2026 IEEE 14th International Workshop on Energy Harvesting & Energy-Neutral Sensing Systems (ENSsys) | 979-8-3195-2133-0/26/$31.00 ©2026 IEEE | DOI: 10.1109/ENSsys71150.2026.00010

Exploring Biodegradable Non-volatile Memories for Intermittent Computing

Matteo Visotto*, Andrea Maioli* and Luca Mottola*†‡
*NESLab, Politecnico di Milano, Milano, Italy
† Connected Intelligence Unit, RI.SE Computer Science, Sweden
‡ Electrical Engineering Department, Uppsala University, Sweden
Email: {matteo.visotto, andrea1.maioli, luca.mottola}@polimi.it

***Abstract*—We explore the use of *non-volatile biodegradable memories* when used for state retention in intermittent computing. These memories are built with *transient electronics*. They feature the ability to physically dissolve in the environment, addressing the arising concerns due to disposal of conventional electronics. Because of their physical construction, however, their *write endurance* and *retention times* are sharply different compared to conventional non-volatile memories. We quantitatively determine the performance of four different non-volatile biodegradable memories when employed with staple benchmarks and system support. We find, for example, that 61% of the tests fail because of write endurance issues, which represent the major hampering factor and yet are caused by specific read/write patterns induced by the underlying system support. With the energy intake expected from available energy sources, nonetheless, we find that in 33% of the experiments, recharging times exceed retention times, causing data losses that produce logical errors in 19% of the executions. These insights point to a necessary re-thinking of the software fabric, which we articulate in the paper by providing an outlook and future directions for work in the field.**

***Index Terms*—Energy Harvesting, Intermittent Computing, Non-volatile Biodegradable Memory**

I. Introduction

The rapid development of technology produces massive amounts of electronics that are developed and eventually discarded . The resulting electronic waste causes environmental concerns [1]. Electronic waste often contains hazardous materials like lead, mercury, and cadmium, which can harm the environment and human health if not properly disposed of.

Transient electronics [2]–[4] features the ability to dissolve or disintegrate in actively or passively-controlled fashions and leave harmless by-products in the environment. As we articulate in Sec. II, this technology uses transient materials including metals, polymers, and biological elements [2]–[4]. Unlike conventional electronics that are designed to remain functional over extended periods, transient electronics are designed to operate over a typically short duration of time and disintegrate fully or partially when no longer needed [5], [6]. A variety of device components, for example, sensors, batteries, and memories [6]–[9], are being prototyped with transient electronics [2]–[4].

Non-volatile biodegradable memories. Transient electronics find applications in domains such as environmental monitoring, precision agriculture, and healthcare [5], [6], [10]. These applications are likely to impose tight energy constraints, for example, due to the risks in employing conventional battery technology or due to size limitations, requiring energy harvesting to power device operation [11]–[13]. This yields an intermittent computing pattern: periods of active computation are interleaved with periods of recharging energy buffers following unpredictable, yet unavoidable energy failures.

Our interest in this paper is precisely to explore the use of a specific kind of transient electronic device, that is, *non-volatile biodegradable memories*, illustrated in Sec. III, in an intermittent computing setting. These memories are built with materials ranging from melanin to sweet potatoes. Their non-volatile nature allows systems to use them as a storage facility to let systems cross energy failures during intermittent executions. On the other hand, *write endurance* and *retention times* of non-volatile biodegradable memories are sharply different compared to mainstream technology, and often on par or even shorter than the expected system lifetime. These features question their application to support intermittent computations.

Quantitative study. We scrutinize the performance of four different non-volatile biodegradable memories when employed with staple system support for intermittent computing. We use eleven benchmarks commonly employed in intermittent computing literature [14]–[17], energy- and time-accurate simulations to ensure reproducibility, and five real-world energy traces. This setup is intentional: we check the application of non-volatile biodegradable memories in the absence of a custom software fabric, that is, using existing designs thought for conventional technology.

Our findings, discussed in Sec. IV, provide two key insights:

1) *write endurance is by far the major hampering factor* regardless of the specific biodegradable memory technology, yet it is so due to specific read/write patterns inherent in the system support we employ;
2) using available energy harvesting technology and storage, *retention times* of some of the non-volatile biodegradable memories we test are *comparable with the times for recharging energy buffers*, which causes data faults that *may* cause errors in executions.

We are arguably the first to quantify the system behavior in an energy-constrained setting when using non-volatile biodegradable memories. Our results indicate that the promise of transient electronics is yet to materialize. The reason for this

979-8-3195-2133-0/26/$31.00 ©2026 IEEE
DOI 10.1109/ENSsys71150.2026.00010

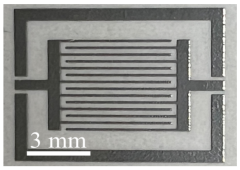

Fig. 1: Cellulose-based moisture sensor [6].

is evidently in how research in this domain remained confined within chemistry, material sciences, and electronics, with little effort invested into the system and software matters. Our quantitative analysis rather indicates that many system-level mechanisms are based on assumptions that become invalid when using transient electronics. We elaborate on these aspects in Sec. V, providing directions for future work. We end the paper in Sec. VI with brief concluding remarks.

II. Background

Transient electronics is a means to address the mounting global crisis of electronic waste [1] and to provide solutions for applications where device recovery is impractical, costly, or biologically invasive [2]. It represent a paradigm shift in device engineering, where the traditional goal of long-term durability is replaced by controlled transience. Unlike conventional hardware, transient electronics is designed to undergo physically or chemically triggered degradation, effectively "disappearing" into their environment after a predefined lifetime [3].

Transient electronics relies on the integration of biocompatible and dissolvable materials; for example, inorganic semiconductors like ultrathin silicon nanomembranes that undergo hydrolysis in aqueous solutions [5]. By tuning the thickness and chemical composition of materials, it is possible to precisely program degradation rates from minutes to years, triggered by external stimuli like moisture, temperature, or pH levels [2].

The scope of transient electronics range from basic passive elements to complex, multimodal systems with diverse real-world applications. The technology offers a sustainable alternative to silicon-based technology, providing a "set-and-forget" solution for applications such as large-scale environmental monitoring and personalized medicine [10].

In healthcare, biodegradable re-absorbable devices are being developed for short-term monitoring of post-operative recovery or localized therapy. For instance, transient light-emitting diodes (LEDs) that naturally dissolve under physiological conditions exist for localized stimulation [18]. They use biodegradable semiconductors that are absorbed by the body to eliminate the need for surgical extraction [10], [18].

In precision agriculture, for example a capacitive cellulose-based moisture sensor [6], shown in Fig. 1, can monitor soil and humidity conditions. It is fabricated by screen-printing interdigitated electrical structures onto a fully biodegradable substrate made of cardstock infiltrated with cellulose nanofibrils. While the sensor has an expected lifetime of a few weeks in the field , the materials themselves degrade rapidly: pure nanocellulose films decompose in 22 to 30 days, and the pulp-based cardstock reaches near 100% degradation in approximately 82 days when placed in high-microbial soil [6].

Prototypes of batteries and memory units based on transient electronics also exist [7]–[9], [19]–[24]. These systems offer advantages for secure hardware, where devices can be engineered to self-destruct upon command to protect data [3].

III. Non-volatile Biodegradable Memory

We explore the use of non-volatile biodegradable memories to support intermittent computations. Notably, most existing designs of non-volatile biodegradable memories [19]–[26] are based on the functioning principles of Resistive Random-access Memories (ReRAM).

ReRAM principles. A ReRAM cell stores data in two stable states: a High-resistance State (HRS) and a Low-resistance State (LRS). Write operations electrically change the resistance of a layer between two electrodes. One such operation forms or ruptures a conductive filament. Filament formation sets the cell to LRS, while rupture resets it to HRS. Data storage thus depends on resistance, not charge. State transitions occur through two operations: SET that switches from HRS to LRS, and RESET that switches from LRS to HRS.

Two metrics characterize the operation of ReRAM chips. Switching impacts *write endurance*, that is, the number of SET and RESET cycles a cell can sustain, because each cycle modifies the filament and the surrounding material. These changes accumulate and eventually impair switching, leading to failure. Data *retention*, on the other hand, measures the ability to preserve the resistance state over time and without power. In ReRAM, filament instability and defect diffusion may cause resistance drift over time, which may result in a *retention fault* that causes a loss of data.

Transient implementation. Biodegradable Resistive switching RAM (BioReRAM) is based on the same functioning principles as regular ReRAM, but is built using transient electronics [8], [9]. Its operation is also characterized with the same key metrics, yet the values are sharply different.

Regular ReRAM sustains up to 10^{12} SET/RESET cycles and retains data for up to ten years [27]–[29]. The materials used for BioReRAM enable biocompatibility, yet they introduce weaker structural stability [8], [9]. The switching process depends on mobile ions and charge trapping in a soft and disordered matrix. Over repeated operations, the internal paths that carry current deform or relax, which reduces the number of reliable write cycles. The stored state drifts because polarization and trapped charges tend to return to equilibrium [19], [20], [23], [24]. Write endurance and retention times are thus far below those of the inorganic counterparts.

Prototypes. Literature exists that reports the use of rose petals as the active layer, enabling switching at low voltage [19]. The material is biodegradable and readily available, making it suitable for disposable applications. However, moisture and natural variability strongly affect the conductive paths. Repeated cycling can permanently alter the tissue, limiting reliable write operations to about 10 and guaranteeing data retention for no more than 7200 seconds.

Using natural aloe vera as the active layer in a memory is highly challenging [22]. Drying temperature and solvent concentration strongly affect switching performance. This sensitivity renders the microscopic structure fragile. Minor changes can disrupt ion transport and filament formation, limiting reliable writes to 21 cycles, even though data retention remains is guaranteed for over 10^4 seconds.

Different results are obtained for instance with an egg albumen film as switching medium [23]. The liquid protein is deposited between two electrodes and then dried, forming a thin layer. When an electric field is applied, charges are trapped and later released inside the organic matrix. This mechanism creates stable high and low resistance states. Memories based on albumen show repeatable programming and data retention up to $10^5 s$.

Want et al. [24] present a BioReRAM prototype that uses orange peel, which dissolves in water. Electrical measurements show competitive HRS/LRS ratios. However, the properties of the material constrain both endurance and retention. This device supports up to 100 write operations and retains data for about 500 seconds without power.

Prototypes exist that use sweet potatoes [25], with the switching mechanism arising from capacitive effects. The measured HRS/LRS loop, however, departs from that of an ideal resistive device. Still, the materials are biocompatible and the cell can retain data for up to 10^4 s without power.

Other approaches rely on melanin produced by bacteria [20]. A thin film is placed between metal electrodes and shows bipolar switching, with SET and RESET operations controlled by the voltage polarity. The material is renewable and biocompatible. The device exhibits reproducible transitions, write endurance up to 20000 cycles, and up to $10^3 s$ of data retention.

Some BioReRAM prototypes use natural tissues or biomaterials, such as mushrooms or maple leaves [21], [26], as the active switching layer. In these devices, hybrid organic–inorganic layers assist charge transport or filament formation, enabling low-voltage switching and environmentally friendly fabrication. However, these devices incorporate inorganic components or partially processed biomaterials, hence they are not fully biodegradable. In addition, the organic matrix is mechanically soft, and the interfaces are delicate, which leads to uncontrolled filament growth, relaxation of trapped charges, and partial loss of the stored data. As a result, the write endurance is significantly lower compared to other BioReRAM prototypes and, in most cases, retention is not documented.

IV. Quantitative Study

Existing prototypes of BioReRAM are few and mostly dedicated to experiments around their ability to dissolve in the environment [8], [9]. Very few attempts exist at exploring their use for data storage. Further, monitoring intermittent executions to collect performance metrics may interfere with the device's intermittent executions and requires specialized hardware and software components [30]–[32]. Reproducing energy harvesting sources using real hardware is challenging, as their behavior is non-deterministic [31], [33], [34].

Similar to existing literature [35], [36], we opt for a simulated environment based on a detailed model of the BioReRAM behavior and real-world energy traces. In the following, Sec. IV-A describes the benchmarks we use, the BioReRAM model, as well as the tools and metrics we employ. In Sec. IV-B, we discuss the results of our experiments.

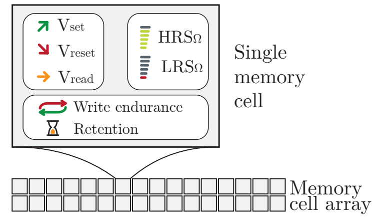

Fig. 2: BioReRAM model.

A. Experimental Setup

The setup we use is a combination of benchmarks, metrics, and tools used in existing work with a novel memory model we develop to represent the BioReRAM behavior.

Benchmarks. We select workloads from MiBench2 [37], a benchmark suite widely used to evaluate intermittent systems [35], [36], [38]. We consider a total of eleven different implementations. The benchmarks run on top of MementOS, applied as-is across all experimental configurations with a loop-latch checkpoint policy [39]. We select MementOS because: *i)* it represents a staple solution for intermittent computing, allowing our conclusions to generalize to the many follow-up systems, *ii)* it represents one of the simplest intermittent framework, allowing a better control and analysis of memory operations, and *iii)* it allows us to pre-compute the worst-case number of non-volatile write operations, useful to select suitable BioReRAM prototypes, as we explain next.

To guarantee consistency, the checkpoint mechanism relies on a two-phase commit policy, often employed in checkpoint-based systems [14], [17], [39], [40]. Two replicas of the state are stored in two separate memory areas. One is the *read copy*, which stores the last consistent checkpoint. The other is the write copy, which receives updates while a new checkpoint is produced. A dedicated bit, the *version bit*, marks what replica is the most recent one. The bit flips only once the system writes a new complete checkpoint, ensuring its consistency.

We determine the checkpoint threshold in MementOS for each application as the minimum energy required to save an entire checkpoint on a given type of BioReRAM, among the ones we test. This energy includes triggering the $SET/RESET$ transition for each bit, which depends on the memory-specific resistive state and voltage.

Memory model. Of the BioReRAM prototypes in Sec. II, we consider those whose features fit a small embedded workload. For example, we require memories to sustain a sufficient number of flips of the versioning bit to complete at least one complete benchmark iteration, in the worst case.

For each benchmark, we compute the maximum number of checkpoints the system can generate. Based on these, we primarily consider the Melanin [20] prototype, which provides the largest write endurance. To study the behavior under tighter constraints, we also include the Albumen [23],

Bio material	Max writes	Retention [s]	HRS [Ω]	LRS [Ω]	$\mathbf{V_{read}}$ [V]	$\mathbf{V_{set}}$ [V]	$\mathbf{V_{reset}}$ [V]
Melanin	$2 \cdot 10^4$	10^3	$3.64 \cdot 10^5$	34	0.2	0.6	-0.6
Albumen	500	10^5	$3.18 \cdot 10^4$	54	-0.2	-0.3	3.6
Sweet Potato	200	10^4	$2.10 \cdot 10^8$	$1.40 \cdot 10^7$	2.9	3.0	-3.0
Orange	100	500	$1.05 \cdot 10^4$	10	1.3	2.0	-1.6

TABLE I: BioReRAM model parameters.

Energy source	Mean power	Duty factor	Gini index
Jogging	35.110 μW	0.011	0.544
Washer	42.383 μW	0.249	0.727
Car	7.099 μW	0.127	0.646
RF	90.189 μW	0.676	0.673
Office	168.982 μW	0.994	0.164

TABLE II: Characterization of energy traces.

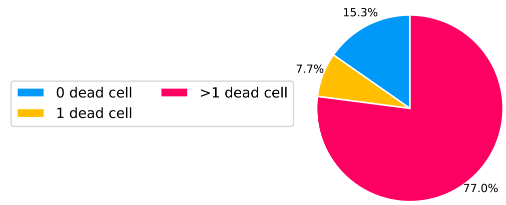

Fig. 3: Number of dead cells at the end of our experiments.

SWEET POTATO [25], and ORANGE [24] prototypes. Despite its limited write endurance, the ORANGE prototype enables the analysis of how retention times affect program execution. Tab. I summarizes the features of BioReRAM prototypes we consider. These data define the basis of our memory abstraction, illustrated in Fig. 2.

We exclude other candidates [19], [21] as they would not allow the system to achieve even a single complete benchmark iteration. Literature related to other prototypes also lacks key parameters that would prevent the design of a realistic model.

Energy traces. We consider five energy traces. We use the RF trace from MementOS [39] and three kinetic energy traces from Bonito [34], specifically JOGGING, WASHER, and CAR, along with the solar indoor trace called OFFICE. These traces reproduce a variety of energy dynamics, both in time and energy content, providing a basis to generalize results.

As summarized in Tab. II, we characterize each energy trace by computing the Duty Factor and the Gini Index. The Duty Factor DF captures the temporal distribution of power above a threshold θ. It quantifies whether the harvested energy is spread over time or concentrated in a small number of high-power events. A value of $DF \approx 1$ indicates that power is evenly distributed along the trace, whereas $DF \ll 1$ denotes energy concentrated in few short-lived spikes. The Gini Index G [41] quantifies the temporal distribution of power, capturing whether high-power events are uniformly spread over the energy trace or clustered in limited intervals. Values of $G \to 0$ denote an even distribution of power over time, whereas $G \to 1$ indicate that most of the energy is concentrated in a small fraction of the trace.

The OFFICE trace is both rich of energy and rather stable. It produces executions with few energy failures, leading to a mostly continuous behavior. The RF trace, although producing intermittent behaviors, is shorter than one minute. To prolong the experiments, we re-play the trace identically over longer periods, which artificially introduces a periodic pattern.

Metrics and events. For each memory cell, we keep track of the number of flips. Once this number exceeds the maximum write endurance, we consider the cell as *dead* and set its value to zero in the simulation. Any additional write operation on the same cell, while consuming energy in the simulation, causes no changes in the data. We also keep track of situations where the system execution unfolds in ways to exceed a cell's retention time. This happens when recharge times exceed the retention times, thus a cell remains without power too long and is unable to retain the logical value. We call these occurrences as *retention faults*, and monitor whether these faults cause errors in the application too.

Consider that energy traces have different durations. Simulations with different energy traces would thus not be comparable. We introduce a reference simulation time t_{sim} to place the different settings we explore on the same ground. We repeat or truncate each energy trace to match t_{sim} and log the system state at time $t = t_{sim}$. This enables us to separate executions that complete without errors within t_{sim}, marked as *successful*, those that fail because of issues with write endurance, and those that fail due to retention faults. We run a total of 220 experiments, with $t_{sim} = 3600s$.

Tools. We use ScEpTIC [42], an open-source emulator for intermittent computing extensively used in existing literature [35], [36], [38]. It emulates executions at cycle level, along with energy sources and consumption, based on an LLVM intermediate representation [43] of the program's source code. The LLVM representation is generated by compiling the code with clang v8.0.1 [44] and default options.

In our experiments, ScEpTIC emulates a TI MSP430-FR5969 MCU [45], a popular MCU for intermittent computing [14], [17], [38], [39], [46], [47], along with a $100\mu F$ capacitor as energy buffer. We model a MAX20361 [48] voltage charge booster to regulate the energy source voltage to $3.65V$, ensuring capacitor's recharge and MCU activation even in presence of low input voltage.

B. Experimental Results

We discuss next the key observations we draw and their grounding in the experimental data.

Observation 1: *Write endurance is the primary cause of memory failures in our experiments.*

Fig. 3 shows the fraction of experiments concluding with zero dead cells compared to those concluding with one or more dead cells. Using real-world energy traces and within one hour of simulation time, most experiments conclude with at least

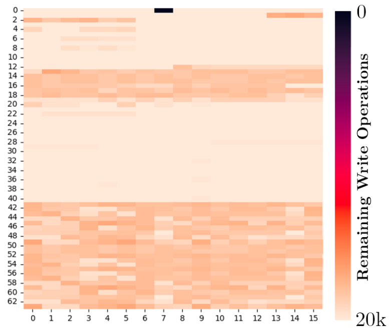

Fig. 4: CRC Fast memory dump.

one dead cell. The implication of this result is two-pronged: it provides an indication that BioReRAM technology should improve write endurance to sustain longer workloads. It also points to the need for dedicated software mechanisms that gracefully handle non-volatile memories with limited write endurance, as is the case of BioReRAM. We return to the latter observation in the remainder of this section.

Observation 2: *Using existing checkpointing techniques, the number of energy failures the system can tolerate is upper bound by the memory's write endurance.*

This observation is rooted in the specific mechanisms used for handling checkpoints in MementOS, common to most system support for intermittent computing [39]. Fig. 4 shows a partial memory snapshot at the end of a test running the CRC Fast benchmark using the MELANIN BioReRAM with the WASHER energy trace. Each block corresponds to a single bit. In this example, only one dead cell is visible: the *versioning bit*. Although checkpoints are written to separate memory regions, the versioning bit flips at every checkpoint. The write endurance of the cell storing this bit is fully consumed by switching between the two checkpoint versions.

Since the versioning bit cannot be updated, the two-phase commit can no longer switch between the two checkpoint versions. Every subsequent state restore then repeatedly loads the same checkpoint. The final behavior depends on which checkpoint the versioning bit is stuck to. If the restored state and available energy allow the program to reach completion and start a new iteration, an output is still produced. Because the same snapshot is always reloaded, however, in the absence of new run-time inputs, this output also never changes. Differently, if an energy failure occurs after the restore operation but before completion of a benchmark iteration, no output is ever produced, preventing detection of the specific system state unless the internal state can be probed somehow.

In addition, when the system can no longer alternate between the two checkpoint versions, one copy necessarily operates as read copy used for recovery, while the other copy always operates as write copy. Every new checkpoint updates the write copy, thus memory cells storing data for the write copy quickly deteriorate because of the limited write endurance. Nonetheless, since restore operations are stuck reading the read copy, the additional dead cells of the write copy cause no additional error, as they are never read.

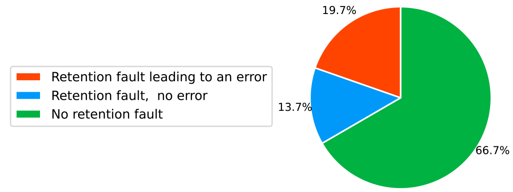

Fig. 5: Retention faults and errors.

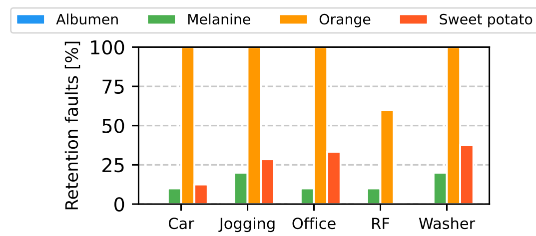

Fig. 6: Retention faults.

This is not the only situation where we observe memory failures that do not necessarily lead to errors in executions.

Observation 3: *In our experiments, about 14% of the data retention faults do not translate into program errors.*

Fig. 5 breaks down how frequently retention failures lead to program errors, that is, erroneous program states that are not reachable in an execution with no memory issues, compared with how often they occur without leading to errors and how often no retention faults occur all together.

We observe no retention faults in the largest fraction of our experiments, indicating that the limited retention of BioReRAM is a secondary issue compared with the limited write endurance. Moreover, we observe a non-negligible fraction of the experiments where retention faults do occur, yet without causing errors in executions. This indicates that these faults are not necessarily critical as they occur on memory cells that do not bear an effect on executions.

A deeper analysis of execution logs reveals where these circumstances occur. The checkpoint size, in particular, is not constant, as it may change, for example, because of variations in heap or stack usage. Because checkpoint data always starts from the same address, when a smaller checkpoint overwrites the previous one, the portion of memory cells at the end of checkpoint area still stores (now obsolete) data of the larger previous checkpoint. These cells are not overwritten and eventually experience retention faults. No execution errors occur because these cells are not part of any valid checkpoint.

Observation 4: *Retention faults depends on both the energy source and the BioReRAM technology.*

Fig. 6 shows the distribution of retention faults by BioReRAM technology and energy source. The ORANGE BioReRAM features a retention time of about $500s$, as reported in Tab. I, that is shorter than recharge times in most cases. As a result, the system would often attempt to restore degraded data, which likely leads to execution errors.

Consider instead the CAR energy trace. As reported in Tab. II, this trace features low mean power and few energy spikes $(DF \rightarrow 0)$ concentrated in short time intervals

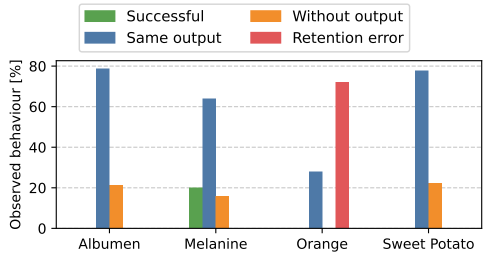

Fig. 7: Summary of experiment outcomes.

($G \rightarrow 1$). With these features, fewer experiments exhibit retention failures across all BioReRAM technologies. Limited input energy increases energy failures and thus the frequency of checkpoint operations. Frequent write operations refresh the memory at a higher rate, mitigating the risk of retention faults.

In contrast, the OFFICE trace features higher mean power and energy spikes ($DF \rightarrow 1$) that are evenly distributed across the trace ($G \rightarrow 0$). As mentioned, this reduces the frequency of energy failures, approximating a continuous rather than intermittent execution. This also entails fewer energy failures and thus infrequent checkpoint operations. More rare write operations leave the memory with no power for longer intervals, increasing the likelihood of retention faults.

Observation 5: *Either because of limited write endurance or short retention times, the specific BioReRAM technology determines the applicability to a specific setting.*

Fig. 7 reports the fraction of experiments that complete successfully versus those ending in some kind of errors, either because of limited write endurance of because of retention faults. We group the results by BioReRAM type.

The link between memory technology and outcome of the experiment is striking. Memories with higher write endurance, such as MELANIN, complete more experiments successfully, confirming that this memory feature is a primary factor in the applicability of BioReRAM technology to intermittent computing. More limited write endurance, however, cause the error rate to increases in our experiments. For instance, BioReRAM technology based on ALBUMEN and SWEET POTATO cannot produce any successful execution.

Retention also impacts the outcome of experiments, although to a lesser extent. Short retention times lead to errors from retention faults, as observed in the case of the ORANGE BioReRAM. This is not the case for the other BioReRAM we test: despite a relatively short retention time, the energy traces we use cause a sufficient refresh of checkpoint data.

V. OUTLOOK AND RESEARCH DIRECTIONS

The results in Sec. IV-B demonstrate how the design criteria employed for conventional intermittent computing are to be re-thought when using BioReRAM technology. For example, checkpointing becomes a destructive force using BioReRAM technology: write operations may rapidly place the system in an erroneous state due to the limited write endurance.

On the other hand, a conservative approach that reduces the number of write operations over time leaves the system vulnerable to retention faults, which possibly also cause errors in the execution, as we show in Sec. IV-B. In essence, any simple technique that would address one limitation, would most likely worsen the behavior with respect to the other.

We argue that BioReRAM cannot be treated as a simple drop-in replacement for silicon-based non-volatile memory. We should rather design software that respects the physical limits of the hardware, while simultaneously engineering the hardware to help address what software cannot handle by itself. Based on these considerations, we provide next a few concrete example of future research in this domain:

State persistence with limited write endurance. We need smart techniques to determine when writing on a memory cell with limited write endurance is absolutely necessary, as opposed to when such an operation may be skipped. One may, for example, conceive a metric to represent the extent of forward progress enabled by a non-volatile memory write used to cross an energy failure. Based on this, the system may judge whether a write operation is sufficiently valuable.

Low overhead wear-leveling mechanisms. Since many intermittent computing platforms lack hardware memory controllers, the burden of prolonging the correct functioning of the memory falls on the software stack. At compile-time, a custom toolchain may find memory access patterns to identify non-volatile variables with frequent write operations, and accordingly injecting instructions to rotate their memory allocation at run-rime, preventing the formation of "hot spots" that would otherwise lead to premature failure of specific memory cells. Concepts of differential checkpointing [49] and techniques exploiting the notion of silent stores [50] may also be employed to lessen the memory pressure.

Opportunistic management of data retention limits. Many BioReRAM technologies refresh the data also during read operations. We may then conceive techniques that opportunistically schedule low-cost read operations in situations of energy surplus. This would ensure the integrity of data using energy that may otherwise go wasted and without the cost of a full SET/RESET cycle, which inevitably also impacts the write endurance of the memory cell.

VI. CONCLUSION

We explored the use of BioReRAM technology to support intermittent computing. We quantified the performance of four BioReRAM prototypes with common intermittent computing benchmarks and system support, using real-world energy traces. A fundamental mismatch exists between the current software fabric and the limitations of BioReRAM technology. Write endurance is the major hampering factor regardless of theBioReRAM prototype, mainly due to the operation of the two-phase checkpoint commit policy in MementOS as well as in many other available systems. Limited retention times also cause data faults as energy provisioning patterns are unable to refresh the data sufficiently often. We concluded the paper by providing directions for future work in the area.

Acknowledgments. This work was partially supported by the Swedish Foundation for Strategic Research (SSF).

References

[1] S. Herat, "Sustainable management of electronic waste (e-waste)," *Clean–Soil, Air, Water*, 2007.

[2] K. K. Fu *et al.*, "Transient electronics: materials and devices," *Chemistry of Materials*, 2016.

[3] W. B. Han *et al.*, "Advanced materials and systems for biodegradable, transient electronics," *Advanced Materials*, 2020.

[4] M. S. Dennison *et al.*, "Biodegradable electronic materials for promoting sustainability in next-generation electronics-a comprehensive review," *Discover Materials*, 2025.

[5] C. Wei *et al.*, "Transient flexible multimodal sensors based on degradable fibrous nanocomposite mats for monitoring strain, temperature, and humidity," *ACS Applied Polymer Materials*, 2024.

[6] A. M. Zaccarin *et al.*, "Fabrication and characterization of soil moisture sensors on a biodegradable, cellulose-based substrate," *IEEE Sensors Journal*, 2024.

[7] V. Galli *et al.*, "A coplanar edible rechargeable battery with enhanced capacity," *Advanced Materials Technologies*, 2024.

[8] F. Y. Rahman *et al.*, "An account of natural material-based nonvolatile memory device," *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, 2023.

[9] S. Mao *et al.*, "Applications of biomemristors in next generation wearable electronics," *Nanoscale Horizons*, 2022.

[10] Ž. a. Janićijević, "Design and development of transient sensing devices for healthcare applications," *Advanced Science*, 2024.

[11] N. A. Bhatti *et al.*, "Energy Harvesting and Wireless Transfer in Sensor Network Applications: Concepts and Experiences," *TOSN*, 2016.

[12] S. Ahmed *et al.*, "The Internet of Batteryless Things," *Commun. ACM*, 2024.

[13] M. Afanasov *et al.*, "Battery-less zero-maintenance embedded sensing at the mithræum of circus maximus," in *SenSys*, 2020.

[14] K. Maeng *et al.*, "Alpaca: intermittent execution without checkpoints," *Proc. ACM Program. Lang.*, 2017.

[15] N. A. Bhatti and L. Mottola, "HarvOS: Efficient Code Instrumentation for Transiently-powered Embedded Sensing," in *IPSN*, 2017.

[16] A. Colin and B. Lucia, "Chain: tasks and channels for reliable intermittent programs," *SIGPLAN Not.*, 2016.

[17] D. Balsamo *et al.*, "Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems," *ESL*, 2015.

[18] D. Lu *et al.*, "Transient light-emitting diodes constructed from semiconductors and transparent conductors that biodegrade under physiological conditions," *Advanced Materials*, 2019.

[19] F. Y.R. *et al.*, "Investigation of non volatile resistive switching behaviour using rose petal," *Materials Today: Proceedings*, 2022.

[20] S. T. Gurme *et al.*, "An organic bipolar resistive switching memory device based on natural melanin synthesized from aeromonas sp. sns," *physica status solidi (a)*, 2018.

[21] L. Zheng *et al.*, "The redox of hydroxyl-assisted metallic filament induced resistive switching memory based on a biomaterial-constructed sustainable and environment-friendly device," *Materials today chemistry*, 2018.

[22] Z. X. Lim and K. Y. Cheong, "Effects of drying temperature and ethanol concentration on bipolar switching characteristics of natural aloe vera-based memory devices," *Physical Chemistry Chemical Physics*, 2015.

[23] Y. Chen *et al.*, "Nonvolatile bio-memristor fabricated with egg albumen film," *Scientific reports*, 2015.

[24] X. Wang *et al.*, "From natural biomaterials to environment-friendly and sustainable nonvolatile memory device," *Chemical Physics*, 2018.

[25] S. Zhu *et al.*, "Non-zero-crossing current-voltage hysteresis behavior induced by capacitive effects in bio-memristor," *Journal of colloid and interface science*, 2020.

[26] S. Ranjan *et al.*, "Passive filters for nonvolatile storage based on capacitive-coupled memristive effects in nanolayered organic–inorganic heterojunction devices," *ACS Applied Nano Materials*, 2020.

[27] Y. Chen, "Reram: History, status, and future," *IEEE Transactions on Electron Devices*.

[28] C. Wang *et al.*, "Conduction mechanisms, dynamics and stability in rerams," *Microelectronic Engineering*, 2018.

[29] Z. Swaidan *et al.*, "Rram endurance and retention: Challenges, opportunities and implications on reliable design," in *ICECS*, 2019.

[30] A. Colin *et al.*, "An energy-interference-free hardware-software debugger for intermittent energy-harvesting systems," *SIGOPS*, 2016.

[31] M. Furlong *et al.*, "Realistic simulation for tiny batteryless sensors," in *ENSsys*, 2016.

[32] F. Österlind *et al.*, "Strawman: Resolving Collisions in Bursty Low-Power Wireless Networks," in *Intl. Conf. on Information Processing in Sensor Networks (IPSN)*, 2012.

[33] J. Hester *et al.*, "Ekho: Realistic and repeatable experimentation for tiny energy-harvesting sensors," in *SenSys*, 2014.

[34] K. Geissdoerfer and M. Zimmerling, "Learning to communicate effectively between battery-free devices," in *NSDI 22*, 2022.

[35] V. Rovelli *et al.*, "Extremis: Static frequency switching for battery-less devices," in *ENSsys*, 2024.

[36] A. Maioli *et al.*, "Dynamic voltage and frequency scaling for intermittent computing," *TOSN*, 2025.

[37] "impedimentToProgress/MiBench2: MiBench ported for IoT devices — github.com," https://github.com/impedimentToProgress/MiBench2, 2025, [Accessed 05-04-2025].

[38] A. Maioli and L. Mottola, "Alfred: Virtual memory for intermittent computing," in *SenSys*, 2021.

[39] B. Ransford *et al.*, "Mementos: System support for long-running computation on rfid-scale devices."

[40] D. Balsamo *et al.*, "Hibernus++: A self-calibrating and adaptive system for transiently-powered embedded devices," *TCAD*, 2016.

[41] C. Gini, *Variabilità e mutabilità: contributo allo studio delle distribuzioni e delle relazioni statistiche.[Fasc. I.]*, 1912.

[42] A. Maioli *et al.*, "Discovering the hidden anomalies of intermittent computing," in *EWSN*, 2021.

[43] "The LLVM compiler infrastructure," https://llvm.org/, 2003 (last access: Jan 5th, 2025).

[44] CLang, "Clang 8.0.1 documentation," 2025. [Online]. Available: https://releases.llvm.org/8.0.1/tools/clang/docs/ReleaseNotes.html

[45] "MSP430FR5969," https://www.ti.com/product/MSP430FR5969, 2025.

[46] B. Lucia and B. Ransford, "A Simpler, Safer Programming and Execution Model for Intermittent Systems," in *PLDI*, 2015.

[47] A. Maioli and L. Mottola, "Intermittence anomalies not considered harmful," in *ENSsys*, 2020.

[48] A. Devices, "Max20361 voltage charge booster," https://www.analog.com/media/en/technical-documentation/data-sheets/max20361.pdf, 2025, [Accessed 05-04-2025].

[49] S. Ahmed *et al.*, "Efficient intermittent computing with differential checkpointing," in *LCTES*, 2019.

[50] G. Bell *et al.*, "Characterization of silent stores," in *PACT*, 2000.