. Note that α is close to the theoretical linear gain of the amplifier (cf. Sect. II-A). The parameter V_b is the bias of each harvester and is adjusted automatically during runtime—it reaches its final value in the first period of zero energy intake, e.g., at night. Figure 2c compares the model with real sensor traces (ground truth) for selected harvesters. Although the model produces average absolute errors of $0.6\,mA$ on some harvesters, it still achieves a better precision than a simple linear approach, which fails to capture the non-linearity near V_b.

C. Evaluation Application

We implemented an evaluation application to record the individual components of our energy flow model, i.e., I_h, V_c, I_n. Each node sends a report to a base station every $30\,s$ containing the supercap voltage V_c, the ambient temperature, and the mean solar current I_h since the last report. To analyze supercap leakage and temperature dependency, one additional pre-charged supercap each is sampled every $30\,s$ using a free ADC port. Consumption per component (MCU, radio, LEDs) is also attached. The nodes are placed outside on a window sill (rain-protected by plastic boxes) to allow for analyzing the impact of temperature changes.

It is only possible to evaluate the model for $V_\mathrm{c} < V_\mathrm{max}$, because we cannot assess the amount of lost energy when the overcharge protection is effective (cf. Sect. II-B). Therefore, consumption I_n must be large enough to prevent this case. On the contrary, I_n must be small enough to ensure $V_\mathrm{c} > V_\mathrm{cut}$ in order to prevent node downtimes and resulting gaps in recorded data. We combine this issue with our intention to evaluate the system model and feasibility of holistic energy assessment for varying loads I_n: each node simulates a duty-cycle adaptation by leaving on its radio for at least $250\,ms$ and at most $25\,s$ after sending its state report. The actual extent of this time is increased, if $V_\mathrm{c} > 2.0\,V$, and decreased, if $V_\mathrm{c} < 1.5\,V$. The in-/decrease operation is performed every $30\,s$.

To analyze the impact of different loads, each node turns on an LED when performing an ADC read operation. In addition, all possible MCU states are tracked. This is particularly important for two reasons caused by the TinyOS implementation of the Iris platform: Firstly, the MCU is always idle when the radio is on, and secondly, the 8-bit asynchronous timer requires the node to wake up briefly every $230\,ms$ to cope with counter overflows.

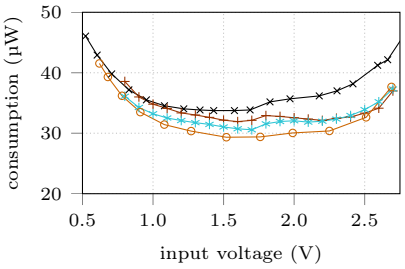
VI. Evaluation

This section evaluates our energy model, the interaction of energy-assessment components in praxis, and the quality of predicting a node's future energetic state.

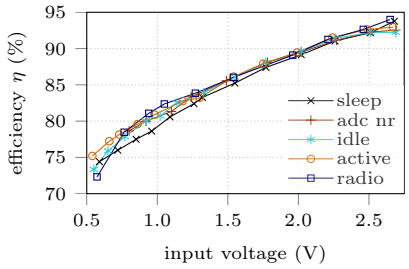
A. Experimental Setup and Methodology

The evaluation is based on deployments of five nodes equipped with our harvester and a $50\,F$ supercap. All nodes run the energy management software and evaluation application introduced in Sect. V. They are placed outside on a window sill (rain-protected by plastic boxes) to allow for analyzing the impact of temperature changes. Overall runtime of the experiment was 3 weeks.

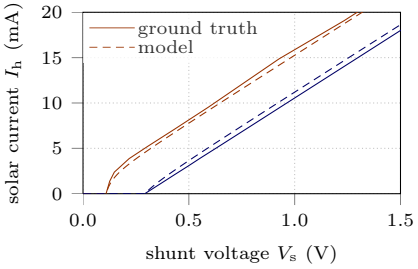
Our analysis is based on $10\,min$ snapshots and averages, resulting in more than 3 000 samples per node. This timing gave a good compromise between sufficient resolution and



(a) Self-consumption of 4 harvesters



(b) Unbiased conversion efficiency η of one (representative) harvester



(c) Self-configuring current sensor model (dashed) vs. empirical data (solid) of two harvesters

Fig. 2: Characteristics of the harvester: self-consumption, power conversion efficiency, accuracy of the current sensor

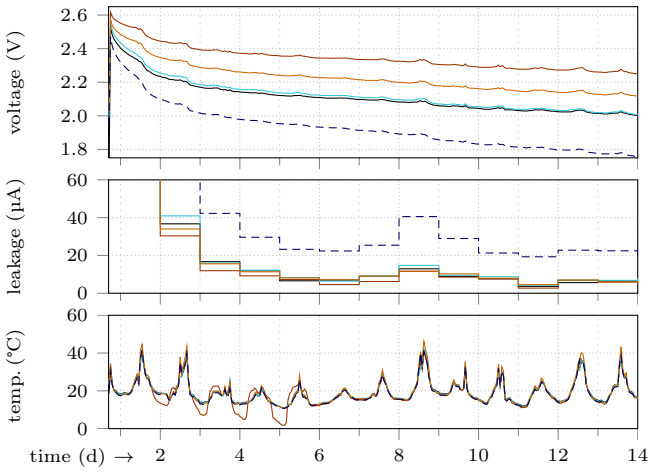
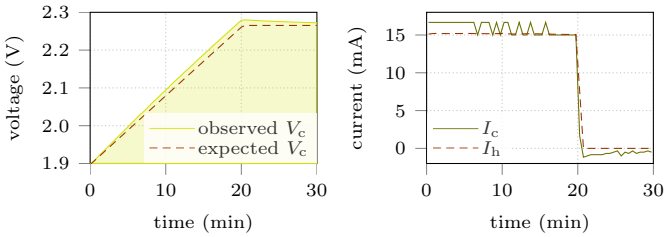


Fig. 3: Leakage analysis of four 50 F supercaps (solid) and one 100 F supercap (dashed) over the first two weeks. Time ticks indicate day borders at midnight



(a) measured vs. modeled voltage

(b) empirical current trace

Fig. 4: Theoretical (dashed) vs. empirical (solid) charging behavior of a 50 F supercap with constant supply of 15.2 mA

noise-reduction—e.g., the change of V_c is smaller than ADC noise for very short timings. All plots are generated directly from these values or derived using the system model equations (cf. Sect IV). The empirical supercap current results from $I_c = C \cdot \Delta V_c / \Delta t$. The capacity C was determined online by the sensor nodes at the beginning of the deployment according to the method suggested in [14]. We calculated model errors by means of $I_h - I_r - I_c$, representing the imbalance of (2). Because I_c may be zero, we did not calculate relative errors.

B. Model Evaluation and Component Interaction

1) Nominal Capacity vs. Actual Capacity: The actual capacity of a supercap may deviate considerably from its nominal value, e.g., due to manufacturing tolerances or aging effects. The online self-calibration in our field test produced values between 49.5 F (node 2) and 44.2 F (node 5). We thus observed a maximum relative error of 11.6% w.r.t. the nominal 50 F. This result proves the need of online self-calibration.

2) Supercap Leakage and Charging Behavior: Our system model does not encompass supercap leakage and temperature dependency (cf. Sect. IV). To assess this assumption, we evaluated the traces of the five additional supercaps. Their voltage course is shown in Fig. 3. The considerable voltage drop on the first day equals an average leakage current of 80 to 250 µA (not visible in the plot due to scaling). This is caused by reorganization effects, which depend on the charging current, initial voltage, and charging time. We further investigated these effects by charging a supercap with our harvester. This supercap has not been used—i.e., charged or discharged—within several days before the experiment. A small, constant charging current linearly increases voltage, as shown in Fig. 4a, but the observable voltage overshoots the expected value. The empirical current I_c is thus larger than expected (see Fig. 4b). After charging, the observed voltage decreases and nears the expected value. Voltage overshoot and reorganization cancel out within a few minutes.

Self-discharge after day 2 (in Fig. 3) is low and shows decreasing variation. Leakage depends on capacity—it is more than twice as large for the 100 F supercap—and is affected by the ambient temperature. Figure 5 portrays the influence of temperature on leakage with higher time resolution. Although the temperature gradient and leakage current appear to be coupled, the effect of temperature leads to tiny leakage currents most of the time. Only large temperature gradients induce currents of more than 30 µA, which is roughly the current consumption of Iris nodes in the sleeping state. Moreover, large temperature gradients occur during day time, in which the solar cell produces a current at least two orders of

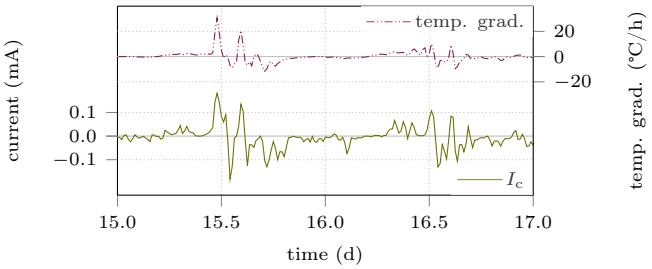


Fig. 5: Supercap leakage vs. temperature gradient

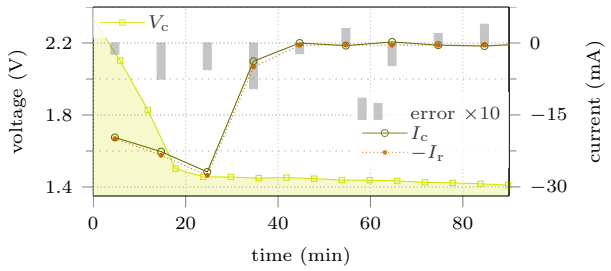


Fig. 6: Consumption vs. supercap current ($I_h = 0$) for node 1 with $C = 49.0\,\mathrm{F}$. Note the $10\times$ error scaling

magnitude in excess to that induced by temperature changes. Finally, the capacity C of the individual supercap is not known precisely, i.e., the error introduced by estimating C will easily exceed any influence of leakage and temperature.

Adding temperature compensation increases accuracy of residual energy assessment slightly but does not improve overall model precision significantly: leakage is rather small, hard to model, and has less influence than calibration errors.

3) Consumption vs. Supercap Current: To analyze the accuracy of the energy consumption tracker, we made an additional indoor run of our field test, in which we disconnected the solar cells at $V_c \approx 2.2\,\mathrm{V}$. The results for one node is depicted in Fig. 6. They indicate that the error is less than 1 mA in each 10 min window. During heavy consumption in the first 25 min, the error is mainly caused by small errors in the actual capacity and modeling η as a constant, generic value. The errors for lower consumption are predominantly caused by measurement noise of V_c, which is evident from the toggling errors and unsteady voltage course. Under these circumstances, consumption tracing matches supercap voltage and current development.

C. Accuracy of Holistic Online Energy Assessment

The previous analysis shows that our system model holds, if either the solar cell is disconnected (only-consuming) or there is no consumer (only-charging). In both cases, I_c matches I_h and I_r with low errors, respectively. To achieve holistic online energy assessment, the system model must hold, if charging and consuming occur at the same time.

1) Case Study: Figure 7 shows the holistic energetic state of node 1 on day 8 of the field test. The system model produces tolerable errors in all situations of the day, with three exceptions. Firstly, the error is high for large absolute values of I_c, if I_r is large at the same time. This can be explained by configuration and calibration errors regarding η and C. Both factors influence I_r and I_h linearly (see Sect. IV). The current sensor is another root of error, since we observed smaller errors for the experiment with disconnected solar cell. This is evident from the peak errors in the period around 11:30, where $I_r \approx 0\,\mathrm{mA}$ and $I_h \approx 16\,\mathrm{mA}$. Secondly, the error is large, when V_c cuts across 2 V or falls below 1.5 V, the thresholds for consumption adaptation. These errors show the inherent, yet intentional error of the model: linearization and averaging. The current I_r and V_c are reciprocal: The higher V_c, the lower I_r, and vice versa. If a node switches, e.g., from low consumption to high, enduring consumption while $I_h \gg 0$, the linearized model using averages produces large errors. If the consumption changes moderately, e.g., at around 13:40, the error is considerably smaller. Thirdly, the errors before 7 h and 18 h stem from too large readings of I_h, which we verified by covering the solar cells for a few minutes.

The presented trace substantiates the applicability of the model. Yet, if the current consumption is unevenly distributed while the solar cell produces large currents, errors of a few mA result. The practical importance of these errors is small in many application scenarios. In contrast to our field test, radio duty cycling shows an even distribution of consumption.

2) Overall Precision: A per-node error analysis is shown in Fig. 8a. Despite hardware deviation, the overall results are homogeneous, thus verifying the applicability of the model. Medians and quantiles are similiar for all nodes, and 80% of all errors are in a narrow region. Large errors stem from short times of massive changes of the consumption profile. The figure reveals an error bias (non-zero median) for some nodes. The error distribution (mean values and standard deviations) for the three current components is shown in Fig. 8b to 8d. Errors w.r.t. I_h are small and mostly positive, explaining the error bias. For $I_c > 0\,\mathrm{mA}$, which basically corresponds to $I_r \approx 0\,\mathrm{mA}$, errors are small with low deviation, thus complying with Sect. VI-B2. Large values of I_r produce large errors. Node 1 exhibits considerably better accuracy than node 5 for heavy consumption. Since both nodes have a similar consumption profile and regulator efficiency, the capacity used with node 5 has presumably decreased during the experiment.

D. Feasibility of Prediction

To answer the question, whether predicting a node's future energetic state is possible, we applied the method presented in Sect. IV-A. We ran one experiment with 24 and 144 time slots each per day. Capacitor voltage V_c was predicted using the consumption and harvesting traces from the field test, but we removed the bias of the solar current sensor. We did not use harvest prediction, because we aim at evaluating the general practicability of our method. Predictions are calculated for one day in advance at the beginning of every slot. For the prediction of the first slot, the algorithm uses the current supercap voltage for y_0, see (5). The following predictions iteratively use the result of the previously predicted V_c.

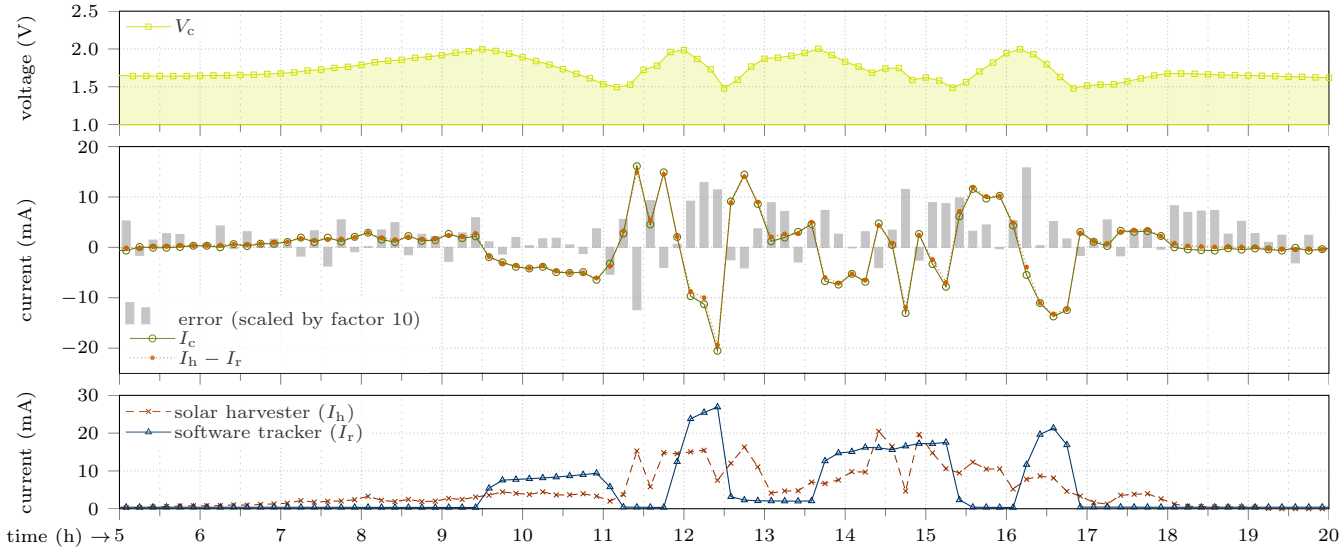
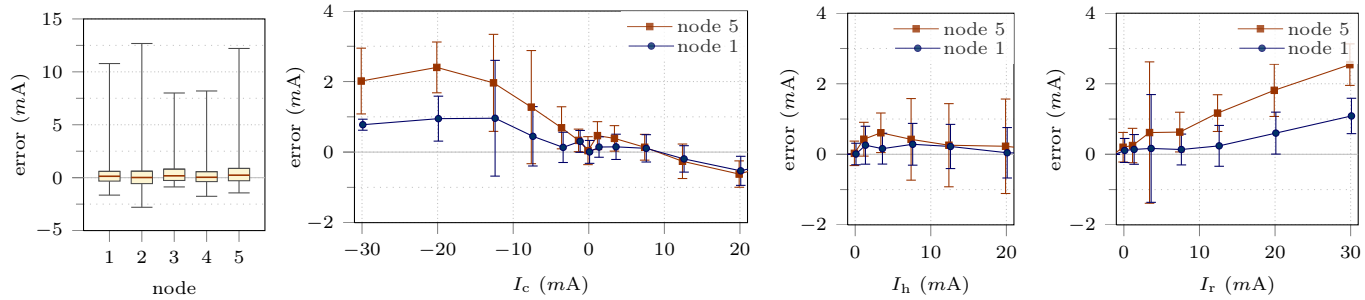


Fig. 7: Trace of sensor node 1



(a) 10%, 50%, 90% quantiles (b) empirical cap current (c) harvester current (d) regulator current

Fig. 8: Overall error analysis (first plot) and clustered mean errors and standard deviations per current component

Fig. 9 shows the 24 h-forecasts generated at midnight for three days—day 2 is the same as in Fig. 7. The figure does not show prediction updates in a day, i.e., all predictions have been created at midnight of the corresponding day. Prediction works fine in the early hours of each day. Note that without removing the solar current sensor bias, predicted values of V_c would be too large. The prediction for the third day produces a maximum error of 0.1 V (even for only 24 slots), which is a quite accurate result considering error propagation and calibration errors. Predictions start to fail as soon as the consumption of the node changes massively when touching the 1.5 V and 2 V thresholds, respectively. The more of these changes occur, the larger the prediction error becomes. As discussed previously, the absolute model error exceeds 2 mA in these cases, resulting in a minimum voltage error of 0.36 V in 60 min (24 slots) for a 50 F supercap. The underlying problem is visible in the close-up (right subfigure) of the second day. Prediction is based on the average consumption of a few mA, but the real consumption course leads to a different picture: Directly after time A, consumption is suddenly and heavily decreased (V_c is falling and stabilizing shortly after passing the 1.5 V threshold at time A). Consumption stays low until reaching the 2 V threshold shortly before time B, when it is heavily increased (V_c quickly falls again). Since the major portion of consumption occurs close to time B, the solar cell charges the cap to a larger V_c, so that I_r is much smaller, cf. (1). Unlike predicted, consumption occurs in a more efficient power point. From time B to C, the effect is smaller, as consumption is more symmetric.

Estimating a sensor node's future energetic course is feasible, because reasonable results are achieved even for small errors of the input variables. The model only fails in case of large errors or heavy, unevenly distributed node loads—i.e., long periods of extremely high current draw followed by periods of low load. However, this situation is unlikely in sensor networks with tiny-sized nodes running adaptive duty-cycling and energy-management algorithms aiming at evenly distributed consumption and stable operation.

E. Portability and Limitations of Results

The presented results are not restricted to our hardware. In contrast, the observations regarding supercap leakage, charging

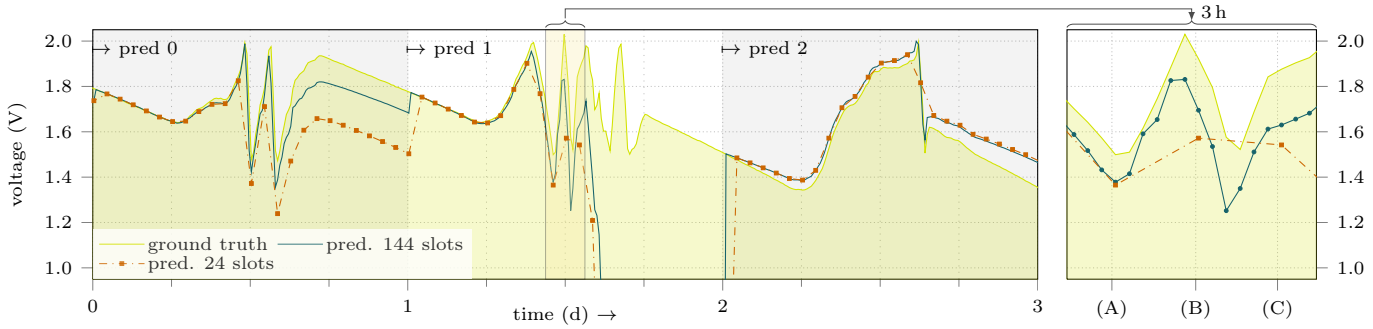


Fig. 9: Predictions created at midnight for a 24 h horizon (24 vs. 144 slots)

behavior, and temperature dependency are valid for similar consumption profiles. Software consumption tracking will yield comparable results for other combinations of sensor nodes and low-power regulators, since our results comply with previous research. We assume that our system model holds for similar hardware designs with direct charging circuits—maximum power-point tracking chargers are not supported.

Predicting a node's future energetic course (e.g., its supercap voltage) is generally feasible, but harvest forecasts are required. Inspired by [2] and the fact that forecasting cannot be exact in a real-world deployment, we think that conservative forecasts should be used, making energy excess more likely than an energy deficit, which might not be compensated on a cloudy or rainy day. In addition, energy-neutral operation has to be defined in a more practical context: How does a predicted, energy-neutral voltage course look like?

VII. Conclusion

This paper contributed to closing the gap towards holistic online energy assessment. We presented a tiny, low-cost energy harvester with a solar cell, supercapacitor, and power-efficient switching regulator. We derived an energy flow system model and devised a method to enable predicting a node's future energetic state. This method allows an implementation suitable for resource-constrained sensor nodes. We implemented an energy assessment layer, based on state-of-the-art energy assessment techniques, for TinyOS and ran a five-node field test for over three weeks to collect real energy traces.

Our evaluation verifies the presented energy model and supports practical collaboration of energy assessment techniques. It shows that online capacity self-calibration on each node is feasible and also required: We experienced a maximum deviation of nominal and actual capacity of more than 11%. Our model simplifications hold: temperature influence and supercap leakage as well as per-node configuration of consumption and regulator efficiency can be neglected. To obtain a valid picture of a node's future energetic state, precise measurements of the energy intake and smooth consumption profiles are essential. Unevenly distributed consumption profiles result in erroneous predictions. This can be avoided by updating predictions upon heavy consumption changes. We find that conservative energy intake forecasts can be paired with our model to derive duty cycles or task schedules ensuring effective while perpetual node operation.

References

[1] M. Ali, B. Al-Hashimi, J. Recas, and D. Atienza. Evaluation and Design Exploration of Solar Harvested-Energy Prediction Algorithm. In *DATE*, 2010.

[2] D. Brunelli, L. Benini, C. Moser, and L. Thiele. Robust and Low Complexity Rate Control for Solar Powered Sensors. In *DATE*, 2008.

[3] A. Dunkels, F. Österlind, N. Tsiftes, and Z. He. Software-based On-line Energy Estimation for Sensor Nodes. In *Emnets*, 2007.

[4] P. Dutta, M. Feldmeier, J. Paradiso, and D. Culler. Energy Metering for Free: Augmenting Switching Regulators for Real-Time Monitoring. In *IPSN*, 2008.

[5] P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler. Trio: Enabling Sustainable and Scalable Outdoor Wireless Sensor Network Deployments. In *IPSN*, 2006.

[6] R. Fonseca, P. Dutta, P. Levis, and I. Stoica. Quanto: Tracking Energy in Networked Embedded Systems. In *OSDI*, 2008.

[7] P. Hurni, T. Braun, B. Nyffenegger, and A. Hergenroeder. On The Accuracy of Software-based Energy Estimation Techniques. In *EWSN*, 2011.

[8] X. Jiang, J. Polastre, and D. Culler. Perpetual Environmentally Powered Sensor Networks. In *IPSN*, 2005.

[9] A. Kansal, J. Hsu, S. Zahedi, and B. S. M. Powermanagement in Energy Harvesting Sensor Networks. *Trans. on Embedded Computing Sys.*, 2007.

[10] C. Moser, L. Thiele, D. Brunelli, and L. Benini. Adaptive Power Management for Environmentally Powered Systems. *IEEE Trans. Computers*, 59(4), 2010.

[11] L. Mottola, G. Picco, M. Ceriotti, S. Guna, and A. Murphy. Not All Wireless Sensor Networks Are Created Equal: A Comparative Study On Tunnels. *Transactions on Sensor Networks (TOSN)*, 7(2), 2010.

[12] J. Recas Piorno, C. Bergonzini, D. Atienza, and T. Simunic Rosing. Prediction and Management in Energy Harvested Wireless Sensor Nodes. In *VITAE*, 2009.

[13] C. Renner, J. Jessen, and V. Turau. Lifetime Prediction for Supercapacitor-powered Wireless Sensor Nodes. In *FGSN*, 2009.

[14] C. Renner and V. Turau. CapLibrate: Self-Calibration of an Energy Harvesting Power Supply with Supercapacitors. In *ARCS*, 2010.

[15] C. Renner and V. Turau. Adaptive Energy-Harvest Profiling to Enhance Depletion-Safe Operation and Efficient Task Scheduling. *Sustainable Computing: Informatics and Systems*, 2(1), 2012.

[16] T. Schmid, P. Dutta, and M. Srivastava. High-Resolution, Low-Power Time Synchronization an Oxymoron No More. In *IPSN*, 2010.

[17] F. Simjee and H. C. P. Everlast: Long-Life, Supercapacitor-Operated Wireless Sensor Node. In *ISLPED*, 2006.

[18] J. Taneja, J. Jeong, and D. Culler. Design, Modeling, and Capacity Planning for Micro-solar Power Sensor Networks. In *IPSN*, 2008.

[19] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong. A Macroscope in the Redwoods. In *SenSys*, 2005.

[20] B. Zhang, R. Simon, and H. Aydin. Energy Management for Time-Critical Energy Harvesting Wireless Sensor Networks. In *SSS*, 2010.