

Exploring the Effect of Energy Storage Sizing on Intermittent Computing System Performance

Jie Zhan, Geoff V. Merrett, *Senior Member, IEEE*, and Alex S. Weddell, *Member, IEEE*

Abstract—Batteryless energy-harvesting devices promise to deliver a sustainable Internet of Things. Intermittent computing is an emerging area, where forward progress of application execution is maintained by saving volatile computing state into non-volatile memory before power interruptions, and restored afterwards. Conventional intermittent computing approaches typically minimize energy storage to reduce device dimensions and interruption periods, but this can result in high state-saving and -restoring overheads and impede forward progress. In this paper, we argue that adding a small amount of energy storage can significantly improve forward progress. We develop a theoretical intermittent computing model that accurately estimates forward progress, with an experimentally validated mean error of 0.5%. Using this model, we show that appropriately sizing energy storage can improve forward progress by up to 65% with a constant current supply, and 43% with real-world photovoltaic sources. Finally, we demonstrate its use in a cost function-based sizing approach, achieving 93% of maximum forward progress while saving 83% of capacitor volume and 90% of interruption periods.

compared to what?

Index Terms—Intermittent computing, energy harvesting, energy storage, forward progress, batteryless, wireless sensor networks, internet of things.

I. INTRODUCTION AND RELATED WORK

INTERNET of Things (IoT) devices are becoming ubiquitous, with forecasts of hundreds of billions being installed in the near future [1]. They are conventionally battery-powered, and thus have constrained lifespans. This necessitates the inconvenience of periodic battery replacement.

Energy-harvesting is a potential solution. Environmentally harvested power is, however, intrinsically variable and intermittent [2]. Traditionally, large energy storage devices such as rechargeable batteries or supercapacitors are used to smooth out the variability in the supply [3]. Unfortunately, these increase cost and device dimensions [4], raise pollution concerns [5], and still have limited lifespans [6].

Recently, *intermittent computing systems* (ICSs) have been proposed as an alternative [7]. Instead of using large energy storage devices to sustain execution, they tolerate power interruptions by saving the state of the system into non-volatile memory (NVM) so that computation can continue when power is restored. They may save this state (e.g. CPU registers and RAM contents) either *statically* at pre-defined points, or *reactively* when the supply is about to fail [7].

by design

The authors are with the School of Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ, UK.

This work was *mainly* supported by the UK Engineering and Physical Sciences Research Council (EPSRC) under EP/P010164/1.

Data and software associated with this paper will be made available online on publication.

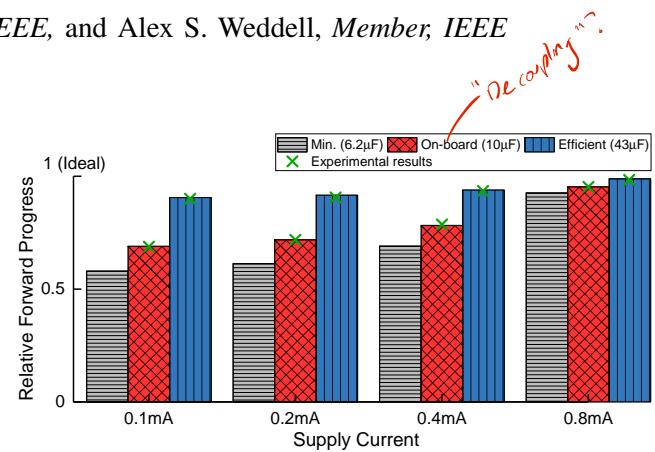


Fig. 1. An efficient energy storage capacitance rather than a minimum or on-board one can improve forward progress, especially when the supply is weak. Results are generated by the proposed model along with experimental comparisons.

Static approaches save state at points determined at design or compile time, either by inserting checkpoints [8], [9] or decomposing a program into atomic tasks [10], [11]. After a power interruption, progress rolls back and resumes from the last saved checkpoint or task boundary. This can introduce issues such as violation of data memory consistency, along with wasting energy on lost and re-executed progress.

Conversely, *reactive approaches* monitor the supply voltage and only save state when it falls below a threshold [12], which is set high enough to reliably save state even with a total and immediate drop-off in harvested energy. They will then enter a low-power mode, in many cases preserving the data in volatile memory and avoiding re-execution and memory inconsistency. These typically make more forward progress than static approaches, e.g. a $2.5 \times$ mean computational speedup [13].

In ICSs, *forward progress* denotes the effective application progress, excluding re-executed progress, lost progress, and state-saving and -restoring operations [14]. The amount of forward progress directly determines application performance, e.g. program iteration rate or task completion time. In this paper, to allow fair comparison, we define normalized forward progress as *the ratio of the effective execution time to the total elapsed time*, without being restricted to a specific workload.

With the goal of minimizing device dimensions and interruption periods, most ICS approaches adopt a minimum amount of energy storage [15]–[19]. This is typically just sufficient for the most energy-expensive atomic operation¹. However, our assertion is that this can be inherently inefficient

¹Atomic operations in ICSs denote operations that should be completed in one continuous period. If an atomic operation is interrupted by a power failure, it should be re-executed rather than resumed. Examples of atomic operations include saving and restoring volatile state, transmitting and receiving packets, and sampling sequences of data from sensors.

in terms of time and energy. We show that a system with minimum energy storage has to frequently go through a cycle of: wake up; restore state; execute program; save state; halt.

We propose that provisioning slightly more energy storage can prolong the operating cycles, reduce the frequency of interruptions, and hence improve forward progress. As shown in Fig. 1, using an efficiently-sized energy storage capacitance ($43\ \mu\text{F}$) achieves up to 55.2% forward progress improvement compared against using the theoretical minimum amount of capacitance ($6.2\ \mu\text{F}$). The relationship between ICS energy storage capacitance and forward progress has not previously been explored, and the challenge of sizing energy storage to improve forward progress while moderating the physical size and interruption periods is largely unaddressed.

This paper presents an approach for sizing energy storage in ICSs, quantifying and trading-off forward progress, capacitor volume, and interruption periods. The main contributions are:

- A theoretical reactive ICS model which accurately estimates forward progress; experimental validation shows a 0.5% mean error (Section III).
- A model-based approach that recommends an appropriate energy storage capacitance in ICSs (Section IV).
- An exploration based on the model, where we analyze the energy storage sizing effect on forward progress with respect to supply current and volatile state size, showing up to 64.9% forward progress improvement (Section V).

On average, the proposed scheme achieves 93% of the maximum forward progress while the capacitor volume remains insignificant as simulated with real energy availability data (Section VI). While most ICSs are designed with a minimal amount of energy storage, this work demonstrates that increasing the energy storage beyond the minimum can bring real benefits. The associated simulation tool, coded in C, is available open-source at (link to be provided on publication).

II. REVIEW OF SIMULATION AND DESIGN EXPLORATION

This needs to be updated - Alex has some additional references .

To explore forward progress of ICSs in a long-term deployment, it is necessary to simulate the intermittent operations in a short time step (microseconds) as well as the overall forward progress in a long-term period (up to years).

A few models have been proposed for exploring system designs and parameters in ICSs to improve forward progress. Su et al. [20] provide a model dedicated to a dual-channel solar-powered nonvolatile sensor node. Jackson et al. [21] propose a model to explore battery usage in ICSs. These two models are configured for simulations with large energy storage from mF-scale supercapacitors to batteries, and thus they cannot respond to frequent power interruptions and accurately estimate forward progress when using minimized energy storage (e.g. $16\ \mu\text{F}$ [15]). In contrast, a set of fine-grained model are also proposed to accurately simulate the frequent micro operations in ICSs. NVPsim [22] and AES [23] are gem5-based simulators for nonvolatile processors, where NVPsim focuses on processor-wise simulation and AES additionally supports system-wise peripherals and modules. EH model [24] compares a range of ICS approaches in a single active period with

TABLE I
MODEL PARAMETERS OF REACTIVE ICS

Input Parameters	
I_{harv}	Energy harvester current supply
C	Energy storage capacitance
Configuration Parameters	
I_{exe}	Execution current draw
I_{sleep}	Sleep current draw
I_r	Restore current draw
I_s	Save current draw
I_{leak}	Leakage current draw
V_r	Restore voltage threshold
V_s	Save voltage threshold
T_r	Restore time overhead
T_s	Save time overhead
Output Parameter	
α_{exe}	Normalized forward progress

the same energy budget and quantify forward progress by the energy spent on the effective execution. Fused [25] is a closed-loop simulator to allow interactions among power consumption, power supply, and forward progress output. However, these fine-grained simulators become time-consuming when processing long-term energy data, especially when multiple system configurations are to be tested iteratively.

Apart from models and simulators, hardware emulators of energy harvesters [26], [27] also support repeatable energy harvesting conditions for experimental comparisons. However, though manifesting practical results, experiments are limited by hardware options and also time-consuming to perform a long-term test.

III. MODEL OF REACTIVE INTERMITTENT COMPUTING

To facilitate the understanding and exploration of reactive ICSs, we present a theoretical model. Parameters of this model are listed in Table I. The model outputs the normalized forward progress α_{exe} with a constant current supply I_{harv} . The model assumes that all configuration parameters remain constant.

For brevity, I_{in} denotes the usable input current as expressed in (1). The effect of capacitor leakage current, I_{leak} , is discussed at the end of Section III-B.

$$I_{in} = I_{harv} - I_{leak} \quad (1)$$

A. Operating Modes of Reactive ICS

The behavior of reactive ICSs can be classified into three operating modes depending on the supply current, as shown in Fig. 2. These are differentiated by the relationship between input current I_{in} and the system's current draw in sleep or active modes. We define the three modes as:

- *Off mode*: When $I_{in} < I_{sleep}$, the system stays inactive. The supply voltage V_{cc} cannot rise above the restore threshold V_r to wake the system and start execution. The sleep current I_{sleep} includes the consumption of voltage monitoring circuits and system idle current.
- *Intermittent mode*: When $I_{sleep} < I_{in} < I_{exe}$, the system executes intermittently after $V_{cc} > V_r$ and before

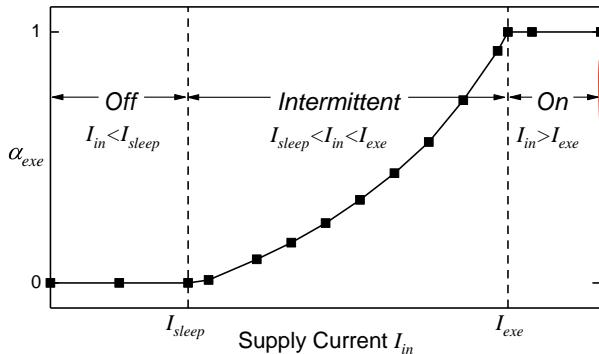


Fig. 2. Operating modes of reactive ICSSs, and achieved forward progress against supply current.

Surf order

$V_{cc} < V_s$. V_{cc} can rise above V_r and the system starts execution. However, the stored energy is then consumed by the load as $I_{in} < I_{exe}$, causing V_{cc} to eventually drop below the save threshold V_s , where the system saves its state and sleeps. The system sleeps until V_{cc} rises to V_r again and then resumes execution. In general, a higher I_{in} leads to more forward progress in this mode, but the exact relationship between I_{in} and forward progress requires further analysis.

- **On mode:** When $I_{in} > I_{exe}$, the system executes constantly as the supply voltage V_{cc} never drops below V_s . V_{cc} grows until I_{in} and I_{exe} are in equilibrium, as I_{in} may decrease due to poor impedance matching, and I_{exe} may increase due to higher CPU current draw, or dissipation through overvoltage protection circuits.

B. Formulating Forward Progress

Next, through theoretical analysis we derive formulations to calculate α_{exe} from I_{in} and energy storage capacitance C . We then explore the effect of capacitor leakage on maximum forward progress.

In the On and Off modes, the normalized forward progress is trivial to find (simply 1 in On mode, 0 in Off mode). In the Intermittent mode, as shown in Fig. 3, the system goes through four intervals in turn, i.e. charging, restoring, executing, and saving, with current consumption of I_{sleep} , I_r , I_{exe} , and I_s in each interval respectively. The normalized forward progress, i.e. effective execution time ratio, is indicated as T_{exe}/T_{period} , where T_{exe} is the time spent on effective execution in one operating cycle and T_{period} is the period of operating cycles. Hence, the forward progress given all supply levels is expressed as:

$$\alpha_{exe} = \begin{cases} 0 & , \text{ Off } (I_{in} < I_{sleep}) \\ \frac{T_{exe}}{T_{period}} & , \text{ Intermittent } (I_{sleep} < I_{in} < I_{exe}) \\ 1 & , \text{ On } (I_{in} > I_{exe}) \end{cases} \quad (2)$$

In the following analysis, we focus on deriving T_{exe}/T_{period} in the Intermittent mode. Let V_{pr} (post-restore) and V_{ps} (post-

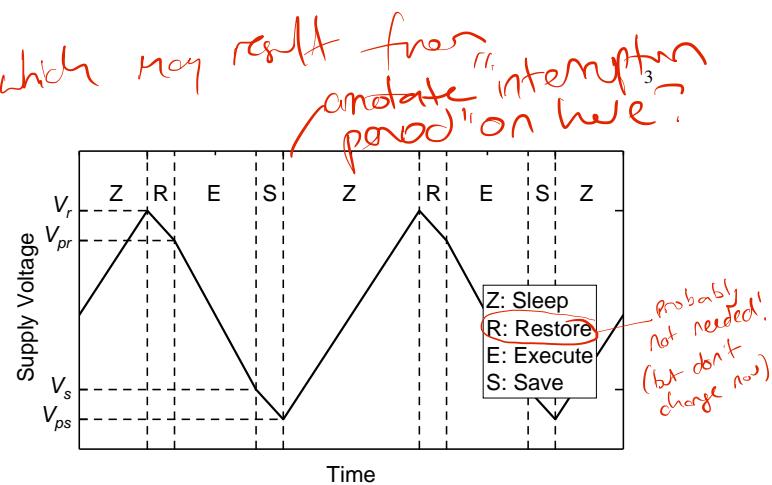


Fig. 3. Operating cycles in the Intermittent mode.

save) denote the voltage after restoring and saving operations. V_{pr} and V_{ps} can be calculated as:

$$V_{pr} = V_r + \frac{T_r(I_{in} - I_r)}{C} \quad (3)$$

$$V_{ps} = V_s + \frac{T_s(I_{in} - I_s)}{C} \quad (4)$$

With (3), the time spent on effective execution T_{exe} in one operating cycle can be expressed as:

$$T_{exe} = \frac{C(V_{pr} - V_s)}{I_{exe} - I_{in}} \quad (5)$$

Analogously, with (4), the charging interval can be described as:

$$T_{charge} = \frac{C(V_r - V_{ps})}{I_{in} - I_{sleep}} \quad (6)$$

With eqs. (5) and (6), the period of an operating cycle is:

$$T_{period} = T_{charge} + T_r + T_{exe} + T_s \quad (7)$$

Finally, combining eqs. (3)–(7), we obtain normalized forward progress α_{exe} in the Intermittent mode as:

$$\alpha_{exe} = \frac{T_{exe}}{T_{period}} = \frac{\frac{C(V_r - V_s) + T_r(I_{in} - I_r)}{I_{exe} - I_{in}}}{\frac{C(V_r - V_s) + T_s(I_{in} - I_{sleep})}{I_{in} - I_{sleep}} + \frac{C(V_r - V_s) + T_r(I_{exe} - I_r)}{I_{exe} - I_{in}}} \quad (8)$$

In the numerator T_{exe} , $C(V_r - V_s)$ represents the amount of charge in the capacitor available for restoring and executing. $T_r(I_{in} - I_r)$ represents the charge used by a restore operation. $I_{exe} - I_{in}$ is the rate of charge consumption from the energy storage during execution.

To explore the effect of energy storage on forward progress, we need to analyze $d\alpha_{exe}/dC$. Here, if we assume that I_{leak} remains constant, α_{exe} keeps increasing and approaches $(I_{in} - I_{sleep})/(I_{exe} - I_{sleep})$ when storage capacitance C increases. Defining $(I_{in} - I_{sleep})/(I_{exe} - I_{sleep})$ as α_{exe_ideal} , $\alpha_{exe} = \alpha_{exe_ideal}$ is an ideal case, where restore and save overheads are absent.

In an electrolytic capacitor, however, I_{leak} typically increases with C with the following relationship [28], [29]:

$$I_{leak} = kCV_{cc} \quad (9)$$

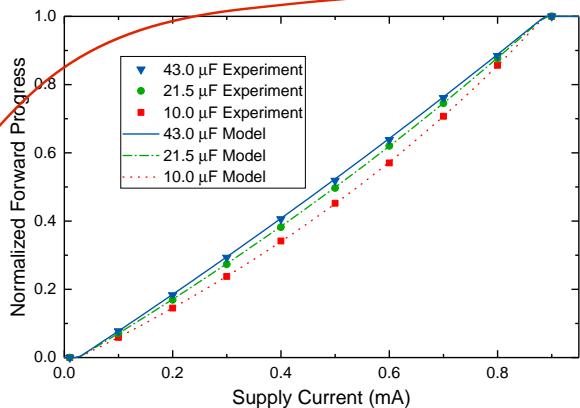


Fig. 4. Model validation with experimental and modelled forward progress.

where k is a constant normally in a range 0.01 to 0.03 ($\frac{A}{FV}$). Combining (9) with (1), dI_{in}/dC is $-kV_{cc}$, meaning I_{in} decreases linearly as C increases. Thus, when C increases, α_{exe} keeps approaching α_{exe_ideal} while α_{exe_ideal} decreases. Hence, there is an optimal capacitance that leads to the maximum α_{exe} considering I_{leak} increases with C .

C. Model Validation

A Texas Instruments MSP430FR6989 development board was used as the experimental platform. The on-board capacitance was measured as $10.0\mu F$, so was the minimum capacitance that could be tested. Further capacitors were added to provide extra energy storage. The leakage current of these capacitors was measured to be less than 10nA at 3V , which is negligible compared to the μA -level current consumption of the platform; hence we omit capacitor leakage in the following experiment and model output. However, this is only applicable to the capacitors and the capacitance range in this experiment; we consider that a leakage model is still necessary for general application. *[bit weak]* $[0.1, 0.2, \dots, 0.8]\text{mA}$ steps

To validate the accuracy of our model, we powered the device with a range of supply currents to operate the device in Intermittent mode ($0.1\text{--}0.8\text{ mA}$, 0.1 mA per step), and repeated the tests with three energy storage capacities: a) on-board $10.0\mu F$ capacitance only; b) $21.5\mu F$ (measured total $11.5\mu F$ added); c) $43.0\mu F$ measured total $(33.0\mu F$ added). We compared the actual forward progress against predictions generated from our model. As shown in Fig. 4, the model-generated output matches closely with the experimental results with only 0.5 % mean absolute percentage error. Hence, the model is able to accurately estimate forward progress for design exploration.

IV. ENERGY STORAGE SIZING APPROACH

We propose a sizing approach which recommends an appropriate energy storage capacitance, trading-off forward progress against capacitor volume and interruption periods. A system model has been implemented, which accepts real long-term data on environmental energy conditions. A range of energy storage capacities are swept through, with the model outputting values for forward progress, capacitor volume, and interruption

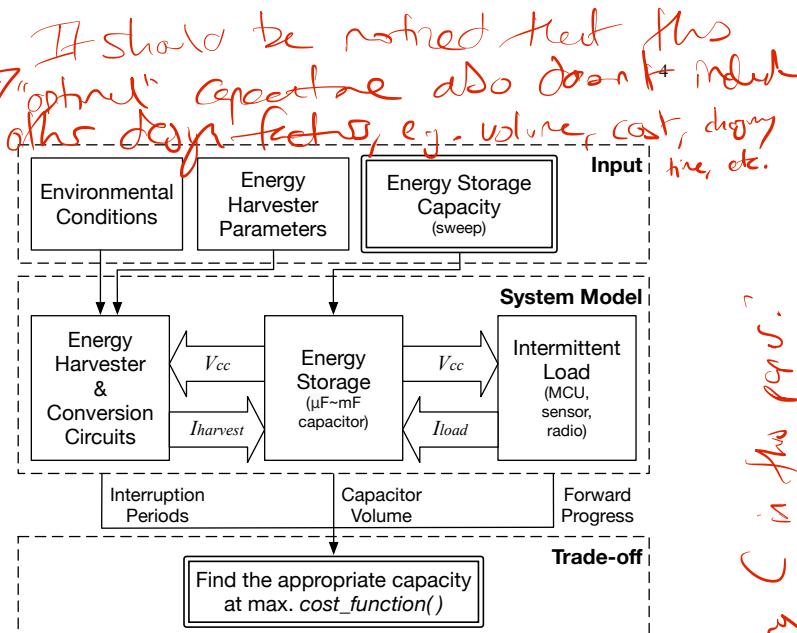


Fig. 5. Structure of the proposed system model and sizing approach.

periods. These are subsequently traded-off in a cost function to indicate the optimal energy storage capacitance. This process is summarized in Fig. 5 with details explained as follows.

- Input:** A time trace of environmental energy conditions in the intended deployment location is provided as an input, along with the energy harvester size; for design exploration, these can optionally be changed to explore variations and scales of harvested power. A pre-defined set of energy storage capacities are swept through.
- System Model and Output:** This contains three modules:
 - Energy Harvester and Conversion Circuits:** The energy harvester module transduces environmental energy into electricity. The harvested power is typically conditioned to provide a suitable voltage for charging the energy storage and supplying the load efficiently. The energy harvester and conversion circuits can be modelled together as a module because they are usually coupled and integrated. In ICSs, conversion circuits may simply be a diode to inhibit backflow of current.
 - Energy Storage:** Energy storage in ICSs is usually in the form of a μF - to mF -scale capacitor. It must be sufficient to complete the most energy-expensive atomic operation, and may be formed only of the decoupling capacitor(s).
 - Intermittent Load:** Includes all the power consumers in an ICS, such as a microcontroller, sensors, and a radio. The outputs from the module are the interruption periods, capacitor volume, and attained forward progress.
- Trade-off** The optimal capacitance is then found through a cost function, e.g. this may trade-off forward progress against capacitor volume and interruption periods.

V. EXPLORATION OF ENERGY STORAGE SIZING

In this section, we configure the reactive ICS model, and then present an exploration of the relationship between α_{exe} and C with respect to I_{in} and volatile state size.

but in previous section, can't I take the only reason for an optimal capacitance?

** Significantly "In the section we have... However, ... this is next section"*

TABLE II
PROFILED MCU PARAMETERS

Parameter	Value
I_{exe}	887 μA
I_{sleep}	26 μA
I_r	971 μA
I_s	811 μA
T_r	1.903 ms
T_s	1.880 ms

A. Model Configuration

1) *Energy Storage*: The energy storage is represented as an ideal capacitor with leakage current. Its terminal voltage is directly applied to the load, so is modelled as:

$$C \frac{dV_{cc}}{dt} = I_{harv} - I_{load} - I_{leak} \quad (10)$$

where I_{load} is the current consumption of the load. In this exploration, we refer to an empirical I_{leak} for an off-the-shelf tantalum capacitor [30], which depends on capacitance C , rated voltage V_{rated} , and terminal voltage V_{cc} [29]:

$$I_{leak} = 0.01\lambda CV_{rated} \quad (A) \quad (11)$$

where λ denotes the ratio of the actual current leakage at V_{cc} to the current leakage at V_{rated} , and λ is approximated as:

$$\lambda = 0.05 \times 20^{\frac{V_{cc}}{V_{rated}}} \quad (12)$$

For this exercise, we assume a typical load of $< 4.0 \text{ V}$, so to minimize leakage, we select a device with $V_{rated} = 10 \text{ V}$ so as to operate between 25–40 % of its rated voltage [29].

2) *Intermittent Load*: We implemented and parameterized a reactive ICS [15] on a TI MSP430FR6989 microcontroller. The parameters are profiled with the MCU running a Dijkstra path finding algorithm with 1696 B RAM usage at 8 MHz. The supply voltage monitoring circuits use the MCUs internal comparator and an external $3 \text{ M}\Omega$ voltage divider. The restore and save voltage thresholds are set as $V_r = 2.4 \text{ V}$ and $V_s = 2.1 \text{ V}$ respectively. The MCU shutdown voltage V_{off} is 1.8 V.

The current draw was profiled with experimental measurements at a range of supply voltages. The variation of I_{sleep} between V_{off} and V_r is 2 %, and for I_{exe} between V_s and 3.3 V is 1.5 %. I_{exe} also has a run-time variation of 2.8 % due to a variable memory access rate. We omit minor variations and use the mean of I_{exe} and I_{sleep} in the model. I_r and I_s are measured at V_r and V_s respectively. Given the voltage thresholds and the current consumption, the minimum energy storage capacity is 6.2 μF . This guarantees that a save and restore operation can complete even if the incoming supply current drops instantaneously to zero. The model parameters in Table II are given as an example, and can be changed for different load characteristics. For example, T_r and T_s can be tuned for different volatile state sizes.

B. Sizing Energy Storage to Improve Forward Progress

1) *Impact of Supply Current*: Increasing the energy storage capacity above the minimum can improve forward progress

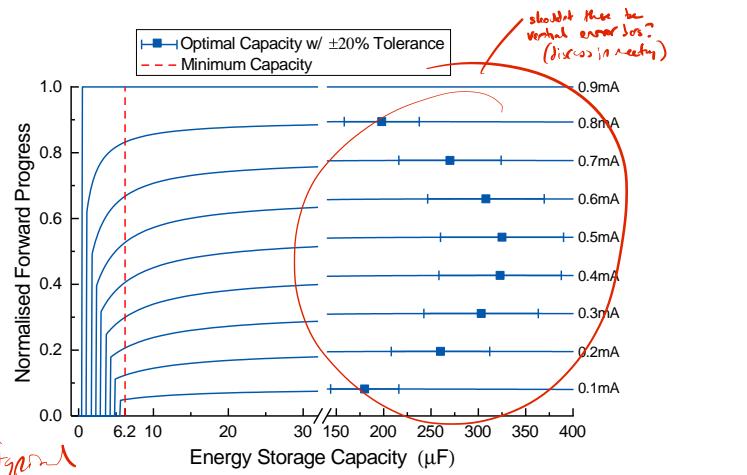


Fig. 6. Forward progress against energy storage capacity at different levels of constant supply current. *Error bars and optimal points just in front of +20% tolerance on C.*

by reducing the frequency of power interruptions, but this improvement may be offset by increased leakage. Fig. 6 shows the relationship between forward progress and energy storage capacity for a range of constant supply currents. Optimal capacitances are shown for each current value.

The minimum capacitance is calculated to deliver correct operation even if the supply current instantaneously drops to zero. If it does not drop to zero, this means that correct operation could have continued even with a smaller capacitance, though designing a system in this way would be inadvisable owing to unpredictability of the supply. This property is illustrated in Fig. 6, in the area on the left of the dashed line. It may be observed that for each of the current values, there is a sudden drop-off towards zero forward progress, illustrating the hazard of setting the capacitor size too small: the stored energy is too low to allow a restore and save to be undertaken.

Typically, commercially-available capacitors have a $\pm 20\%$ tolerance. The effect of this variation on forward progress is shown to be negligible ($< 0.23\%$) in Fig. 6. However, it must be pointed out that the effect would be much more pronounced if operating at the minimum calculated capacitance value. For this reason, it is recommended that a tolerance is applied to the minimum capacitance value when designing systems.

Fig. 7 shows that ~~up to 69%~~ improvement in forward progress can be achieved when using the optimal capacitance instead of the minimum. However, it may not be desirable to set the capacitance solely for maximizing forward progress, because there are often trade-offs with other factors including increased interruption periods and dimensions (see Sec. VI-C).

While a large improvement can be delivered with the optimal capacitance, ~~much smaller capacitances (mean 31.2% of the optimal value)~~ can give 95 % of the maximum improvement (Fig. 7). For example, reducing from 325 μF to 90 μF gives 95 % of the maximum improvement with a 0.5 mA supply.

2) *Impact of Volatile State Size*: The size of volatile state differs across applications with different amounts of RAM usage, and hence incurs varying time and energy overheads for restore and save operations. We measured time overheads of restore and save operations in the minimum case (64B register data and a 160B stack) and the maximum case (64B register

95% of the gain can still be obtained with significantly smaller capacities (mean 31.2% of the opt val).

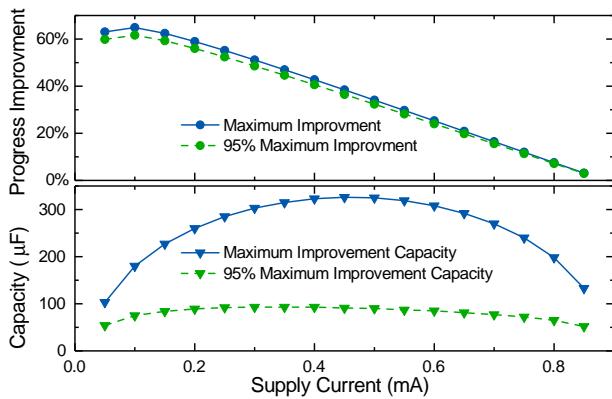


Fig. 7. Maximum forward progress improvement by sizing energy storage given a spectrum of supply current (normalized by the minimum capacity case), with the corresponding maximum and sub-maximum (95 % of maximum) capacitance.

TABLE III
LINEAR SCALING RANGE OF VOLATILE STATE SIZE AND RESTORE/SAVE TIME OVERHEADS

State Size (Registers + SRAM)	Restore Time	Save Time
64B + 160B (lower bound)	232 μs	208 μs
64B + 2048B (upper bound)	2.298 ms	2.274 ms

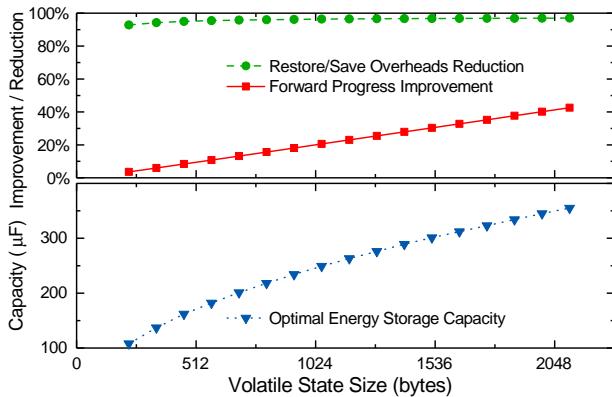


Fig. 8. Impact of RAM usage (linear to restore/save overheads) on sizing energy storage with 0.4 mA current supply. Improvement and reduction are normalized by the minimum capacity case.

data and 2048B RAM) respectively as shown in Table III. As the restore and save time overheads are expected to be linear [31], the model can be tuned for various volatile state sizes by linearly scaling the profiled values.

An example of this is plotted in Fig. 8. The forward progress improvement by sizing energy storage increases with the volatile state size, and the optimal capacity grows accordingly. The improvement becomes insignificant when the volatile state size is small because the restore and save overheads are already negligible. For example, when the workload uses the least volatile state (the leftmost point), the maximum progress improvement is only 3.6 % although the restore and save overheads are reduced by 92.9 %.

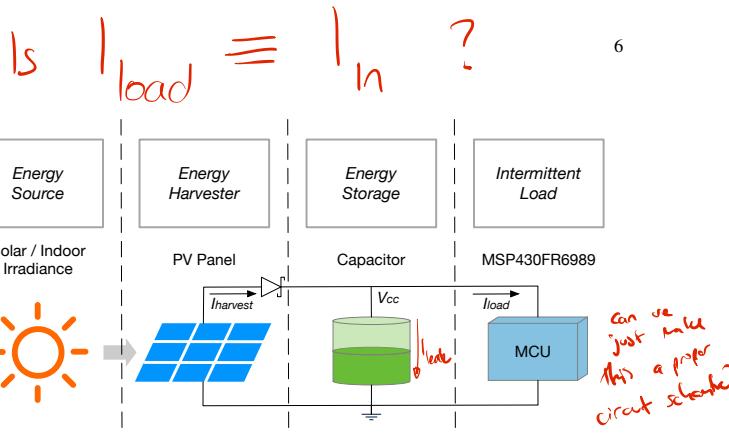


Fig. 9. System model configuration of a PV-based ICS.

C. Validation of Sizing Effects

As shown in Fig. 1, the efficiently-sized energy storage capacitance (43 μF) improves forward progress by up to 55 % and 30 % compared to the minimum and on-board capacitance respectively. We notice that this improvement becomes significant when the supply current attenuates because the saving and restoring overheads occupy a larger proportion of the available energy. Also, this achieves at least 90 % of the ideal forward progress. The ideal case switches between sleeping and executing without restoring and saving overheads (mentioned in Section III-B). These results illustrate the importance of this technique, in particular for conditions where the supply current is low.

VI. DEMONSTRATION OF SIZING APPROACH under real-world conditions (?)

In this section, we model an ICS with a photovoltaic (PV) energy harvester to explore the energy storage sizing effect in real-world energy conditions, and demonstrate use of the proposed sizing approach.

A. System Model Configuration

We integrate the validated reactive ICS model into a system model with a PV energy-harvesting supply as shown in Fig. 9. The energy storage model and the intermittent load model are as presented in Section V. We use a converter-less supply circuit with only a diode at the energy harvester output.

The energy source conditions are imported from NREL outdoor solar irradiance data [32] and EnHANTs indoor irradiance data [33]. To encompass different energy environments, four light source datasets across different environments are used. To convert irradiance into harvested power, we adopt a datasheet-based PV cell model [34]. This model takes the parameters available in common PV cell datasheets, so can easily be reconfigured to suit various PV cells. We refer to a commercial solar cell [35] for PV cell properties as shown in Table IV. We set four cells in series (with $V_{oc} = 3.56V$) to match the operating voltage of the MCU (maximum 3.6V), and model energy harvester sizing by scaling the cell area. A Schottky diode is connected to the energy harvester output in order to prevent current backflow.

B. Exploration with Real-World Energy Source Conditions

In real-world deployments, ambient energy source conditions are dependent on time and location. The energy harvester

I don't think you're signing me

7
Section / heading: Sizing the energy harvester

with min. energy storage

TABLE IV
PV CELL MODEL PROPERTIES

Parameter	Value
Open-Circuit Voltage	0.89 V/cell
Short-Circuit Current	14.8 mA/cm ²
MPP Voltage	0.65 V/cell
MPP Current	12.1 mA/cm ²

need to

and storage should be sized to achieve the desired forward progress across the range of expected conditions.

For the purposes of this exploration, three levels of baseline mean forward progress (α_{exe}) are set as 0.1, 0.2, and 0.3. We use the system model to find the PV panel area that achieves expected forward progress under the different energy source conditions. We scale the PV panel area with the minimum energy storage to find the PV panel area that achieves each target α_{exe} . As specified in Fig. 10, the energy harvester sizes that achieve the desired α_{exe} may span orders of magnitude given different energy source conditions, (from mm² for outdoor sources to cm² for indoor sources).

We analyze the sizing effect of energy storage on forward progress given real-world energy conditions. Fig. 10 shows 7.8–43.3% forward progress improvement by sizing energy storage under the given real-world energy conditions and baseline energy harvester sizes. Another takeaway is optimizing energy storage can either improve forward progress for a given energy harvester size, or reduce the energy harvester size that achieves the target forward progress. Given strong energy sources (e.g. Denver 2018 and Hawaii 2018 outdoor solar sources), increasing energy harvester size efficiently improves forward progress with minor dimensional overheads, e.g. tens of mm²; however, given weak energy sources (e.g. EnHANTs Setup A and Setup D indoor light sources), optimizing energy storage capacitance can save tens of cm² of PV panel area to achieve the same forward progress.

Besides forward progress, we also explore how the capacitor size can change the interruption periods. When interrupted by insufficient power supply, an ICS enters an interruption period where it saves its volatile state, waits for supply voltage to recover, and restores the state to resume execution, without making any forward progress. Applications that require frequent sensing may be negatively affected by long interruption periods. We measure an interruption period as the period between two successive execution periods. We record all the interruption periods during a one-year simulation with 10–50 μ F capacitors, the Denver 2018 dataset, and an 80 mm² PV panel. Fig. 11 presents the distribution of all the interruption periods. With larger energy storage provided, the interruption period is generally prolonged at every percentile. For example, the 90th percentile of interruption periods increases from 32.2 ms at 10 μ F to 123.4 ms at 50 μ F at an approximate rate of 23 ms per 10 μ F. Facilitated by the simulator, developers are enabled to estimate whether the distribution of interruption periods meet their application requirement.

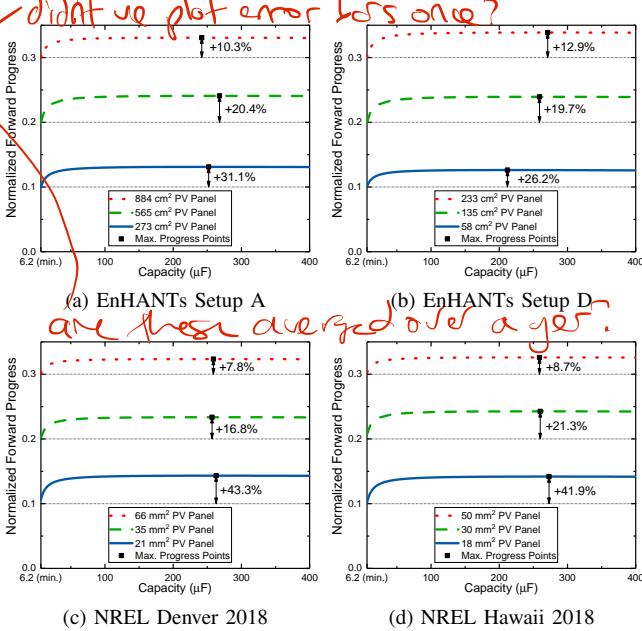


Fig. 10. Forward progress improvement by sizing energy storage given different PV panel areas under real-world energy source conditions. The model is able to find the PV panel area required for achieving the target mean forward progress.

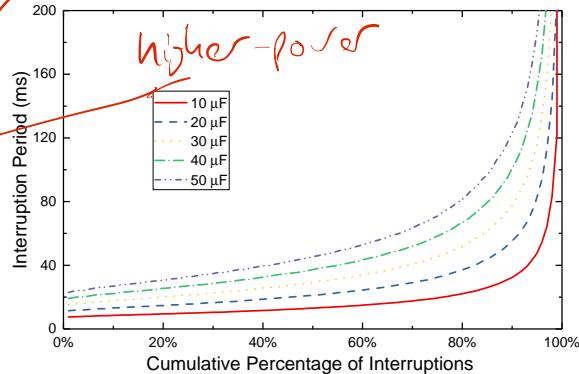


Fig. 11. Distribution of interruption periods.

C. Trading off Forward Progress, Dimensions, and Interruption Periods

Although increasing the storage capacitance improves forward progress, larger capacitance increases both dimensions and interruption periods. We evaluate the overheads of increased capacitor dimensions and interruption periods, and then trade them off against forward progress using a cost function to suggest an optimal capacitor size.

1) *Metric of Dimensions:* The overhead of capacitor dimensions is evaluated by characteristics of off-the-shelf tantalum capacitors. We narrow down the range of sample capacitors within a set of characteristics: low-profile, 10V rated voltage, and surface-mount package, and select six series of capacitors². The volume and capacitance of these devices are plotted

²The series of capacitor considered were: AVX TACmicrochip, AVX F92, Vishay 572D, Vishay 591D, and Vishay 592D.

subject "Sizing the energy storage" // heavy edited the guy harvester sizes for the baseline found project we have the model approach to swap storage capacitors.

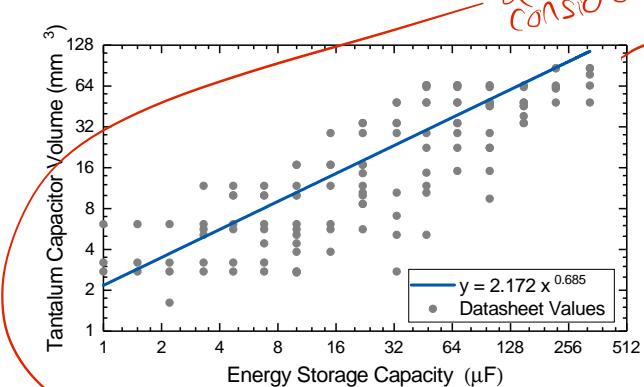


Fig. 12. Tantalum capacitor volume against capacitance for the six series of capacitors analyzed.

in Fig. 12. We use the regression of these data to represent the general capacitance-volume relationship.

2) *Metric of Interruption Periods*: Applications may have various requirements on interruption periods. To demonstrate the usage of our sizing approach, we take the 90th percentile of all interruption periods as an example metric of interruption periods, denoted as $T_{\text{interrupt}}$. This metric indicates 90 % of interruption periods are shorter than $T_{\text{interrupt}}$. This metric can be adapted for particular application requirements.

3) *Trade-offs*: From the previous observations (Fig. 7) we can see that to achieve the optimal progress improvement costs much more storage capacitance (mean 3.2×) than to achieve 95 % improvement. A trade-off is necessary to improve forward progress while restricting the overheads of increased capacitor volume and interruption periods.

As a part of the proposed sizing approach, the cost function in eq. (13) is used to trade-off forward progress, capacitor volume, and interruption periods.

$$f = \frac{\alpha_{\text{exe}}}{k_1} - \left(\frac{v_{\text{cap}}}{k_2} \right)^2 - \left(\frac{T_{\text{interrupt}}}{k_3} \right)^2 \quad (13)$$

where α_{exe} denotes normalized forward progress, v_{cap} represents capacitor volume and $T_{\text{interrupt}}$ represents interruption periods. k_1 , k_2 , and k_3 are independent factors used for normalizing each metric, and they are empirically determined according to applications. In this example, the undesirable parameters are expressed as quadratics to give an increasing cost to higher values. While only three parameters are considered here, others (such as the energy harvester size) could be included for a system-wise sizing scenario. As an example, we configure the function by setting $k_1 = 0.2$, $k_2 = 200 \text{ mm}^3$, and $k_3 = 500 \text{ ms}$.

The effect of the trade-off is plotted in Fig. 13 using the Denver 2018 energy source dataset. Compared to the capacitor size that solely maximizes forward progress, on average, an appropriately-sized capacitor achieves 93 % of the maximum forward progress, while saving 83 % of capacitor volume and 91 % of interruption periods. Compared to the minimum storage case, the appropriately-sized capacitor improves forward progress by 11.7–124.1 % with energy storage increased from 6.2 μF to 30 μF .

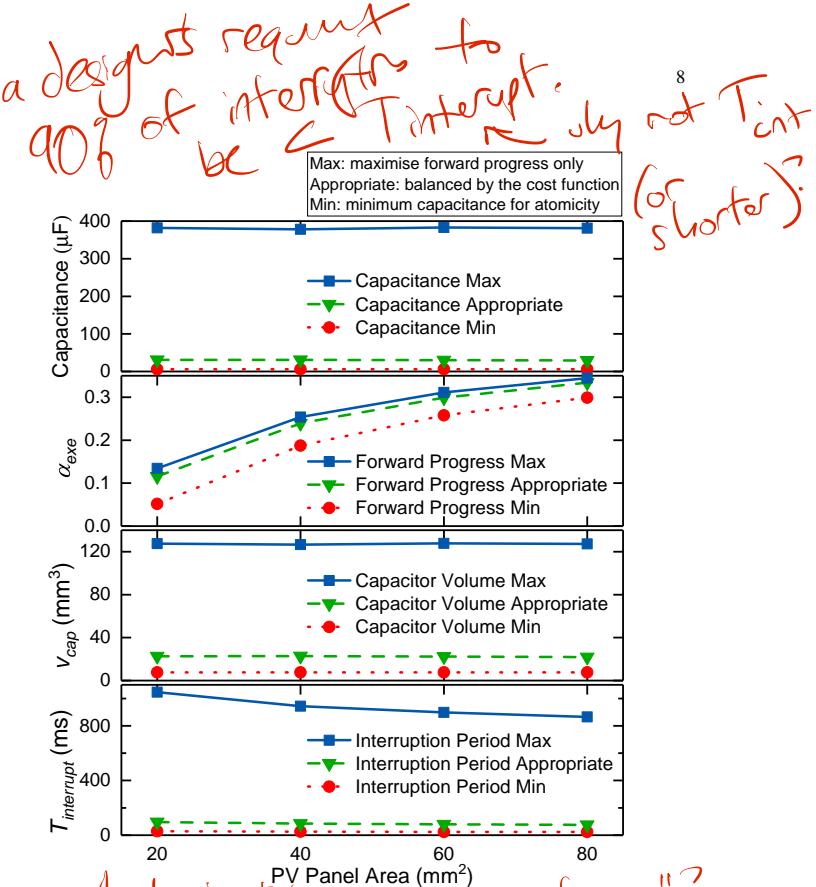


Fig. 13. The sizing approach trades off forward progress, capacitor volume, and interruption periods. The results are plotted against a range of PV panel area, given Denver 2018 energy source dataset.

As shown in Fig. 12, the closest available capacitance that satisfies the 6.2 μF minimum capacitance is 6.8 μF , whereas the closest available capacitance to the 30 μF appropriate capacitance are 33 μF . The minimum volume for these three capacitors are both 2.75 mm^3 , which means using the optimal capacitance, instead of the minimum one, may not incur dimensional overhead. The regressed volume of the above two capacitance values are 8.1 mm^3 and 23.8 mm^3 respectively. However, the selection of capacitors can be dependent on factors other than physical volume, such as reliability, operation temperature, and more specific application needs. These factors can also be added into the cost function if necessary.

VII. CONCLUSIONS

While conventional ICSs have used minimal levels of capacitance, this paper has shown that system performance can be significantly improved by increasing the amount of energy storage. The work includes a simulator which is available to download, enabling researchers to experiment with energy storage sizes to optimize the operation of systems. A cost function can be incorporated, allowing various aspects of system performance to be traded-off. Our conclusion is that energy storage should be carefully designed, rather than minimized or indiscriminately picked, to efficiently operate ICSs.

REFERENCES

- [1] P. Sparks, “The route to a trillion devices,” ARM Ltd., Tech. Rep., 2017.
- [2] Sravanthi Chalasani and J. M. Conrad, “A survey of energy harvesting sources for embedded systems,” in *IEEE SoutheastCon 2008*, Apr. 2008, pp. 442–447.

Suggest you're consistent in nomenclature. Some subscripts one single letters, others a few, others all words.

- [3] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, Sep. 2007.
- [4] 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 2008.
- [5] G. Liu, Y. Yu, J. Hou, W. Xue, X. Liu, Y. Liu, W. Wang, A. Alsaedi, T. Hayat, and Z. Liu, "An ecological risk assessment of heavy metal pollution of the agricultural ecosystem near a lead-acid battery factory," *Ecological Indicators*, vol. 47, pp. 210 – 218, 2014.
- [6] F. Akhtar and M. H. Rehmani, "Energy replenishment using renewable and traditional energy resources for sustainable wireless sensor networks: A review," *Renewable and Sustainable Energy Reviews*, vol. 45, pp. 769 – 784, 2015.
- [7] S. T. Sliper, O. Cetinkaya, A. S. Weddell, B. Al-Hashimi, and G. V. Merrett, "Energy-driven computing," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 378, no. 2164, p. 20190158, 2020.
- [8] B. Ransford, J. Sorber, and K. Fu, "Mementos: System support for long-running computation on RFID-scale devices," in *Proceedings of the Sixteenth International Conference on Architectural Support for Programming Languages and Operating Systems*, ser. ASPLOS XVI, New York, NY, USA: ACM, 2011, pp. 159–170.
- [9] N. A. Bhatti and L. Mottola, "Harvos: Efficient code instrumentation for transiently-powered embedded sensing," in *2017 16th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2017, pp. 209–220.
- [10] A. Y. Majid, C. D. Donne, K. Maeng, A. Colin, K. S. Yildirim, B. Lucia, and P. Pawełczak, "Dynamic task-based intermittent execution for energy-harvesting devices," *ACM Trans. Sen. Netw.*, vol. 16, no. 1, Feb. 2020.
- [11] K. Maeng, A. Colin, and B. Lucia, "Alpaca: Intermittent execution without checkpoints," *Proc. ACM Program. Lang.*, vol. 1, no. OOPSLA, pp. 96:1–96:30, Oct. 2017.
- [12] D. Balsamo, A. S. Weddell, A. Das, A. R. Arreola, D. Brunelli, B. M. Al-Hashimi, G. V. Merrett, and L. Benini, "Hibernus++: A self-calibrating and adaptive system for transiently-powered embedded devices," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 12, pp. 1968–1980, 2016.
- [13] K. Maeng and B. Lucia, "Supporting peripherals in intermittent systems with just-in-time checkpoints," in *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*, ser. PLDI 2019. New York, NY, USA: ACM, 2019, pp. 1101–1116.
- [14] K. Maeng, X. Li, K. Swaminathan, Y. Zheng, S. Li, Y. Liu, Y. Xie, J. J. Sampson, and V. Narayanan, "Nonvolatile processor architectures: Efficient, reliable progress with unstable power," *IEEE Micro*, vol. 36, no. 3, pp. 72–83, May 2016.
- [15] D. Balsamo, A. S. Weddell, G. V. Merrett, B. M. Al-Hashimi, D. Brunelli, and L. Benini, "Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems," *IEEE Embedded Systems Letters*, vol. 7, no. 1, pp. 15–18, March 2015.
- [16] H. Jayakumar, A. Raha, and V. Raghunathan, "QUICKRECALL: A low overhead HW/SW approach for enabling computations across power cycles in transiently powered computers," in *2014 27th International Conference on VLSI Design and 2014 13th International Conference on Embedded Systems*, Jan 2014, pp. 330–335.
- [17] J. Hester, L. Sitanayah, and J. Sorber, "Tragedy of the coulombs: Federating energy storage for tiny, intermittently-powered sensors," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 5–16.
- [18] D. Balsamo, B. J. Fletcher, A. S. Weddell, G. Karatzios, B. M. Al-Hashimi, and G. V. Merrett, "Momentum: Power-neutral performance scaling with intrinsic mppt for energy harvesting computing systems," *ACM Trans. Embed. Comput. Syst.*, vol. 17, no. 6, Jan. 2019.
- [19] K. Maeng and B. Lucia, "Adaptive dynamic checkpointing for safe efficient intermittent computing," in *13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18)*. Carlsbad, CA: USENIX Association, Oct. 2018, pp. 129–144.
- [20] F. Su, Y. Liu, X. Sheng, H. G. Lee, N. Chang, and H. Yang, "A task failure rate aware dual-channel solar power system for nonvolatile sensor nodes," *ACM Trans. Embed. Comput. Syst.*, vol. 18, no. 4, pp. 33:1–33:21, Jul. 2019.
- [21] N. Jackson, J. Adkins, and P. Dutta, "Capacity over capacitance for reliable energy harvesting sensors," in *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*, ser. IPSN '19. New York, NY, USA: ACM, 2019, pp. 193–204.
- [22] Yizi Gu, Y. Liu, Y. Wang, H. Li, and H. Yang, "NVPsim: A simulator for architecture explorations of nonvolatile processors," in *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, Jan 2016, pp. 147–152.
- [23] T. Wu, L. Zhang, H. Yang, and Y. Liu, "An extensible system simulator for intermittently-powered multiple-peripheral iot devices," in *Proceedings of the 6th International Workshop on Energy Harvesting & Energy-Neutral Sensing Systems*, ser. ENSsys '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1–6.
- [24] J. San Miguel, K. Ganeshan, M. Badr, C. Xia, R. Li, H. Hsiao, and N. Enright Jerger, "The EH model: Early design space exploration of intermittent processor architectures," in *2018 51st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Oct 2018, pp. 600–612.
- [25] S. T. Sliper, W. Wang, N. Nikoleris, A. Weddell, and G. Merrett, "Fused: closed-loop performance and energy simulation of embedded systems," in *IEEE International Symposium on Performance Analysis of Systems and Software*, ser. ISPASS 2020, Boston, MA, USA, 2020.
- [26] J. Hester, T. Scott, and J. Sorber, "Ekho: Realistic and repeatable experimentation for tiny energy-harvesting sensors," in *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, ser. SenSys '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 1–15.
- [27] K. Geissdoerfer, M. Chwalisz, and M. Zimmerling, "Shepherd: A portable testbed for the batteryless iot," in *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*, ser. SenSys '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 83–95.
- [28] Nichicon, "UPW aluminum electrolytic capacitors." [Online]. Available: <http://www.nichicon.co.jp/english/products/pdfs/e-upw.pdf>
- [29] R. Faltus, "Low leakage current aspect of designing with tantalum and niobium oxide capacitors." [Online]. Available: https://www.avx.com/docs/techinfo/Low_Leakage_Current_Aspect_Designing_Tantalum_Niobium_Oxide_Capacitors.pdf
- [30] AVX, "AVX TAJ low profile series tantalum capacitors." [Online]. Available: http://datasheets.avx.com/TAJ_LOW_PROFILE.pdf
- [31] S. T. Sliper, D. Balsamo, N. Nikoleris, W. Wang, A. S. Weddell, and G. V. Merrett, "Efficient state retention through paged memory management for reactive transient computing," in *Proceedings of the 56th Annual Design Automation Conference 2019*, ser. DAC '19. New York, NY, USA: ACM, 2019, pp. 26:1–26:6.
- [32] A. Andreas and T. Stoffel, "NREL solar radiation research laboratory (SRRL): Baseline measurement system (bms); golden, colorado (data)," *NREL Report NO.DA-5500-56488*, 1981.
- [33] M. Gorlatova, A. Wallwater, and G. Zussman, "Networking low-power energy harvesting devices: Measurements and algorithms," *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1853–1865, Sep. 2013.
- [34] S. Vergura, "A complete and simplified datasheet-based model of PV cells in variable environmental conditions for circuit simulation," *Energies*, vol. 9, no. 5, 2016.
- [35] Panasonic, "Amorton amorphous silicon solar cells." [Online]. Available: https://www.panasonic-electric-works.com/cps/rde/xocr/pew_eu_en/ca_amorton_solar_cells_en.pdf

Jie Zhan Biography text here.

Alex S. Weddell Biography text here.

Geoff V. Merrett Biography text here.