Momentum: Power Neutral Performance Scaling with Intrinsic MPPT for Energy Harvesting Computing Systems

DOMENICO BALSAMO, BENJAMIN J. FLETCHER, ALEX S. WEDDELL,
GIORGOS KARATZIOLAS, BASHIR M. AL-HASHIMI, and GEOFF V. MERRETT,
University of Southampton, UK

Recent research has looked to supplement or even replace the batteries in embedded computing systems with energy harvesting, where energy is derived from the device's environment. However, such supplies are generally unpredictable and highly variable, and hence systems typically incorporate large external energy buffers (e.g. supercapacitors) to sustain computation; however, these pose environmental issues and increase system size and cost. This paper proposes *Momentum*, a general power-neutral methodology, with intrinsic system-wide maximum power point tracking, that can be applied to a wide range of different computing systems, where the system dynamically scales its performance (and hence power consumption) to optimize computational progress depending on the power availability. *Momentum* enables the system to operate around an efficient operating voltage, maximizing forward application execution, without adding any external tracking or control units. This methodology combines at run-time 1) a hierarchical control strategy which utilizes available power management controls (such as dynamic voltage and frequency scaling, and core hot-plugging) to achieve efficient power-neutral operation, 2) a software-based maximum power point tracking scheme (unlike existing approaches, this does not require any additional hardware), which adapts the system power consumption so that it can work at the optimal operating voltage, considering the efficiency of the entire system rather than just the energy harvester, and 3) experimental validation on two different scales of computing system: a low power microcontroller (operating from the already-present 4.7 μF decoupling capacitance) and a multi-processor system-on-chip (operating from 15.4 mF added capacitance). Experimental results from both a controlled supply and energy harvesting source show that *Momentum* operates correctly on both platforms, and exhibits improvements in forward application execution of up to 11% when compared to existing power-neutral approaches, and 46% compared to existing static approaches.

CCS Concepts: • **Computer systems organization** → *Embedded systems*;

Additional Key Words and Phrases: Energy Harvesting, Performance Adaptation, Power Neutrality, Maximum Power Point Tracking, Transient Computing, Embedded Computing Systems

ACM Reference Format:
Domenico Balsamo, Benjamin J. Fletcher, Alex S. Weddell, Giorgos Karatziolas, Bashir M. Al-Hashimi, and Geoff V. Merrett. 2018. Momentum: Power Neutral Performance Scaling with Intrinsic MPPT for Energy Harvesting

This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) Grants EP/P010164/1, EP/L000563/1 and EP/K034448/1 (the PRiME Programme www.prime-project.org).
Author's addresses: D.Balsamo, B. J. Fletcher, A. S. Weddell, G. Karatziolas, B. M. Al-Hashimi and G. V. Merrett are with the Pervasive Systems Centre, Electronics and Computer Science, University of Southampton, UK..
Experimental data used in this paper can be found at DOI:10.5258/SOTON/D0679 (http://doi.org/10.5258/SOTON/D0679).
Authors' address: Domenico Balsamo; Benjamin J. Fletcher; Alex S. Weddell;
Giorgos Karatziolas; Bashir M. Al-Hashimi; Geoff V. Merrett,
University of Southampton, University Road, Southampton, SO17 1BJ, UK.

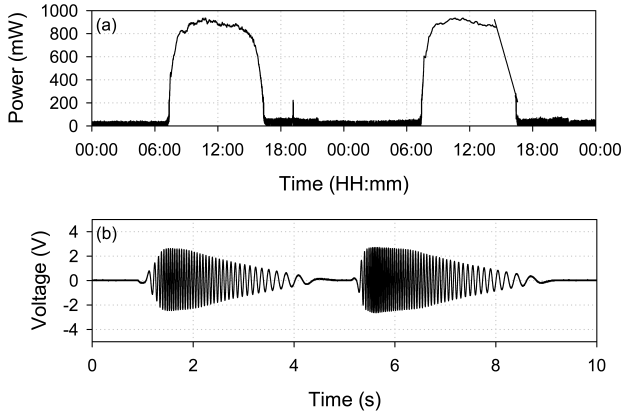

Fig. 1. Experimentally obtained power and voltage outputs from (a) a photovoltaic cell and (b) a micro wind turbine.

Computing Systems. *ACM Trans. Embedd. Comput. Syst.* 1, 1, Article 1 (January 2018), 25 pages. https://doi.org/10.1145/3281300

1 INTRODUCTION

Emerging application domains such as IoT are driving the need to have computing systems embedded in their environment. These include ultra low-power embedded devices (e.g. IoT sensors) through to low-power mobile platforms (e.g. IoT edge devices and data aggregators). Motivated by the limited device lifetimes achievable when powering these systems using batteries, research has recently looked to replace batteries with Energy Harvesting (EH), where energy is scavenged from the device's environment.

A primary challenge in developing embedded systems powered by EH is the unpredictable nature of the source [1], which usually exhibits high temporal and spatial variability. This is illustrated in Fig. 1, where experimentally obtained voltage and power output traces are shown for two typical EH sources. The output from each of these sources varies by many orders of magnitude over the experimental time period, and both would also likely exhibit significant spatial variability. To smooth this variablility, EH systems typically incorporate large external energy buffers (such as rechargeable batteries or supercapacitors) to sustain computation [2]. This approach is known as *energy neutral* operation. Technically, the definition of energy neutral operation includes any EH system that offers unlimited operation by ensuring that its stored energy never completely depletes; however, in practice, the term has come to refer to systems that attempt to balance the long-term energy consumption against the harvested energy over a period of time (e.g. a day) such that the energy consumed equals the energy harvested [3].

Energy neutrality can smooth the long-term variability in EH supplies, but it also presents some drawbacks. Energy storage devices require time to charge, pose environmental issues and deteriorate in performance over time [4]. Additionally, energy-neutral EH systems also require Maximum Power Point Tracking (MPPT) circuitry in order to maximize power extraction under variable energy source conditions. The MPPT circuit relies on additional energy storage to decouple the workload (i.e. computing unit) from the EH source dynamics, so that the workload performance is not affected by the source. Due to the requirement for additional storage and MPPT circuits, energy-neutral systems also suffer from increased volume, weight and cost [5]. Attempting to use a buffer that stores large amounts of energy inevitably leads to high losses due to power harvesting

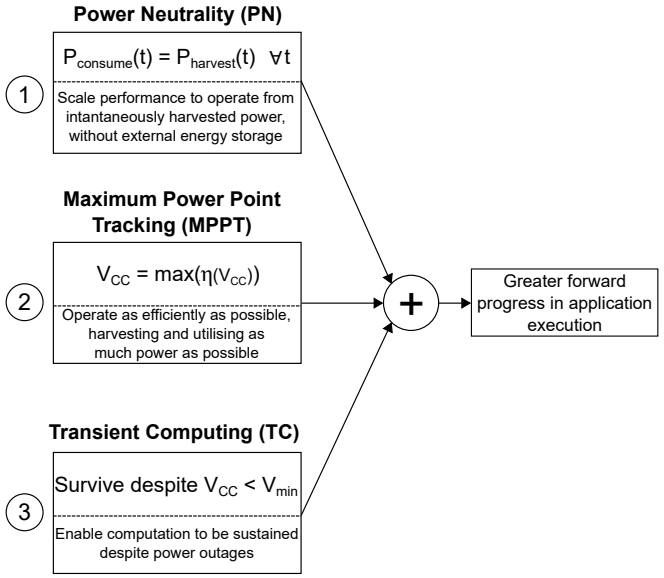

Fig. 2. Merits of the proposed *Momentum* methodology to energy harvesting computing systems.

costs, self-discharge, and converter inefficiencies. Energy storage devices also have limited lifetimes (measured in time or number of cycles) meaning they can constrain the operational lifetime of systems. In systems with stringent volume constraints such as wearables, implantable devices for medical applications [6], and RFID devices [7], it is desirable to minimize the size of the system and by removing, or dramatically reducing, energy storage elements. In such applications, the energy storage and associated power-conversion circuitry can form the majority of the device volume [8].

To overcome these problems, an alternative approach has been proposed whereby systems operate directly from the energy harvesting source, without needing any external energy storage or MPPT circuitry. This emerging class of storage-less systems presents an alternative to the *energy-neutral* paradigm: *power-neutral* operation, where the instantaneous power consumption of the system is dynamically adjusted such that it matches the instantaneous harvested power [9]. When considering an 'ideal' power neutral system, no additional capacitance (C) is required because the instantaneously consumed power is equal to the instantaneous harvested power. However, there is a practical minimum to this, whereby capacitance is present for other purposes e.g. power supply decoupling, parasitic capacitance in electronic components, or capacitance to support the system whilst changing operating point, or saving the system state.

Due to the diverse range of computing scales (from the ultra-low power Microcontrollers (MCUs) found in autonomous sensor nodes, to large Multi-Processor System-on-Chips (MP-SoCs) used for high performance, complex data collection and analysis), applying the concept of power-neutrality efficiently to different scales of systems presents a significant challenge. In this paper, we present *Momentum*[1]: a methodology combining the following novel aspects:

- A general approach for enabling Power Neutrality (PN), through dynamically scaling system performance, that can be applied to computing systems of different scales, from mobile to embedded ((1) in Fig. 2).

[1] *Momentum* relates to the magnitude of harvested power, which is driving the system forward at a specific speed, or frequency. The aim is to keep this as high as possible in response to changing power supply dynamics.

- The combination of this PN approach with a software-based maximum power point tracking scheme (without requiring any external hardware for tracking or control units) that considers the efficiency of the entire system ((2) in Fig. 2). This approach benefits from the unique opportunity to modulate the operating point, and hence power consumption, such that power neutrality is achieved and the overall system efficiency is maximized.
- A practical validation on embedded (an ultra low-power MCU operating from only the already-present 4.7 μF decoupling capacitance) and mobile (a low-power 8-core MP-SoC found in typical smart-phones, requiring only 15.4 mF of added energy storage) platforms. The results show that *Momentum* works effectively on each platform, from both controlled energy sources and real energy harvesting devices.

The *Momentum* methodology also incorporates existing state-of-the-art Transient Computing (TC) approaches ((3) in Fig. 2) in cases where the power from EH is insufficient for the system to operate even in its lowest mode, enabling computation to be sustained despite power outages. This is achieved by saving the system's state before a power failure occurs, and restoring it once the power supply recovers. As shown in Fig. 2, the combination of these three contributions results in greater forward progress in application execution.

Momentum is designed to be broadly applicable across a range of computing scales and energy harvesting sources, rather than being customised for a specific platform and/or source. Because of this, less emphasis in the remainder of the paper is placed on modelling of the energy harvesting source, in favour of detailed characterisation of each energy harvesting source (Section 3.2) to assess the performance of *Momentum*.

Background and related works are presented in Section 2. *Momentum* is then detailed in Section 3 and modelled in Section 4, where experimental parameters are determined for the practical validation presented in Section 5. Results from these experiments on a Texas Instruments MCU and an ODROID XU-4 MP-SoC with real energy harvesting sources are presented in Section 6.

2 BACKGROUND AND RELATED WORK

This section surveys previous work, summarising existing research surrounding control strategies for power neutral systems, MPPT, and transient computing.

2.1 Control Strategies for Power Neutral Operation

Different controls to adjust the power consumption of systems at runtime exist with their implementation and availability dependent on the type of processing unit present in the system. In smaller single-core MCUs, Dynamic Power Management (DPM) and dynamic frequency scaling (DFS) can be used, whilst in larger MP-SoCs more sophisticated techniques exist such as dynamic voltage and frequency scaling (DVFS) [10] and core hot-plugging [11]. Many runtime approaches to manage and adapt these controls have been proposed for energy-aware and charging-aware power management on single-core embedded systems [12] and multi-core embedded systems [13, 14]. However, these schemes are focused on minimising power consumption given specific performance constraints, rather than maximising performance given specific power constraints.

SolarTune [15], a storageless system with a multicore CPU directly coupled to an EH source, uses harvesting-aware runtime task scheduling (in the same way as [16]) to adjust system performance based on the predicted availability of harvested energy. This, however, relies upon accurate prediction of future power availability, making it unsuitable for unpredictable EH sources (such as wind or indoor PV cells). Additionally, *SolarTune* incorporates a backup power supply for when the harvested power is insufficient. Task-based approaches have also been proposed, where the system uses sufficient energy storage for the execution of small tasks [7, 17]. These small tasks (such as

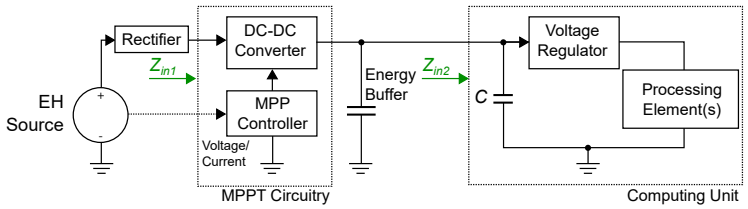

Fig. 3. Typical SoA MPPT system architecture where the energy buffer is used to decouple the dynamics of source from the workload (computing unit).

sampling or transmitting data) are only performed when enough energy is stored in the small capacitor. These approaches however, do not incorporate any adaptation of the system's performance and hence suffer from reduced control granularity. Moreover, the use of these approaches results in additional overhead whilst saving the system's state between tasks.

To address these issues, more recent research has proposed the *power-neutral* operating paradigm (similar to the *power elastic* approach [18], which is focussed on managing computational workloads according to existing power constraints), which dynamically adjust the system's performance in real-time whilst tracking the available harvested power by maintaining a stable operating voltage [9, 19]. These existing approaches, however, aim to maintain a single static operating voltage which does not necessarily correspond to the maximum power point of the system and hence do not perform as efficiently as they could.

2.2 Maximum Power Point Tracking Approaches

Fig. 3 shows a typical energy harvesting system employing hardware-based MPPT. This consists of a switching-mode power converter and Maximum Power Point (MPP) controller to adjust the impedance Z_{in1} and therefore provide maximum power to the energy buffer and the workload (i.e. computing unit). The MPP controller implements this adjustment, by measuring the voltage and the current of the EH source, and drives the DC-DC converter by using Pulse Width Modulation (PWM) [20] or Pulse Frequency Modulation (PFM) control schemes [21]. This introduces significant energy loss associated with the DC-DC converter, which typically ranges between 40% (in the case of PWM schemes, as they generally present poor efficiency due to the switching loss) and 10% (in the case of PFM schemes) [22], along with additional hardware and software control. This arrangement de-couples the energy harvesting source from the workload [23].

These units (such as those adopted in [24] and [7]) use a boost converter with integrated MPPT to convert the input voltage to a suitable level for charging an energy buffer, and therefore a buck converter is used (as a voltage regulator in Fig. 3) to provide the required output voltage.

However, this architecture significantly increases the weight, size and cost of devices due to the MPP control circuitry and large energy buffer. In addition to this, typical MPPT configurations only allow for control of the first stage (MPPT Circuitry) impedance (Z_{in1} as shown in Fig. 3) as the first and second stages are de-coupled. The overall system efficiency is, however, governed by the impedances (Z_{in1} and Z_{in2}) of both stages (MPPT Circuitry and Computing Unit) and therefore the performance of the voltage regulator in stage 2 can have a significant impact, as this regulator is often an inefficient linear regulator which is built-in on-chip. An alternative approach, which provides greater efficiency, is to implement an entirely customised energy harvesting system in VLSI. In [25], the authors discuss state-of-the-art micro-scale energy harvesting systems and conclude that in order to ensure efficient operation, the entire system needs to be optimized in

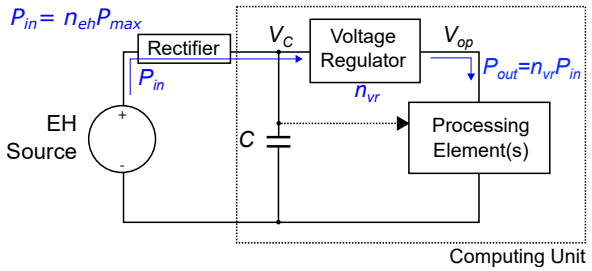

Fig. 4. Schematic of the proposed EH powered system using *Momentum*.

a holistic way. However, such thorough customisation limits the system to one specific energy harvesting source and application.

For example, in [26], the authors demonstrate high efficiency maximum power point tracking for micro-power applications. The proposed hardware introduces minimal power overheads, however is highly complex and finely tuned for a specific energy harvesting source. Additionally, although both of these papers propose a holistic approach to designing the 'front-end' energy harvesting circuitry (by considering the transducer, power converter and energy buffer type), little consideration is given to the power profile of the attached computing unit and its application.

2.3 Transient Computing Techniques

When considering embedded systems powered by energy harvesting, it is important to accommodate the unpredictable, and often transient, nature of the power source. Various approaches for transient computing exist to facilitate system state retention and restore, which allow the system to continue executing where it left off after a supply interruption [27]. Broadly, these can be classified into two categories; software-based approaches [28, 29], where the system's state (e.g. core and general-purpose registers, and main RAM memory) is saved into a Non-Volatile Memory (NVM) before a power failure occurs and restored once the power supply recovers, and hardware-assisted approaches [30, 31], where the entire system is designed to be non-volatile (e.g. a unified low-power NVM system with non-volatile core and general-purpose registers). As *Momentum* aims to be broadly applicable, without dictating a specific hardware platform, it incorporates a software-based transient computing approach, *Hibernus* [28], which has a low time and energy overhead whilst saving the system's state [32]. Additionally, recent advances in non-volatile memory technology mean that software-based transient computing techniques are becoming much more power efficient as NVM read and write operations require less power [33, 34].

2.4 Summary

This section has surveyed previous research surrounding power neutrality, MPPT and transient computing techniques, identifying the following three unsolved challenges:

- Existing power neutral control schemes aim to stabilise the operating voltage at a fixed value, which does not necessarily correspond to the maximum power point of the system.
- Because systems typically operate under energy neutral conditions, external tracking or control units are required for maximum power point tracking, adding complexity. Additionally, this external circuitry needs to be customised to the energy harvesting source.
- Existing power neutral control schemes are 'specialised' for particular hardware platforms, and hence are not transferable between computing units of different scales.

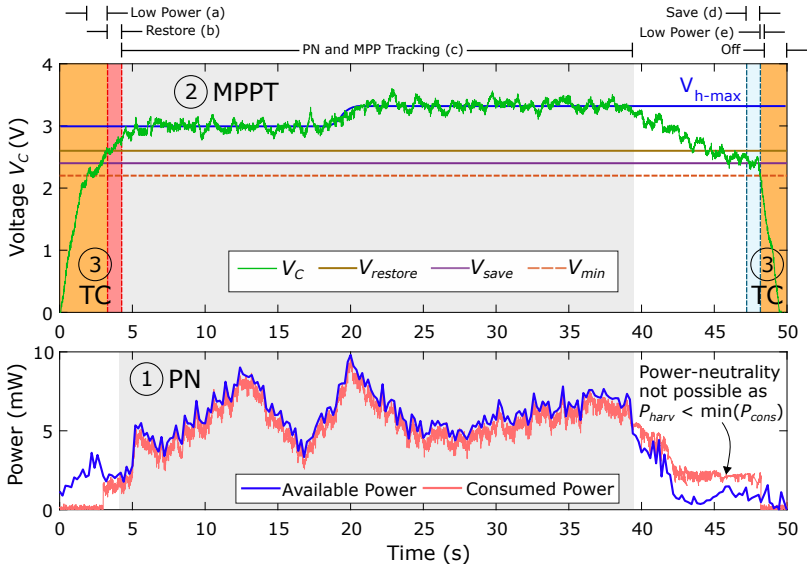

Fig. 5. Conceptual illustration of an EH system under power-neutral operation, showing V_C, the available harvested power, and the consumed power.

Momentum provides a solution to these challenges, in the form of a power-neutral control scheme (Section 3.1) where the system's operating point is dynamically adjusted in accordance with the changing maximum power point. This is performed without additional energy storage or customised hardware taking into account the whole system efficiency. Whilst optimised hardware would inevitably lead to improved efficiency when tuned for a specific application [25, 26], *Momentum* is a broadly-applicable control approach which does not require any customised hardware and can be implemented on existing platforms using only a few additional 'off-the-shelf' components. Moreover, *Momentum* is broadly applicable to systems of varying scale taking into account the overall system efficiency (Section 3.2).

3 MOMENTUM DESIGN APPROACH

Fig. 4 illustrates the proposed EH powered system using *Momentum*, where a *computing unit* is directly coupled to a time-varying EH source. Here, V_C is the voltage across the small decoupling capacitance C which is tracked to perform MPPT, and V_{op} is the voltage that the processing element (i.e. the MCU) is being operated at which ensures maximum computational efficiency.

Compared to the system architecture presented in Fig. 3, this system does not require an additional MPP controller, DC-DC converter or external energy buffer, as the MPPT is controlled by the proposed scheme. Here, *computing unit* refers to one or more processing elements of any scale (including anything from an ultra-low power single core MCU, to a large heterogeneous MP-SoC) in addition to a voltage regulator (used to maintain the desired pre-defined voltage level for the processing elements) and a small capacitance C (to guarantee the stability of the system). At its minimum, C is just the system's decoupling capacitance.

The 'rectifier' element shown in Fig. 4 represents the rectification circuitry required to convert the EH source voltage to a positive DC voltage and prevent back-flow of energy to the energy harvester, and should be chosen for compatibility with the adopted energy harvesting source. For example, when incorporating a PV cell or TEG which provides a uni-polar output, a simple low

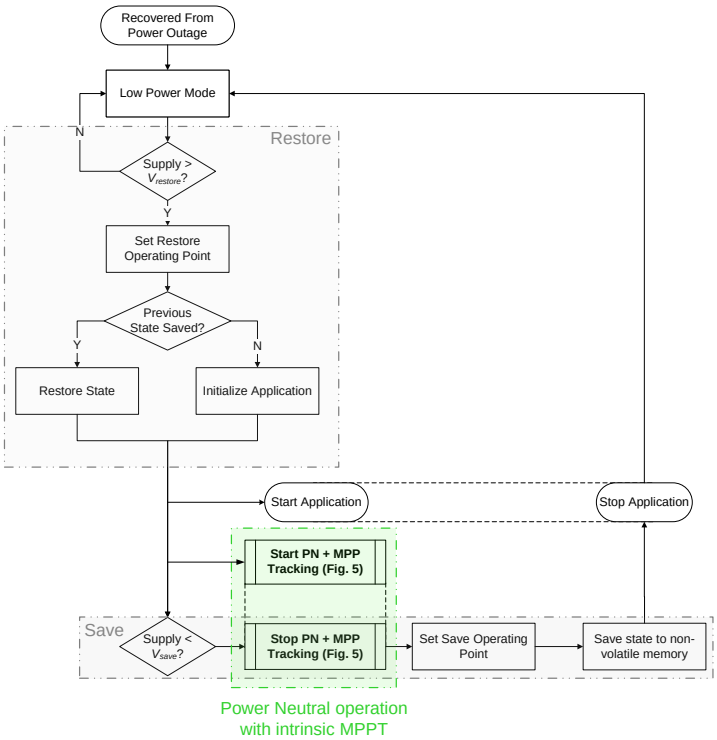

Fig. 6. Overall operation of *Momentum*, highlighting the novel power-neutral software-based MPPT scheme.

voltage drop Schottky diode can be used. When considering a system powered by an AC source (e.g. when harvesting energy using piezoelectric elements or micro wind-turbines), an AC-DC rectifier must be employed. This can be a simple full-wave AC-DC rectifier with passive diodes, or a hybrid rectifier which combines passive and active rectifiers together. Each of these solutions can be used in tandem with *Momentum* (without modification of the algorithm, simply by tuning the system efficiency profile), however detailed consideration of these approaches is beyond the scope of this work.

Fig. 5 illustrates the ideal behaviour of *Momentum*. The first task is to efficiently restore the previous state of the system. Initially, as the voltage, V_C, is below the minimum operating voltage, V_{min}, the processing element(s) are powered-down. As power becomes available, the processing element(s) are powered through the voltage regulator and the system is placed in a low-power mode ((a) in Fig. 5). It remains in this state until V_C reaches the state restore threshold $V_{restore}$. At this time, the system attempts to restore a previous state ((b) in Fig. 5) where a prior save has been successful, else, the system is initialised.

In order for the system to operate under power neutrality, whilst tracking the MPP ((c), in Fig. 5), the system reacts to maintain V_C at the voltage corresponding to the maximum system efficiency ($V_{\eta-max}$), and hence MPP. This is achieved by on-the-fly adjustment of the system operating point with respect to the voltage V_C. This is discussed further in Section 3.1. The maximum efficiency considered here is the product of the EH source efficiency, η_{eh} shown in Fig. 4 and the intermediate voltage regulation efficiency, η_{vr}. This is discussed further in Section 3.2.

Whilst the supply voltage remains above the state-save threshold, V_{save}, the system continues to operate by dynamically adjusting the operating point according to the incoming power (power-neutrality, Section 3.1) and the efficiency of the overall system (maximum efficiency tracking, Section 3.2).

Power-neutral operation is only feasible when the harvested power is greater than the system's minimum power consumption. For this reason, *Momentum* incorporates the facility for saving the system's state when a power outage is imminent. When the harvested power is not sufficient to achieve power neutrality, even at the processor's minimum operating point, the system's state is saved to non-volatile memory ((d), in Fig. 5) when the supply voltage drops below V_{save} (transient operation in Section 3.3). The system remains in a low-power mode until V_C recovers ((e), in Fig. 5). For the reasons outlined above, power neutral systems are not suitable for use in applications which mandate continuous operation. An example of this could be a system for monitoring pollution levels of a particular area during the daytime.

Fig. 6 illustrates the operation of *Momentum*. When compared with previously published works, *Momentum* adds system-wide MPPT, enabled intrinsically through power neutral operation (highlighted on Fig. 6). Aside from this, *Momentum* also facilitates system save and restore, building upon previous works [9, 28] as shown in Fig. 6. The restore area shows how the system's state is recovered, following a power failure. After recovering the system's state, the main application and PN + MPP tracking algorithm are started simultaneously. Finally, the save area shows how the system's state is saved when the harvested power becomes insufficient.

3.1 Power Neutral Tracking

Fig. 5 illustrates the concept of power neutrality where the consumed power follows the available harvested power over time. In order to achieve this, the system must react by identifying the correct operating point and hence power consumption.

To facilitate operating point adjustment, two dynamic voltage thresholds are used, V_{high} and V_{low}. These thresholds track the voltage, V_C, which exists across the capacitance C [2].

V_{high} and V_{low} are dynamically alterable, however a constant potential, V_{width}, exists between them. Consideration of the values for V_{high} and V_{low} are provided in the following section, 3.2. Depending on the type of computing unit within the system, the nature of this run-time performance scaling varies. This can be achieved through simple dynamic frequency scaling (DFS) or using more sophisticated strategies such as dynamic voltage and frequency scaling (DVFS) and dynamic power management (DPM) through enabling/disabling different processing elements and peripherals at runtime. The combination of these different performance scaling strategies results in a number of fixed operating points (OP), where each has its own corresponding power consumption.

The *Momentum* control approach facilitates operation at any of these fixed OPs (between the lowest and the highest). Due to static power consumption, operating a system at its highest OP may often be the most energy efficient solution (when compared with operating it at a lower OP for a longer time). However, the power neutral control scheme tracks the available power and hence cannot always operate at the highest OP as this will increase the likelihood of power failure. Power failures occur when the consumed power exceeds the harvested power at a given instant and as a result, the system's state must be saved and eventually rebooted, introducing significant overhead in terms of power consumption and downtime.

In general, the latency associated with DFS (or DVFS) is lower than that associated with DPM (empirical validation of this assumption is provided in Section 5.2) activities such as enabling/disabling processing elements or peripherals at runtime. Therefore, the proposed methodology incorporates

[2]Consideration of the size of this capacitance, C, is provided in Section 5.2

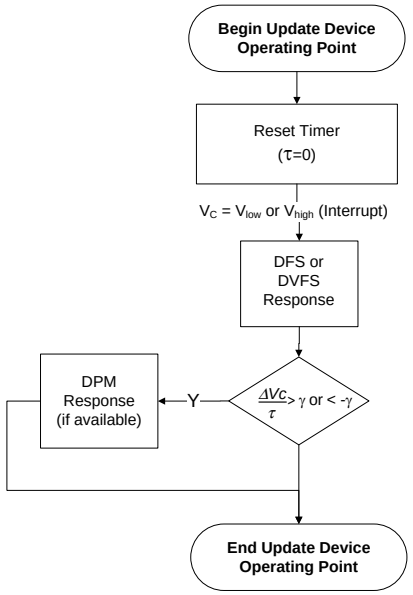

Fig. 7. Power-neutral performance scaling approach using DFS (or DVFS) and DPM.

a hierarchy of power management activities, ranked by their latencies such that DFS (or DVFS) will be performed most readily and frequently to deal with *micro* variations in the harvested power supply, whereas DPM is applied to deal with *macro* variation in the harvested supply where this is supported by the platform.

Fig. 7 shows how the operating point is updated in the event of a threshold V_{low} or V_{high} being reached. Here, derivative control is applied to determine the DPM control response. *Macro* variation in the EH supply is detected using a timer τ which measures the time elapsed since the previous update and hence estimates the derivative $dV_C/dt \approx \Delta V_C/\tau$. If this value is larger than a specific gradient threshold parameter γ, the control algorithm also considers DPM, else, only DFS (or DVFS) is considered. Algorithms which use this PN approach on both a MCU and MP-SoC are presented in Section 5, and hence more detailed information is explained there.

3.2 Maximum Efficiency Characterisation

As highlighted in Section 2.2, existing approaches for MPPT only allow maximum power transfer between the EH source and the large energy buffer, and rely on voltage regulators between the energy buffer and the workload (i.e. computing unit), to convert energy as efficiently as possible from the energy buffer to the load. This was shown in Fig. 3. However, these voltage regulators are often built-in on-chip and are often inefficient linear regulators due to their low cost. *Momentum* addresses this issue by considering the whole system efficiency as the voltage, V_C, can be directly controlled.

3.2.1 Overall System Efficiency. Considering again the schematic shown in Fig. 4, the overall system efficiency is governed by two factors; the efficiency of the voltage regulation (η_{vr}), and the efficiency of the EH source (η_{eh}).

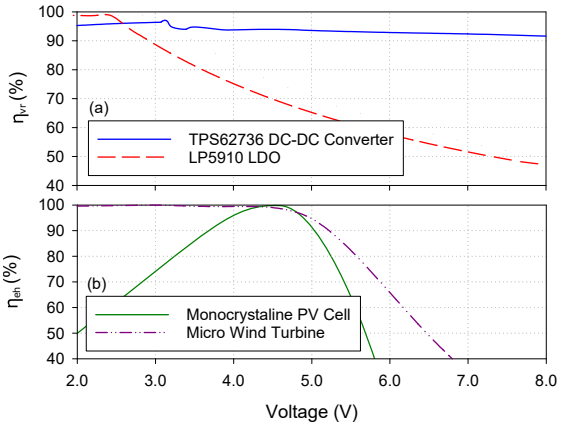

Fig. 8. Efficiency of (a) typical LDO and DC-DC regulators [V_{out} = 2.5V, I_{load} = 1mA] and (b) PV and wind EH sources [Solar Irradiance = 200W/m^2, Wind Speed = 7.4m/s].

We will first consider the efficiency of the voltage regulation, η_{vr}, given by

$$\eta_{vr} = \frac{P_{out}}{P_{in}} \tag{1}$$

where P_{in} is the power delivered by the EH source, and P_{out} the power delivered to the CPU cores. Fig 8 (a) illustrates this efficiency, η_{vr} for two types of typical voltage regulator (DC-DC and LDO voltage regulators) for a single value of V_{out} and I_{load}. In practice, this efficiency will vary as the operating conditions (V_{out} and I_{load}) change.

In a similar way, to maximise P_{in} such that the maximum source power P_{max} is delivered, the efficiency of the energy harvester must also be considered, which will vary as the operating conditions (such as the operating voltage and power available to harvest) fluctuate. For example, when considering a typical photovoltaic (PV) cell, the power available to harvest is proportional to the solar irradiance or, for a micro wind turbine, proportional to the wind speed. The efficiency of the EH source is given by

$$\eta_{eh} = \frac{P_{in}}{P_{max}} \tag{2}$$

Fig 8 (b) shows how this efficiency, η_{eh}, varies with voltage for a typical PV cell and micro wind turbine. Here, readings are shown for one set of operating conditions.

Combining these, the overall system efficiency will be a product of η_{eh} and η_{vr}, as shown below

$$\eta_{sys} = \frac{P_{out}}{P_{max}} = \left(\frac{P_{out}}{P_{in}}\right) \cdot \left(\frac{P_{in}}{P_{max}}\right) = \eta_{eh}\eta_{vr} \tag{3}$$

and will therefore vary with the operating voltage V_C, load current I_{load}, and the power available to harvest, P_{max}.

As an example of this, Fig. 9 shows the overall efficiency $\eta_{eh}\eta_{vr}$ of the system presented in Fig. 4 where the EH source is a PV cell, the rectifier is a Schottky diode, and the voltage regulator is a switching (DC-DC) converter. Fig. 10 shows the same system with a low drop-out (LDO) linear regulator. In this second case, the voltage regulator has a significant impact on the overall system efficiency and MPP location. Five efficiency curves are shown corresponding to five increasing values of solar irradiance from Ir_0 to Ir_4. The maximum efficiency $(V_{\eta-max})$ for each value of irradiance is marked, and varies significantly with respect to solar irradiance.

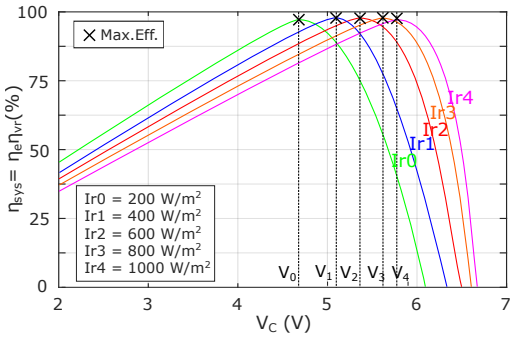

Fig. 9. Overall system efficiency characteristics (with PV EH and DC-DC converter) plotted against output voltage for five values of solar irradiance (I_r).

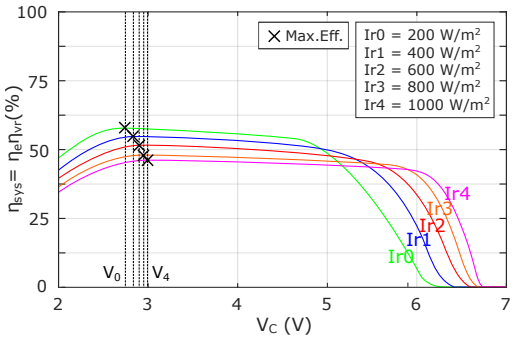

Fig. 10. Overall system efficiency characteristics (with PV EH and LDO regulator) plotted against output voltage for five values of solar irradiance (I_r).

3.2.2 Maximum Efficiency Tracking Approach. Fig. 11 shows the power output characteristics and the maximum efficiency (Max. Eff.) for the system presented in Fig. 9. In order for a system to operate with maximum efficiency, it is important that the system's operating point closely matches the MPP for a given value of irradiance. Therefore, the proposed power output tracking scheme will maintain the system operating voltage between $V_{high}(t)$ and $V_{low}(t)$ which are defined as:

$$V_{high}(t) = V_{\eta-max}(t) + \frac{V_{width}}{2} \tag{4}$$

and

$$V_{low}(t) = V_{\eta-max}(t) - \frac{V_{width}}{2} \tag{5}$$

where $V_{\eta-max}(t)$ is the voltage corresponding to the MPP at a given time t.

To reduce the software overheads of this approach, the domain of $V_{high}(t)$ and $V_{low}(t)$ is quantised in N discrete values, $V_{high}[i]$ and $V_{low}[i]$, with i varying between *0* and *N-1*. These values are selected aiming for the system to operate in the region of maximum efficiency (dark grey area in Fig. 11). To achieve this, the power domain is also quantised into N discrete sectors (from $Sect_0$ to $Sect_{N-1}$), each of these with associated values of $V_{high}[i]$ (triangle) $V_{low}[i]$ (circle). Both parameters V_{width} and N are determined through simulation in Section 4.

Fig. 12 illustrates the operation of the power neutral tracking scheme. Initially, the power is estimated in order to classify the power sector in which the device is operating. According to this classification, thresholds V_{high} and V_{low} are then set. If V_C crosses V_{high} or V_{low}, a voltage interrupt is generated, causing the system to scale its own performance (and hence power consumption, as described in Section 3.1).

All previous related work requires additional hardware to perform this estimation, either by evaluating the harvested power output (e.g. by sampling the energy harvesting source status periodically) or the harvesting conditions (e.g. using a pilot harvester). *Momentum* tracks the maximum power point without using any external tracking or control units by exploiting the off-line MPP characterisation curve, in addition to the power neutrality relationship:

$$P_{in} \approx P_{out} \tag{6}$$

(where P_{in} is the harvested power after regulation and P_{out} is the power consumed by the processing element) to gain knowledge of the harvested power. The consumed power (P_{out}) can be estimated

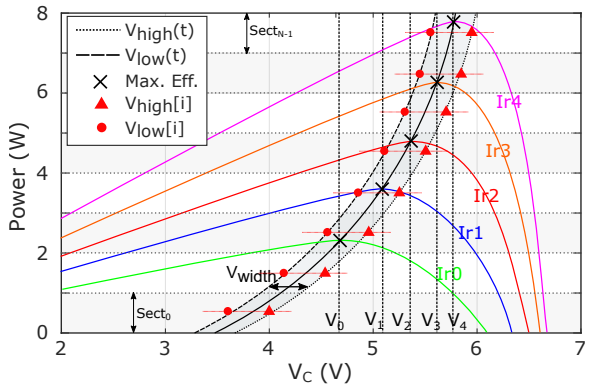

Fig. 11. Typical system power characteristics plotted against V_C for the same five values of solar irradiance (I_r).

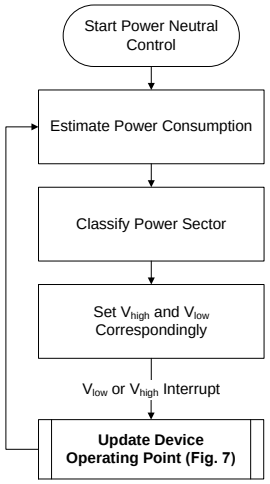

Fig. 12. The general control approach flowchart for PN tracking.

at runtime using existing power models [35] and hence the harvested power (P_{in}) can be derived using Eqn. 6.

3.3 Transient Operation

To deal with the unstable EH source output, *Momentum* also includes transient operation where the system's state is saved to non-volatile memory (NVM) as the power supply falls below V_{save} and resumes operation when the power supply recovers (see Fig. 5). The number of save and restore operations depends on the intermittency of the power source, meaning that it falls below the minimum operating point of the system. Depending on the type of computing unit within the system, transient operation can be included either as a customised library such as *Hibernus* [28] or, where present, through support from the OS (e.g. Ubuntu). A highly intermittent or very

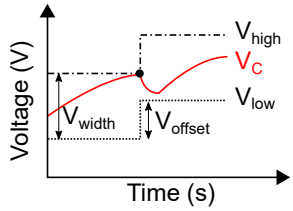

Fig. 13. Example of threshold adjustment in response to a gradually rising input voltage.

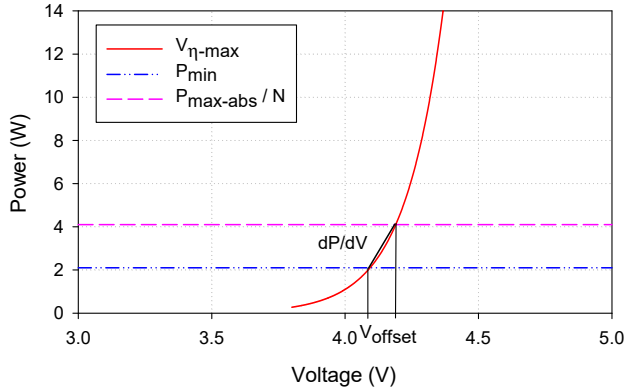

Fig. 14. The dependency of V_{offset} on N, with respect to the solar cell's MPP curve.

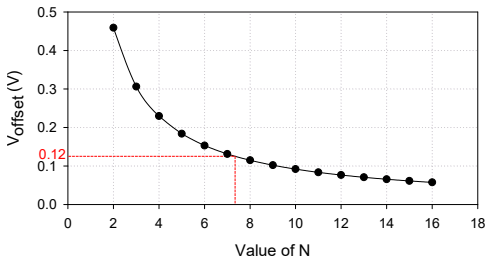

Fig. 15. Relationship between the V_{offset} and N, determined by Equation 8 for the PV array from Fig. 14.

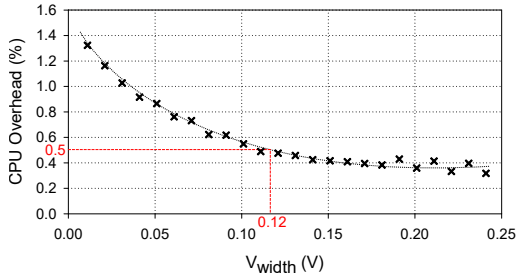

Fig. 16. Relationship between the CPU overhead and V_{width}, determined through simulation using a 24 hour solar data trace.

current-constrained source will therefore result in greater overhead being incurred, while a constant and stable source will have virtually no overhead as a result of the proposed approach.

4 SYSTEM MODELLING AND SIMULATION

In order to determine suitable values for the algorithmic parameters N and V_{width}, we take into account the stability of the proposed approach in addition to the number of system interruptions which are incurred.

We will first consider the stability of the system to successfully track V_C. The system's response to a gradually rising voltage V_C is shown in Fig. 13. As V_C meets the upper threshold V_{high} the performance of the system is increased (hence a momentary drop in V_C is observed as the change

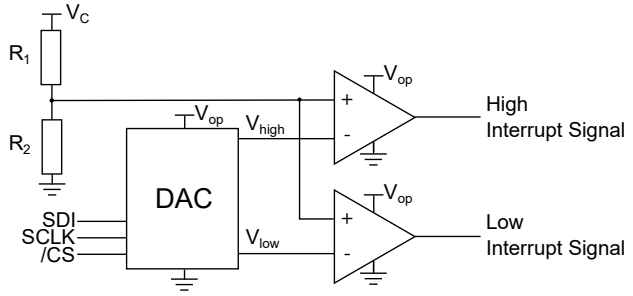

Fig. 17. Schematic showing PMU hardware.

in operating point causes an energy overhead), following this, both thresholds are increased by a value V_{offset} according to the next available operating point.

For the system to be stable, and the thresholds to track the voltage V_C, the worst case offset V_{offset} must satisfy

$$V_{offset} < V_{width} \tag{7}$$

In addition to this, V_{offset} is also related to the number of discrete power sectors, N, and the maximum efficiency curve used in the system. Fig. 14 shows this for a large PV cell connected to a DC-DC converter. It can be observed that the largest value of V_{offset} is present in the lowest ranked power sector (between P_{min} and $P_{max-abs}/N$). Here, V_{offset} is given by:

$$V_{offset} \approx \frac{P_{max-abs}}{N \frac{dP_{max-eff}}{dV}} \tag{8}$$

where $P_{max-abs}$ is the absolute maximum power that can be obtained. For the PV cell with characteristics shown in Fig. 14, V_{offset} has been evaluated for multiple values of N and plotted as shown in Fig. 15.

On the other side, to find suitable values for V_{width} and V_{offset}, we also need to consider the trade-off which exists between accurate tracking of the MPP, and the number of system interruptions (i.e. CPU overhead due to the time and energy needed for setting a new operating point) which are incurred. To evaluate this, the control approach was simulated with operating parameters relating to a typical MP-SoC. Fig. 16 shows these simulation results, plotting the CPU time overhead as a function of V_{width}. To minimise the software overhead of the proposed approach whilst still efficiently tracking the maximum efficiency, the acceptable CPU overhead has to be decided. For example, if it is decided that 0.5% CPU time is acceptable (less than 1% is typically acceptable), this corresponds to V_{width} = 0.12V.

From Equation 7, a value of V_{offset} less than 0.12V must therefore be selected for the system to be stable. As above, V_{offset} should be less than 0.12V to ensure system stability, corresponding to N equal to or greater than 8 (see Fig. 15). The same evaluation procedure can also be followed for a typical MCU, and yields the values V_{width} = 0.16V, N = 8.

5 EXPERIMENTAL CASE STUDY

In order to demonstrate that the proposed methodology is valid across different embedded computing scales, it was implemented on both MCU and MP-SoC platforms. In both cases, in order to minimise the software overhead, external low-power circuitry is used to generate hardware interrupts corresponding to V_{high} and V_{low}. A schematic of this power management unit (PMU) is shown in 17. Here, a 10-bit digital-to-analogue converter (DAC) controlled by the processor is used

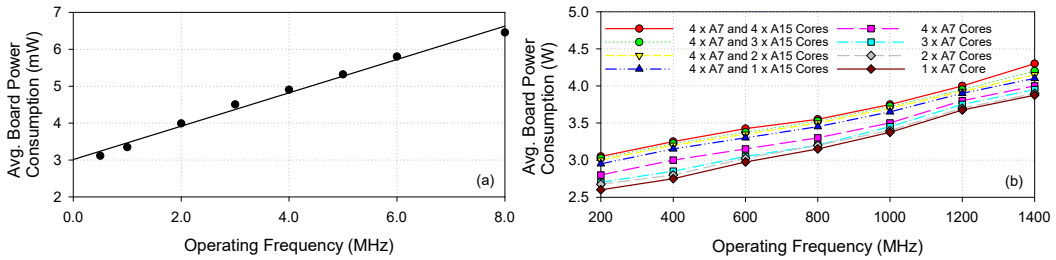

Fig. 18. Experimentally obtained power vs operating point characteristics for both testing platforms for (a) the TI MSP430 MCU and (b) the ODROID XU-4 MP-SoC.

to set the voltage thresholds V_{high} and V_{low}. Two low-power analogue comparators are then used to compare these voltage levels to V_C (post scaling by the potential divider R_1, R_2). In the event that V_C crosses a voltage threshold, the comparator output changes and an interrupt is generated.

For testing on an MCU, the TI MSP430FR5739 processor was used, which incorporates a 16-bit RISC CPU operating between 2V and 3.6V supporting operating frequencies up to 24MHz. Tests with this MCU were performed using a 50cm^2 PV cell as the EH source, and benchmarked using a typical IoT-like application; a Fast Fourier Transform (FFT) analysis of three arrays, each holding 128 × 16-bit samples of tri-axial accelerometer data. Fig. 18 (a) shows the power-performance characteristics of the MCU across multiple operating frequencies whilst running the FFT application. Here, the power consumption varies linearly with frequency.

For testing on an MP-SoC, the ODROID XU4, built around the Samsung Exynos5422 big.LITTLE processor, was used. This processor operates between 4V and 5.7V supporting frequencies in the range 0.2GHz to 2GHz. For testing purposes, a 1340cm^2 outdoor PV array was used to power the MP-SoC whilst running a face recognition algorithm using OpenCV to provide parallelised and CPU-intensive computation [36] (again, a typical long running IoT-like sensing application). Whilst the PV array used for evaluation is reasonably large, the PV array and additional capacitance adopted in this work form a proof-of-concept demonstration that *Momentum* can be applied to computing units of varying scale. Technological developments in the coming years should allow a much more compact form factor. Fig. 18 (b) shows a similar characterisation for the MP-SoC whilst running the face recognition algorithm.

For the MCU, no additional startup circuitry was required for booting the system reliably as the current consumption whilst $V_{op} > V_{min}$ is negligible. In this case the system naturally boots itself to continue operation after a power failure. For larger systems, such as the MP-SoC, the current drawn whilst the supply is lower than the minimum operating voltage (V_{min}) can be significant. *Momentum* therefore assumes that the intermediate voltage regulator (cf. Fig. 4) holds the computing unit in reset until $V_{op} > V_{min}$.

5.1 Power-neutral control scheme

The following sections outline the power neutral control scheme when applied to the MCU and MP-SoC platforms. Due to the architectural differences between the two platforms, performance modulation varies from simple DFS (on the MCU) to DVFS and DPM (on the MP-SoC). To estimate the power consumption at run-time the MCU uses a simple look-up table, considering the linear dependency between the power consumption and the operating frequency (Fig. 18 (a)), while the MP-SoC uses run-time CPU power modelling shown to be accurate ±3.8% [35].

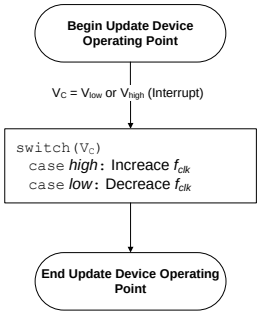

Fig. 19. DFS control algorithm for a typical single-core microcontroller.

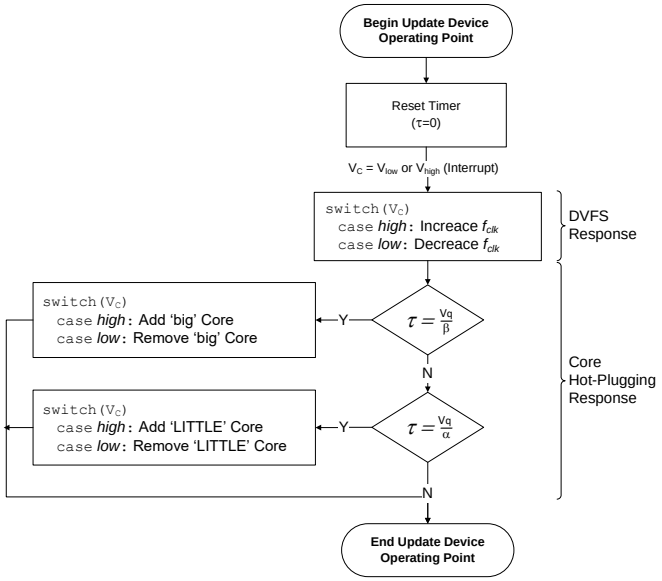

Fig. 20. Power-neutral performance scaling approach using DVFS and core hot-plugging for a heterogeneous multicore SoC.

5.1.1 MCU. Fig. 19 shows the operation of the DFS control algorithm for the MCU. The algorithm waits for the supply voltage to be equal to V_{high} or V_{low}. If the supply voltage is equal to V_{low} a voltage interrupt is generated and the system responds by reducing the operating frequency, f_{clk}. In a similar way, if the voltage rises to become equal to V_{high}, f_{clk} is increased.

5.1.2 MP-SoC. For the MP-SoC, we also provide DPM through the enabling/disabling of CPU cores at runtime (also known as *core hot-plugging*). In heterogeneous systems such as this, two or more CPU clusters with complementary power-performance characteristics are present, e.g. ARM big.LITTLE. This provides a greater degree of flexibility when considering runtime performance scaling. In order to leverage this, the proposed operating point selection algorithm takes into consideration this kind of architecture.

Fig. 20 shows the way in which the performance scaling response is determined for the MP-SoC. Initially a timer τ is started. Following this, when V_C becomes equal to V_{high} or V_{low} a hardware interrupt is generated. This prompts two forms of performance scaling response (1) DVFS and (2) Core Scaling.

Due to the delay associated with DVFS typically being lower than that associated with DPM [37], the first response is to scale the system's operating voltage-frequency to deal with *micro* variation in the harvested power supply. More specifically, the frequency (and consequently the associated voltage) is updated when an interrupt occurs to a new value between $(f_0, f_1 \ldots f_{N-1})$, depending on the previous operating frequency.

Following this, to deal with *macro* variation in the harvested supply, derivative control is applied to calculate a core hot-plugging response so that the number of active cores is proportional to dV_C/dt.

In order to explain the core hot-plugging response for a heterogeneous architecture, two ternary *core scaling factors*: CS_b and CS_L (for 'big' and 'LITTLE' cores respectively) are introduced. The core scaling factors may take one of three values 0, 1 or -1 where 1 denotes the addition of a core, -1 denotes the removal of a core, and 0 denotes no alteration.

Two constant gradient threshold parameters α and β are also defined for *LITTLE* and *big* cores respectively, which represent the minimum gradient required to warrant a change in the existing core configuration.

$$CS_b = \begin{cases} 1 & \text{if } \frac{dV_C}{dt} > \beta \\ -1 & \text{if } \frac{dV_C}{dt} < -\beta \\ 0 & \text{otherwise} \end{cases}, CS_L = \begin{cases} 1 & \text{if } \frac{dV_C}{dt} > \alpha \\ -1 & \text{if } \frac{dV_C}{dt} < -\alpha \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

In order to estimate $\frac{dV_C}{dt}$, an approximation is used shown in Eq. 10.

$$\frac{dV_C}{dt} \approx \frac{\Delta V_C}{\Delta \tau} = \frac{V_{width}}{\tau} \tag{10}$$

Combining this with (9), the response when the V_{low} threshold is met is given by

$$CS_b = \begin{cases} -1 & \text{if } \tau < \frac{V_q}{\beta} \\ 0 & \text{otherwise} \end{cases}, CS_L = \begin{cases} -1 & \text{if } \tau < \frac{V_q}{\alpha} \\ 0 & \text{otherwise} \end{cases}$$

and the response when the V_{high} threshold is met is given by

$$CS_b = \begin{cases} 1 & \text{if } \tau < \frac{V_q}{\beta} \\ 0 & \text{otherwise} \end{cases}, CS_L = \begin{cases} 1 & \text{if } \tau < \frac{V_q}{\alpha} \\ 0 & \text{otherwise} \end{cases}$$

5.2 Energy Storage

In order for the system to react in a stable way to fluctuations in the harvested supply, a small amount of capacitance must be added. This is only present to support the system in the short-term whilst reacting, and hence does not make the system energy neutral (it is not intended to smooth the supply variability over a long period of time). To calculate the minimum value of this capacitance, C, the latencies associated with performance scaling on both systems were considered. Table 1 shows the latency from receiving the interrupt to scaling the performance on the MCU using DFS, while Tables 2 and 3 show the latency from receiving the interrupt to scaling the performance using DVFS and DPM (respectively) on the ODROID XU-4 platform.

Using these data, the minimum value for the capacitance C can be found by considering the worst case scenario, where the system must adapt its performance from the highest operating point

to the lowest operating point due to a sudden and significant decline in harvested power. If C is large enough that it can hold sufficient charge to sustain the processor through this period, then the system should respond robustly to any input.

When considering the MCU, this charge simply corresponds to the worst case frequency scaling latency[3], multiplied by the current draw of the system and so the minimum capacitance C is around 50nF. However, a minimum decoupling capacitance of 4.7 µF is already recommended by manufacturers for this MCU, and hence no additional storage was added. This value is also sufficient to support the system when hibernating.

For the MP-SoC board, both the voltage and frequency scaling latencies δ_f and core hot-plugging latencies δ_{core} must be considered. The minimum charge required is therefore given by:

$$Q = \int_{t=0}^{t=\delta_f} I_1(t)dt + \int_{t=\delta_f}^{t=\delta_f+\delta_{core}} I_2(t)dt \tag{11}$$

This was practically evaluated for the ODROID XU-4 and found to be around 0.46 C, corresponding to a capacitance C of 15.4mF. A 47mF supercapacitor was added to the platform for the experiments in this paper as it was the next highest component size readily available for purchase. For a system with power consumption between 2.5-4W, 47mF still represents a tiny capacitance that could only sustain system operation for around 0.2s. In both cases, as the worst case latency associated with performance modulation (δ_f for the MCU and $\delta_f + \delta_{core}$ for the MP-SoC) is so small, the parasitic leakage across the capacitor over this time is negligible and so omitted in Eqn. 11. If the source varies rapidly (with a period in the same order of magnitude as that of the latencies), the approach will not be efficient due to the latency of system operation scaling. However, in the case of the MCU these latencies are negligible, and the system works reliably, even with rapidly varying EH sources (e.g. a micro wind turbine). In the case of the ODROID, the latencies are slightly higher but still acceptable for EH sources such as PV cells.

The CPU state is typically not the only volatile state in the system; for example attached peripherals such as sensors or communication modules will also require their state to be saved on a power failure (when power neutral operation can no longer continue). The approach that we present in this paper is supplementary to existing transient approaches that have been proposed to accomodate this [38].

6 RESULTS AND ANALYSIS

Testing with both boards was performed with controlled variable voltage supplies (Section 6.1) in addition to real photovoltaic energy harvesting sources (Section 6.2). This methodology is then compared against the state-of-art (Section 6.3).

6.1 Response to a Controlled Supply

Both systems were initially tested using a controlled voltage supply in order to verify the basic operation of the proposed control scheme. Fig. 21 (a) shows the behavior of the MCU to a varying supply voltage and Fig. 21 (b) shows a similar plot for the MP-SoC with the addition of the system's core scaling response. The control scheme can be seen to operate successfully on both platforms, modulating frequency (and hence power consumption) with respect to the input voltage V_C. In the case of the MP-SoC, it can also be observed that DPM is triggered much less often than DVFS suggesting that the system is effectively and appropriately selecting long-term and short-term performance scaling responses.

[3]For experiments in this paper, the maximum frequency of the MSP430 is limited to 8 MHz to avoid having wait states while accessing FRAM (the maximum frequency of which is 8 MHz).

Table 1. Average measured frequency scaling latency on the TI MSP430FR5739 MCU.

Frequency Range	Scaling Up Latency (us)	Scaling Down Latency (us)
1.0 MHz ⟷ 3.0 MHz	53.7	39.8
1.0 MHz ⟷ 8.0 MHz	58.6	20.0
1.0 MHz ⟷ 5.3 MHz	62.2	24.2

Table 2. Average measured frequency scaling latency on the ODROID XU-4 with all CPU cores enabled.

Frequency Range	Scaling Up Latency (ms)	Scaling Down Latency (ms)
0.2 GHz ⟷ 0.4 GHz	2.5	2.2
0.8 GHz ⟷ 1.0 GHz	2.2	2.1
1.2 GHz ⟷ 1.4 GHz	2.6	2.6

Table 3. Average measured core scaling latency on the ODROID XU-4 operating at 0.2 GHz.

Number of Cores	Latency Whilst Adding Core (ms)	Latency Whilst Removing Core (ms)
1 Core ⟷ 2 Cores	39.6	70.1
3 Cores ⟷ 4 Cores	35.1	45.4
7 Cores ⟷ 8 Cores	36.5	44.2

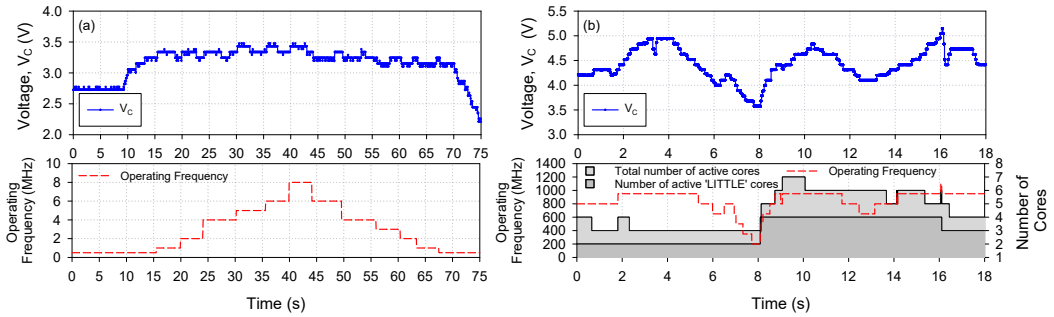

Fig. 21. System performance using a controlled variable voltage supply for (a) the TI MSP430 MCU and (b) the ODROID XU-4 MP-SoC.

6.2 Response to an Energy Harvesting Supply

Testing was then performed with the systems powered by energy harvesting, using the PV cells mentioned above (Section 6).

6.2.1 Power Neutrality. Fig. 22 shows the estimated available power and the power consumption of each system for comparison. The power was estimated using an additional 'pilot' PV cell placed adjacent to the active cell ensuring that the incident solar irradiance is comparable. The open circuit voltage of this adjacent cell was then measured and the corresponding MPP values were obtained. As shown in Fig. 22, the power consumed by the device closely matches the available harvested power meaning that the system is operating under power neutrality.

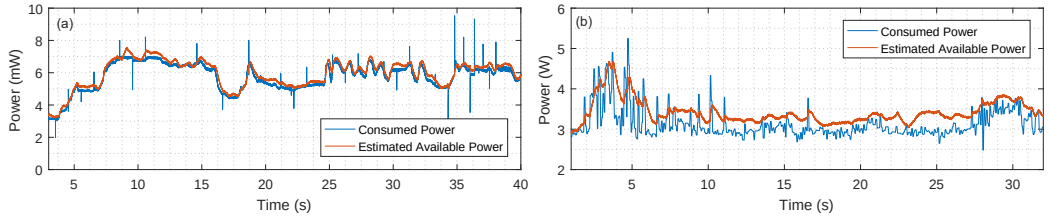

Fig. 22. Results of available and consumed power over time showing correct power neutral behaviour, for (a) the TI MSP430 MCU and (b) the ODROID XU-4 MP-SoC.

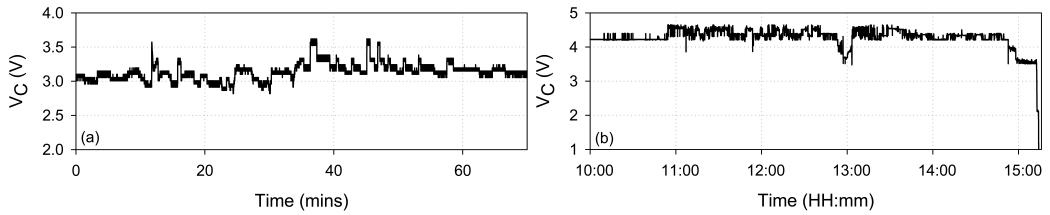

Fig. 23. V_C over time whilst testing the system with PV energy harvesting for (a) the TI MSP430 MCU and (b) the ODROID XU-4 MP-SoC.

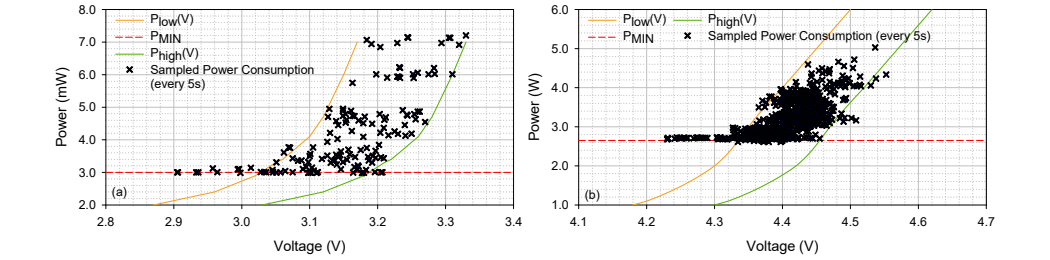

Fig. 24. Power-Voltage characteristics of the system over time, compared with the maximum efficiency point curves showing correct MPPT behaviour, for (a) the TI MSP430 MCU and (b) the ODROID XU-4 MP-SoC.

6.2.2 MPPT. Fig. 23 (a) shows how the MCU performs when powered by an EH supply over time. Here, the voltage V_C remains stable, remaining at, or very close to, $V_{\eta-max}$ for the duration of the test on the MCU showing correct MPPT behaviour. Fig. 23 shows the same, but this time for the MP-SoC. In this case, V_C again remains stable (around $V_{\eta-max}$), until 15:00 at which point the harvested power becomes insufficient to sustain power neutrality.

Fig. 24 shows the measured power consumption of the system over time, sampled every 5 seconds for (a) the MCU and (b) the MP-SoC platforms. Here, the power consumption remains almost entirely within the bounds of V_{high} and V_{low} where the time spent outside these bounds is generally where the harvested power supply is too low for the system to operate at the maximum efficiency voltage; this demonstrates correct MPP tracking behaviour. It can be observed that the points in Fig. 24 (a) are grouped in discrete operating levels, whereas the operating points in Fig. 24 (b) (for the MP-SoC) are more evenly distributed. This is due to the fact that the power consumption of the MCU is relatively constant at a given operating point however, on the MP-SoC, the power consumption varies more significantly even at a constant operating point due to the changing demands of the application and other Kernel tasks.

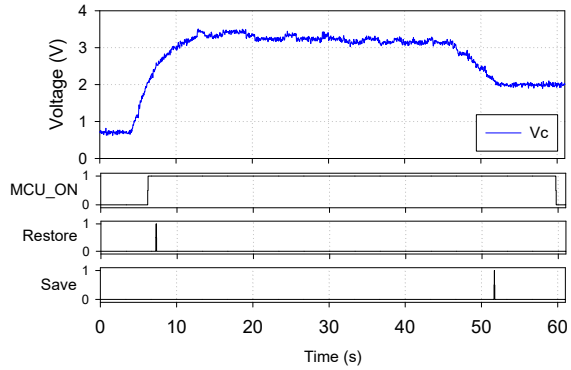

Fig. 25. Demonstration of transient operation on the MSP430 MCU.

Table 4. Efficiency comparisons against hardware MPPT.

MPPT Technique	System Efficiency (%)
Y. Kim *at al.* [40]	45-63%
Y. Wang *at al.* [39]	84-92%
C. Lu *at al.* [41]	33-40%
D. Brunelli *at al.* [20]	50-80%
Y. Y. Chou *at al.* [42]	<82%
Momentum	>90%

6.2.3 Transient Operation. Fig. 25 shows the transient property of *Momentum*, whilst running on the MSP430 MCU. Here, the system is on, however it remains in low-power mode until the voltage V_C reaches the restore threshold, $V_{restore}$ (2.3V in this case). Once the system's state has been restored, the power neutral tracking commences and continues until the voltage drops below the save threshold, V_{save} (2.1V in this case). After saving the system's state, the MCU is placed in low-power mode before the power outage, demonstrating correct TC behaviour.

6.3 Comparison Against State-of-Art

6.3.1 Comparison Against Hardware MPPT. There are two main metrics to evaluate the quality of an MPPT approach: the efficiency of the power conversion circuit (that typically includes DC-DC converter, MPP controller, energy buffer and voltage regulator), and the accuracy of the MPPT strategy. Table 4 compares the efficiency of the *Momentum* methodology, with established hardware-based MPPT approaches[39] (a converter-less PV power system with the MPPT that directly supplies power to the load without the power converters or the energy storage element), [40] (conventional MPPT baseline, which includes two cascaded converters and a supercapacitor), [41] (low-overhead MPPT algorithm for micro-scale solar energy harvesting systems), [20] (using an MPPT circuit which applies PWM control based upon an additional pilot-cell) and [42] (using a PFM control strategy).

Here, it can be observed that *Momentum* outperforms each of these existing approaches achieving a system efficiency greater than 90%. This is due to the fact that *Momentum* is not affected by the limited efficiency of power conversion circuitry, as it adopts an entirely software-based approach.

Table 5. Forward application execution when using *Momentum*, compared against existing approaches, for the MCU.

Performance Scaling Technique	Number of FFTs per second
Static Approach (*Hibernus*) [28]	1.07
Graceful Performance Modulation for PN TC Systems [9]	1.65
Momentum	1.83

Table 6. Forward application execution when using *Momentum*, compared with the state-of-art for the MP-SoC.

Performance Scaling Technique	Billions of Instructions Per Second
Linux Powersave	0.69
MP-SoC Power-neutral Approach [19]	1.17
Momentum	1.27

The small efficiency losses that are present are due to the additional PMU circuit used to enable power-neutrality, which could be built-in on-chip.

The accuracy of the MPPT strategy used in *Momentum* was also measured, and was found to track the corresponding MPP within ±5% for over 99% of the testing duration, comparable with other state-of-the-art MPPT algorithms.

6.3.2 Comparison Against Static and Power-neutral approaches. *Momentum* was also compared to other existing static approaches ([28] and default Powersave Linux governor) and power-neutral control schemes, [9] and [19], which are state-of-the-art power neutral control schemes for MCU and MP-SoC systems.

The results from these tests are shown in Table 5 (for the MCU) and Table 6 (for the MP-SoC). In both tests the same EH conditions (controlled irradiance condition) were used to evaluate performance whilst running *Momentum* and the previously reported approaches. *Momentum* outperformed these existing static and power neutral approaches, allowing the system to execute a higher number of instructions over the same amount of time due to the addition of dynamic MPPT. For the MCU running an FFT, the number of instructions executed was increased by 46% when compared to a static approach and 11% compared to a previous power-neutral approach. Despite the power required for memory access being dependent on the hardware configuration (e.g. type of memory), the proposed scheme significantly reduces the number of complete system interruptions, and hence the number of read and write operations to NVM. This has been previously demonstrated in [9], and Table 5 documents a further improvement when compared to this prior work.

For the MP-SoC running a face recognition algorithm, the number of instructions executed was increased by 42% when compared to the Linux Powersave governor, and 9% compared to a previous power-neutral approach.

7 CONCLUSIONS

In this paper, we have proposed *Momentum*, a broadly-applicable methodology for maximising forward application execution in energy harvesting computing systems, without introducing significant control overhead. The methodology combines three aspects: 1) a general approach for enabling power-neutral operation by dynamically scaling system performance, 2) a software-based MPPT scheme that benefits from the unique opportunity to modulate the voltage around which power-neutrality is achieved, such that the overall system efficiency is maximized, and 3) transient computing approaches to provide resilience against power outages. Performance scaling is provided through DFS and DPM, and operates through a hierarchical control approach. The methodology significantly reduces circuit complexity compared to existing energy-neutral systems, as it doesn't require complex power conversion or bulky energy storage components. Furthermore, compared to

alternative MPPT approaches, it requires no hardware to be added to the system. The methodology was validated experimentally on two different mobile (a low-power 8-core MP-SoC found in typical smart-phones) and embedded (an ultra low-power MCU operating from only the decoupling capacitance) platforms powered by PV energy harvesting. The results show that *Momentum* works correctly and effectively on both platforms, from both controlled energy sources and real energy harvesting devices, improving forward application execution by 11% when compared to existing power-neutral approaches, and 46% when compared to existing static approaches.

REFERENCES

[1] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *IEEE Communications Surveys Tutorials*, vol. 13, no. 3, pp. 443–461, Third 2011.

[2] F. Ongaro, S. Saggini, and P. Mattavelli, "Li-ion battery-supercapacitor hybrid storage system for a long lifetime, photovoltaic-based wireless sensor network," *IEEE Trans. Power Electron.*, vol. 27, no. 9, Sept 2012.

[3] A. Kansal *et al.*, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, Sept 2007.

[4] V. Raghunathan *et al.*, "Design considerations for solar energy harvesting wireless embedded systems," in *Int. Symp. on Information Processing in Sensor Networks (IPSN)*, Apr 2005, pp. 457–462.

[5] J. A. Paradiso and T. Starner, "Energy scavenging for mobile and wireless electronics," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 18–27, Jan 2005.

[6] A. Hammoud *et al.*, "Towards an implantable bio-sensor platform for continuous real-time monitoring of anti-epileptic drugs," in *38th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug 2016, pp. 2982–2985.

[7] S. Naderiparizi *et al.*, "Wispcam: A battery-free rfid camera," in *IEEE Int. Conf. on RFID (RFID)*, April 2015, pp. 166–173.

[8] P. D. Mitcheson *et al.*, "Energy harvesting from human and machine motion for wireless electronic devices," *Proceedings of the IEEE*, vol. 96, no. 9, pp. 1457–1486, Sept 2008.

[9] D. Balsamo *et al.*, "Graceful performance modulation for power-neutral transient computing systems," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 35, no. 5, pp. 738–749, May 2016.

[10] M. E. Salehi *et al.*, "Dynamic voltage and frequency scheduling for embedded processors considering power/performance tradeoffs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 10, pp. 1931–1935, Oct 2011.

[11] R. Ye and Q. Xu, "Learning-based power management for multi-core processors via idle period manipulation," in *17th Asia and South Pacific Design Automation Conference*, Jan 2012, pp. 115–120.

[12] G. Anastasi *et al.*, "Extending the lifetime of wireless sensor networks through adaptive sleep," *IEEE Trans. on Industrial Informatics*, vol. 5, no. 3, pp. 351–365, Aug 2009.

[13] S. Elmalaki *et al.*, "A case for battery charging-aware power management and deferrable task scheduling in smartphones," in *Proceedings of the 6th USENIX Conference on Power-Aware Computing and Systems*, ser. HotPower'14. Berkeley, CA, USA: USENIX Association, 2014, pp. 4–4.

[14] A. Das, B. M. Al-Hashimi, and G. V. Merrett, "Adaptive and hierarchical runtime manager for energy-aware thermal management of embedded systems," *ACM Trans. Embed. Comput. Syst.*, vol. 15, no. 2, pp. 24:1–24:25, Jan. 2016.

[15] Y. Wang *et al.*, "Solartune: Real-time scheduling with load tuning for solar energy powered multicore systems," in *Int. Conf. on Embedded and Real-Time Computing Systems and Applications*, Aug 2013, pp. 101–110.

[16] C. Moser *et al.*, "Real-time scheduling for energy harvesting sensor nodes," *Real-Time Systems*, vol. 37, no. 3, pp. 233–260, 2007.

[17] A. Gomez *et al.*, "Wearable, energy-opportunistic vision sensing for walking speed estimation," in *Sensors Applications Symposium*. Glassboro, New Jersey, USA: IEEE, Mar 2017.

[18] F. Xia *et al.*, "Towards power-elastic systems through concurrency management," *IET computers & digital techniques*, vol. 6, no. 1, pp. 33–42, 2012.

[19] B. J. Fletcher, D. Balsamo, and G. V. Merrett, "Power neutral performance scaling for energy harvesting MP-SoCs," in *Proc. Conf. Design, Automation & Test in Europe (DATE'17)*, March 2017.

[20] D. Brunelli *et al.*, "Design of a solar-harvesting circuit for batteryless embedded systems," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 56, no. 11, pp. 2519–2528, Nov 2009.

[21] B. Sahu and G. A. Rincon-Mora, "An accurate, low-voltage, cmos switching power supply with adaptive on-time pulse-frequency modulation (pfm) control," *IEEE Trans.on Circuits and Systems I: Regular Papers*, vol. 54, no. 2, pp. 312–321, Feb 2007.

[22] Y. Choi *et al.*, "Dc-dc converter-aware power management for low-power embedded systems," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 8, pp. 1367–1381, Aug 2007.

[23] A. S. Weddell, G. V. Merrett, and B. M. Al-Hashimi, "Photovoltaic sample-and-hold circuit enabling mppt indoors for low-power systems," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 59, no. 6, pp. 1196–1204, Jun 2012.
[24] A. Gomez *et al.*, "Dynamic energy burst scaling for transiently powered systems," in *Proc. Conf.Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2016, pp. 349–354.
[25] C. Lu *et al.*, "Efficient design of micro-scale energy harvesting systems," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 3, pp. 254–266, Sept 2011.
[26] C. Lu *et al.*, "Vibration energy scavenging system with maximum power tracking for micropower applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 11, pp. 2109–2119, Nov 2011.
[27] B. Lucia *et al.*, "Intermittent computing: Challenges and opportunities," in *Summit on Advances in Programming Languages (SNAPL)*, 2017.
[28] D. Balsamo *et al.*, "Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems," *IEEE Embedded Systems Letters*, November 2014, the underlying data is available from http://dx.doi.org/10.5258/SOTON/389749.
[29] B. Ransford, J. Sorber, and K. Fu, "Mementos: System support for long-running computation on rfid-scale devices," *SIGARCH Comput. Archit. News*, vol. 39, no. 1, pp. 159–170, Mar. 2011.
[30] H. Jayakumar *et al.*, "Quickrecall: A hw/sw approach for computing across power cycles in transiently powered computers," *J. Emerg. Technol. Comput. Syst.*, vol. 12, no. 1, pp. 8:1–8:19, Aug. 2015. [Online]. Available: http://doi.acm.org/10.1145/2700249
[31] K. Ma *et al.*, "Nonvolatile processor architectures: Efficient, reliable progress with unstable power," *IEEE Micro*, vol. 36, no. 3, pp. 72–83, May 2016.
[32] A. R. Arreola *et al.*, "Approaches to transient computing for energy harvesting systems: A quantitative evaluation," in *Proceedings of the 3rd International Workshop on Energy Harvesting & Energy Neutral Sensing Systems*, ser. ENSsys '15. ACM, 2015, pp. 3–8.
[33] H. Wang *et al.*, "A graphene oxide quantum dots embedded charge trapping memory with enhanced memory window and data retention," *IEEE Journal of the Electron Devices Society*, vol. 6, pp. 464–467, 2018.
[34] J. S. Meena *et al.*, "Overview of emerging nonvolatile memory technologies," *Nanoscale Research Letters*, vol. 9, no. 1, p. 526, Sep 2014.
[35] M. J. Walker *et al.*, "Accurate and stable run-time power modeling for mobile and embedded cpus," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 1, pp. 106–119, Jan 2017.
[36] S. Tiwari, "Webcam-face-detect," Oct 2016. [Online]. Available: https://github.com/shantnu/Webcam-Face-Detect
[37] S. Holmbacka *et al.*, "Accurate energy modeling for many-core static schedules with streaming applications," *Microprocessors and Microsystems*, vol. 43, pp. 14–25, Jun 2016.
[38] A. R. Arreola, D. Balsamo, G. V. Merrett, and A. S. Weddell, "Restop: Retaining external peripheral state in intermittently-powered sensor systems," in *Sensors*, 2018.
[39] Y. Wang *et al.*, "Storage-less and converter-less photovoltaic energy harvesting with maximum power point tracking for internet of things," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 2, pp. 173–186, Feb 2016.
[40] Y. Kim *et al.*, "Maximum power transfer tracking for a photovoltaic-supercapacitor energy system," in *Proceedings of the 16th ACM/IEEE International Symposium on Low Power Electronics and Design*, 2010, pp. 307–312.
[41] C. Lu *et al.*, "Low-overhead maximum power point tracking for micro-scale solar energy harvesting systems," in *25th International Conference on VLSI Design*, Jan 2012, pp. 215–220.
[42] Y. Y. Chou *et al.*, "Multi-input energy harvesting interface for low-power biomedical sensing system," in *2014 International Symposium on Next-Generation Electronics (ISNE)*, May 2014, pp. 1–2.