Offline Guarantee and Online Management of Power Demand and Supply in Cyber-Physical Systems

Eugene Kim, Youngmoon Lee, Liang He, Kang G. Shin
Department of Electrical Engineering and Computer Science
The University of Michigan – Ann Arbor, MI 48109-2121, U.S.A.
{kimsun,ymoonlee,lianghe,kgshin}@umich.edu

Jinkyu Lee
Department of Computer Science and Engineering
Sungkyunkwan University (SKKU), Republic of Korea
jinkyu.lee@skku.edu

***Abstract*—Since modern electric systems require to support various power-demand operations for user applications and system maintenance, they need advanced power management that jointly considers power demand by the operations and power supply from various sources, such as batteries, solar panels, and supercapacitors. In this paper, we develop a power scheduling framework for a reliable energy storage system with multiple power-supply sources and multiple power-demand operations. First, we provide an offline power-supply guarantee such that every power-demand operation completes its execution in time while the sum of power required by individual operations does not exceed the total power supplied by the entire energy storage system at any time. We find similarities between this and a real-time scheduling problem, and make a power-supply guarantee using real-time scheduling techniques. Second, we propose online power management that efficiently utilizes the surplus power (available at run-time) for system performance improvement. Our experimental results on a prototype demonstrate that the proposed framework not only guarantees the required power-supply, but also enhances system performance by up to 33.1%.**

I. Introduction

Most electric systems, such as electric vehicles, mobile robots, nano satellites, and drones, have to perform various mechanical/electrical operations for their intended applications. For example, a drone needs to (i) operate the flight motors to fly, (ii) power sensors, coolers, and communication modules, (iii) operate the camera and stepper motors to take pictures and deliver parcels, and (iv) perform the computation necessary to maintain its stability (see Fig. 3 in the supplementary file [1]). These *multiple power-demand operations* impose different power demands on the system and may be triggered at different times, requiring the system to effectively provide time-varying supply of power [2–7]. Existing studies on the power scheduling problem focused on the reduction of peak power by scheduling multiple power-demand operations and analysis thereof [8–11].

However, a complete solution of the power scheduling problem also needs to characterize energy storages/sources because it affects power capability for the power-demand operations. We address this need by targeting hybrid energy storage systems (HESSes) comprised of *multiple power-supply sources and storages*, such as batteries, supercapacitors, and renewable energy sources, whose concept and implementation have been proposed and explored in [5, 12–17]. A combination of high-energy-density batteries and high-power-density supercapacitors enables the system to supply the required time-varying power for a longer time, improving system sustainability. Moreover, renewable power sources (e.g., solar, wind and geothermal energy) enhance the system's power capacity and reduce the load intensity on batteries and supercapacitors, thus prolonging their lifetime [18–22].

Researchers have studied the optimal design of a HESS to meet the power demands at minimum cost [13, 23–26]. They first explored possible storage configurations (i.e., size, type, connection) of multiple power-source supplies and storages, and then selected the best configuration by comparing the performance and the energy production cost. Search for an optimal design iterates to generate candidate configurations until one of the candidates satisfies the pre-defined performance and cost criteria [24]. However, these approaches do not guarantee power-sufficiency during operation, as the electric load and criteria are determined based on the load history or the designer's experience and intuition.

To ensure the power sufficiency, we will develop a power scheduling and analysis framework for a reliable energy storage system with multiple power-supply sources and multiple power-demand operations, which achieves the following goals.

- G1. **Offline power-supply guarantee:** We provide an offline guarantee to complete every operation before its deadline (i.e., *operation-level power guarantee*) while keeping the amount of power supplied to the entire system no smaller than the sum of power required by individual operations at any time instant (i.e., *system-level power guarantee*).
- G2. **Online power management:** We develop online power management that effectively utilizes the difference between the worst-case supplied/demanded power and the actual one for system performance improvement.

This is a typical *cyber-physical systems* (CPS) problem in that we should address power scheduling and its analysis in the cyber space, based on comprehensive understanding of the physical characteristics of power supply and demand.

To achieve G1, we find similarities between the power-supply guarantee problem and a real-time scheduling problem that determines the execution order of real-time tasks; while the operation-level power guarantee corresponds to the task-level deadline satisfaction, the system-level power guarantee matches the computing platform's capacity constraint. Using the techniques of real-time scheduling, we solve the power-supply guarantee problem in two steps. First, we address the case of multiple power-demand operations with a single,

978-1-5090-5303-2/16 $31.00 © 2016 IEEE
DOI 10.1109/RTSS.2016.23

uniform power-supply source (e.g., a battery pack), and develop a scheduling framework and offline power guarantee analysis. Based on the scheduling and analysis framework, we address the general problem—multiple power-demand operations with multiple power-supply sources. For the second step, we develop two scheduling frameworks to utilize additional sporadic power-supply sources: one for sharing additional power-supply sources by all power-demand operations, and the other for assigning the sources to only some of power-demand operations. Our solution for G1 not only demonstrates that the technique of real-time scheduling helps solve a CPS problem, but also addresses the design problem of a HESS by finding a combination of energy storages at minimum cost.

In addition to making the offline power-supply guarantee (G1), we develop online power management (G2), which aims at increasing the utilization of the energy generated by renewable power-supply sources. That is, scheduling multiple power-supply sources and power-demand operations according to G1 necessarily yields surplus energy since G1 considers the worst case, i.e., the largest power demand and the smallest power supply. While we can utilize the surplus energy for improving various performance aspects of a HESS, we focus on the reduction of the peak power of our main energy storage (i.e., a battery pack) to improve its capacity and lifetime [16, 25, 27–30].

We have prototyped a HESS-powered system (Fig. 5 as described in Section VII and Fig. 1 in the supplementary file [1]) to validate the proposed solutions for G1 and G2. The prototype consists of power-consuming components such as wheel motors, stepper motors, coolers, sensors and converters, whose operations are controlled by an application. The prototype is powered by a HESS consisting of Lithium-ion batteries, ultra-capacitors (UCs), and solar panels. When designing this prototype, we determine the sizes of batteries, UCs and solar panels that ensure the worst-case power sufficiency based on the offline power-supply guarantee G1. At run-time, the master board schedules the power-demanding operations with fixed-priority scheduling, and determines the power distribution among batteries, UCs, and solar panels according to online power management G2. Our experimental results show that the prototype not only supplies a sufficient amount of power to the components during their operation, and but also reduces the battery's peak power by 33.1%.

In summary, this paper makes the following main contributions:

- Design of a power scheduling and analysis framework that consists of (i) offline power-supply guarantee, which is the first power guarantee analysis applicable to the design of a HESS, and (ii) online power management for a HESS to enhance the energy storage's performance,
- Demonstration of the effectiveness of the proposed framework via in-depth, realistic experiments, and
- Solution of an important CPS problem using real-time scheduling techniques.

The paper is organized as follows. Section II describes the characteristics of power-supply sources and power-demand operations, and Section III formally states our main problem. Sections IV and V present the power scheduling and analysis framework for the offline power-supply guarantee, while Section VI details online power management. Section VII implements our solutions on a prototype and evaluates their power sufficiency and reduction of peak power dissipation. Finally, the paper concludes with Section VIII.

II. Characteristics of Power-Demand Operations and Power-Supply Sources

As the first step for offline power-supply guarantee and online power management, we investigate the characteristics of power-demand operations and power-supply sources.

A. Power-demand operations

We consider typical mechanical/electrical operations that are executed repeatedly to complete a given set of tasks. For example, an operation that maintains a constant vehicle speed would accelerate/decelerate motors repeatedly based on the online collected speed data. In this closed-loop control system, the operation updates the required acceleration and requests it to the motor controller at every control period — the operation requires a certain level of power, and must be completed within its control period.

We model the power-demand operations as follows. A system has a set of mechanical/electrical operations $\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_{n^{\mathtt{d}}}\}$, where $n^{\mathtt{d}}$ is the number of operations.[1] The power usage of λ_i is modeled by the minimum inter-arrival time $T_i^{\mathtt{d}}$, its maximum power consumption $P_i^{\mathtt{d}}$, and the maximum length of execution $L_i^{\mathtt{d}}$. Each operation is assumed to be non-preemptive; unlike a computing task executed on a processor, a preemption of an operation on a power system either yields incorrect system behavior or incurs significant cost. The former happens because the physical state of devices before a preemption is not necessarily identical to that after the preemption, e.g., revolutions of a motor per minute. The latter also makes sense in that most power-consuming devices are deployed in a distributed manner, so a preemption and its re-activation entail non-trivial communication/operation overhead such as sending a message via communication channel, pausing the mechanical/electrical operation, and sending back a preemption completion message to the master controller. Each operation is also assumed to be rigid, i.e., providing more power does not reduce the execution length.

Considering the characteristics described so far, λ_i with parameters $T_i^{\mathtt{d}}$, $P_i^{\mathtt{d}}$, and $L_i^{\mathtt{d}}$ invokes its instances as follows. Each instance's release time is separated from the predecessor by at least T_i time units, and each instance should be finished within T_i time units after its release. Once starting to execute, each instance performs its execution during at most L_i time units without any preemption, and the amount of actual power consumption at t within the interval is denoted by $P_i^{\mathtt{d}}(t)$, which is less than or equal to $P_i^{\mathtt{d}}$.[2]

[1] Throughout the paper, we use the superscripts $\mathtt{d}$ and $\mathtt{s}$ for power demand and power supply, respectively.

[2] Throughout the paper, $P(t)$ implies the actual power demand/supply at t, and P represents the maximum/minimum power demand/supply. Also, we use the term of an "operation" for an "instance" of the operation, when no ambiguity arises.

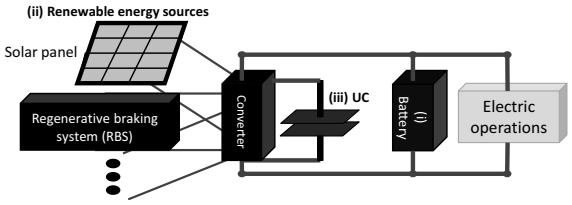

Fig. 1. Battery-centric hybrid energy storage system model

Finally, each operation has its own priority based on the importance of the operation; we consider fixed-priority scheduling [31] using the predefined operation-level priorities.

B. Power-supply sources

We consider a battery-centric hybrid energy storage system (HESS) that is comprised of three parts: (i) an energy-dense lithium-ion battery pack, (ii) a set of auxiliary renewable energy sources, and (iii) a power-dense energy buffer as shown in Fig. 1. Although relatively simple, this architecture contains essential parts for advanced energy management, and each part has been studied extensively [5, 12, 16, 32–34].

While batteries are widely used as the main energy storage due to their capability to store a large amount of energy and deliver the required power, renewable energy sources such as an RBS (Regenerative Braking System) and a solar panel, supply power sporadically. For instance, an RBS in a vehicle can supply power only when the vehicle decelerates, and the amount of generated power is dependent on braking torque that each brake can provide. A solar panel generates the energy when sunlight strikes a solar cell. On the other hand, the energy buffer accommodates surges of recharging current to protect the battery and provide power when the power requirement is high. Fig. 4 in the supplementary file [1] is an example showing how the energy buffer works for accommodating power supply from renewable energy sources and powering requested operations.

Described below are notations and characteristics of above-mentioned power-supply sources and storages. Let $\Gamma_{\texttt{bat}}$ and $\Gamma_{\texttt{UC}}$ denote the battery pack and the energy buffer corresponding to (i) and (iii), respectively, and they can supply a certain amount of power, which are limited to their individual power capabilities $P^{\texttt{s}}_{\texttt{bat}}$ and $P^{\texttt{s}}_{\texttt{UC}}$, respectively; $P^{\texttt{s}}_{\texttt{bat}}(t)$ $(\leq P^{\texttt{s}}_{\texttt{bat}})$ and $P^{\texttt{s}}_{\texttt{UC}}(t)$ $(\leq P^{\texttt{s}}_{\texttt{UC}})$ denote the actual power supply at t of the corresponding sources. Let $L^{\texttt{s}}_{\texttt{UC}}$ denote the minimum time interval for $\Gamma_{\texttt{UC}}$ to supply power as much as $P^{\texttt{s}}_{\texttt{UC}}$ when the system starts, i.e., at $t = 0$, $\Gamma_{\texttt{UC}}$ has at least $P^{\texttt{s}}_{\texttt{UC}} \cdot L^{\texttt{s}}_{\texttt{UC}}$ amount of energy.[3] We assume that the target system halts when there is no energy in the battery pack, meaning that we do not need to specify the interval length of $\Gamma_{\texttt{bat}}$.

As to (ii), let $\Gamma = \{\Gamma_1, \Gamma_2, ...\Gamma_{n^{\texttt{s}}}\}$ denote a set of auxiliary renewable energy sources, where $n^{\texttt{s}}$ (≥ 0) is the number of elements in the set. Each energy source Γ_i is modeled with the maximum inter-arrival time $T^{\texttt{s}}_i$, the minimum supplied power $P^{\texttt{s}}_i$, and the minimum length of supply duration $L^{\texttt{s}}_i$. That is, Γ_i generates at least $P^{\texttt{s}}_i$ amount of power during at least $L^{\texttt{s}}_i$ time units at the end of every period whose length is no larger than $T^{\texttt{s}}_i$. Let $P^{\texttt{s}}_i(t)$ is the actual supplied power at t from Γ_i. Power generation from Γ_i may be unpredictable, which makes it difficult to derive the parameters of Γ_i. However, if we imagine power generation from solar panels during daytime, we can derive the parameters that express conservative behaviors in terms of the amount of power generation; we will show an example of Γ_i using our prototype in the evaluation section (Fig. 2 in the supplementary file [1]). Note that parameters that provide "guarantee" of power supply (i.e., $P^{\texttt{s}}_{\texttt{bat}}$ amount of power for all time and $P^{\texttt{s}}_i \cdot L^{\texttt{s}}_i$ amount of energy for every $T^{\texttt{s}}_i$ time units) will be utilized for the offline power guarantee, while any surplus energy (the information which is available only at run-time) will be stored in $\Gamma_{\texttt{UC}}$ and used for online management for minimizing battery peak power. Also, we assume that the energy $\Gamma_{\texttt{UC}}$ can store is sufficiently large so as to accommodate power generated by all the renewable energy sources.

[3] For example, the battery pack supplies energy to the energy buffer when an electric vehicle turns on the ignition.

III. Problem Statement and Solution Approach

We first present a formal problem statement for achieving offline power-supply guarantee and online power management. Then, we give high-level explanation how to solve the problem.

Problem statement. Using the characteristics of power-demand operations and power-supply sources described in Section II, we formally state the problem to be solved as:

Given a set of mechanical/electrical operations λ, the battery pack $\Gamma_{\texttt{bat}}$, the energy buffer $\Gamma_{\texttt{UC}}$, a set of auxiliary renewable energy sources Γ, and the operation interval $[0, t_{\texttt{max}})$,

Determine $P^{\texttt{s}}_{\texttt{UC}}(t)$ and $\{P^{\texttt{d}}_i(t)\}_{i=1}^{n^{\texttt{d}}}$ for all $t \in [0, t_{\texttt{max}})$ for achieving the following objective function.

$$\text{Minimize (S0)} \quad \int_0^{t_{\texttt{max}}} P^{\texttt{s}}_{\texttt{bat}}(t) + P^{\texttt{s}}_{\texttt{bat-loss}}(t)\, dt,$$

$$\text{Subject to (S1)} \quad \sum_{\lambda_i \in \lambda} P^{\texttt{d}}_i(t) - P^{\texttt{s}}_{\texttt{UC}}(t) - \sum_{\Gamma_i \in \Gamma} P^{\texttt{s}}_i(t) = P^{\texttt{s}}_{\texttt{bat}}(t) \leq P^{\texttt{s}}_{\texttt{bat}},$$

(S2) $P^{\texttt{s}}_{\texttt{UC}}(t) \leq P^{\texttt{s}}_{\texttt{UC}}$,

(S3) Every instance of each operation $\lambda_i \in \lambda$ finishes its execution within $T^{\texttt{d}}_i$ time units after its release.

$\{P^{\texttt{s}}_i(t)\}_{i=1}^{n^{\texttt{s}}}$ are not control knobs in that their generated power is immediately stored in $\Gamma_{\texttt{UC}}$ or served for power-demand operations. $P^{\texttt{s}}_{\texttt{bat}}(t)$ is also not, because it is determined once $P^{\texttt{s}}_{\texttt{UC}}(t)$ and $\{P^{\texttt{d}}_i(t)\}_{i=1}^{n^{\texttt{d}}}$ are determined as shown in S1. For each $P^{\texttt{d}}_i(t)$, we determine the time instant at which each λ_i starts to execute. When it comes to $P^{\texttt{s}}_{\texttt{UC}}(t)$, we determine the amount of supplied power from $\Gamma_{\texttt{UC}}$ to λ at each time instant t.

$P^{\texttt{s}}_{\texttt{bat-loss}}(t)$ in S0 denotes the amount of power dissipation at t caused by the power supply of the battery pack; in other words, we lose $P^{\texttt{s}}_{\texttt{bat-loss}}(t)$ of power due to supplying $P^{\texttt{s}}_{\texttt{bat}}(t)$ of power from the battery pack at t. Therefore, S0 implies the amount of energy used and dissipated by the battery pack; since the former (i.e., the amount of energy used) is not a control knob, we should reduce the latter (i.e., the amount of energy dissipated) using a simple circuit-based battery model [35]: $P^{\texttt{s}}_{\texttt{bat-loss}}(t) = I^2_{\texttt{bat}}(t) \cdot R_{\texttt{bat}}(t)$, where $R_{\texttt{bat}}(t)$ is a battery's internal resistance, and $I_{\texttt{bat}}(t)$ is the battery's

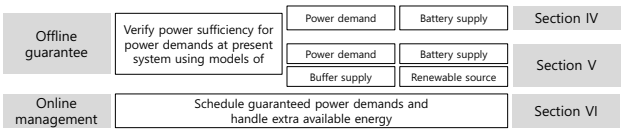

Fig. 2. Outline of our power scheduling and analysis framework that consists of offline power guarantee and online power management

discharge/charge current for supplying $P_{\mathtt{bat}}(t)$. Since power dissipation is quadratically proportional to discharge/charge battery current, the minimization of battery peak current may reduce energy dissipation of the battery pack, potentially extending the battery operation-time.

Solution approach. To achieve the above objective, we develop a power scheduling and analysis framework that consists of two steps, as shown in Fig. 2. In the first step, we focus on satisfaction of constraints S1, S2 and S3, achieving offline power-supply guarantees to complete every operation before its deadline while keeping the amount of power supplied to the entire system no smaller than the sum of power required by individual operations at any time instant. To this end, we consider two sub-steps: one for the single, uniform power-supply source, and the other for the multiple power-supply sources, both of which use parameters that exhibit the worst-case behaviors, e.g., P_i^{d}, not $P_i^{\mathrm{d}}(t)$. First, in Section IV we will consider scheduling of λ only with $\Gamma_{\mathtt{bat}}$, and develop a power scheduling framework and its power guarantee analysis that determines whether or not λ is always scheduled on time without any power shortage under fixed-priority scheduling. Then, we extend the results for considering additional power-supply sources Γ. Adding Γ and using the fact that $\Gamma_{\mathtt{UC}}$ has the minimum energy when the system starts, in Section V we will suggest two approaches that utilize the power from Γ: one for sharing the power by all operations, and the other for assigning the power to some operations. We also develop the scheduling framework and power guarantee analysis for both approaches.

For the second step, Section VI aims to minimize S0 without compromising S1, S2 and S3. Since the first step is performed based on parameters that exhibit the worst-case behaviors, a lot of energy is actually stored in $\Gamma_{\mathtt{UC}}$, coming from the difference between P_i^{s} and $P_i^{\mathrm{s}}(t)$. To effectively utilize the difference for minimizing S0, in Section VI we propose an online power management framework to reduce battery peak dissipation, thereby improving battery performance. Since greedy usage of the energy buffer may cause energy shortage when it is needed to mitigate the subsequent surges of power demand, we determine power supply of the energy buffer ($P_{\mathtt{UC}}^{\mathrm{s}}(t)$) effectively, by considering the current status of the energy buffer and the history of power demand.

IV. Scheduling of Multiple Power-Demand Operations with a Uniform Supply

In this section, we present how to schedule a set of power-demand operations (λ) when the battery pack ($\Gamma_{\mathtt{bat}}$) is the sole power-supply source. We first develop a scheduling framework that considers the characteristics of λ and $\Gamma_{\mathtt{bat}}$ described in Section II. Under this scheduling framework, we then develop an offline power guarantee analysis that determines whether every operation in λ powered by the battery pack is performed before its deadline without suffering any power shortage.

Algorithm 1 Scheduling Framework

The following steps are performed whenever at least one mechanical/electrical operation is finished or released at t:

1: For each λ_k finished at t, $Q_{\mathtt{run}} \leftarrow Q_{\mathtt{run}} \setminus \{\lambda_k\}$
2: For each λ_k released at t, $Q_{\mathtt{ready}} \leftarrow Q_{\mathtt{ready}} \cup \{\lambda_k\}$
3: Sort $Q_{\mathtt{ready}}$ by given operation-level fixed-priorities
4: **for** $\lambda_k \in Q_{\mathtt{ready}}$ (a higher priority operation is chosen earlier) **do**
5: **if** $P_k^{\mathrm{d}} \leq P_{\mathtt{bat}}^{\mathrm{s}} - \sum_{\lambda_i \in Q_{\mathtt{run}}} P_i^{\mathrm{d}}$ **then**
6: $Q_{\mathtt{run}} \leftarrow Q_{\mathtt{run}} \cup \{\lambda_k\}$
7: $Q_{\mathtt{ready}} \leftarrow Q_{\mathtt{ready}} \setminus \{\lambda_k\}$
8: **end if**
9: **end for**

A. Scheduling framework

We would like to schedule a set of multiple operations (λ) so as to achieve a system-level and operation-level power guarantee under the uniform supply from the battery pack $\Gamma_{\mathtt{bat}}$. By "system-level power guarantee," we mean that the sum of power demand at any time instant should be no larger than maximum power capability of the battery pack, i.e., $\sum_{\lambda_i \in \lambda} P_i^{\mathrm{d}}(t) \leq P_{\mathtt{bat}}^{\mathrm{s}}$, which is equivalent to S1.[4] To achieve the operation-level power guarantee, every operation should receive sufficient power within its period, which is S3.

Our scheduling framework employs the work-conserving policy based on the worst-case power demand and operation-level fixed-priority scheduling policy. The former implies that an operation λ_k can start to execute as long as its worst-case power demand P_k^{d} (as opposed to the actual power demand $P_k^{\mathrm{d}}(t)$) is no larger than the difference between the battery capability $P_{\mathtt{bat}}^{\mathrm{s}}$ and the sum of the worst-case power demand of currently-executing operations $\sum_{\lambda_i \in Q_{\mathtt{run}}} P_i^{\mathrm{d}}$. The latter implies that the scheduling framework prioritizes the operations that satisfy the above condition.

Since each operation exhibits the non-preemptive behavior as described in Section II, an operation can start its execution only when it is released or the other operation is finished, and each operation, once started, continues its execution until the completion. Therefore, the scheduling framework can be expressed (as in Algorithm 1) by describing actions for the situations. Lines 1–3 update the ready queue $Q_{\mathtt{ready}}$ that contains operations ready to execute, and the running queue $Q_{\mathtt{run}}$ for currently-executing operations. Lines 4–9 select operations to be started and move them from $Q_{\mathtt{ready}}$ to $Q_{\mathtt{run}}$. Although the scheduling framework prioritizes operations based on their priorities, a lower-priority operation λ_j in $Q_{\mathtt{ready}}$ can start its execution earlier than a higher-priority operation λ_k in $Q_{\mathtt{ready}}$, if the remaining power capability of the system (i.e., $P_{\mathtt{bat}}^{\mathrm{s}} - \sum_{\lambda_i \in Q_{\mathtt{run}}} P_i^{\mathrm{d}}$) is larger than P_k^{d} but, no larger than P_j^{d}, as described in line 5.

Analogy with real-time scheduling. The scheduling framework presented in Algorithm 1 is similar to gang scheduling [36] in the area of real-time scheduling, where P_i^{d} corresponds to the number of threads to be parallelized for a real-time task, and L_i^{d} matches the WCET (Worst-Case Execution

[4] Since this section considers the sole supply of $\Gamma_{\mathtt{bat}}$, we remove terms of the energy buffer and a set of renewable power-supply sources in S1.

Time) of each thread. Note that P_i^{d} is a continuous variable, but the number of threads is a discrete value. While existing studies for gang scheduling have focused on preemptive scheduling [36–38], we need non-preemptive gang scheduling. In Section IV-B, we will develop an offline power guarantee analysis, corresponding to the schedulability analysis of non-preemptive gang scheduling, which is, to the best of our knowledge, the first attempt in the real-time scheduling field.

We also note that the similarity to gang scheduling holds for the case of this section—multiple power-demand operations only with the uniform supply; it is a new type of scheduling problem to address the general case of multiple power-demand operations and multiple power-supply sources, which will be addressed in Section V.

B. Offline power guarantee analysis

Since each operation is non-preemptive, we need to check if each operation λ_k can start its execution no later than $(T_k^{\mathrm{d}} - L_k^{\mathrm{d}})$ time units after its release; once it starts to execute, it completes execution within its period without any preemption. Therefore, we focus on the interval of length $(T_k^{\mathrm{d}} - L_k^{\mathrm{d}} + \epsilon_t)$, which begins at the time of λ_k's release, and check whether the sum of energy consumed by other operations within the interval is strictly less than $(P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot (T_k^{\mathrm{d}} - L_k^{\mathrm{d}} + \epsilon_t)$, where ϵ_t and ϵ_p denote the time and power quantum (the smallest unit). Since the smallest power that prevents λ_k from execution at each time instant is $(P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p)$, the above condition guarantees the start of λ_k's execution to be no later than $(T_k^{\mathrm{d}} - L_k^{\mathrm{d}})$ time units after its release.

The remaining step is then to calculate $I_{k \leftarrow i}(\ell)$, the amount of energy demanded by instances of λ_i within an interval of length ℓ that contributes to prevention of λ_k from starting its execution. Note that for calculation of $I_{k \leftarrow i}(\ell)$, we limit the power demand at each time instant to $(P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p)$, since we need to know whether the sum of all power demands at each time instant is no smaller than $(P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p)$. Now, we will describe how to calculate an upper bound of $I_{k \leftarrow i}(\ell)$.

First, if λ_i has a higher priority than λ_k, then the upper-bound of $I_{k \leftarrow i}(\ell)$ is the maximum energy demanded by λ_i in an interval of length ℓ, which is calculated as

$$W_i(\ell) = \min(P_i^{\mathrm{d}}, P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \times \min\left(\ell, \left(N_i(\ell) \cdot L_i^{\mathrm{d}} + \min(L_i^{\mathrm{d}}, \ell + T_i^{\mathrm{d}} - L_i^{\mathrm{d}} - N_i(\ell) \cdot T_i^{\mathrm{d}})\right)\right), \quad (1)$$

where $N_i(\ell) = \lfloor \frac{\ell + T_i^{\mathrm{d}} - L_i^{\mathrm{d}}}{T_i^{\mathrm{d}}} \rfloor$. This calculation is similar to the workload calculation of real-time scheduling [39]. Briefly, $N_i(\ell)$ implies the number of instances of λ_i, and each of their periods is completely included within the interval of length ℓ (including the first instance of λ_i), which contributes $\min(P_i^{\mathrm{d}}, P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot N_i(\ell) \cdot L_i^{\mathrm{d}}$ of energy to $W_i(\ell)$. The second part of $W_i(\ell)$ represents the contribution of the operation whose period is partially included in the interval of length ℓ. Fig. 5 in the supplementary file [1] shows an example of $W_i(\ell)$ with $N_i(\ell) = 2$, which contributes $\min(P_i^{\mathrm{d}}, P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot 2 \cdot L_i^{\mathrm{d}}$ of energy, and the third instance of λ_i contributes $\min(P_i^{\mathrm{d}}, P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot \min(L_i^{\mathrm{d}}, \ell + T_i^{\mathrm{d}} - L_i^{\mathrm{d}} - N_i(\ell) \cdot T_i^{\mathrm{d}})$ of energy to $W_i(\ell)$.

Second, if λ_i has a lower priority than λ_k, then we consider two sub-cases. Since each operation is non-preemptive, λ_i can execute before the start of λ_k's execution, if λ_i starts its execution before the release of λ_k. In this case, the energy demand is upper-bounded by $\min(P_i^{\mathrm{d}}, P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot \min(L_i^{\mathrm{d}} - \epsilon_t, \ell)$, which is an upper-bound of $I_{k \leftarrow i}(\ell)$ for the first sub-case of $P_i^{\mathrm{d}} \geq P_k^{\mathrm{d}}$. If $P_i^{\mathrm{d}} < P_k^{\mathrm{d}}$, then an upper-bound of $I_{k \leftarrow i}(\ell)$ can be larger than the first sub-case. That is, due to our worst-case-based work-conserving policy, it is possible for λ_i to start its execution before the start of λ_k's execution, whenever λ_k does not satisfy Line 5 of Algorithm 1 but λ_i does. In this case, we use the general upper-bound $W_i(\ell)$ as an upper-bound of the second sub-case.

Combining all the results discussed so far, we develop a power guarantee analysis as follows.

Lemma 1: Suppose that every $\lambda_k \in \lambda$ satisfies Eq. (2). Then, every instance of every operation $\lambda_k \in \lambda$ finishes its execution within its period of length T_k^{d}, while guaranteeing the sum of power demands at any time instant is no larger than the battery power capability (i.e., S1 and S3 hold).

$$\sum_{\lambda_i \in \lambda \setminus \{\lambda_k\}} I_{k \leftarrow i}(T_k^{\mathrm{d}} - L_k^{\mathrm{d}} + \epsilon_t) < (P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot (T_k^{\mathrm{d}} - L_k^{\mathrm{d}} + \epsilon_t), \quad (2)$$

where $I_{k \leftarrow i}(\ell) = W_i(\ell)$, if λ_k has a higher-priority than λ_i or $P_i^{\mathrm{d}} < P_k^{\mathrm{d}}$; $I_{k \leftarrow i}(\ell) = \min(P_i^{\mathrm{d}}, P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}} + \epsilon_p) \cdot \min(L_i^{\mathrm{d}} - \epsilon_t, \ell)$ otherwise.

Proof: As discussed so far, $I_{k \leftarrow i}(T_k^{\mathrm{d}} - L_k^{\mathrm{d}} + \epsilon_t)$ in Eq. (2) is an upper-bound of the amount of energy demanded by instances of λ_i in an interval of length $(T_k^{\mathrm{d}} - L_k^{\mathrm{d}} + \epsilon_t)$. Therefore, if Eq. (2) holds, then there exists an instant t_1 within the interval, such that the sum of power demands is less than or equal to $(P_{\mathtt{bat}}^{\mathrm{s}} - P_k^{\mathrm{d}})$. This means that λ_k can start its execution no later than $(T_k^{\mathrm{d}} - L_k^{\mathrm{d}})$ time units after its release, implying that λ_k finishes its execution within its period. ■

The lemma works not only for an offline power guarantee for the case of the sole supply, but also for a basis to develop an offline power guarantee for the general case to be discussed in Section V. Also, the lemma can be used for addressing a design problem: calculation of the minimum capability of the battery cell that can supply given λ by finding the minimum $P_{\mathtt{bat}}^{\mathrm{s}}$ that satisfies the lemma for given λ.

V. Scheduling of Multiple Power-Demand Operations with Multiple Power-Supply Sources

This section addresses a more general situation than Section IV, in which additional power is sporadically generated from multiple power-supply sources such as an RBS and a solar panel, and immediately stored in the energy buffer or used for power-demand operations. We will first address a scheduling challenge due to the existence of sporadic additional power supply. Then, we present two approaches, depending on how to distribute the additional power supply to power-demand operations.

A. A scheduling challenge

Unlike the situation where a battery pack is the only power-supply source discussed in Section IV, a straightforward

approach cannot yield a system-level power guarantee, as shown in the following example.

Example 1: Suppose that additional power is supplied by Γ_1 in $[t_1, t_2)$, while the battery pack is the only supply in $[t_0, t_1)$ and $[t_2, t_3)$, where $t_0 < t_1 < t_2 < t_3$, as shown in Fig. 6 (a) in the supplementary file [1]. Also, there are two operations λ_1 and λ_2 ready to execute at t_0, and $\lambda_1 \leq P^{\mathtt{s}}_{\mathtt{bat}}$ and $\lambda_1 + \lambda_2 > P^{\mathtt{s}}_{\mathtt{bat}}$, as shown in Fig. 6 (b) in the supplementary file [1]. Suppose that we apply the worst-case-based work-conserving policy in Algorithm 1, implying we start execution of an operation in the ready queue as long as the system has enough remaining power supply to accommodate the worst-case power demand of the operation. Then, λ_2 can start its execution at t_1, but there is a problem at t_2, at which power supplied by Γ_1 ends. In $[t_2, t_3)$, the total amount of power demand is strictly larger than that of power supply, entailing either eviction of one of "non-preemptive" operations, or risking power shortage in executing operations, both of which are considered as a system failure.

Example 1 shows the need for a more fine-grained way to handle additional sporadic power-supply sources. To meet this need, we consider two policies depending on how to distribute the additional supply as follows.

- Calculate the additional "guaranteed" uniform supply $P^{\mathtt{s}}_{\mathtt{uni}}$, meaning that additional power-supply sources (and the energy buffer) can *always* provide power as much as $P^{\mathtt{s}}_{\mathtt{uni}}$ (as the battery pack provides up to $P^{\mathtt{s}}_{\mathtt{bat}}$). This entails the calculation of $P^{\mathtt{s}}_{\mathtt{uni}}$; once it is calculated, we can reuse the power scheduling and analysis framework presented in Section IV, by adding $P^{\mathtt{s}}_{\mathtt{uni}}$ to the existing uniform supply $P^{\mathtt{s}}_{\mathtt{bat}}$. In this case, all operations can share the power generated by additional power-supply sources.
- Assign additional power to a partial set of operations. Power generated by additional power-supply sources (and stored in the energy buffer) is used only when the operations in the partial set are executed. This entails the way to divide power generated by additional power-supply sources for individual power-demand operations.

In what follows, we will detail the above two approaches, including their scheduling frameworks.

B. Uniform supply approach

In this approach, we calculate the additional guaranteed uniform supply $P^{\mathtt{s}}_{\mathtt{uni}}$ from additional power-supply sources. After calculating $P^{\mathtt{s}}_{\mathtt{uni}}$, we can reuse the scheduling framework in Algorithm 1. That is, we just change the $P^{\mathtt{s}}_{\mathtt{bat}}$ term in Line 5 to $P^{\mathtt{s}}_{\mathtt{bat}} + P^{\mathtt{s}}_{\mathtt{uni}}$. The main issue of this approach is to accurately calculate $P^{\mathtt{s}}_{\mathtt{uni}}$; the larger $P^{\mathtt{s}}_{\mathtt{uni}}$, the more operations to be accommodated.

The basic idea to obtain $P^{\mathtt{s}}_{\mathtt{uni}}$ is to calculate the amount of the minimum supplied energy in $[0, t)$ by considering the fact that the energy buffer has at least $P^{\mathtt{s}}_{\mathtt{UC}} \cdot L^{\mathtt{s}}_{\mathtt{UC}}$ of energy at 0 and each additional supply generates power at the end of its instances' periods. If we divide this amount by t and take the minimum, we guarantee to supply power as much as $P^{\mathtt{s}}_{\mathtt{uni}}$ in $[0, t)$, as stated in the following lemma.

Lemma 2: We can calculate $P^{\mathtt{s}}_{\mathtt{uni}}$ using the following equation.

$$P^{\mathtt{s}}_{\mathtt{uni}} = \min_{0 \leq t \leq \mathsf{LCM}} \frac{f(\Gamma_{\mathtt{UC}}, t) + \sum_{\Gamma_i \in \Gamma} f(\Gamma_i, t)}{t}, \tag{3}$$

where LCM is the least common multiple of $\{T^{\mathtt{s}}_i\}_{\Gamma_i \in \Gamma}$,

$$f(\Gamma_{\mathtt{UC}}, t) = P^{\mathtt{s}}_{\mathtt{UC}} \cdot \min(t, L^{\mathtt{s}}_{\mathtt{UC}}), \text{and} \tag{4}$$

$$\begin{aligned} f(\Gamma_i, t) =& P^{\mathtt{s}}_i \cdot \left\lfloor \frac{t}{T^{\mathtt{s}}_i} \right\rfloor \cdot L^{\mathtt{s}}_i \\ &+ P^{\mathtt{s}}_i \cdot \max\left(0, t - \left\lfloor \frac{t}{T^{\mathtt{s}}_i} \right\rfloor \cdot T^{\mathtt{s}}_i - (T^{\mathtt{s}}_i - L^{\mathtt{s}}_i)\right). \end{aligned} \tag{5}$$

Proof: Since the amount of energy in the energy buffer at $t = 0$ is at least $P^{\mathtt{s}}_{\mathtt{UC}} \cdot L^{\mathtt{s}}_{\mathtt{UC}}$, the amount of the supplied energy from the energy buffer in $[0, t)$ is $P^{\mathtt{s}}_{\mathtt{UC}} \cdot t$ if $t \leq L^{\mathtt{s}}_{\mathtt{UC}}$, and at least $P^{\mathtt{s}}_{\mathtt{UC}} \cdot L^{\mathtt{s}}_{\mathtt{UC}}$ otherwise, which is recorded in $f(\Gamma_{\mathtt{UC}}, t)$ of Eq. (4). For given t, $\lfloor \frac{t}{T^{\mathtt{s}}_i} \rfloor$ means the number of instances of λ_i whose periods are completely included in $[0, t)$, and each instance generates energy no smaller than $P^{\mathtt{s}}_i \cdot L^{\mathtt{s}}_i$. The second term of Eq. (5) presents the minimum energy generated by the last instance whose period is partially included in $[0, t)$. Therefore, $\sum_{\Gamma_i \in \Gamma} f(\Gamma_i, t)$ represents the amount of generated energy by Γ in $[0, t)$. Since we assume that the capacity of the energy buffer is sufficiently large, we can always use power as much as the lower-bound of $\frac{f(\Gamma_{\mathtt{UC}}, t) + \sum_{\Gamma_i \in \Gamma} f(\Gamma_i, t)}{t}$ for $0 \leq t \leq \mathsf{LCM}$. ■

If LCM is very large or time-complexity is critically important, e.g., for online admission control for operations, we need a tractable way to calculate $P^{\mathtt{s}}_{\mathtt{uni}}$, which is covered in Lemma 1 of the supplementary file [1].

Finally, we can check a power guarantee of this approach by applying Lemma 1 for all $\lambda_k \in \lambda$ and replacing $P^{\mathtt{s}}_{\mathtt{bat}}$ with $P^{\mathtt{s}}_{\mathtt{bat}} + P^{\mathtt{s}}_{\mathtt{uni}}$.

C. Dedicated supply approach

In this approach, we can determine $\lambda^{\mathtt{ded}}$, a set of operations completely powered by a set of additional power-supply sources Γ and the energy buffer $\Gamma_{\mathtt{UC}}$. Once we determine $\lambda^{\mathtt{ded}}$, operations in $\lambda \setminus \lambda^{\mathtt{ded}}$ can be executed according to Algorithm 1, and their power guarantee is judged by Lemma 1 with $\lambda \setminus \lambda^{\mathtt{ded}}$. On the other hand, each operation in $\lambda^{\mathtt{ded}}$ is fully supplied by Γ with $\Gamma_{\mathtt{UC}}$, and does not use power from $\Gamma_{\mathtt{bat}}$. Our policy is to execute each operation in $\lambda^{\mathtt{ded}}$ at the end of each period, which accommodates more operations in $\lambda^{\mathtt{ded}}$ (because this policy uses less initial energy from the energy buffer). Formally, $\lambda_k \in \lambda^{\mathtt{ded}}$ starts its execution at $r + T^{\mathtt{d}}_k - L^{\mathtt{d}}_k$, where r is the release time of an instance of λ_k, and $\Gamma_{\mathtt{UC}}$ supplies $P^{\mathtt{d}}_k(t)$ ($\leq P^{\mathtt{d}}_k$) amount of power to λ_k in $[r + T^{\mathtt{d}}_k - L^{\mathtt{d}}_k, r + T^{\mathtt{d}}_k)$.

Then, the remaining step is to determine $\lambda^{\mathtt{ded}}$. The basic idea is to calculate the maximum energy demanded by $\lambda^{\mathtt{ded}}$ in $[0, t)$ and the minimum energy supplied by Γ and $\Gamma_{\mathtt{UC}}$ in $[0, t)$. We check whether the former is not larger than the latter at all times, which is stated in the following lemma.

Lemma 3: Every instance of every operation $\lambda_k \in \lambda_{\mathtt{ded}}$ finishes its execution at the end of each period (e.g., $[r + T^{\mathtt{d}}_k -$

$L_k^{\mathrm{d}}, r + T_k^{\mathrm{d}})$ where r is the release time of an instance of λ_k), only with Γ and $\Gamma_{\mathtt{UC}}$, if the following inequality holds for all $t \in [0, \mathsf{LCM})$.

$$\sum_{\lambda_i \in \lambda^{\mathtt{ded}}} f(\lambda_i, t) \leq f(\Gamma_{\mathtt{UC}}, t) + \sum_{\Gamma_i \in \Gamma} f(\Gamma_i, t), \text{and} \tag{6}$$

$$f(\lambda_i, t) = P_i^{\mathrm{d}} \cdot \left\lfloor \frac{t}{T_i} \right\rfloor \cdot L_i^{\mathrm{d}} + P_i^{\mathrm{d}} \cdot \max\left(0, t - \left\lfloor \frac{t}{T_i^{\mathrm{d}}} \right\rfloor \cdot T_i^{\mathrm{d}} - (T_i^{\mathrm{d}} - L_i^{\mathrm{d}})\right). \tag{7}$$

Proof: Since $f(\lambda_i, t)$ in Eq. (7) exhibits the same formula as Eq. (5), it calculates the maximum energy demanded by λ_i in $[0, t)$ when its instances are executed at the end of their periods. Therefore, the LHS (Left-Hand Side) of Eq. (6) is the maximum energy demanded by $\lambda^{\mathtt{ded}}$ in $[0, t)$. On the other hand, the RHS (Right-Hand Side) of the equation is the minimum energy supplied by Γ and $\Gamma_{\mathtt{UC}}$ in $[0, t)$ as explained in Lemma 2. Therefore, the lemma follows. ■

Note that if time-complexity is important, we can use another necessary condition presented in Lemma 2 in the supplementary file [1].

The remaining problem is then how to select $\lambda^{\mathtt{ded}}$ that satisfies Lemma 3 (or Lemma 2). Here we describe a simple, but effective heuristic. We sort $\lambda_i \in \lambda$, based on the ratio of the LHS to the RHS of Eq. (2). If the ratio of λ_i is larger than that of λ_k, we interpret that λ_i is more difficult to satisfy Eq. (2) than λ_j. Therefore, starting from $\lambda^{\mathtt{ded}} = \emptyset$, we repeat the following step until there is no operation to be moved: we select an operation λ_j with the largest ratio among operations in $\lambda \setminus \lambda^{\mathtt{ded}}$ such that $\lambda^{\mathtt{ded}} \cup \{\lambda_j\}$ satisfies Lemma 3, and then add λ_j to $\lambda^{\mathtt{ded}}$.

Finally, we can check the power guarantee of this approach by checking Lemma 1 only with $\lambda_i \in \lambda \setminus \lambda^{\mathtt{ded}}$. Note that for $\lambda_i \in \lambda^{\mathtt{ded}}$, we automatically guarantee their power sufficiency in that $\lambda^{\mathtt{ded}}$ is constructed so as to supply all power demands in $\lambda^{\mathtt{ded}}$ by Γ and $\Gamma_{\mathtt{UC}}$ without $\Gamma_{\mathtt{bat}}$.

VI. Online Power Management

If one performs a set of power-demand operations under a set of power-supply sources that satisfy the offline power guarantee analysis in Section V, there will be extra energy stored in the energy buffer as the analysis is based on the minimum (not actual) power supply. Thus, we propose an online power management framework, which adaptively controls power of the energy buffer effectively to reduce the battery's peak power that achieves the goal in Section III.

Our framework periodically controls power of the energy buffer, as shown in Fig. 3 and Algorithm 2. At t_0, the beginning of each period (for online power management) of length t_{p}, we calculate the amount of energy in $\Gamma_{\mathtt{UC}}$, which is necessary for an offline power guarantee for the current period $[t_0, t_0 + t_{\mathrm{p}})$ (denoted by $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$) as follows. For the uniform power-supply approach in Section V-B, $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$ is simply calculated by $t_{\mathrm{p}} \cdot P^{\mathrm{s}}_{\mathtt{uni}}$ since the offline power guarantee exploits the property that $P^{\mathrm{s}}_{\mathtt{uni}}$ of power is *always* supplied by Γ. For the dedicated supply approach in Section V-C, we calculate

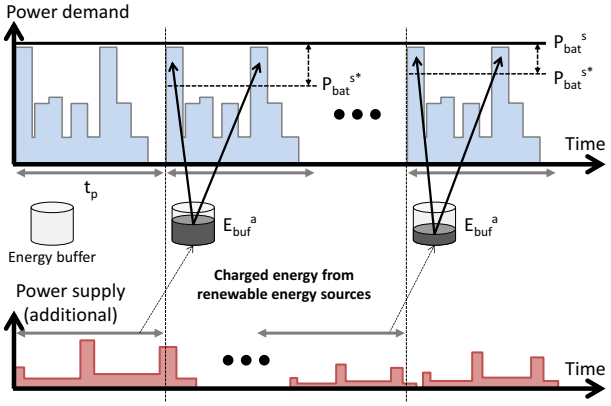

Fig. 3. Online power management

Algorithm 2 Online Power Management

The following steps are performed at t_0, the beginning of each period of length t_{p},

1: Calculate $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$ depending on the uniform/dedicated supply approach.
2: Calculate $E_{\mathtt{buf}}(t_0)$ by $\frac{1}{2} \cdot C_{\mathtt{buf}} \cdot V^2_{\mathtt{buf}}(t_0)$.
3: // Calculate $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$ as follows.
4: $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0) \leftarrow P^{\mathrm{s}}_{\mathtt{bat}}$ // The initial threshold is the battery capacity.
5: $E_{\mathtt{add}} \leftarrow 0$
6: **while** $E_{\mathtt{buf}}(t_0) - E^{\mathrm{m}}_{\mathtt{buf}}(t_0) \geq E_{\mathtt{add}}$ **do**
7: $\quad P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0) \leftarrow P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0) - \epsilon_p$
8: $\quad E_{\mathtt{add}} = \int_{t_0 - t_p}^{t_0} \mathsf{max}(\Sigma P^{\mathrm{d}}_{\mathrm{i}}(t) - P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0), 0)\ dt$
9: **end while**

$E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$ using the amount of energy consumed by $\lambda^{\mathtt{ded}}$ based on their maximum power demand parameters (i.e., P_i^{d}, not $P_i^{\mathrm{d}}(t)$) during the interval.

Once $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$ is calculated, we can utilize the energy from $\Gamma_{\mathtt{UC}}$ up to the difference between the amount of total energy stored in $\Gamma_{\mathtt{UC}}$ at t_0 (denoted by $E_{\mathtt{buf}}(t_0)$) and $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$. Note that we can measure the amount of energy in $\Gamma_{\mathtt{UC}}$ by monitoring the voltage level of $\Gamma_{\mathtt{UC}}$ at t_0 (denoted by $V_{\mathtt{buf}}(t_0)$), using $E_{\mathtt{buf}}(t_0) = \frac{1}{2} \cdot C_{\mathtt{buf}} \cdot V^2_{\mathtt{buf}}(t_0)$, where $C_{\mathtt{buf}}$ is a constant representing the capacitance of $\Gamma_{\mathtt{UC}}$ [16]. Lines 1 and 2 of Algorithm 2 represent the calculation of $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$ and $E_{\mathtt{buf}}(t_0)$.

Then, we utilize the extra energy up to as much as $E_{\mathtt{buf}}(t_0) - E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$, for reducing the battery's peak power. We use energy from the energy buffer if the power usage of the battery pack is larger than a threshold $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$. We determine $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$ for the current period $[t_0, t_0 + t_{\mathrm{p}})$, using the history of the previous period $[t_0 - t_{\mathrm{p}}, t_0)$, which is the amount of actual energy consumption by λ during that period. As shown in Lines 3–9 of Algorithm 2, we repeat the following process: for given $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$, we check if the additional amount energy to keep the peak demand no larger than $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$ (denoted by $E_{\mathtt{add}}$) is not larger than the available energy $E_{\mathtt{buf}}(t_0) - E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$. If yes, we increase $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$ by ϵ_p and repeat the process; otherwise, we stop the process.

Finally, for given $E^{\mathrm{m}}_{\mathtt{buf}}(t_0)$ and $P^{\mathrm{s}\star}_{\mathtt{bat}}(t_0)$, the online power management framework controls the energy stored in the energy buffer within a period $[t_0, t_0 + t_{\mathrm{p}})$ as follows. Suppose that the battery pack should supply X amount of power if there is no supply from $\Gamma_{\mathtt{UC}}$ at t for the purpose of the peak power

reduction.[5] Then, the energy buffer supplies $X - P^{s\star}_{\mathtt{bat}}(t_0)$ of power, only when $X \geq P^{s\star}_{\mathtt{bat}}(t_0)$ and $E_{\mathtt{buf}}(t) \geq E^{m}_{\mathtt{buf}}(t_0)$ hold.

VII. Evaluation

We now evaluate our offline power-supply guarantee analysis and online power management, focusing on whether or not they meet the goals stated in Section III. We first introduce a prototype consisting of sub-devices and a HESS, and the required power-demand operations. We then present experimental results, demonstrating the HESS's power-supply guarantee and reduction of energy dissipation. Fig. 4 shows the overall evaluation process.

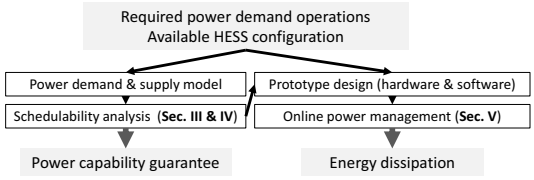

Fig. 4. The evaluation process used

A. Prototype design

We have built a prototype which is equipped with wheels, wheel motors, stepper motors, coolers and a HESS including a pack of lithium-ion batteries, a pack of UCs, an RBS and solar panels as shown in Fig. 5. The required power demand operations and specifications of energy sources/storage are detailed in Tables I and II, and the supplementary file [1]. We can determine the parameters of power-demand operations and power-supply sources from their power demand/supply profiles and specifications. Based on the parameters, we determine the optimal number of batteries achieving a power-supply guarantee for the system via the power guarantee analysis in Sections IV and V. We have then executed various sequences of operations while recording battery states to evaluate the proposed system.

1) System architecture of the prototype: The architecture is required to execute operations of user applications or the system maintenance, and assign the available resources for their execution. Our prototype system consists of a single master and multiple local controllers. The master controller is responsible for scheduling real-time operations using scheduling frameworks in Sections IV and V, and sending messages to the local controller over the CAN bus [40]. Several Arduino boards are used as local controllers to actuate sub-devices according to the messages from the master controller. Our HESS consists of lithium-ion batteries ($\Gamma_{\mathtt{bat}}$), UCs ($\Gamma_{\mathtt{UC}}$), switched-mode converters, regenerative braking system (Γ_3), solar panels (Γ_1, Γ_2), and controllers that can monitor the state of the HESS and communicate with the master controller, all of which are detailed in the supplementary file [1]. The master controller can also regulate power supply of each energy storage via the converter control. Table I shows power supply models from renewable energy sources.

[5]The amount of energy $\Gamma_{\mathtt{UC}}$ (the energy buffer) should supply at t for an offline power guarantee is already figured in X.

Tasks	$T^{\mathrm{s}}(s)$	$L^{\mathrm{s}}(s)$	$P^{\mathrm{s}}(W)$
Γ_1	6	2	3
Γ_2	6	1.5	3
Γ_3	1.5	0.2	0.2
$\Gamma_{\mathtt{UC}}$		3	3
$\Gamma_{\mathtt{bat}}$			40

TABLE I. Power supply

Tasks	$T^{\mathrm{d}}(s)$	$L^{\mathrm{d}}(s)$	$P^{\mathrm{d}}(W)$
λ_1	5	1	12
λ_2	3	1	9.6
λ_3	2	1	6
λ_4	2	1	6
λ_5	3	2	7.2

TABLE II. Power demand operations

2) Applications and tasks: To make our experiments more realistic, we have obtained real driving data from "The US Environmental Protection Agency (EPA)" [41]. Our drive application is programmed to operate wheel motors to achieve the driving profile. Our motor control (λ_1) depends on the PID controller to achieve the required speed, and its control interval (T^{d}_1), maximum acceleration time (time to achieve the target speed, L^{d}_1), and the maximum power (P^{s}_1) are 5s, 1s, and 12W, respectively. To reflect various user applications, we also ran applications that sporadically actuate motors (λ_4 and λ_5). Some applications actuate stepper motors to control position (λ_2), while others (λ_3) use thermal fins to regulate temperature for system thermal stability. The power demand operations are shown in Table II.

B. Evaluation results

Our offline power-supply guarantee analysis and online power management are evaluated in terms of power guarantee and the amount of energy dissipation reduction, respectively.

1) Offline power-supply guarantee analysis: As mentioned in the description of our prototype, we schedule the five operations using the three scheduling frameworks in Sections IV, V-B and V-C, and record the minimum battery capability needed to pass the power guarantee analysis, as follows.

- PS1: the minimum required battery capability under the scheduling framework in Section IV (no additional renewable power sources),
- PS2-UNI: the minimum required battery capability under the scheduling framework in Section V-B (uniform supply approach with renewable sources), and
- PS2-DED: the minimum required battery capability under the scheduling framework in Section V-C (dedicated supply approach with renewable sources).

Fig. 6 shows the battery power profile under the scheduling framework in Section IV. As references, we compare PS1 with MAX and AVG, which denote the sum of the maximum power demand (i.e., $\sum_{\lambda_i \in \lambda} P^{\mathrm{d}}_i$), and the average of the maximum power demand (i.e., $\sum_{\lambda_i \in \lambda} \frac{P^{\mathrm{d}}_i \cdot L^{\mathrm{d}}_i}{T^{\mathrm{d}}_i}$). AVG could not supply the sufficient power in the worst case, because most electric systems do not require power constantly. While MAX and PS1 guarantee power capability only with the battery pack, PS1 (34.8W) reduces the required battery capability, compared

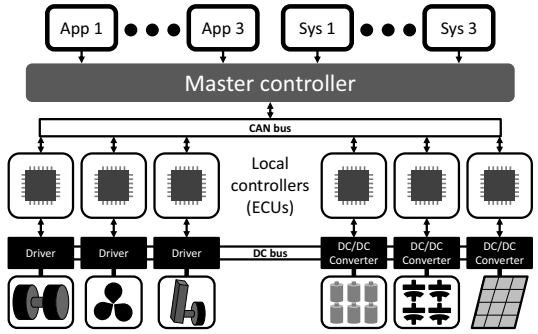

Fig. 5. Prototype overview

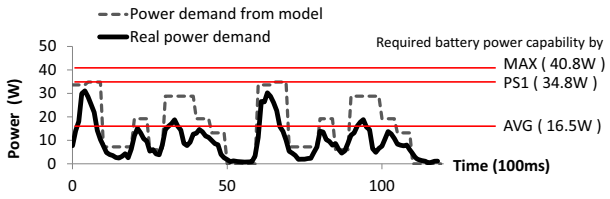

Fig. 6. Power demand profile when the battery pack is the sole power supply

to MAX (40.8W). Note that PS1 can be used as a battery capability when it is difficult to estimate power supply from renewable energy sources.

When it comes to PS2-UNI and PS2-DED, they further reduce the required battery capability (consuming 29.7W and 25.3W, respectively). Between the two, PS2-DED is the most reasonable for a power guarantee, in that it can reduce the required battery capability by 38%, compared to MAX, a naive approach without utilizing renewable energy sources.

2) Online power management: We now compare the average dissipation power in batteries during an operation interval $[0, 600)$, under our online power management (OPM) and the best-effort approach (BE) that enforces the use of buffer energy as long as energy remains in the buffer. Since we have two scheduling frameworks for buffer usage, we have four approaches to compare: PS2-UNI & BE, PS2-UNI & OPM, PS2-DED & BE and PS2-DED & OPM. Note the average dissipation is calculated by $P_{\texttt{loss}} = \frac{1}{t_{\max}} \int_0^{t_{\max}} P^{\texttt{s}}_{\texttt{bat-loss}}(t)\, dt = \frac{1}{t_{\max}} \int_0^{t_{\max}} I^2_{\texttt{bat}}(t) \cdot R_{\texttt{bat}}(t)\, dt$, as mentioned in Section III.

Table III shows the energy dissipation during the operation interval. For a given underlying scheduling framework, OPM significantly reduces the energy dissipation over BE. That is, under PS2-UNI and PS2-DED, the amounts of energy dissipation reduction by OPM over BE are 33.1% and 14.4%, respectively. This observation can be explained using Fig. 7 that presents the battery discharge current profile under PS2-UNI & BE and PS2-UNI & OPM. From the figure, we can easily observe that PS2-UNI & BE fails to supply power when the peak current occurs, while PS2-UNI & OPM effectively reduces the peak current. For example, if we focus on a time instant that exhibits the highest peak current (around $t = 90$), PS2-UNI & OPM successfully reduces the peak current, while PS2-UNI & BE cannot. This is because PS2-UNI & BE consumes much energy when power demand is low (around $t = 80$), as shown in the figure.

Schemes	$P_{\texttt{loss}}(mW)$
PS2-UNI & BE	526
PS2-UNI & OPM	352
PS2-DED & BE	390
PS2-DED & OPM	334

TABLE III. AVERAGE DISCHARGE STRESS ENERGY DISSIPATION $P_{\texttt{loss}}$

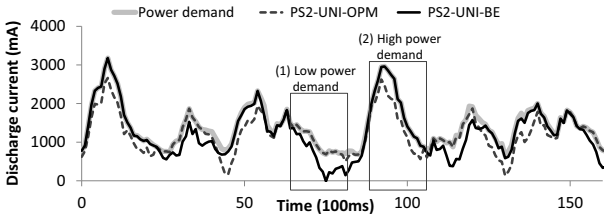

Fig. 7. Battery discharge current profile under PS2-UNI & BE and PS2-UNI & OPM, which shows that our online power management helps reduce peak battery discharge current considering energy generation rate and power demand history

On the other hand, if we compare the underlying scheduling frameworks, PS2-DED outperforms PS2-UNI. However, the gap between the two is significantly reduced if we apply OPM (352 versus 334, as opposed to 526 versus 390). This also substantiates the effectiveness of OPM in reducing energy dissipation.

VIII. CONCLUSION

In this paper, we have proposed a power scheduling framework—as a guideline for the design of a HESS—that ensures power sufficiency for the worst-case power demand and supply. To improve the runtime performance of the HESS further, we have also designed an online power management framework that utilizes the surplus energy from a real-time power supply, reducing the battery's peak power and hence extending its lifetime. We have validated the design with a HESS-powered prototype system running realistic applications, demonstrating power sufficiency with a lower-cost HESS and higher energy-efficiency.

In future, we would like to develop a design framework for general energy storage systems based on a power guarantee analysis. It will search for the optimal configurations of energy storage systems at their design or replacement time while considering power demand history and power supply's state-of-health. We also plan to build a power/energy management system that not only schedules power demand and supply, but also monitors and pro/diagnoses energy storages/sources.

ACKNOWLEDGEMENT

The work reported in this paper was supported in part by the NSF under grants CNS-1446117 and CNS-1138200. This work was also supported by LG Chem Ltd., and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2014R1A1A1035827). This work was also supported by the MISP (Ministry of Science, ICT & Future Planning), Korea, under the National Program for Excellence in SW (R2215-16-1005) supervised by the IITP (Institute for Information & communications Technology Promotion). Jinkyu Lee is the corresponding author.

REFERENCES

[1] E. Kim, J. Lee, L. He, Y. Lee, and K. G. Shin. (2016) Supplement: Offline guarantee and online management of power demand and supply in cyber-physical systems. [Online]. Available: https://kabru.eecs.umich.edu/wordpress/wp-content/uploads/supp_psched.pdf

[2] P. D. Blair, "Modeling energy and power requirements of electric vehicles," *Energy Conversion*, vol. 18, no. 3, pp. 127 – 134, 1978.

[3] G. Cai, L. Feng, B. M. Chen, and T. H. Lee, "Systematic design methodology and construction of {UAV} helicopters," *Mechatronics*, vol. 18, no. 10, pp. 545 – 558, 2008.

[4] E. Pastor, J. Lopez, and P. Royo, "Uav payload and mission control hardware/software architecture," *IEEE Aerospace and Electronic Systems Magazine*, vol. 22, no. 6, pp. 3–8, June 2007.

[5] C. Chan and K. Chau, "An overview of power electronics in electric vehicles," *Industrial Electronics, IEEE Transactions on*, vol. 44, no. 1, pp. 3–13, Feb 1997.

[6] T.-K. Lee, B. Adornato, and Z. Filipi, "Synthesis of real-world driving cycles and their use for estimating phev energy consumption and charging opportunities: Case study for midwest/u.s." *Vehicular Technology, IEEE Transactions on*, vol. 60, no. 9, pp. 4153–4163, 2011.

[7] E. Kim, J. Lee, and K. G. Shin, "Real-time prediction of battery power requirements for electric vehicles," in *ACM/IEEE 4th International Conference on Cyber-Physical Systems (ICCPS13)*, Philadelphia, PA, Apr 2013.

[8] Z. Li, P. C. Huang, A. K. Mok, T. Nghiem, M. Behl, G. Pappas, and R. Mangharam, "On the feasibility of linear discrete-time systems of the green scheduling problem," in *Real-Time Systems Symposium (RTSS), 2011 IEEE 32nd*, Nov 2011, pp. 295–304.

[9] D. Caprino, M. L. D. Vedova, and T. Facchinetti, "Peak shaving through real-time scheduling of household appliances," *Energy and Buildings*, vol. 75, pp. 133 – 148, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378778814001248

[10] T. Facchinetti and M. L. D. Vedova, "Real-time modeling for direct load control in cyber-physical power systems," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 689–698, Nov 2011.

[11] G. Benetti, M. Delfanti, T. Facchinetti, D. Falabretti, and M. Merlo, "Real-time modeling and control of electric vehicles charging processes," *IEEE Transactions on Smart Grid*, vol. 6, no. 3, pp. 1375–1385, May 2015.

[12] S. Pay and Y. Baghzouz, "Effectiveness of battery-supercapacitor combination in electric vehicles," in *Power Tech Conference Proceedings, 2003 IEEE Bologna*, vol. 3, June 2003, pp. 6 pp. Vol.3–.

[13] R. Kaiser, "Optimized battery-management system to improve storage lifetime in renewable energy systems," *Journal of Power Sources*, vol. 168, no. 1, pp. 58 – 65, 2007.

[14] S. Park, Y. Kim, and N. Chang, "Hybrid energy storage systems and battery management for electric vehicles," in *Design Automation Conference (DAC), 2013 50th ACM / EDAC / IEEE*, May 2013, pp. 1–6.

[15] J. Cao and A. Emadi, "A new battery/ultracapacitor hybrid energy storage system for electric, hybrid, and plug-in hybrid electric vehicles," *Power Electronics, IEEE Transactions on*, vol. 27, no. 1, pp. 122–132, Jan 2012.

[16] E. Kim, K. Shin, and J. Lee, "Real-time discharge/charge rate management for hybrid energy storage in electric vehicles," in *Real-Time Systems Symposium (RTSS), 2014 IEEE*, Dec 2014, pp. 228–237.

[17] M. Ortuzar, J. Moreno, and J. Dixon, "Ultracapacitor-based auxiliary energy system for an electric vehicle: Implementation and evaluation," *Industrial Electronics, IEEE Transactions on*, vol. 54, no. 4, pp. 2147–2156, Aug 2007.

[18] J. K. Shiau, D. M. Ma, P. Y. Yang, G. F. Wang, and J. H. Gong, "Design of a solar power management system for an experimental uav," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 4, pp. 1350–1360, Oct 2009.

[19] N. Baldock and M. MokhtarzadehDehghan, "A study of solarpowered, highaltitude unmanned aerial vehicles," *Aircraft Engineering and Aerospace Technology*, vol. 78, no. 3, pp. 187–193, 2006.

[20] M. Z. Jacobson and M. A. Delucchi, "Providing all global energy with wind, water, and solar power, part i: Technologies, energy resources, quantities and areas of infrastructure, and materials," *Energy Policy*, vol. 39, no. 3, pp. 1154 – 1169, 2011.

[21] J. Paska, P. Biczel, and M. Kos, "Hybrid power systems an effective way of utilising primary energy sources," *Renewable Energy*, vol. 34, no. 11, pp. 2414 – 2421, 2009.

[22] S. Letendre, R. Perez, and C. Herig, "Battery-powered, electric-drive vehicles providing buffer storage for pv capacity value," in *proceedings of the solar conference*, 2002, pp. 105–110.

[23] R. Baos, F. Manzano-Agugliaro, F. Montoya, C. Gil, A. Alcayde, and J. Gmez, "Optimization methods applied to renewable and sustainable energy: A review," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 4, pp. 1753 – 1766, 2011.

[24] O. Erdinc and M. Uzunoglu, "Optimum design of hybrid renewable energy systems: Overview of different approaches," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 3, pp. 1412 – 1425, 2012.

[25] E. Kim, J. Lee, and K. G. Shin, "Modeling and real-time scheduling of large-scale batteries for maximizing performance," in *Real-Time Systems Symposium, 2015 IEEE*, Dec 2015, pp. 33–42.

[26] W. Zhou, C. Lou, Z. Li, L. Lu, and H. Yang, "Current status of research on optimum sizing of stand-alone hybrid solarwind power generation systems," *Applied Energy*, vol. 87, no. 2, pp. 380 – 389, 2010.

[27] M. Majima, S. Ujiie, E. Yagasaki, K. Koyama, and S. Inazawa, "Development of long life lithium ion battery for power storage," *Journal of Power Sources*, vol. 101, no. 1, pp. 53 – 59, 2001.

[28] S. S. Choi and H. S. Lim, "Factors that affect cycle-life and possible degradation mechanisms of a li-ion cell based on licoo2," *Journal of Power Sources*, vol. 111, no. 1, pp. 130 – 136, 2002.

[29] H. Kim and K. G. Shin, "Scheduling of battery charge, discharge, and rest," in *Proceedings of the 30th IEEE Real-Time Systems Symposium*, 2009, pp. 13–22.

[30] L. Benini, A. Macii, E. Macii, M. Poncino, and R. Scarsi, "Scheduling battery usage in mobile systems," *IEEE Transactions on VLSI systems*, vol. 11, no. 6, pp. 1136–1143, 2003.

[31] C. Liu and J. Layland, "Scheduling algorithms for multi-programming in a hard-real-time environment," *Journal of the ACM*, vol. 20, no. 1, pp. 46–61, 1973.

[32] C. Lv, J. Zhang, Y. Li, and Y. Yuan, "Regenerative braking control algorithm for an electrified vehicle equipped with a by-wire brake system." SAE International, 04 2014.

[33] H. Gao, Y. Gao, and M. Ehsani, "Design issues of the switched reluctance motor drive for propulsion and regenerative braking in ev and hev." SAE International, 08 2001.

[34] R. Carter and A. Cruden, "Strategies for control of a battery/supercapacitor system in an electric vehicle," in *Power Electronics, Electrical Drives, Automation and Motion, 2008. SPEEDAM 2008. International Symposium on*, June 2008, pp. 727–732.

[35] J. Zhang, S. Ci, H. Sharif, and M. Alahmad, "Modeling discharge behavior of multicell battery," *Energy Conversion, IEEE Transactions on*, vol. 25, no. 4, pp. 1133–1141, 2010.

[36] S. Kato and Y. Ishikawa, "Gang EDF scheduling of parallel task systems," in *Proceedings of IEEE Real-Time Systems Symposium*, 2009, pp. 459–468.

[37] J. Goossens and V. Berten, "Gang FTP scheduling of periodic and parallel rigid real-time tasks," in *Proceedings of the 18th International Conference on Real-Time Networks and Systems*, 2010.

[38] V. Berten, P. Courbin, and J. Goossens, "Gang fixed priority scheduling of periodic moldable real-time tasks," in *Proceedings of the Junior Researcher Workshop Session of the 19th International Conference on Real-Time and Network Systems*, 2011.

[39] M. Bertogna and M. Cirinei, "Response-time analysis for globally scheduled symmetric multiprocessor platforms," in *Proceedings of IEEE Real-Time Systems Symposium*, 2007.

[40] S. Corrigan. Introduction to the controller area network (can). http://www.ti.com/lit/an/sloa101a/sloa101a.pdf.

[41] Dynamometer drive schedules. [Online]. Available: https://www.epa.gov/vehicle-and-fuel-emissions-testing/dynamometer-drive-schedules