Maximizing the Lifetime of Embedded Systems Powered by Fuel Cell-Battery Hybrids

Jianli Zhuo, *Student Member, IEEE,* Chaitali Chakrabarti, *Senior Member, IEEE,* Kyungsoo Lee, *Student Member, IEEE,* Naehyuck Chang, *Senior Member, IEEE,* and Sarma Vrudhula, *Senior Member, IEEE*

***Abstract*—Fuel cell (FC) is a viable alternative power source for portable applications; it has higher energy density than traditional Li-ion battery and thus can achieve longer lifetime for the same weight or volume. However, because of its limited power density, it can hardly track fast fluctuations in the load current of digital systems. A hybrid power source, which consists of a FC and a Li-ion battery, has the advantages of long lifetime and good load following capabilities. In this work, we consider the problem of extending the lifetime of a fuel-cell-based hybrid source that is used to provide power to an embedded system which supports DVS (dynamic voltage scaling). We propose an energy-based optimization framework that considers the characteristics of both the energy consumer (the embedded system) and the energy provider (the hybrid power source). We use this framework to develop algorithms that determine the output power level of the FC and the scaling factor of the DVS processor during task scheduling. Simulations on task traces based on a real-application (Path Finder) and a randomized version demonstrate significant superiority of our algorithms with respect to a conventional DVS algorithm which only considers energy minimization of the embedded system.**

***Index Terms*—Dynamic voltage scaling (DVS), embedded system, fuel cell (FC), hybrid power, and task scheduling**

I. Introduction

ENERGY minimization is an important design criteria for embedded systems used in portable applications. In the past two decades, significant amount of work has been done in reducing the energy consumption from the system level all the way to the circuit level. Even though minimizing the energy consumption of the embedded system can improve the lifetime of the portable device, the steady increase in the computing/communicating abilities of these devices demand power sources with higher energy density, and energy management strategies that actively take into account the characteristics of the power source.

Among the different alternative energy sources (solar, fuel, etc.), a fuel cell (FC) is considered to be one of the most stable and viable sources. Typically, an FC uses hydrogen (H_2) and oxygen (O_2) to generate electrical power, and the by-products are water and heat. The basic characteristics of an FC are:

- High energy density: compared to widely used Li-ion batteries, FCs may deliver 4 to 6X more energy for the same size and weight.
- Slow and limited load following: the power range of the FC output is limited, and is much lower than the traditional Li-ion batteries.

While FCs have been extensively used in stationary applications with high energy density, such as power plants, and automotive applications, their application in portable embedded systems is only now being investigated. The two FC candidates for portable applications are DMFC (the direct methanol FC) [1] and PEMFC (polymer electrolyte membrane FC, also referred as proton exchange membrane FC) [2], [3]. An FC alone cannot meet the power demands of an embedded system because of the large and frequent fluctuations. So a practical solution is to use a hybrid power source where the FC provides the average power and a secondary energy source (which could be a rechargeable battery, or a super-capacitor [4], or a combination [5]) provides the variance in the power profile.

Most of the prior work on FC control has been in the context of hybrid automobiles. These include thermal, hydration and air-supply management polices [6], [7], FC models to capture the chemical kinetics, and mechanisms to control FC operations [6], [8], [9]. Control techniques that coordinate the operations of the FC, battery and super capacitor to achieve longer battery lifetime and higher system efficiency have also been proposed in [9], [10], [11], [12], [13]. These include a PI (Proportional-Integral) control based scheme in [9], an integrated system approach based on game theory in [10], policies for power distribution among FC/battery/ultra-capacitor in [11], and battery charge/discharge switching policies in [12], [13].

Unfortunately, techniques developed for hybrid automobiles cannot be directly applied to portable embedded systems. Automotive systems are reactive, and the required load current acts as a reference signal that the hybrid power system must track faithfully. However, in an embedded system, the tasks can be executed in any manner as long as they meet a specified performance level. Furthermore, the load current drawn by the embedded system is a function of parameters such as the processor supply voltage, the power state of the peripherals - all of which can be controlled.

In this work, we consider a portable platform where the hybrid power source is built with a sodium borohydride ($NaBH_4$)

Manuscript received February 2, 2007; revised September 21, 2007. This work was funded in part by the NSF grant (CSR-EHS 05059540), the Consortium for Embedded Systems, ASU, and LG Yonam Foundation.

J. Zhuo and C. Chakrabarti are with the Department of Electrical Engineering, Arizona State University, Tempe, AZ (email: chaitali@asu.edu).

K. Lee and N. Chang are with the Department of Computer Science and Engineering, Seoul National University, Seoul, Korea (email: naehyuck@elpl.snu.ac.kr).

S. Vrudhula is with the Department of Computer Science and Engineering, Arizona State University (email: vruduhula@asu.edu).

IEEE Digital Object Identifier TBD.

PEMFC system and a Li-ion battery. Here the embedded system consists of a DVS (dynamic voltage scaling) enabled processor and devices, such as disk, SDRAM memory, flash memory, etc. An embedded system where the devices are DPM (dynamic power management) enabled have been considered in [14], [15]. Our objective is to maximize the operating lifetime of of the FC, which is equivalent to minimize the fuel consumption. The fuel cell/battery hybrid power source introduces additional non-linearities that cannot be addressed by only considering the energy consumption of the embedded system.

This paper is an extension of our earlier work presented in [16], [17]. We first describe an energy-based optimization framework that considers the energy characteristics of the hybrid power source as well as the energy demands of the embedded system. We develop control algorithms based on the optimization framework for the case when the FC works at fixed output power level as well as for the case when it works at multiple power levels. When the FC works at fixed output power, Algorithm *fc_scale* is used to control the speed of the DVS processor subject to the deadline and battery constraints. When the FC works at multiple output levels, Algorithm *fc_scale_ctrl* controls not only the speed of the DVS processor but also the FC output power level, resulting in a significant increase in the FC lifetime. This is an extension of Our contributions are summarized below:

- Introduction of an FC based hybrid power source for embedded systems.
- Development of an optimization framework to minimize the energy consumption of the hybrid system.
- Development of FC-efficient voltage scaling algorithms *fc_scale* and *fc_scale_ctrl* based on the optimization framework.
- Performance validation with real-application based task traces and synthetic task traces.

The rest of the paper is outlined as follows. Section II describes the FC operation, FC-battery hybrid power source and embedded system powered by such a source. Section III introduces the energy optimization framework for the FC-battery hybrid system followed by static scheduling algorithms that enhance the lifetime of the system. The experimental results for both real-application-based task traces and synthetic task traces are given in Section IV. Section V concludes the paper.

II. Background

In this section, we first introduce the basics of the fuel cell/battery hybrid system, including the FC stack operation (Section II-A), the hybrid power source (Section II-B), and the FC system efficiency (Section II-C). Then we describe the energy flow of the embedded system powered by the hybrid source (Section II-D) followed by the definitions used in our optimization framework (Section II-E).

A. Fuel cell

FCs have higher energy density (so longer lifetime) and low environmental side-effect. Their energy densities are typically >2000 Wh/Kg compared to that of batteries which is typically <200 Wh/Kg [18]. Further more, FCs are clean power sources and emission free. Even though the chemical energy to electrical energy conversion efficiency in an FC is about only half of that of a Li-ion battery, the PEMFC package that we use here is expected to generate power 4X to 10X longer than a Li-ion battery package with the same size and weight [19], [20].

The core of the FC system is the FC stack. The FC uses H_2 and O_2 to convert chemical energy to electricity by an electrochemical process. At the anode, the reaction is given by $H_2 \rightarrow 2H^+ + 2e^-$. The H^+s move towards the cathode through the ion conducting electrolyte (membrane). The e^-s find a path to the cathode through the external circuit thus performing electrical work. At the cathode, the reaction is $2H^+ + 2e^- + \frac{1}{2}O_2 \rightarrow H_2O$.

The electrical energy produced by the FC is calculated by the difference between the Gibbs free energy of the product (H_2O) and the inputs (H_2 and O_2) [21]. According to the reaction $H_2 + \frac{1}{2}O_2 \rightarrow H_2O$ in a PEMFC, the energy in "per mole" form is given by

$$\Delta\overline{g}_f = (\overline{g}_f)_{H_2O} - (\overline{g}_f)_{H_2} - \frac{1}{2}(\overline{g}_f)_{O_2}, \quad (1)$$

where $(\overline{g}_f)_X$ is the Gibbs free energy of X per mole. When there is no loss in the energy conversion, the open circuit voltage is

$$V_{open} = \frac{-\Delta\overline{g}_f}{2F}, \quad (2)$$

where $F = 96485.3415$ C/mol is the Faraday constant, and the factor 2 is because one mole of H_2 produces 2 moles of electrons. The open circuit voltage of a single cell is typically around 1.2 V. The real open circuit voltage is less than the theoretical no loss value calculated by Equation (2). As the current density increases, the open circuit voltage reduces as shown in Fig. 1. This reduction is due to activation losses, fuel crossover and internal current, ohmic losses, and mass transport or concentration losses [21], [22].

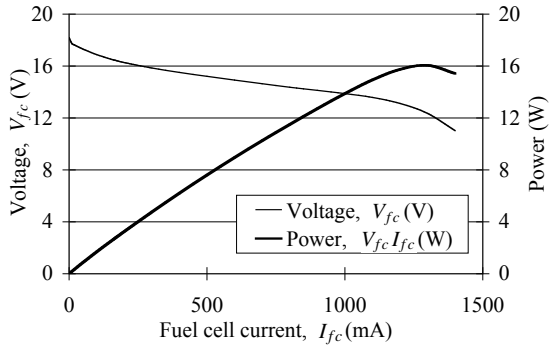


Fig. 1. Measured I-V-P curves for the BCS 20 W, 20 stack, room-temperature PEMFC (@2 psig H_2 pressure).

The FC output current, I_{fc}, can be increased by increasing the area of the membrane. The FC output voltage, V_{fc}, can be multiplied by stacking multiple cells. Fig. 1 describes the polarization (I-V and I-P) curves for the BCS 20 W, 20 stack, room-temperature PEMFC applying 2 psig H_2 pressure. The

power $V_{fc} \times I_{fc}$ first increases corresponding to the region where the voltage curve has a mild slope, and then decreases corresponding to the region where the voltage drops rapidly. The FC output power can be varied in a certain range referred as the *load following range* if we control the fuel input flow rate. If the FC works at fixed output power, the operating point is set at $\frac{2}{3}$ of the maximum power based on stability and cost-efficiency considerations. Note that FC output power is also associated with temperature, pressure and humidity [22].

B. Hybrid power source

In general, an FC has "slow" electrodes, i.e., its reaction does not occur as quickly as the load current varies, and so the electrodes polarize and the FC loses voltage. As a result, for applications where the load current fluctuates, the FC has to be supported by a battery which has "fast" electrodes and is a much better load follower. A *hybrid* system consisting of an FC and a battery provides the high energy density of FCs and the high power density of batteries [13], [23]. A hybrid system also results in both components being smaller. Fig. 2 shows the advantage of using a hybrid power source over a fuel-cell-only source or a battery-only source. When the load current variance is very large (e.g., the maximum load current is 3X of the average load current), a fuel-cell-only system has be over-designed in order to ensure system safety. This would result in an increase both the size and weight of the FC. The battery can handle the current variance very well, but it cannot last as long as an FC with the same size due to its low energy density. An FC in conjunction with a small size battery is more efficient with respect to power/energy density and size/weight.

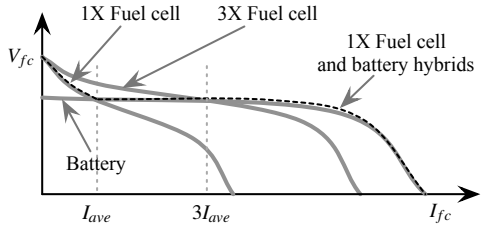


Fig. 2. Advantage of using a hybrid power source.

The structure of the proposed FC-battery hybrid power source is shown in Fig. 3. It consists of the following components:

Fuel cell stack: As described in Section II-A, the FC stack is the core of the FC system. The anode is fed with H_2 generated from the fuel processor, and the cathode is fed with O_2 from the ambient air. The FC stack is capable of self-humidification, and the stack temperature is maintained at around 45°C for high efficiency.

Fuel processor: As H_2 is difficult to store and carry for human-portable applications, on-site H_2 generation with safe chemicals is very important. We choose $NaBH_4$ as the fuel for the PEMFC because it is stable, can easily be converted to H_2 by the catalyst Rutherium[24], and generates wet and warm H_2 that is conducive to efficient FC operation. The chemical reaction for H_2 generation is as follows:

$$NaBH_4 + 2H_2O \xrightarrow{Ru} 4H_2 + NaBO_2. \quad (3)$$

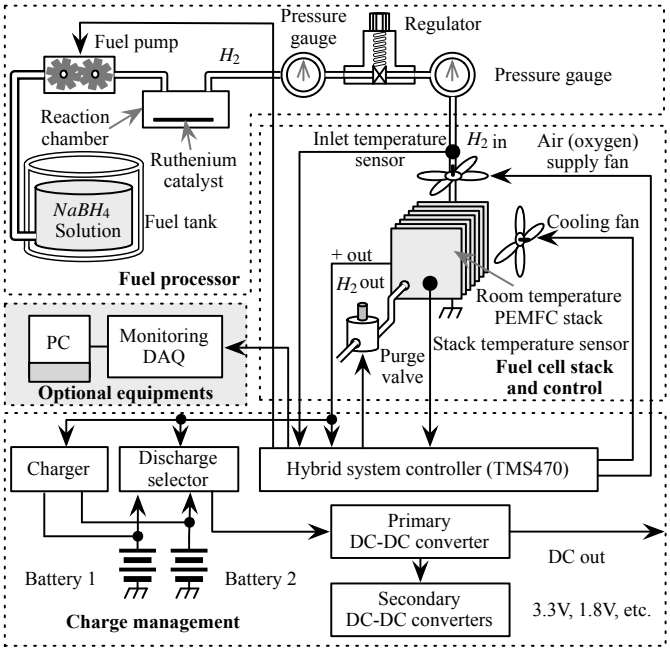


Fig. 3. FC-battery hybrid power source.

The amount of $NaBH_4$ delivered through the fuel pump determines the amount of H_2 generated.

Purge valve: The room-temperature 20 W FC stack requires 2 to 3 psig H_2 pressure at the anode end. While this is achieved by a dead-end configuration, a purge valve (solenoid valve) is used to make the channel periodically open-end and flush out the accumulated H_2O.

Cathode fan: Since 2 to 3 psig H_2 pressure is formed at the anode side, a similar amount of O_2 pressure is required at the cathode side. This pressure is generated by a blower fan.

Cooling fan: As the efficiency of the FC stack is temperature dependent, the stack temperature must be maintained at a particular value for maximum efficiency. We use a cooling fan for robust operation against the ambient temperature variation.

DC-DC converter: DC-DC converters are mandatory components for voltage regulation against load current and input voltage variation. We use a primary converter along with secondary converters to make the FC system a stand-alone power source. The primary DC-DC converter is chosen to be LTC1625 which is a switching voltage regulator and has an efficiency of $> 85\%$ for currents above 100 mA and low efficiency otherwise. The secondary DC-DC converters are selected depending on the voltage requirement of the embedded system.

Battery: The rechargeable Li-ion battery is used as the secondary power source. Since frequent switching between charging and discharging mode reduces the cycle-lifetime of the battery significantly, we use two batteries where one is in charging mode and the other is in discharging mode.

Microprocessor: An TI TMS470 microprocessor controls the operation of the controller components such as the fuel pump, purge valve, cathode fan, cooling fan, etc. In addition, it supervises the charge management including the battery charging/discharging.

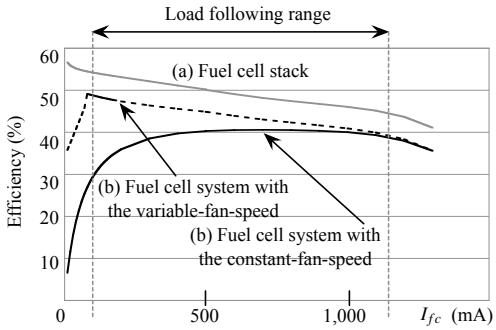


Fig. 4. Measured FC stack efficiency and system efficiency.

C. Fuel cell system efficiency

The FC system efficiency, η_s, is defined as the FC system output power, P_F, divided by Gibbs free energy per second (denoted as ΔE_{Gibbs}). ΔE_{Gibbs} is the product of the fuel flow rate and $\Delta \bar{g}_f$ in Equation (1). Since the fuel flow rate is proportional to the FC stack current, I_{fc}, we have $\Delta E_{Gibbs} \propto I_{fc}$, and η_s is given by

$$\eta_s = \frac{P_F}{\Delta E_{Gibbs}} \propto \frac{P_F}{I_{fc}}. \tag{4}$$

The efficiency, η_s, is affected by the PEMFC stack efficiency, the DC-DC converter efficiency and the loss due to the controller current.

The PEMFC stack efficiency is given by the stack output power divided by ΔE_{Gibbs}, i.e., $\frac{V_{fc} \times I_{fc}}{\Delta E_{Gibbs}} \propto V_{fc}$. It follows the same trend as the stack voltage V_{fc} (see Fig. 1). The measured PEMFC stack efficiency is shown in Fig. 4(a).

The loss due to the controller power comes from the cathode air blow fan, cooling air blow fan, purge valve (solenoid), microcontroller, etc. The loss is dominated by the cathode air blow fan (0.4 W) and the cooling air blow fan power (0.4 W). The purge valve, the microcontroller, and the fuel pump consume very little power.

Fig. 4 plots η_s as a function of the FC output current. When the cathode air blow fan and the cooling fan operate at the optimal speed that is determined by the output current, the system efficiency is shown in Fig. 4(b). A simpler mechanism is to operate the cathode air blow fan at constant speed (regardless of the FC current). In such a system, η_s is virtually a constant within the load following range. In fact, the system efficiency is around 39% with a variation of $\pm 4\%$ in the operation range 0.3 A - 1.2 A, as shown in Fig. 4(c). To simplify the analytical analysis, we present the optimization framework for the case when the system efficiency is a constant. The framework can be modified to handle the case when the system efficiency is not a constant, and we include an experimental validation for this case.

D. System overview

The main blocks of the embedded system powered by hybrid power source are shown in Fig. 5. The hybrid power source consists of a PEMFC system with the output power P_F, and a Li-ion battery. The charging and discharging of the battery is controlled by the charge management system (CMS). At the consumer end, there is a DVS processor based embedded system whose load power, P_L, can be changed by DVS. When $P_F < P_L$, the Li-ion battery discharges to provide an additional power amount of $P_L - P_F$ to the embedded system; when $P_F > P_L$, the Li-ion battery is charged. If the battery gets fully charged and $P_F > P_L$, the excess power is dissipated through the bleeder circuit. Note that in FC systems that support load following, P_F can be decreased by adjusting the fuel flow rate, thereby reducing the bleeder current.

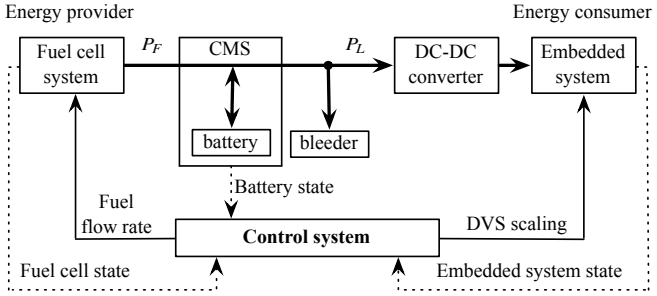


Fig. 5. The schematic view of the embedded system powered by the hybrid power source

The control system shown in Fig. 5 is in charge of optimizing the energy performance of the hybrid system based on the state information of the embedded system, the battery and the FC. If the FC works at fixed power level (fixed fuel flow rate, and thus P_F is a constant), the control system only controls the DVS policy of the embedded system. If the FC supports load following (with fuel flow rate control), the control system jointly controls the DVS policy of the embedded system and the FC system output power.

The energy flow of the hybrid system is shown in Fig. 6. At the energy provider end, E_{fc} is the energy provided by the

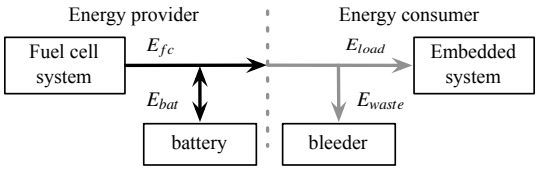


Fig. 6. The energy flow of the system shown in Fig. 5

FC system, and E_{bat} is the energy provided by the battery. The battery works as an energy buffer. If the initial state of the battery is B^{ini} and the final state is B^{end}, then the energy provided by the battery is $E_{bat} = B^{ini} - B^{end}$ (if $E_{bat} < 0$, the battery stores energy). At the energy consumer end, E_{load} is the energy consumed by the embedded system, and E_{waste} is the energy wasted through the bleeder. Recall that E_{waste} occurs when $P_F > P_L$ but the battery is fully charged. The energy flow is summarized by the following equation.

$$E_{total} = E_{fc} + E_{bat} = E_{load} + E_{waste}. \tag{5}$$

The operational lifetime of the the hybrid system is determined by the fuel consumption of the FC. Since a given volume (or weight) of fuel can provide a certain amount of energy E_{Gibbs}, if the amount of E_{Gibbs} for completing a certain number of tasks is less, the lifetime of the FC becomes longer. So the objective of extending the lifetime of the FC is transformed to minimizing E_{Gibbs} during task execution. Since

we assume that the FC system efficiency is constant, E_{Gibbs} is proportional to $E_{fc}+E_{bat}$. Thus our objective transforms to minimizing $E_{total}=E_{fc}+E_{bat}=E_{load}+E_{waste}$.

E. Definitions

We begin with the notations that have been used in the rest of the paper. The hybrid power source is characterized by the FC power, P_F, that has a value in the range $[P_F^{min}, P_F^{max}]$, and the energy capacity of the Li-ion battery, B^{max}. For a task T_k, the FC power is $P_{F,k}$ (or simply P_F when it is constant), the battery energy state is B_k^{ini} at the beginning and B_k^{end} at the end of the execution of task T_k.

The power model of load power P_L is expressed as

$$P_L = C \cdot V_{dd}^2 \cdot f + P_{on} + V_{dd} \cdot I_{leakage}, \quad (6)$$

where the first term on the right hand side is the dynamic power consumption, the second term is the intrinsic power, and the third term is the leakage power. Both P_{on} and $I_{leakage}$ are assumed to be constant and do not change with voltage scaling [25], [26]. The relation between frequency and the supply voltage is $f \propto \frac{(V_{dd}-V_t)^\gamma}{V_{dd}}$, where γ has a value between 1 to 2, and V_t is the threshold voltage. Here, we assume $\gamma=2$ and $V_{dd} \gg V_t$, and so $f \propto V_{dd}$.

Let s_k be the frequency scaling factor while executing T_k, and $s_k \geq 1$. Note that $s_k=1$ corresponds to the highest operating frequency. The task execution time is then $s_k \times \tau_k$, where τ_k is the worst case execution time at the highest frequency. The total power consumption of T_k when scaled by s_k, $P_{L,k}(s_k)$ can be rephrased as

$$P_{L,k}(s_k) = s_k^{-3}\alpha_1 P_{L,k}(1) + \alpha_2 P_{L,k}(1) + s_k^{-1}(1-\alpha_1-\alpha_2)P_{L,k}(1), \quad (7)$$

where $P_{L,k}(1)$ is the total power consumption at $s_k=1$, α_1 is the ratio of dynamic power to total power, and α_2 is the ratio of intrinsic power to the total power at $s_k=1$.

The metric that is used to measure the performance of difference policies is the total energy consumption of the system, E_{total}. The scaling factor which minimizes E_{total} for task T_k (FC-battery hybrid system aware DVS) is called the optimal scaling factor, s_k^{opt}. The scaling factor which only considers minimizing E_{load} (conventional DVS) is called s_k^{load}. Table I provides a comprehensive list of all the parameters.

TABLE I
DEFINITION OF THE FC SYSTEM PARAMETERS

E_{Gibbs}	Gibbs free energy, provided by the fuel consumption
ΔE_{Gibbs}	Gibbs free energy per unit time
V_{fc}	FC stack voltage
I_{fc}	FC stack current
η_s	FC system efficiency
T_k	$k-th$ task in the task profile
$P_{F,k}$	FC system output power for task T_k
P_F	FC system output power when set to a constant value
Ω	$\Omega=[P_F^{min}, P_F^{max}]$, the FC load following range
s_k	voltage/frequency scaling factor of T_k, $s_k \geq 1$
τ_k	worst case execution time of T_k
$P_{L,k}(s_k)$	the load power when task T_k is scaled by s_k
α_1	ratio of the dynamic power when s_k=1
α_2	ratio of the intrinsic power when s_k=1
B^{max}	energy capacity of the battery
B_k^{ini}	battery energy state when task T_k starts
B_k^{end}	battery energy state when task T_k finishes
E_{load}	energy consumed by the embedded system load
E_{waste}	energy wasted through bleeder bypass
E_{total}	total energy consumption, given by $E_{load}+E_{waste}$
E_{fc}	energy provided by the FC system
E_{bat}	energy provided by the battery
s_k^{opt}	scaling factor which minimizes E_{total} for task T_k
s_k^{load}	scaling factor which minimizes E_{load} for task T_k

III. ENERGY OPTIMIZATION OF THE HYBRID SYSTEM

In this section, we present an energy optimization framework for the embedded system powered by an FC-battery hybrid source. We first introduce a conventional energy minimization algorithm, *en_scale*, which aims at minimizing E_{load}, the energy consumption of the embedded system (Section III-B). Then we present the algorithm *fc_scale* which minimizes the total energy consumption, E_{total}, when the FC has a fixed output level (Section III-C). Next we present the optimization framework and corresponding algorithm *fc_scale_ctrl* when the FC operates at multiple output levels (Section III-D).

To simplify the analysis, we make the following assumptions: (1) The FC system has constant efficiency as described in the Section II-C, (2) the battery charge management system has 100% efficiency, (3) the task sequence is determined apriori, we only consider voltage/frequency assignment, and (4) the load power and the FC output power do not change during the execution of a single task.

A. Motivational example

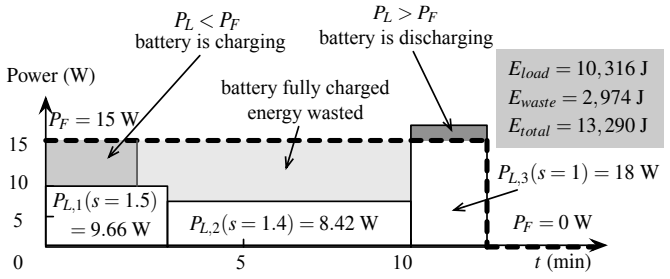
Consider a DVS system whose frequency can be scaled from 1 to 2 in steps of 0.1. The FC load following range is [4 W, 15 W]. The battery capacity is $B^{max}=1500$ J and the initial state of the battery is $\frac{1}{2}B^{max}$. The task configuration is given in Table II.

TABLE II
TASK PARAMETERS OF THE MOTIVATIONAL EXAMPLE

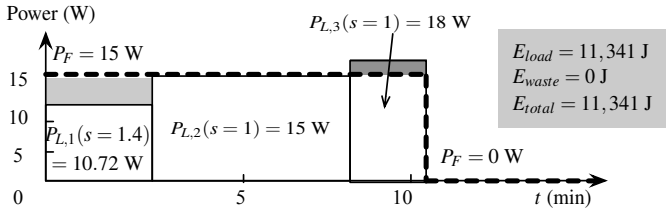
T_k	τ_k	$P_L(1)$	α_1	α_2	D
T_1	2 min	20 W	0.64	0.16	12 min
T_2	5 min	15 W	0.60	0.20	12 min
T_3	2 min	18 W	0.25	9.55	12 min

Case I – E_{load}-efficient DVS: We first apply the conventional energy minimization policy on the embedded system, that is, minimizing E_{load}. For the given task specifications, $s_1=1.5$, $s_2=1.4$, and $s_3=1$. The corresponding load power values are $P_{L,1}=9.66$ W, $P_{L,2}=8.42$ W and $P_{L,3}=18$ W, and consequently $E_{load}=10,316$ J. If the FC output power is fixed, $P_F=15$ W, there is wasted energy $E_{waste}=2,974$ J, and then the total energy consumption is $E_{total}=13,290$ J. The power profile is shown in Fig. 7(a).

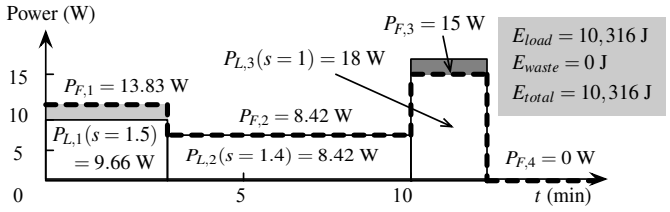
Case II – E_{total}-efficient DVS with fixed P_F: When the FC output power is fixed at $P_F=15$ W, we may set $s_1=1.4$



(a) E_{load} minimization: Algorithm *en_scale*



(b) E_{total} minimization with fixed P_F: Algorithm fc_scale



(c) E_{total} minimization with variable P_F: Algorithm fc_scale_ctrl

Fig. 7. Motivational example: (a) E_{load} minimization, (b) E_{total} minimization with fixed P_F, and (c) E_{total} minimization with variable P_F.

($P_{L,1} = 10.72$ W) and $s_2 = s_3 = 1$. The corresponding power profile and energy metrics are shown in Fig. 7(b). Now $E_{load} = 11,341$ J which is about 10% higher than that of Case I. However since E_{waste} has dramatically reduced to 0, $E_{total} = 11,341$ J, which is 15% lower than that of Case I.

Case III – E_{total}-efficient DVS with variable P_F: Now if we control both P_F and P_L, we can get the profile shown in Fig. 7(c). For task T_1, P_F is set to 13.83 W, and P_L is scaled to 9.66 W; for task T_2, P_F is set to 8.42 W, and P_L is scaled to 8.42 W; and for task T_3, P_F is set to 15 W, and P_L is 15 W. In this particular example, the load energy is only 10,316 J which is the minimal value according to Case I, and much lower than that of Case II. Since $E_{waste} = 0$, E_{total} reduces to 10,316 J, which is 22.4% lower than that of Case I, and 9.1% lower than that of Case II. Thus the lifetime of the FC can be extended significantly if the FC supports load following and the algorithm has the capability to exploit this feature.

B. E_{load}-efficient DVS: Algorithm en_scale

We first describe an algorithm for minimizing E_{load}. We include this for the sake of completeness and also to provide a baseline algorithm to compare against algorithms that have been optimized for FC operation.

E_{load} minimization for a single task: When there is only one task T_k, the load energy is calculated by

$$E_{load}(s_k) = P_{L,k}(s_k) \times s_k \times \tau_k. \quad (8)$$

Since $P_{L,k}(s_k) = s_k^{-3}\alpha_1 P_{L,k}(1) + \alpha_2 P_{L,k}(1) + s_k^{-1}(1 - \alpha_1 - \alpha_2)P_{L,k}(1)$ according to Equation (7), the scaling factor which minimizes E_{load} is given as $s_k^{load} = \sqrt[3]{\frac{2 \times \alpha_1}{\alpha_2}}$ if there is no deadline constraint. If s_k^{max} is the maximum scaling factor determined by the deadline constraint, then the scaling factor s_k^{load} has to be bounded by s_k^{max}.

E_{load} minimization for a sequence of tasks: Let us assume that there are n tasks, $(T_1, T_2, \cdots, T_n)$; all tasks arrive at time 0 and share the same deadline D. The load energy minimization is achieved by

$$\text{minimize: } E_{load}(s_1, \cdots, s_n) = \sum_{k=1}^{n} \big(P_{L,k}(s_k) \cdot s_k \tau_k\big), \quad (9)$$

$$\text{subject to: } \sum_{k=1}^{n} (s_k \times \tau_k) \le D. \quad (10)$$

We can analytically determine $\{s_1^{load}, s_2^{load}, \cdots, s_n^{load}\}$ that minimize $E_{load}(s_1, \cdots, s_n)$ by the Lagrange multiplier method. Since the objective function, $E_{load}(s_1, \cdots, s_n)$, is a sum of convex functions, $E_{load}(s_k)$, and the deadline constraint is also a convex function, we introduce a Lagrange multiplier λ and construct a function $f(s_1, s_2, \cdots, s_n)$ such that

$$f(s_1, \cdots, s_n) = E_{load}(s_1, \cdots, s_n) - \lambda\Big((\sum_{k=1}^{n} s_k \tau_k) + x^2 - D\Big). \quad (11)$$

We include x^2 in the function because the deadline constraint is not a equation.

We calculate the partial derivative of $f(s_1, s_2, \cdots, s_n)$ with respect to each s_k, x and λ, and set the partial differential functions to value zero.

$$\frac{\partial f}{\partial s_k} = \Big(\big(\alpha_2 - 2\alpha_1 s_k^{-3}\big)P_{L,k}(1) - \lambda\Big)\tau_k = 0, \forall k \le n, \quad (12)$$

$$\frac{\partial f}{\partial \lambda} = \sum_{k=1}^{n} (s_k \times \tau_k) + x^2 - D = 0, \quad (13)$$

$$\frac{\partial f}{\partial x} = 2 \times x \times \lambda = 0. \quad (14)$$

The solution of Equation (14) can be either $\lambda = 0$ or $x = 0$. If $\lambda = 0$, then the simultaneous equations in Equation (12) have an unconstrained solution: $s_k = \sqrt[3]{\frac{2 \times \alpha_1}{\alpha_2}} = s_k^{load}$. If $\lambda \neq 0$ and $x = 0$, the solution $(s_1, s_2, \cdots, s_n, \lambda)$ for the $n+1$ functions in Equations (12) and (13) exists, but it is difficult to find analytically. However, Equation (12) tells us that the value of $(s_1, s_2, \cdots, s_n)$ that minimizes the objective function should satisfy $\big(P_{L,k}(s_k) \times s_k\big)'_{s_k} = \big(P_{L,j}(s_j) \times s_j\big)'_{s_j} \le 0, \forall j, k \le n$. If $\alpha_1 : \alpha_2$ is identical for all tasks, s_k^{load} is also identical for all tasks and the value can be obtained by evenly distributing the slack to all tasks. When $\alpha_1 : \alpha_2$ is different from each other, we can find these values numerically. For the load energy minimization of tasks with individual deadlines, please refer to the algorithms presented in [27].

C. E_{total}-efficient DVS with fixed P_F: Algorithm fc_scale

As mentioned earlier, in order to extend the lifetime of the FC, we should minimize $E_{total} = E_{load} + E_{waste}$. The proposed

fuel-efficient scaling algorithm *fc_scale* minimizes E_{total} and is built on top of Algorithm *en_scale* which minimizes E_{load}. In this section, we consider the case when the FC works at fixed output power level, P_F.

For task T_k, if $P_F > P_{L,k}(s_k)$, ideally the battery can be charged to $B_k^{ini} + \big(P_F - P_{L,k}(s_k)\big) \times s_k \times \tau_k$. Since the battery charge cannot exceed B^{max}, we have

$$B_k^{end} = \min\Big(B^{max}, B_k^{ini} + \big(P_F - P_{L,k}(s_k)\big) \times s_k \times \tau_k\Big). \quad (15)$$

The wasted energy occurs after the battery is fully charged, and is calculated by

$$E_{waste} = \max\Big(0, B_k^{ini} + \big(P_F - P_{L,k}(s_k)\big) \times s_k \times \tau_k - B^{max}\Big). \quad (16)$$

When $P_F < P_{L,k}(s_k)$, the battery is discharged to $B_k^{ini} - \big(P_{L,k}(s_k) - P_F\big) \times s_k \times \tau_k$. In order to ensure the feasibility, we include a battery constraint:

$$\text{Battery constraint: } \big(P_{L,k}(s_k) - P_F\big) \times s_k \times \tau_k \le B_k^{ini}. \quad (17)$$

The above battery constraint will determine the lower bound of the feasible scaling factor values, s_k^{min}. Note that we have to set $s_k \ge s_k^{min}$, otherwise the battery will die before T_k completes.

We determine s_k^{opt} of the tasks one by one, in the order in which they were scheduled. Specifically, s_k^{opt} is determined by considering s_k^{load} (the scaling factor that minimizes E_{load}), s_k^{min}, and E_{waste}. Fig 8 describes the algorithm flow.

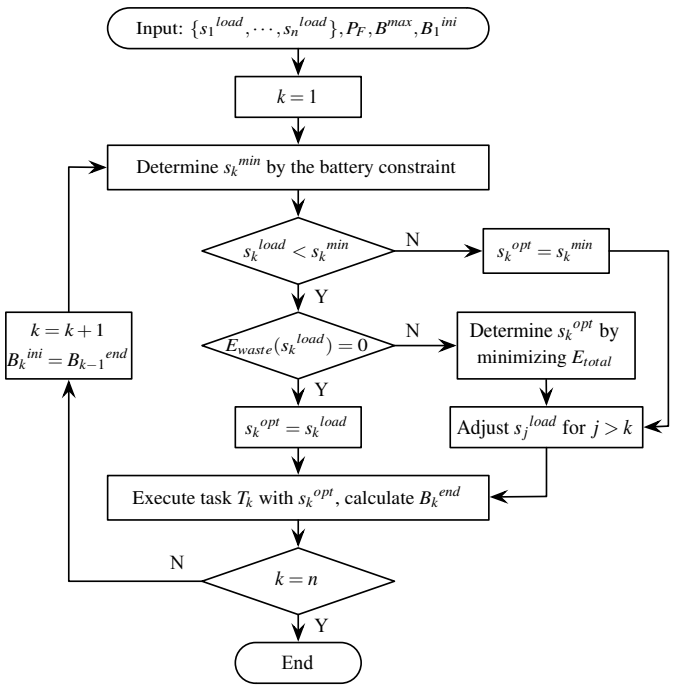


Fig. 8. Flow chart of Algorithm *fc_scale*.

- If $s_k^{load} < s_k^{min}$, we have to set $s_k^{opt} = s_k^{min}$ to satisfy the battery constraint. Note that $E_{waste} = 0$ because the battery is in discharge mode. Since now $s_k^{opt} > s_k^{load}$, it not only increases the load energy consumption of T_k, but also requires additional slack time from other tasks. So we have to adjust the pre-set s_j^{load} for all future tasks T_j where $j \ne k$ to compensate for the extra slack taken by T_k. The slack re-distribution can be done by the same method as Algorithm *en_scale*.
- If $E_{waste}(s_k^{load}) > 0$ which means $P_{L,k}(s_k^{load})$ is too low, we first set $s_k = s_k^{load}$ and then reduce s_k until $E_{total}(s_k)$ is not decreasing anymore. Since E_{total} is a convex function, there exists such a scaling factor; we refer to it as s_k^{opt}. It is obvious that the battery constraint is fulfilled in this case because the battery is charged. There is an extra slack released from T_k since $s_k^{opt} < s_k^{load}$, which can be absorbed by the remaining tasks.
- Otherwise we set $s_k^{opt} = s_k^{load}$, because in this case $s_k^{load} \ge s_k^{min}$ and $E_{waste}(s_k^{load}) = 0$.

The energy state of the battery, B_k^{end}, is updated and the process repeated for the next task.

Illustrative example: We illustrate how to calculate s_k^{opt} by Algorithm *fc_scale* using the motivational example shown in Fig 7. The scaling factors determined by Algorithm *en_scale* are $s_1^{load} = 1.5$, $s_2^{load} = 1.4$ and $s_3^{load} = 1$ (s_1^{load}, $s_2^{load} \le s_k^{load} = \sqrt[3]{\frac{2 \times \alpha_1}{\alpha_2}}$ due to the deadline constraint). When we set $s_1 = s_1^{load} = 1.5$, we have $P_L = 9.66$ W, $E_{load} = 1{,}738.7$ J, $E_{waste} = 750 + (15 - 9.66) \times 1.5 \times 2 \times 60 - 1500 = 211.3$ J, and finally $E_{total} = 1{,}950$ J.

Since E_{waste} is not zero, we try to decrease the scaling factor to find the minimum value of E_{total}. As shown in Table III, we get $s_1^{opt} = 1.4$, $P_{L,1}(s_k^{opt}) = 10.72$ W. The corresponding energy metrics are $E_{load} = 1{,}801.3$ J, $E_{waste} = 0$ and $E_{total} = 1{,}801.3$ J. Task T_1 finishes at 2.8 min, and $B_1^{end} = 750 + (15 - 10.72) \times 1.4 \times 2 \times 60 = 1{,}366.32$ J. We apply the same method on T_2 and T_3 one by one, and finally achieve the profile shown in Fig 7(b).

TABLE III
DETERMINING THE OPTIMAL SCALING FACTOR FOR TASK T_1 IN ALGORITHM *fc_scale* FOR THE TASK PROFILE SHOWN IN FIG. 7(*b*)

Scaling factor	1.5	1.4	1.3
$P_{L,1}$ (W)	9.66	10.72	12.10
E_{load} (J)	1738.70	1801.30	1888.10
E_{waste} (J)	211.30	0.00	0.00
E_{total} (J)	1950.00	**1801.30**	1888.10

D. E_{total}-efficient DVS with variable P_F: Algorithm fc_scale_ctrl

Next consider the case when the FC supports load following. For such a system, E_{load}, E_{waste}, and the battery constraint are given by

$$E_{load}(s_k) = P_{L,k}(s_k) s_k \tau_k, \quad (18)$$

$$E_{waste}(P_{F,k}, s_k) = \max\big(0, B_k^{ini} + \big(P_{F,k} - P_{L,k}(s_k)\big) s_k \tau_k - B^{max}\big), \quad (19)$$

$$\text{Battery constraint: } \big(P_{L,k}(s_k) - P_{F,k}\big) s_k \tau_k \le B_k^{ini}. \quad (20)$$

E_{load} is independent of the FC system output power $P_{F,k}$, and is minimized by scaling factor s_k^{load} which is given by Algorithm *en_scale*. If we can find $P_{F,k}$ such that $E_{waste}(P_{F,k}, s_k^{load}) = 0$, $E_{total}(P_{F,k}, s_k^{load})$ can be minimized. To achieve $E_{waste}(P_{F,k}, s_k^{load}) = 0$, the value of $P_{F,k}$ has to satisfy $\big(P_{F,k} - P_{L,k}(s_k^{load})\big) \times s_k^{load} \times \tau_k \le B^{max} - B_k^{ini}$. The value of $P_{F,k}$

should also satisfy the charge constraint in Equation (20). So the range of $P_{F,k}$ is $\Phi_k = \left[P_{F,k}^{low}, P_{F,k}^{high}\right]$ where

$$P_{F,k}^{low} = P_{L,k}(s_k^{load}) - \frac{B_k^{ini}}{s_k^{load} \times \tau_k}, \quad (21)$$

$$P_{F,k}^{high} = P_{L,k}(s_k^{load}) + \frac{B^{max} - B_k^{ini}}{s_k^{load} \times \tau_k}. \quad (22)$$

The value of $P_{F,k}$ should be chosen by considering both the range Φ_k and the load following range Ω. (Recall that $\Omega = [P_F^{min}, P_F^{max}]$). If Φ_k overlaps with Ω ($\Phi_k \bigcap \Omega \neq \emptyset$), $P_{F,k}$ could be set to any value in the overlapped region $\Phi_k \bigcap \Omega$. In our algorithm, we choose the largest value in $\Phi_k \bigcap \Omega$ since that would allow the battery to charge more and would relax the battery constraint of T_{k+1}; When there is no overlap such that $\Phi_k \bigcap \Omega = \emptyset$, we set $P_{F,k}$ to P_F^{min} when $P_F^{min} > P_{F,k}^{high}$, or P_F^{max} when $P_F^{max} < P_{F,k}^{low}$, and re-calculate s_k^{opt} by minimizing $E_{total}(P_{F,k}, s_k)$ using the the same flow as that in the case when the FC output if fixed. The flow of the algorithm *fc_scale_ctrl* is shown in Fig 9.

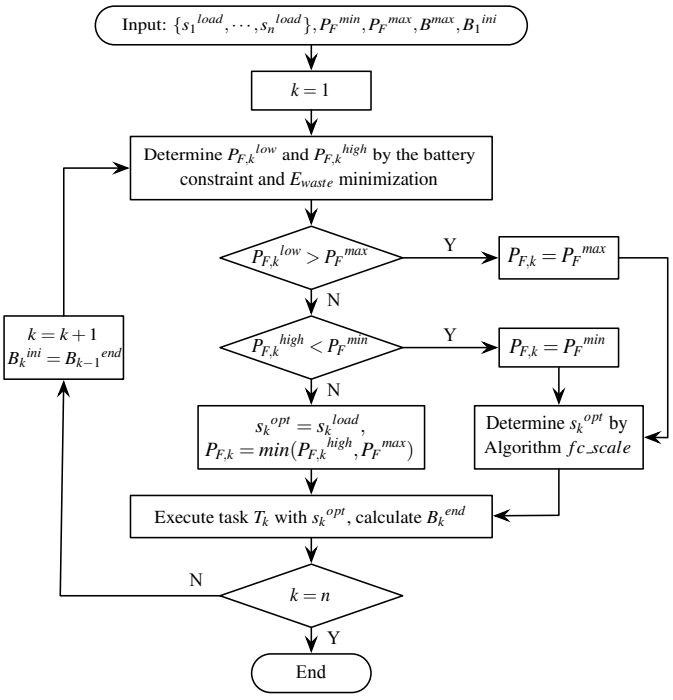


Fig. 9. Flow chart of Algorithm *fc_scale_ctrl*.

Illustrative example: We illustrate how to calculate s_k^{opt} by Algorithm *fc_scale_ctrl* using the motivational example shown in Fig 7. By Algorithm *en_scale*, $s_1^{load} = 1.5$, $P_L = 9.66$ W and the range of $P_{F,k}$ is set to [5.49 W, 13.83 W] according to Equations (21) and (22). Since the range is supported by the load following range [4 W, 15 W], we choose $P_F = 13.83$ W for T_1. Task T_1 finishes at 3 min, and $B_1^{end} = 1,500$ J (fully charged without wasted energy). We apply the same method on T_2 and T_3, and finally get the profile shown in Fig 7(c).

IV. Experimental result

In this section, we compare the performance of the proposed algorithms with respect to the energy metrics, E_{load}, E_{waste} and E_{total}. We first apply the algorithms on task traces based on a real application, Path Finder, and then apply them on a randomly generated task trace. The power source setup and DVS setup are the same as that in the motivational example (Section III-A) except when specifically noted.

A. Experiment 1: Path Finder

We consider the Path Finder of a land warrior [28]. Its power consumption is contributed by the processing unit which is DVS scalable, and the communication unit, which is always on and not DVS scalable. The power and time specifications (at the highest frequency) are presented in Table IV. The processing unit executes three tasks sequentially, i.e., T_1 (path route processing) followed by T_2 (data compression and decompression) followed by T_3 (other processing). We refer to the sequence of $T_1 \rightarrow T_2 \rightarrow T_3$ as a task cycle. The task cycle is periodic with a period of 10s.

TABLE IV
Power and time specifications of Path Finder

Component[1]	P_{active} (W)	P_{idle} (W)	Active time	$P_{average}$ (W)
PAN	4.42	1.70	11.0%	2.00
GPS	0.77	0.40	81.3%	0.70
HFP	7.00	0.94	1.0%	1.00
DS	3.00	0.27	1.0%	0.30

(a) Communication unit

Task	Dynamic power (W)	Intrinsic power (W)	WCET (sec)
T_1	15.0	1.5	5.00
T_2	2.0	1.5	0.26
T_3	4.0	1.5	1.00

(b) Processing unit

Based on the knowledge of the task trace, we choose constant output power used in *en_scale* and *fc_scale* as $P_F = 10$ W, and the load following range used in Algorithm *fc_scale_ctrl* as [4 W, 10 W]. The energy metrics for a two-hour task trace (720 task cycles) simulation are shown in Table V(a). Here the percentage values are the values of E_{total} normalized by that of Algorithm *en_scale*. We see that since Algorithm *en_scale* does not consider the characteristics and state of the power source, a very large portion of energy is wasted: $E_{waste}/E_{total} = 14.9\%$. Algorithm *fc_scale* considers the characteristics of the power source and has significant lower E_{waste}. As a result, it has a lower E_{total} even though its E_{load} value is 7.6% higher than that of Algorithm *en_scale*. Algorithm *fc_scale_ctrl* utilizes the load following capability of the FC and applies joint optimization of both of the embedded system and the FC system. In this particular task trace, Algorithm *fc_scale_ctrl* completely eliminates the wasted energy. Its total energy saving is 8.6% compared to Algorithm *fc_scale*, and 14.9% compared to Algorithm *en_scale*.

Fuel cell over-design: Now assume that the FC output power is configured to a higher value to ensure a safe and robust system operation, i.e., avoid system power failure when the load power increases unexpectedly. In this case, assume that

[1]PAN stands for Primary Area Network, HFP stands for Handheld Flat Panel, and DS stands for Data Storage.

TABLE V
EXPERIMENT 1: PATH FINDER APPLICATION

Energy	*en_scale*	*fc_scale*	*fc_scale_ctrl*
E_{total} (KJ) (%)	71.19 (100.0%)	66.24 (93.1%)	60.58 (85.1%)
E_{load} (KJ)	60.58	65.19	60.58
E_{waste} (KJ)	10.61	1.05	0.00

(a) $P_F = 10$ W with the load following range of [4 W, 10 W] and 11 discrete voltage levels

Energy	*en_scale*	*fc_scale*	*fc_scale_ctrl*
E_{total} (KJ) (%)	107.14 (100.0%)	85.89 (80.2%)	60.58 (56.5%)
E_{load} (KJ)	60.58	77.13	60.58
E_{waste} (KJ)	46.56	8.76	0.00

(b) $P_F = 15$ W with the load following range of [4 W, 15 W] and 11 discrete voltage levels

Energy	*en_scale*	*fc_scale*	*fc_scale_ctrl*
E_{total} (KJ)(%)	104.79 (100.0%)	85.90 (82%)	64.32 (61.4%)
E_{load} (KJ)	64.32	77.16	64.32
E_{waste} (KJ)	40.47	8.74	0.00

(c) $P_F = 15$ W with the load following range of [4 W, 15 W] and 5 discrete voltage levels

the FC output power is set to $P_F = 15$ W for Algorithms *en_scale* and *fc_scale*, and the load following range used in Algorithm *fc_scale_ctrl* is [4 W, 15 W]. The corresponding simulation results are shown in Table V(b). Since Algorithm *en_scale* does not consider the value of P_F at all, its E_{waste} and E_{total} increase significantly, compared to the $P_F = 10$ W case. On the other hand, Algorithm *fc_scale* is aware of the higher P_F, and correspondingly adjusts s_k to reduce E_{total}, resulting in 19.8% saving compared to Algorithm *en_scale*. While the energy metrics E_{load}, E_{waste} and E_{total} of Algorithm *fc_scale* increase significantly compared to the case when $P_F = 10$ W, the energy metrics of Algorithm *fc_scale_ctrl* remain the same. This is because *fc_scale_ctrl* dynamically adjusts the FC output power in the load following range, and a larger load following range ([4 W, 15 W] vs. [4 W, 10 W]) can never deteriorate the performance. The energy saving of *fc_scale_ctrl* is now significantly high, 43.5% compared to *en_scale* and 29.5% compared to *fc_scale*.

Coarse-grain discrete voltage levels: In the previous experiments, we assumed that the supply voltage can be scaled from 1 to 2 in steps of 0.1. Since most commercial DVS systems only support coarse-grain frequency levels, we consider the case when the DVS system supports only five scaling factors: 1, 1.2, 1.5, 1.8, and 2. We set $P_F = 15$ W for the fixed power source configuration, and set the load following range to [4 W, 15 W] for the variable power output configuration. The simulation results are shown in Table V(c). The energy savings achieved by Algorithms *fc_scale* and *fc_scale_ctrl*, are slightly reduced as expected, but Algorithm *fc_scale_ctrl* still achieves large energy savings (38.6%) compared to *en_scale*.

B. Experiment 2: Random task traces

In Experiment 1, T_1 dominates the whole task trace in terms of both power and time duration. So we create a task trace with a more evenly distributed load, which is typical of embedded system applications, and compare the performance of the three algorithms for this task trace.

We consider periodic task cycles with a period of 20 s. Each task cycle consists of five tasks. For each task, the task execution time is randomly chosen from [2 s, 4 s]. The task-dependent scalable power is chosen in the range of [10 W, 20 W], and an additional task-independent non-scalable power consumption is chosen as 4 W (as in Experiment 1). For the scalable power unit, the ratio of the dynamic power to the total power, α_1, is chosen from [0.4, 0.8] and the ratio of the intrinsic power to the total power, α_2, is set to $0.8 - \alpha_1$. All the random data are generated based on uniform distribution. The battery configuration is the same as that of Experiment 1, and the DVS scaling factor is from 1 to 2 with steps of 0.1.

For this randomly generated task trace, when $P_F = 10$ W, the battery is exhausted during task execution, and the FC power cannot support the load demands. So, we increase the FC output power up to $P_F = 15$ W for Algorithms *en_scale* and *fc_scale*, and choose the load following range [4 W, 15 W] for Algorithm *fc_scale_ctrl*.

We simulate the task trace for four hours, and the results are shown in Table VI. The energy savings are lower compared to Experiment 1 but the trends are similar. The total energy savings achieved by Algorithms *fc_scale* and *fc_scale_ctrl* are 16.1% and 27.5%, respectively, compared to Algorithm *en_scale*.

TABLE VI
EXPERIMENT 2: RANDOM TASK TRACES

Energy	*en_scale*	*fc_scale*	*fc_scale_ctrl*
E_{total} (KJ) (%)	214.31 (100.0%)	179.89 (83.9%)	155.31 (72.5%)
E_{load} (KJ)	155.31	179.25	155.31
E_{waste} (KJ)	59.00	0.64	0.00

C. Experiment 3: η_s is not constant

The system efficiency, η_s, is assumed to be a constant (39% according to Fig. 4(c)) for both Experiments 1 and 2. So E_{Gibbs}, which is directly proportional to the fuel consumption of the FC, is calculated by $E_{Gibbs} = E_{total}/\eta_s = E_{total}/0.39$. Now we consider the case when η_s is not constant, but a linear function in the load following range as shown in Fig. 4(b). For such a system, minimization of the fuel consumption is no longer equivalent to minimization of E_{total}. Instead, it is equivalent to minimizing E_{Gibbs}. The optimization framework is similar to the flow introduced in Section III except that the efficiency factor η_s has to be included in every step of the calculation, and s_k^{opt} is now calculated numerically.

The experiment setting for this case is the same as that of Experiment 2. The energy metrics of a four-hour task trace simulation are shown in Table VII. Note that we use E_{Gibbs} as the primary metric to evaluate the performance of the different algorithms. We also include E_{total} values for comparison. We see that for Algorithm *fc_scale_ctrl*, the normalized value of E_{Gibbs} (67.7%) is lower than the normalized value of E_{total} (72.4%). This is because the FC system efficiency η_s increases as FC output level (P_F) decreases, and Algorithm *fc_scale_ctrl* has the capability to assign a lower FC output power level corresponding to higher efficiency.

TABLE VII
EXPERIMENT 3: LINEAR EFFICIENCY MODEL, η_s IS AS IN FIG.4(b)

Energy	*en_scale*	*fc_scale*	*fc_scale_ctrl*
E_{Gibbs} (KJ) (%)	570.58 (100.0%)	447.50 (78.4%)	386.24 (67.7%)
E_{total} (KJ) (%)	213.97 (100.0%)	167.81 (78.4%)	154.97 (72.4%)
E_{load} (KJ)	154.97	164.04	154.97
E_{waste} (KJ)	59.00	3.77	0.00

V. CONCLUSION

In this paper, we addressed the problem of maximizing the lifetime of a fuel-cell based hybrid system. We showed that maximizing the lifetime is equivalent to minimizing the total energy drawn from the hybrid power source. We developed an energy based optimization framework, and proposed control algorithms based on this optimization framework for the case when the FC works at fixed output power level as well as for the case when it works at multiple power levels. When the FC works at a fixed output power, we proposed Algorithm *fc_scale* to control the speed of the DVS processor; when the FC works at multiple power levels, we proposed Algorithm *fc_scale_ctrl* to jointly control the DVS processor speed and the output power of the FC system. We validated the performance of the proposed algorithms on both real-application based task traces and synthetic task traces, for both constant efficiency and linear efficiency configurations of the FC system. For our experimental settings, Algorithm *fc_scale* can achieve up to $\sim 20\%$ fuel savings, and Algorithm *fc_scale_ctrl* can achieve up to $\sim 40\%$ fuel savings compared to the algorithm which is optimized for energy minimization of the embedded system.

ACKNOWLEDGMENT

We give special thanks to Dr. Ken Han and his colleagues at Korean Institute of Industrial Technology (KITECH), and Dr. Dominic Gervasio and his colleagues at Arizona State University (ASU), for their support in providing us with the PEMFC stack and the H_2 generator.

REFERENCES

[1] C. Xie, J. Pavio, J. Hallmark, J. Bostaph, and A. Fisher, "Key requirements of micro fuel cell system for portable electronics," in *37th Intersociety Energy Conversion Engineering Conf. (IECEC)*, 2002.

[2] H. L. Maynard and J. P. Meyers, "Miniature fuel cells for portable power: Design considerations and challenges," *Journal of Vacuum Science & Technology*, vol. 20, no. 4, pp. 1287–1297, July 2002.

[3] D. R. P. et.al., "Development of a soldier-portable fuel cell power system," *Journal of Power Sources*, vol. 108, pp. 28–34, 2002.

[4] W. Gao, "Performance comparison of a fuel cell-battery hybrid powertrain and a fuel cell-ultracapacitor hybrid powertrain," *IEEE Trans. on Vehicular Tech.*, vol. 54, no. 3, pp. 846–855, May. 2005.

[5] J. P. Zheng, T. R. Jow, and M. S. Ding, "Hybrid power sources for pulsed current applications," *IEEE Trans. Aerospace and Electronic Sys.*, vol. 37, no. 1, pp. 288–292, 2001.

[6] A. Vahidi, A. Stefanopoulou, and H. Peng, "Model predictive control for starvation prevention in a hybrid fuel cell system," in *Proc. of American Control Conf. (ACC)*, vol. 1, 2004, pp. 834–839 vol.1.

[7] A. Vahidi, I. Kolmanovsky, and A. Stefanopoulou, "Constraint management in fuel cells: a fast reference governor approach," in *Proc. of American Control Conf. (ACC)*, vol. 6, 2005, pp. 3865–3870.

[8] Y. Guezennec, T.-Y. Choi, G. Paganelli, and G. Rizzoni, "Supervisory control of fuel cell vehicles and its link to overall system efficiency and low-level control requirements," in *Proc. of American Control Conf. (ACC)*, vol. 3, 2003, pp. 2055–2061.

[9] A. Nasiri, V. S. Rimmalapudi, A. Emadi, D. J. Chmielewski, and S. Al-Hallaj, "Active control of a hybrid fuel cell-battery system," in *Proc. of Intl'l Power Electronics and Motion Control Conf. (IPEMC)*, vol. 2, 2004, pp. 491–496.

[10] M. J. Gielniak and Z. J. Shen, "Power management strategy based on game theory for fuel cell hybrid electric vehicles," in *Proc. Vehicular Technology Conf. (VTC)*, vol. 6, 2004, pp. 4422–4426.

[11] E. Ozatay, B. Zile, J. Anstrom, and S. Brennan, "Power distribution control coordinating ultracapacitors and batteries for electric vehicles," in *Proc. American Control Conf. (ACC)*, vol. 5, 2004, pp. 4716–4721.

[12] L. Gao, Z. Jiang, and R. A. Dougal, "Evaluation of active hybrid fuel cell/battery power sources," *IEEE Trans. on Aeorspace and Electronic Systems*, vol. 41, no. 1, pp. 346–355, Jan. 2005.

[13] Z. Jiang, L. Gao, and R. A. Dougal, "Flexible multiobjective control of power converter in active hybrid fule cell/battery power sources," *IEEE Trans. on Power Electronics*, vol. 20, no. 1, pp. 244–253, Jan 2005.

[14] J. Zhuo, C. Chakrabarti, K. Lee, and N. Chang, "Dynamic power management with hybrid power sources," in *Proc. of DAC*, June 2007, pp. 871–876.

[15] J. Zhuo, C. Chakrabarti, and N. Chang, "Energy management of dvs-dpm enabled embedded systems powered by fuel cell-battery hybrid source," in *Proc. of ISLPED*, August 2007, pp. 322–327.

[16] J. Zhuo, C. Chakrabarti, N. Chang, and S.Vrudhula, "Extending the lifetime of fuel cell based hybrid systems," in *Proc. of DAC*, July 2006, pp. 562–567.

[17] ——, "Maximizing the lifetime of embedded systems powered by fuel cell-battery hybrids," in *Proc. of ISLPED*, October 2006, pp. 424–429.

[18] C. K. Dyer, "Fuel cells and portable electronics," in *Symposium on VLSI circuits (digest of technical papers)*, June 2004, pp. 124–127.

[19] D. Gervasio, S. Tasic, and F. Zenhausern, "A room temperature micro-hydrogen-generator," *Journal of Power Sources*, vol. 149, pp. 15–21, 2005.

[20] D. Gervasio, "Fuel-cell system for hand-carried portable power," *International Fuel Cell R&D Forum*, Nov. 2005.

[21] J. Larminie and A. Dicks, *Fuel Cell Systems Explained*. John Wiley & Sons, LTD, 2000.

[22] J. T. Pukrushpan, A. G. Stefanopoulou, and H. Peng, *Control of Fuel Cell Power Systems: Principles, Modeling Analysis, and Feedback Design*. Springer, 2004.

[23] L. P. Jarvis, P. J. Cygan, and M. P. Roberts, "Fuel cell/lithium-ion battery hybrid for manportable applications," in *Proc. of Battery Conf. on Applications and Advances*, 2002, pp. 69–72.

[24] Y. Kojima, "Hydrogen storage and generation using sodium borohydride," *R&D Review of Toyota CRDL*, vol. 40, pp. 31–36, 2005.

[25] N. Kim and etl, "Leakage current: Moore's law meets static power," *Computer*, vol. 36, pp. 68–75, Dec. 2003.

[26] R. Jejurikar, C. Pereira, and R. Gupta, "Leakage aware dynamic voltage scaling for real-time embedded systems," in *Proc. of DAC*, June 2004, pp. 275–280.

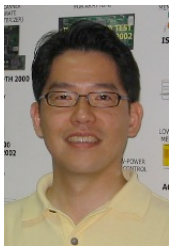
[27] Y. Cho, N. Chang, C. Chakrabarti, and S. Vrudhula, "High-level power management of embedded systems with application-specific energy cost functions," in *Proc. of DAC*, July 2006, pp. 568–573.

[28] NRC (National Research Council), *Meeting the Energy Needs of Future Warriors Committee of Soldier Power/Energy Systems*. Washington, D.C.: National Academy Press, 2004.

Jianli Zhuo received her BS and MS degrees from Department of Automation, Tsinghua University (Beijing), in 1999 and 2001, respectively. She was with Bell Labs, Lucent Technologies (Beijing) from 2001 to 2003. She received her Ph.D degree from Department of Electrical Engineering, Arizona State University, in 2007. She is currently with Synopsys Inc.. Her research interests are in computer-aided design, including low power design, VLSI design and embedded system design.

Chaitali Chakrabarti (SM'02) is a Professor in the Department of Electrical Engineering at Arizona State University. She received the B.Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, in 1986 and 1990, respectively. Her research interests include the areas of low power embedded systems design including memory optimization, high level synthesis and compilation, and VLSI architectures and algorithms for signal processing, image processing, and communications. She is a recipient of the Research Initiation Award from the National Science Foundation in 1993, a Best Teacher Award from the College of Engineering and Applied Sciences from ASU in 1994, and the Outstanding Educator Award from the IEEE Phoenix Section in 2001. She has served on the program committees of ICASSP, ISCAS, ASAP, SIPS, HPCA, ISLPED, and DAC. She currently serves as an Associate Editor of the Journal of VLSI Signal Processing Systems and the IEEE Transactions on VLSI Systems. She is currently the TC Chair of the sub-committee on Design and Implementation of Signal Processing Systems, IEEE Signal Processing Society.

Kyungsoo Lee received the B.S. degree from the School of Computer Science and Engineering, Seoul National University, Korea, in 2004, where he is currently pursuing the Ph.D. degree at the same university. His current research interests include system-level low-power design and embedded systems design.



Naehyuck Chang (SM'05) received BS, MS and PhD degrees all from Department of Control and Instrumentation, Seoul National University, 1989, 1992 and 1996, respectively. He has been with School of Computer Science and Engineering since 1997 and currently is an Associate Professor. His research interest includes Embedded Systems and Low-Power Systems. He published over 70 technical papers in these areas. He serves or served as Technical Program Committee of ACM SIGDA and IEEE Circuits and Systems Society conferences and symposiums such as DAC, ICCAD, ISLPED, ISQED, ASP-DAC, GLSVLSI and so on. He is currently an Associate Editor of IEEE TCAD, Journal of Low-Power Electronics and Journal of Embedded Computing. He is a Senior Member of Association for Computing Machinery.

Sarma Vrudhula (SM'01) is the Consortium for Embedded Systems (CES) Chair Professor in the department of Computer Science and Engineering at Arizona State University. He received the B.Math (Honors) from the University of Waterloo, Ontario, Canada, in 1976 and his M.S. and Ph.D degrees in electrical engineering from the University of Southern California in 1980 and 1985, respectively. His primary research area in Computer-Aided Design and Design Automation for digital VLSI. His current research activities statistical analysis and optimization for power and performance in the presence of process variations, low power circuit design, system level power and thermal modeling, and dynamic thermal management of multi-core processors, threshold logic synthesis and optimization, applications of circuit CAD to modeling of biological processes. He has published over 150 papers on these topics in peer-reviewed conferences and journals. He has served on the technical program committees of many national and international conferences in VLSI CAD, and on government review panels. He serves as an Associate Editor for the IEEE Transactions on CAD and ACM Transactions on Design Automation of Electronics Systems (TODAES).