Optimal Power Allocation for Hybrid Energy Harvesting and Power Grid Coexisting System With Power Upper Bounded Constraints

Peter He, *Member, IEEE*, Lian Zhao, *Senior Member, IEEE*, and Bala Venkatesh, *Senior Member, IEEE*

***Abstract*—As one of the green energy resources, the technique of energy harvesting harnesses energy from its surrounding environment. In this setting, a power grid is also utilized to serve as a supplementary source to regulate the not-so-stable harvested energy supply of the system. The power allocated to the user(s) from the sum of the harvested energy and the power grid is subject to peak power constraints. The background of these constraints comes from field requirements, such as avoiding the saturation of power allocated to the user(s), avoiding system level out-of-band power leakage, and reducing interference with other transmitter(s) due to the nonlinearity generated via the transmitting mechanisms to the user(s). The proposed problem considers simultaneously 1) the hybrid paradigm of both energy harvesting and grid power supplies, and 2) the peak power constraints in such systems. For our proposed problem, the most efficient known-to-date and popular convex optimization method of primal-dual interior method (PD-IPM) only computes an ϵ solution, not an optimal solution, even with more computations. The novelty of the proposed algorithms is that they compute the exact solutions with the low degree polynomial computational complexity. To the best of the authors' knowledge, under the same assumptions, no prior publication, including PD-IPM, can arrive at such results. Numerical examples also illustrate efficiency of the proposed algorithms.**

***Index Terms*—Energy harvesting, exact solution, optimal power allocation, optimization theory and methods, smart power grid, water-filling algorithm with mixed constraints.**

I. Introduction

A. Background

WIRELESS devices are normally powered by batteries, which need to be either replaced or recharged periodically. One possible technique to overcome this limitation is to harvest energy from the surrounding environment, via energy harvesting devices such as vibration absorption devices, solar energy, wind energy, thermal energy, and other clean forms of energy sources [1]. In developing green communication systems, energy harvesting has become a preferred choice for supporting "the green communication". Such a system is normally modelled as a sequence of epochs, at the ends of which, the fading gain and the harvested energy arrival are observed or predictable. This system setting leads to new design and insights in a wireless link with a rechargeable transmitter and fading channels [2]. On prediction, many techniques can be used, such as the time series prediction. In this paper, the system parameters are assumed to be predictable, like others, *e.g.* [3]–[5].

There has been much recent research effort on understanding data transmission in this kind of systems, for example, the investigation of power allocation policies [6]–[9], medium access control protocols [10], adaptive opportunistic routing protocols [11], [12], network throughput of a mobile ad hoc network powered by energy harvesting [13], and energy management in wireless sensor networks [8], [14], *etc*. In [15], optimality of a variant of the back pressure algorithm using energy queues is discussed in the setting of wireless networks with rechargeable batteries. In [16], transmitters with energy harvesting and batteries with finite energy are considered for minimizing the weighted sum of the outage probabilities under a set of predetermined transmission rates over a finite horizon. It applied such an approximation while assuming high signal-to-noise ratios and obtained a near-optimal offline solution. As the fundamental work for transmission with energy harvesting in wireless communications, [3] investigated the throughput maximization problem with full side information and proposed some approaches that made use of the water-filling algorithm to solve the Karush-Kuhn-Tucker (KKT) conditions [17] of the target optimization problem. Different from these previous works, we recursively applied our proposed Geometric Water-Filling (GWF) [18] to solve the throughput maximization problem and transmission time minimization problem, as demonstrated in the more recent paper [19].

Since the available harvested energy depends on many environmental conditions, it is not considered as a stable energy source. Energy drawn from a smart power grid is needed and is considered as a supplementary source to regulate the overall energy flow of the system. With this kind of hybrid energy source systems, the proposed problem of optimal power allocation to maximize system throughput turns out to be far more complicated. Recently, [20] investigated the issues of power allocation problems to minimize the grid power consumption mainly with predictable energy and data arrival, and analyzed the structure of the optimal power allocation policy in some special cases. In

Manuscript received April 7, 2016; revised July 20, 2016 and August 15, 2016; accepted August 15, 2016. Date of publication September 8, 2016; date of current version February 7, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yonggang Wen.

The authors are with the Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada (e-mail: phe@ee.ryerson.ca; lzhao@ee.ryerson.ca; bala@ee.ryerson.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSIPN.2016.2607123

the most recent research [21], an efficient algorithm was proposed to compute the solution to the optimal power allocation problem, which is the MIMO case of the energy harvesting and power grid hybrid systems.

B. Our Work

In this paper, the GWF and recursion machinery are further exploited to obtain the (exact) optimal solution to the maximum throughput problems for the hybrid energy harvesting and power grid communication system with additional upper bound power constraints, *i.e.*, peak power constraints. The background of these constraints comes from field requirements, such as, avoiding the saturation of power allocated to the user(s) and avoiding system level out-of-band power leakage, via the transmitting mechanisms to the user(s). This set of the peak power constraints leads to more challenges in (exactly) solving the target problem. To the best of the authors' knowledge, no solution has been reported in the open literature.

Besides proposing a more difficult problem, this paper proposes algorithms that solve this difficult problem with the following characteristics: 1) the proposed algorithms can compute the exact optimal solution, with the polynomial computational complexity, against the ϵ solution (will be elaborated in more details in Numerical Results section); 2) their optimality is strictly proven; and 3) the proposed approach can be exploited to solve the minimum time and the minimum sum power problems, and the corresponding stochastic (*i.e.*, online,) systems, as well. The last issue is our on-going research topics. In this paper, we shall focus on the first two points only.

The peak power constraints mentioned in this paper form a "cubical" constraint in a higher dimensional Euclidean space [22]. Pontryagin [22] pointed out that this cubical non-trivia constraint sets modern optimization apart from the classical ones. The difference between modern optimization and the classical ones lies in the fact: the solution of the former is often at the boundary of a compact set; while that of the latter is an interior point of an open set. As a direct consequence, the optimality condition of the former is considered to be much more complicated than that of latter. [22] utilizes the famous maximum principle in an attempt to solve this class of modern problems with the cubical constraint [22, p. 3]. To solve an optimization problem, method A is regarded as being better than method B if method A can compute the exact solution with less finite amount of computation. Under this meaning, to the best of the authors' knowledge, our proposed algorithms are considered to be the best way so far to solve the target problem. At the same time, we prove optimality of the obtained solution, and present complexity analysis to show the differences from existing methods. The proposed problem is shown to be much more difficult and its solution is shown to be quite different. The proposed problem can be further regressed into a problem where energy harvesting is the sole energy source with the peak power constraints of the harvested power or energy. For this individual case of our general problem, our proposed algorithms could also handle it successfully, unlike Primal-Dual Interior point Method (PD-IPM), the most efficient and popular convex optimization method known to date.

Algorithm 1: Pseudocode of HPA1 (for EH with Peak Power Constraints).

1: Initialize:
   $L = n = 1, K, P \longleftarrow E_{in}(1), \overline{P}_1, w_1, a_1$;
2: Output the result for epoch 1:
   $s_1^* =$ GWFPP$(1, 1, w_1, a_1, P)$, *i.e.*, $s_1^* = E_{in}(L)$,
   $\overline{s}_1^* =$HPA1(1)$= \min\{\overline{P}_1, s_1^*\}$;
3: **for** $L = 2 : 1 : K$ **do**
4: Input: $\{E_{in}(L), \overline{P}_L, w_L, a_L\}$;
5: $E_{in} \longleftarrow E_{in}(L) + \sum_{k=n}^{L-1}(E_{in}(k) - \overline{s}_k^*)^+$;
6: $\{s_k'\}_{k=1}^{L-1} =$ HPA1$(L-1)$;
7: **if** $E_{in} \geq \overline{P}_L$ **then**
8: $\overline{s}_L^* = \overline{P}_L$, HPA1$(L) = \{s_1', \cdots, s_{L-1}', \overline{s}_L^*\}$ and Continue;
9: **end if**
10: **for** $n = L : -1 : 1$ **do**
11: $W = \{w_j\}_{j=n}^{L}$; $A = \{a_j\}_{j=n}^{L}$;
12: $S_T = E_{in}$;
13: $\{\{s_k^*\}_{k=n}^{L}, k^*\} =$ GWFPP$(n, L, W, A, S_T, \{\overline{P}_k\}, \delta = 0)$;
14: **if** $n > 1$ **then**
15: $c_k'(s_k') = \frac{1}{a_k w_k} + \frac{s_k'}{w_k}$,
   $k_{\mathrm{e}}^* = \max\{\arg(\max\{c_k' | s_k' > 0, 1 \leq k \leq n-1\})\}$,
   or else $k_{\mathrm{e}}^* = k^*$;
16: **end if**
17: **if** $\overline{P}_n \geq s_n^*$ and $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}^*}{w_{k^*}} \geq \frac{1}{a_{k_{\mathrm{e}}^*} w_{k_{\mathrm{e}}^*}} + \frac{{s_{k_{\mathrm{e}}^*}}'}{w_{k_{\mathrm{e}}^*}}$ **then**
18: $\overline{s}_n^* = s_n^*$, HPA1$(L) = \{s_1', \cdots, s_{n-1}', \overline{s}_n^*, \cdots, \overline{s}_L^*\}$
   Break and go to Line 25;
19: **end if**
20: $\{s_k'\}_{k=n}^{L} =$ GWFPP$(n, L, S_T, \{\overline{P}_k\}, \delta = 1)$;
21: $E_{in} = s_{n-1}' + S_T$;
22: **end for**
23: $\{\overline{s}_k^* = s_k'\}_{k=n}^{L}$;
24: HPA1(L)$= \{s_1', \ldots, s_L'\}$;
25: **end for**
26: Output HPA1(K): $\{s_k^*\}_{k=1}^{K} =$HPA1(K).

Algorithm 2: Statement of HPA2 (for EH+PG), Based on HPA1 with Peak Power Constraints.

1: Compute $\{s_{(H,i)}^*\}_{i=1}^{K}$, as follows:
2: Input the parameters $\{E_{in}(i), \overline{P}_i, w_i, a_i\}_{i=1}^{K}$, from (2);
3: Using HPA1(K);
4: $\{s_{(H,i)}^*\}_{i=1}^{K}$=HPA1(K);
5: Compute $\{s_{(G,i)}^*\}_{i=1}^{K}$, as follows:
6: Input the parameters $\{E_{in}(i), \overline{P}_i, w_i, a_i\}_{i=1}^{K}$, from (2);
7: Update $E_{in}(1) \longleftarrow E_{in}(1) + E_{(G,\text{total})}$;
8: Using HPA1(K);
9: $\{s_i^*\}_{i=1}^{K}$=HPA1(K);
10: $\{s_{(G,i)}^* \longleftarrow (s_i^* - s_{(H,i)}^*)\}_{i=1}^{K}$;
11: Define HPA2(K), as follows:
12: HPA2(K)$|_I = \{s_{(H,i)}^*\}_{i=1}^{K}$;
   HPA2(K)$|_{II} = \{s_{(G,i)}^*\}_{i=1}^{K}$;
13: Exit the algorithm;

In the remaining of this paper, system model and problem statement are presented in Section II. Energy harvesting power allocation HPA1 for maximum throughput with the peak power constraints, is investigated in Section III. Its extensions: Hybrid power allocation Algorithm 1 and Algorithm 2 (HPA2

Algorithm 3: Statement of HPA2-R (for EH+PG), Recursion Version.

1: Initialize:
   $L = n = 1, K, \overline{P}_1, w_1, a_1; \quad E \longleftarrow E_{(G,\text{total})}, P \longleftarrow E_{in}(1);$
2: Output the result for epoch 1:
3: $s_1^* =$ GWFPP$(1, 1, w_1, a_1, P)$, *i.e.*, $s_1^* = E_{in}(L)$,
4: $s_{H,1}^* =$HPA2-R$(1)|_I = \min\{\overline{P}_1, s_1^*\}$,
5: $E_{in} \longleftarrow E + P;$
6: $s_1^* =$ GWFPP$(1, 1, w_1, a_1, E_{in})$, *i.e.*, $s_1^* = E_{in}$,
7: HPA2-R$(1)|_{II} = \min\{\overline{P}_1, s_1^*\}$;
8: **for** $L = 2:1:K$ **do**
9: Input: $\{E_{in}(L), \overline{P}_L, w_L, a_L\}$;
10: $E_{in} \longleftarrow E_{in}(L) + \sum_{k=n}^{L-1}(E_{in}(k) - \overline{s}_k^*)^+;$
11: $\{s'_k\}_{k=1}^{L-1} =$HPA2-R$(L-1)|_I$;
12: **if** $E_{in} \geq \overline{P}_L$ **then**
13: $\overline{s}_L^* = \overline{P}_L$, HPA2-R$(L)|_I = \{s'_1, cdots, s'_{L-1}, \overline{s}_L^*\}$
   and go to Line 30;
14: **end if**
15: **for** $n = L : -1 : 1$ **do**
16: $W = \{w_j\}_{j=n}^L; A = \{a_j\}_{j=n}^L;$
17: $S_T = E_{in};$
18: $\{\{s_k^*\}_{k=n}^L, k^*\} =$ GWFPP$(n, L, W, A, S_T, \{\overline{P}_k\}, \delta = 0)$;
19: **if** $n > 1$ **then**
20: $k_\text{e}^* = \max\{\arg(\max\{s'_k > 0, 1 \leq k \leq n-1\})\}$, or else $k_\text{e}^* = k^*$;
21: **end if**
22: **if** $\overline{P}_n \geq s_n^*$ and $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}^*}{w_{k^*}} \geq \frac{1}{a_{k_\text{e}^*} w_{k_\text{e}^*}} + \frac{s_{k_\text{e}^*}{}'}{w_{k_\text{e}^*}}$ **then**
23: $\overline{s}_n^* = s_n^*$, HPA2-R$(L)|_I = \{s'_1, \cdots, s'_{n-1}, \overline{s}_n^*, \cdots, \overline{s}_L^*\}$
   Break and go to Line 30;
24: **end if**
25: $\{s'_k\}_{k=n}^L =$ GWFPP$(n, L, S_T, \{\overline{P}_k\}, \delta = 1)$;
26: $E_{in} = s'_{n-1} + S_T;$
27: **end for**
28: $\{\overline{s}_k^* = s'_k\}_{k=n}^L;$
29: HPA2-R(L)$|_I = \{s'_1, \ldots, s'_L\}$;
30: $E_{in} \longleftarrow E + E_{in};$
31: $\{s'_k\}_{k=1}^{L-1} =$HPA2-R$(L-1)|_{II}$;
32: **if** $E_{in} \geq \overline{P}_L$ **then**
33: $\overline{s}_L^* = \overline{P}_L$, HPA2-R$(L)|_{II} = \{s'_1, cdots, s'_{L-1}, \overline{s}_L^*\}$
   and then Continue;
34: **end if**
35: **for** $n = L : -1 : 1$ **do**
36: $W = \{w_j\}_{j=n}^L; A = \{a_j\}_{j=n}^L;$
37: $S_T = E_{in};$
38: $\{\{s_k^*\}_{k=n}^L, k^*\} =$ GWFPP$(n, L, W, A, S_T, \{\overline{P}_k\}, \delta = 0)$;
39: **if** $n > 1$ **then**
40: $k_\text{e}^* = \max\{\arg(\max\{s'_k > 0, 1 \leq k \leq n-1\})\}$, or else $k_\text{e}^* = k^*$;
41: **end if**
42: **if** $\overline{P}_n \geq s_n^*$ and $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}^*}{w_{k^*}} \geq \frac{1}{a_{k_\text{e}^*} w_{k_\text{e}^*}} + \frac{s_{k_\text{e}^*}{}'}{w_{k_\text{e}^*}}$ **then**
43: $\overline{s}_n^* = s_n^*$, HPA2-R$(L)|_{II} = \{s'_1, \cdots, s'_{n-1}, \overline{s}_n^*, \cdots, \overline{s}_L^*\}$
   Break and go to Line 50;
44: **end if**
45: $\{s'_k\}_{k=n}^L =$ GWFPP$(n, L, S_T, \{\overline{P}_k\}, \delta = 1)$;
46: $E_{in} = s'_{n-1} + S_T;$
47: **end for**
48: $\{\overline{s}_k^* = s'_k\}_{k=n}^L;$
49: HPA2-R$(L)|_{II} = \{s'_1, \ldots, s'_L\}$;
50: **end for**
51: Output HPA2-R(K):

$$\{s_{H,k}^*\}_{k=1}^K = \text{HPA2-R}(K)|_I;$$
$$\{s_{G,k}^*\}_{k=1}^K = \text{HPA2-R}(K)|_{II} = \{s'_k - s_{H,k}^*\}_{k=1}^{n-1} \cup \{\overline{s}_k^* - s_{H,k}^*\}_{k=n}^K.$$

and HPA2-R), are investigated in Section IV. Numerical examples and computational complexity analysis are presented in Section V. Section VI concludes the paper.

II. Problem Statement

In this section, the model of energy harvesting and power grid coexisting system with the power upper bound constraints in a fading channel is first presented, followed by the optimization problem to maximize the throughput in such a system. For convenience and without loss of generality, the process is assumed to be a discrete time process.

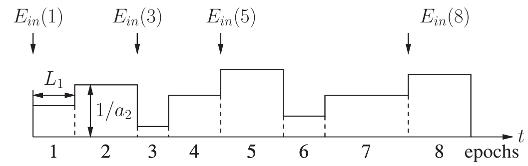


Fig. 1. System model, $K = 8$ epochs in $(0, T]$

As shown in Fig. 1, the system model depicts the time period from $(0, T]$ including K epochs. Let L_i and a_i denote the time duration and the fading channel gain of the ith epoch, where $i = 1, \ldots, K$. Without loss of generality, assume $L_i > 0, a_i > 0, \forall i$. At the beginning of the ith epoch, the harvested energy that is available is denoted by $E_\text{in}(i)$, and it is depicted as $E_\text{in}(i) \geq 0$. Besides the harvested energy $E_\text{in}(i)$, the transmission is also connected with the smart power grid. Let $E_{(G,\text{total})}$ denote the energy budget of the total energy, supported by the power grid.

We assume that $\overline{P}_i \geq 0$, for the peak power constraint on the powers from the energy harvesting and the power grid for the ith epoch, $\forall i$.

For the dual-energy-source system, we restrict our attention to power management strategies with constant transmit power in each epoch, and find the optimal one among these strategies. Therefore, let us denote the transmit power at epoch i by s_i ($i = 1, \ldots, K$), which consists of the power from harvested energy, $s_{H,i}$, and the power from the smart power grid, $s_{G,i}$. The objective is to maximize the total throughput of the user by the deadline T, *i.e.*, within the K epochs. We have causal constraints that the currently harvested energy cannot be used in the previous epochs but only in the following epochs. For simplification, an infinite energy storage capacity is assumed. Hence, the optimization problem to maximize the throughput in this hybrid system can be written as:

$$\begin{aligned}
\max_{\{s_{H,i}, s_{G,i}\}_{i=1}^K} \quad & \sum_{i=1}^K \frac{L_i}{2} \log\left(1 + a_i(s_{H,i} + s_{G,i})\right) \\
\text{Subject to:} \quad & 0 \leq s_{H,i}, \forall i; \\
& 0 \leq s_{G,i}, \forall i; \\
& s_{H,i} + s_{G,i} \leq \overline{P}_i, \forall i; \\
& \sum_{i=1}^l L_i s_{H,i} \leq \sum_{i=1}^l E_\text{in}(i), \\
& \quad \text{for } l = 1, \ldots, K; \\
& \sum_{i=1}^K L_i s_{G,i} \leq E_{(G,\text{total})}.
\end{aligned} \tag{1}$$

As a side note, the logarithm function used by the objective function above takes the number of 2 as its base. Also, a_i in the objective function is the reciprocal of the noise variance times the square of channel gain, $\forall i$.

In this optimal power allocation problem, the first two constraints account for the nonnegative powers from harvested energy and grid respectively; the fourth constraint accounts for the causal requirement; and the fifth constraint reflects the maximum energy available from the smart power grid. Different from [21], the insertion of the third constraint reflects the peak power constraints.

The observed properties of the optimal harvested power allocation can be interpreted by water-filling vividly: $E_{\text{in}}(i)$ units of water is filled into a rectangle container with bottom width $\frac{L_i}{2}, \forall i$. Note that the last weighted power sum constraint from energy harvesting (the fourth constraint in Problem (1)) cannot be guaranteed to be equality, unlike the case when there is no peak power constraint. Furthermore, for unifying parameter notation, through a change of variables, we can obtain an equivalent target problem as follows:

$$\begin{aligned}\max_{\{s_{H,i},s_{G,i}\}_{i=1}^K} \quad & \sum_{i=1}^{K} w_i \log\left(1 + a_i(s_{H,i} + s_{G,i})\right) \\ \text{subject to:} \quad & 0 \le s_{H,i},\ \forall i; \\ & 0 \le s_{G,i},\ \forall i; \\ & s_i = s_{H,i} + s_{G,i} \le \overline{P}_i,\ \forall i; \\ & \sum_{i=1}^{l} s_{H,i} \le \sum_{i=1}^{l} E_{\text{in}}(i), \forall l; \\ & \sum_{i=1}^{K} s_{G,i} \le E_{(G,\text{total})}, \end{aligned} \tag{2}$$

where $\frac{L_i}{2} \longrightarrow w_i, \frac{a_i}{L_i} \longrightarrow a_i,\ L_i s_{H,i} \longrightarrow s_{H,i},\ L_i s_{G,i} \longrightarrow s_{G,i}$ and $L_i \overline{P}_i \longrightarrow \overline{P}_i$, for any i. Note that the symbol "$\longrightarrow$" is the assignment operator from the left side to the right side; and the symbol "$\longleftarrow$" is that from right side to left side. Without consideration of trivial cases, $\overline{P}_i > 0$, $E_{\text{in}}(i) > 0$ and $E_{(G,\text{total})} > 0$ can be assumed.

Due to the existence of the peak power constraints (as in the third constraint in Problem (1) or (2)), the water level non-decreasing condition (monotonicity) discussed in [19] doesn't hold anymore.

III. Algorithm HPA1 Computing the Case Without Power Grid

This section presents the proposed Power allocation algorithm: HPA1, to compute the optimal solution to maximize throughput under energy harvesting with the peak power constraints but without power grid. HPA1 cannot directly compute an optimal solution to the target problem (2) yet. However, HPA1 will serve as a precursor for Algorithm HPA2 that handles the target problem (2).

A. Preparations for HPA1

This subsection introduces tools needed for HPA1.

1) GWFPP (without switching): Since HPA1 uses the algorithm: Geometric Water Filling with Peak Power constraints (GWFPP) with Switching, GWFPP without switching proposed in [18] is first revisited below for gradual transition to GWFPP with Switching.

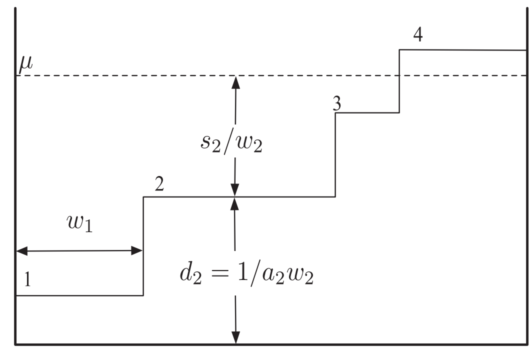


Fig. 2. Illustration for s_i/w_i, and $1/(a_i \cdot w_i)$.

GWFPP aims at computing the solution to the following problem.

$$\begin{aligned}\max_{\{s_k\}_{k=1}^K} \quad & \sum_{k=1}^{K} w_k \log(1 + a_k s_k) \\ \text{subject to:} \quad & 0 \le s_k \le \overline{P}_k,\ \forall k; \\ & \sum_{k=1}^{K} s_k \le P, \end{aligned} \tag{3}$$

where P is the total power for allocation. For introducing GWFPP without Switching, the results for GWF (Geometric Water-filling) [18] are summarized below. GWF aims to solve the same problem as (3) with relaxation of the individual peak power constraint, *i.e.*, $\overline{P}_k$ is large enough. The water level step index k^*, denoting the index of the highest step below water level, is obtained by [18]

$$k^* = \max\left\{k \,|P_2(k) > 0,\ 1 \le k \le K \text{ and } k \in E\right\} \tag{4}$$

where E is an index set that is a subset of $\{1, \ldots, K\}$, and $P_2(k)$ is a middle variable, denoting the water volume above the kth step, and is given as

$$P_2(k) = \left[P - \sum_{i \in \{1 \le i \le k-1, i \in E\}} \left(\frac{1}{a_k w_k} - \frac{1}{a_i w_i}\right) w_i\right]^+, \quad \text{for } 1 \le\ k \le K \text{ and } k \in E. \tag{5}$$

Then the power allocated for the k^* step is

$$s_{k^*} = \frac{w_{k^*}}{\sum_{i \in \{1 \le i \le k^*, i \in E\}} w_i} P_2(k^*). \tag{6}$$

The completed solution is then

$$\begin{cases} s_i = \left[\dfrac{s_{k^*}}{w_{k^*}} + \left(\dfrac{1}{a_{k^*} w_{k^*}} - \dfrac{1}{a_i w_i}\right)\right] w_i, \\ \quad 1 \le i \le k^* \text{ and } i \in E; \\ s_i = 0, \quad k^* < i \le K \text{ and } i \in E. \end{cases} \tag{7}$$

For conveniently understanding, the key terms s_i/w_i and $1/(a_i \cdot w_i)$ are illustrated by Fig. 2.

Equipped with GWF algorithm, GWFPP without Switching is formally listed below.

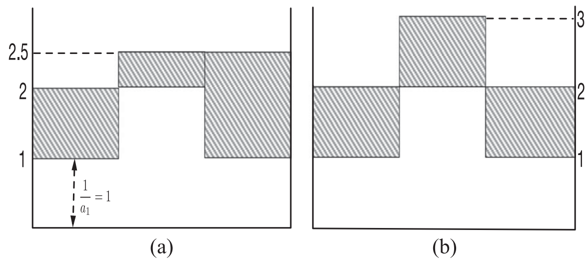


Fig. 3. Illustration for GWFPP with switching.

Algorithm GWFPP:

Input: vector $\{d_i = \frac{1}{a_i w_i}\}, \{w_i\}, \{\overline{P}_i\}$ for $i = 1, 2, \ldots, K$, the set $E = \{1, 2, \ldots, K\}$, and P.

1) Utilize (4)-(7) to compute $\{s_i\}$.
2) The set Λ is defined by the set $\{i | s_i > \overline{P}_i, i \in E\}$. If Λ is the empty set, output $\{s_i\}_{i=1}^K$; else, $s_i = \overline{P}_i$, as $i \in \Lambda$.
3) Update E with $E \setminus \Lambda$ and P with $P - \sum_{t \in \Lambda} \overline{P}_t$. Then return to 1) of the **GWFPP**.

Remark 1: Algorithm GWFPP is a dynamic power distribution process. The state of this process is the difference between the individual peak power sequence and the current power distribution sequence obtained by (4)-(7). The control of this process is to use the last statement of 2) in GWFPP and the first statement of 3) in GWFPP. A new state is updated in the next time stage. An optimal dynamic power distribution process through GWFPP with the state feedback is thus formed. Since the finite set E is getting smaller and smaller until the set is empty, Algorithm GWFPP carries out K loops to compute the optimal solution at most.

2) GWFPP with Switching:

Definition 1: (GWFPP with Switching): Let L and K be two positive integers with $L \leq K$, to denote the index of the starting channel and the ending channel, respectively. For (3) and GWFPP, we define a switch variable, δ. If the peak power constraint for the last channel, *i.e.*, the Kth channel, is relaxed, the used GWFPP is said to have the switch variable $\delta = 0$; otherwise, $\delta = 1$.

Note that, only letting $\overline{P}_K \gg 0$ in (3), *i.e.*, large enough, GWFPP corresponds to the switch $\delta = 0$; while each of $\{\overline{P}_k\}$ keeping the same as that in (3), without being relaxed, GWFPP corresponds to the switch $\delta = 1$. In detail, according to Definition 1 of GWFPP with Switching and its explanation above, GWFPP with Switching, as an algorithm, computes the two classes of exact solutions to the optimization problem (3) and its another form, respectively. Further, (3) is the optimization problem that is solved by GWFPP with the switch variable $\delta = 1$; while $\overline{P}_K$ is large enough, *i.e.*, the constraint $0 \leq s_K \leq \overline{P}_K$ in (3) is relaxed into $0 \leq s_K$, this new problem, as another form of (3), is solved by GWFPP with the switch variable $\delta = 0$. Therefore, the switch variable has been used for the two different problems with similar structures. The motivation to propose GWFPP with Switching is that such an algorithm will be used for HPA1, HPA2 and HPA2-R in the following, for convenience and clarity. GWFPP with Switching is illustrated in Fig. 3 by using a simple example for its clearly being understood. Here $K = 3, a_1 = a_3 = 1, a_2 = \frac{1}{2}, \overline{P}_1 = \overline{P}_3 = 1, \overline{P}_2 = 2, P = 3$, and $\{w_i = 1\}_{i=1}^3$. Fig. 3(a) corresponds to $\delta = 0$ with the corresponding throughput of $\log 6.25$; while Fig. 3(b) corresponds to $\delta = 1$ with the corresponding throughput of $\log 6$.

GWFPP with Switching can be viewed as a mapping from the parameters to the solution $\{s_i\}_{i=L}^K$ and to the highest water level step index: $k^* \in E$, where E is the finally obtained non-empty index set in GWFPP with Switching, which can be expressed as (from [18]),

$$\{\{s_i\}_{i=L}^K, k^*\} = \text{GWFPP}\left(L, K, \{w_i, a_i\}_{i=L}^K, P, \{\overline{P}_i\}_{i=L}^K, \delta\right), \tag{8}$$

where the parameters, $\{w_i, a_i, \overline{P}_i\}$ and P, have the same meaning as they had in Eqn. (3). Further, if E is an empty set, GWFPP with Switching only outputs the part of $\{s_i\}_{i=L}^K$. To emphasize the first part of computing $\{s_i\}_{i=L}^K$ by GWFPP with Switching, we also write

$$\{s_i\}_{i=L}^K = \text{GWFPP}\left(L, K, \{w_i, a_i\}_{i=L}^K, P, \{\overline{P}_i\}, \delta\right). \tag{9}$$

Furthermore, without confusion, GWFPP$(L, K, \{w_i, a_i\}_{i=1}^K, P, \{\overline{P}_i\}, \delta)$ can be simply written as

$$\text{GWFPP}\left(L, K, P, \{\overline{P}_i\}, \delta\right), \tag{10}$$

due to the subordinate status of the water level step index: k^*.

As a motivation to raise GWFPP with Switching, the correct order of the switching function being utilized determines that we can obtain the optimal solution to the target problem (2) by the proposed algorithm. Here, the order of the switching function means the way in which the switch variable $\delta = 0$ or 1 of GWFPP is arranged in chronological sequence. This detail will be shown by Proposition 1 and Proposition 2 in next Section.

B. Energy Harvesting Without Power Grid, Algorithm HPA1

In this subsection, we will propose a novel algorithm to first solve a simpler form of problem (2), as follows:

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i \leq \overline{P}_i, \forall i; \\ & \sum_{i=1}^l s_i \leq \sum_{i=1}^l E_{\text{in}}(i), \forall l. \end{aligned} \tag{11}$$

Note that Problem (11) does not include power supply from the power grid. This proposed new algorithm is referred to as the energy Hybrid Power Allocation algorithm HPA1, which will serve as a function block to solve our target problem (2). Since HPA1 depends on the final index of the epochs: K, sometimes to emphasize this dependence, we also use HPA1(K) to denote HPA1. The proposed HPA1(K) is stated as follows:

Algorithm: HPA1 (for EH with Peak Power Constraints):

1. Initialize: Let $L = n = 1$, and it is given that $\{K, E_{\text{in}}(1), \overline{P}_1, w_1, a_1\}$, where L is the (index) number of the final epoch for the sub-process that starts from epoch 1; and n is a variable to represent the number of an epoch in the sub-process.

Output HPA1(1):

$$s_1^* = \text{GWFPP}(1, 1, w_1, a_1, P), \text{ i.e.}, s_1^* = E_{\text{in}}(L),$$
$$\overline{s}_1^* = \text{HPA1}(1) = \min\{\overline{P}_1, s_1^*\}, \tag{12}$$

where $\overline{s}_1^* = \text{HPA1}(1)$ means that $\overline{s}_1^*$ is the value of HPA1(1).

2. Implement Recursion: (from HPA1(L-1) to HPA1(L))

0) Let $L = 2$.

1) The item is to do preparation.

$$\begin{aligned}&\text{Input: } \{E_{\text{in}}(L), \overline{P}_L, w_L, a_L\}.\\ &\text{Then assign: } E_{\text{in}} \longleftarrow E_{\text{in}}(L) + \textstyle\sum_{k=n}^{L-1}(E_{\text{in}}(k) - \overline{s}_k^*)^+,\\ &\{s_k'\}_{k=1}^{L-1} = \text{HPA1}(L-1).\end{aligned} \tag{13}$$

2) If $E_{\text{in}} \geq \overline{P}_L$,

$$\begin{aligned}&\text{output: HPA1}(L) = \{s_1', \ldots, s_{L-1}', \overline{s}_L^*\},\\ &\text{where } \overline{s}_L^* = \overline{P}_L \text{ under the given HPA1}(L-1).\end{aligned} \tag{14}$$

If $L = K$, HPA1(K) is completed; else $L \longleftarrow L + 1$ and go to (13).

3) If $E_{\text{in}} < \overline{P}_L$, each of the sub sub-processes from epoch n to epoch L is investigated by the following, where n decreases successively from L to 1. Thus, let $n = L$, now.

3.1) First, for the sub sub-processes from epoch n to epoch L, its initial values and assignment are made as follows.

$$\begin{aligned}&W = \{w_j\}_{j=n}^L; A = \{a_j\}_{j=n}^L; S_T = E_{\text{in}}; \text{ and}\\ &\{\{s_k^*\}_{k=n}^L, k^*\} = \text{GWFPP}(n, L, W, A, S_T, \{\overline{P}_k\},\\ &\delta = 0).\end{aligned} \tag{15}$$

3.2) Second, for the complementary process of the sub sub-process denoted by $\{n, n+1, \ldots, L\}$, which starts from epoch 1 upto epoch $n-1$ as $n > 1$, the maximum subscript, of the maximum water level that is denoted by c', is denoted by k_{e}^* and computed through the following procedures.

$$\begin{aligned}c_k'(s_k') =& \frac{1}{a_k w_k} + \frac{s_k'}{w_k}, 1 \leq k \leq n-1; \text{ and}\\ k_{\text{e}}^* =& \max\{\arg(\max\{c_k' | s_k' > 0, 1\\ &\leq k \leq n-1\})\}.\end{aligned} \tag{16}$$

Thus, k_{e}^* is the maximum index. If k_{e}^* does not exist, *e.g.* $n = 1$, as a trivial case, then

$$k_{\text{e}}^* = k^*. \tag{17}$$

3.3) Third,

$$\begin{aligned}&\text{if } \overline{P}_n \geq s_n^* \text{ and } \tfrac{1}{a_{k^*} w_{k^*}} + \tfrac{s_{k^*}^*}{w_{k^*}} \geq \tfrac{1}{a_{k_{\text{e}}^*} w_{k_{\text{e}}^*}} + \tfrac{s_{k_{\text{e}}^*}{}'}{w_{k_{\text{e}}^*}}\\ &\text{then } \overline{s}_n^* = s_n^*,\end{aligned} \tag{18}$$

and $\text{HPA1}(L) = \{s_1', \cdots, s_{n-1}', \overline{s}_n^*, \cdots, \overline{s}_L^*\}$. If $L = K$, HPA1(K) is completed; else $L \longleftarrow L + 1$ and go to (13). If the condition in (18) does not hold, the following two assignments are done:

$$\begin{aligned}\{s_k'\}_{k=n}^L =& \text{ GWFPP}(n, L, S_T, \{\overline{P}_k\}, \delta = 1); \text{ and}\\ E_{\text{in}} =& \; s_{n-1}' + S_T.\end{aligned} \tag{19}$$

Then $n \longleftarrow (n-1)$ and go to (15).

3.4) Fourth, an assignment and HPA1(L) are implemented below.

$$\begin{aligned}\{\overline{s}_k^* =& \; s_k'\}_{k=n}^L; \text{ and}\\ \text{HPA1}(L) =& \; \{s_1', \ldots, s_L'\};\end{aligned} \tag{20}$$

$L \longleftarrow L + 1$ and go to (13).

Therefore, Output HPA1(K): $\{s_k^*\}_{k=1}^K = \text{HPA1(K)}$.

At the same time, the pseudo code of HPA1 is attached at the end of this paper.

C. Optimality of HPA1

This subsection discusses optimality of the proposed HPA1.

Remark 2: HPA1 aims to compute the exact solution to (11). The case without the peak power constraint, has the water levels that are monotonic in the subscript of the epochs; but (11) does not have this monotonicity, due to existence of the peak power constraint. It leads to the difficulty to compute the exact solution. This point is reflected by the fact: there is no prior algorithm reported in the open literature, including our previous publications, that can compute the exact solution to (11), to the best of the authors' knowledge. HPA1 using GWFPP with Switching can overcome this difficulty and obtain the exact solution. As a motivation, HPA1 lays the basis for computing the exact solution to the target problem, with Lemmas 1–2 discussed below.

As a main idea, HPA1 uses GWFPP with Switching, recursively. The designed two values for the switch variable in HPA1 aim at not permitting the currently harvested energy to be used for the previous epochs, nor breaking the peak power constraints.

Remark 3: Besides the time series prediction and others, for a stochastic system, if the system parameters are random variables or sequences, the predictive parameters may also be chosen as their expected values. Using the proposed algorithm(s) can obtain the exact solution under the assumed parameters. This using does not lose or relax the essence of energy harvesting: the causality constraint of energy harvesting earlier, and using the harvested energy later. The causality constraint makes the problem rather difficult for exact solution. Thus, due to such a difficulty from energy harvesting, the causality constraint has to be relaxed by publications often, *e.g.* [5], [23]. [23] used its implied constraint (4) to take the place of (or telax) its original constraints (2) and (3); and [5] relaxed its original problem (P1) into (P2). Our approach can overcome this difficulty, indeed.

Remark 4: HPA1 is an optimal dynamic power distribution process. The dynamics of this recursive process is shown by the generalized state equation:

$$\begin{aligned}\text{HPA1}(L+1) =& \; G[\text{HPA1}(L), \text{GWFPP}(n, L+1, \delta)|_I],\\ &\text{for } L = \; 1, \ldots, K-1,\end{aligned} \tag{21}$$

where n is the index of the starting epoch of the currently processing window, and $G[\cdot]$ is a function determined by the algorithm. Note that the concept of dynamic processes is not identical to that of dynamic programming. The value of n stems from HPA1(L) due to (14) and the second line of (20). In this process, HPA1(L) can be regarded as the generalized system state at the time stage (or epoch) L; GWFPP$(n, L+1, \delta)$ can be regarded as the generalized system control at the time stage (or epoch) L; and then HPA1 $(L+1)$, as a state at the next time stage, can be derived or determined from its previous state and control. Due to optimality of GWFPP and mentioned forwarding dynamic recursive process of HPA1, we may obtain the following conclusion of HPA1 (K).

Proposition 1: HPA1 can compute the optimal exact solution to problem (11) within finite loops.

Proof: See Appendix VI. ∎

As a summary of the algorithm: HPA1(K), we clarify the following two points: (1) HPA1 can compute the optimal solution only from the causal information in finite steps. In essence, via the definition, an analytic or closed solution can be obtained by HPA1. *It is not required to directly solve any non-linear system, consisting of many equations and inequalities in multiple dual variables*; (2) Since it considers the peak power constraints, HPA1 can compute the exact optimal solution under the energy harvesting system with the peak power constraints.

D. Illustration of HPA1

In the following, we first analyze the power allocation of problem (11), *i.e.*, the algorithm: HPA1(K). For problem (11), due to the existence of the peak power constraints, the water level is no longer monotonically increasing in the indexes of the epochs [19].

In HPA1, the first step is for initialization and parameter input, where L denotes the index of current processing epoch. (12) outputs the allocated power at epoch 1 by HPA1. Beginning from the second step of implementing recursion to (20), HPA1(K) sequentially processes from the second epoch to the Kth epoch. The inner "For" loop ((15)-(19)) updates the power levels for the current processing epoch (L) and its previous ($L-n$) epochs to form a processing window (where n is the loop index shown in (15) and will be elaborated below) with the number of epochs which are included in the window: $(L-n+1)$. The GWFPP algorithm is applied to this window to find a common water level (shown at (15)). Then the "If" clause compares the water level of this processing window with both the previous maximum epoch's water level and the current peak power upper bound (in (18)). If the conditions are satisfied, then update power allocated at epoch L, output the entire HPA1(L) and move to process the next epoch (see (20)). If the conditions are not satisfied, the window is expanded by 1 epoch in the left side (the loop going back to (15) by decreasing n).

The seed of the recursive algorithm, HPA1(1), is given in (12). The logical block ((13)–(20)), including one inner loop ((15)–(19)), can be illustrated by Fig. 4 where it is assumed that the current processing epoch $L = 6$ with $E_{\text{in}}(6) = 0$. The power level transition is shown in the vertical bars in Fig. 4(a) based on HPA1(5). That is to say, the optimal power allocation for the first 5 epochs has been completed as shown by the upper layer of the shadowed areas in Fig. 4(a); while the lower darker layer illustrates the effect of fading gains. Epoch 6 is now under processing. From Fig. 4(a), it is assumed that epoch 3 (from number 2 to number 3 in the horizontal axis) has the saturated power that reaches $\overline{P}_3$, and $\overline{P}_6 = 1.2$. Since there is no harvested energy input in epoch 6, the power level for epoch 6 is zero and the water level is just the fading level for this processing window. That is to say, (16) calculates that $k_e^* = 5$ and then (18) compares the water level of current processing window with that of k_e^*th epoch. Since the comparison in (18) does not hold, the algorithm goes back to (15) by decreasing n to 5 and then the processing window is extended to include epochs 5 and 6 as shown in Fig. 4(b). Fig. 4(b) also shows the power allocation from GWFPP(5,6) in (15). Still, the comparison of water level non-decreasing in (18) does not hold, and then the algorithm returns to (15) again by decreasing $n = 4$. As shown in Fig. 4(c), the processing window consists of epochs 4 to 6. Opposite to the previous comparison, the water level comparison condition and peak power constraint are satisfied. The power update is carried out at epoch 6 as shown in Fig. 4(c). As a result, HPA1$(L = 6)$ is solved which is recursively obtained from HPA1$(L = 5)$.

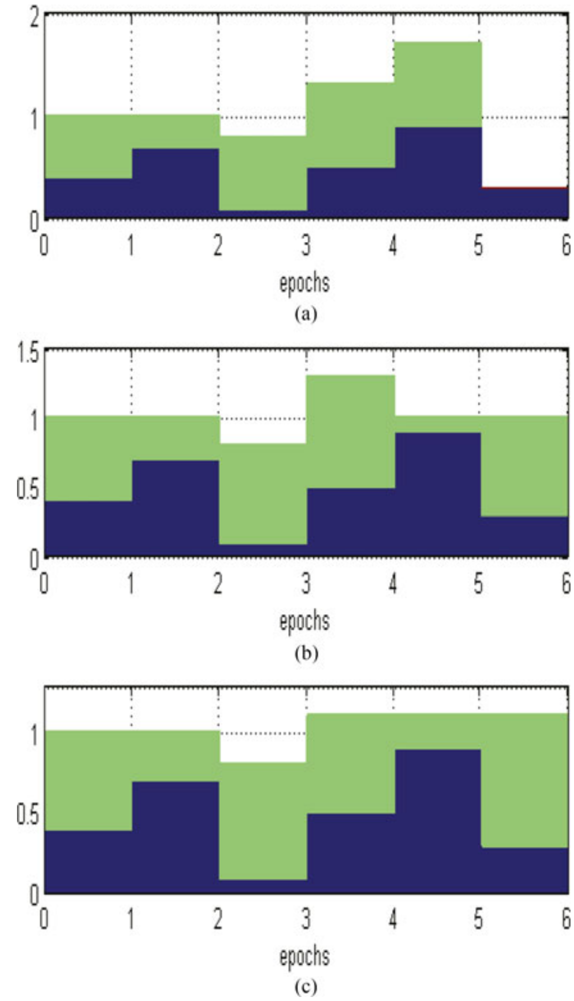


Fig. 4. Illustration for Algorithm HPA1(K) (Lines 10-22 for $L = 6$ and $K > 6$), harvested energy having been allocated up to epoch 5; (a) $n = 6$; (b) $n = 5$; (c) $n = 4$.

Furthermore, a summation is used in (13). If the lower limit of a summation is greater than the upper limit, the result of this summation is defined as zero, as well known.

Through this mechanism of recursion, the solution, $\{s_i^*\}_{i=1}^K = \text{HPA1}(K)$, is obtained (see the final line of HPA1(K)).

Note that the proposed algorithm eliminates the procedure to directly solve the non-linear system in multiple variables and dual variables of the KKT conditions of (11), provides exact solutions via finite computation steps, and offers helpful insights to the problem and the solution.

IV. Algorithms HPA2 and HPA2-R Solving the Target Problem

It is seen that the target problem (2) implies the following optimization:

$$\max_{\{s_i\}_{i=1}^K} \sum_{i=1}^K w_i \log(1 + a_i s_i)$$
$$\text{subject to: } 0 \le s_i \le \overline{P}_i, \forall i;$$
$$\sum_{k=1}^i s_k \le E_{(G,\text{total})} + \sum_{k=1}^i E_{\text{in}}(k), \forall i. \quad (22)$$

The introduction of this implied optimization problem is motivated by two aspects: it can be utilized for solving the target problem; and its exact solution can be efficiently computed by the same algorithm HPA1 without designing others, via mapping $E_{(G,\text{total})} + E_{\text{in}}(1)$ into the harvested energy for epoch 1 of (11). It is interesting that by solving the implied optimization problem, we can compute an optimal solution to the original problem. However, to establish this, two lemmas are needed. Lemma 1 is proposed to offer a relationship between the solution to problem (11) and another solution to the implied optimization problem. Successively, lemma 2 claims a relationship between the difference of the two mentioned solutions and $E_{(G,\text{total})}$. At last, an optimal solution to the target problem (2) is proposed via Proposition 2.

Lemma 1: let the solution to (22) be denoted by $\{\overline{s}_k^*\}_{k=1}^K$, and the solution to (11) be denoted by $\{s_k^*\}_{k=1}^K$. Then $\overline{s}_k^* \ge s_k^*, \forall k$.

Proof: Each of the two maximization problems of (11) and (22) has a unique solution, due to these two convex optimization problems with the strictly concave objective functions, respectively. This uniqueness is utilized for the following proof.

First, let us introduce an increment optimization problem:

$$\max_{\{\triangle s_i\}_{i=1}^K} \sum_{i=1}^K w_i \log(1 + a_i s_i^* + a_i \triangle s_i)$$
$$\text{subject to: } 0 \le \triangle s_i \le \overline{P}_i - s_i^*, \forall i;$$
$$\sum_{i=1}^K \triangle s_i \le E_{(G,\text{total})}. \quad (23)$$

Then, for the optimal solution to (23), denoted by $\{\triangle s_i\}_{i=1}^K$, there are dual variables $\{\lambda, \{\overline{\sigma}_i, \underline{\sigma}_i\}_{i=1}^K\}$ such that the following KKT conditions of (23) hold,

$$\begin{cases} \dfrac{1}{\frac{1}{a_i w_i} + \frac{s_i^* + \triangle s_i}{w_i}} = \lambda + \overline{\sigma}_i - \underline{\sigma}_i, \forall i; \\ \triangle s_i \ge 0, \underline{\sigma}_i \triangle s_i = 0, \underline{\sigma}_i \ge 0, \forall i; \\ \triangle s_i \le \overline{P}_i - s_i^*, \overline{\sigma}_i (s_i^* + \triangle s_i - \overline{P}_i) = 0, \overline{\sigma}_i \ge 0, \forall i; \\ \sum_{i=1}^K \triangle s_i \le E_{(G,\text{total})}, \lambda \left(\sum_{i=1}^K \triangle s_i - E_{(G,\text{total})}\right) = 0, \\ \lambda \ge 0. \end{cases} \quad (24)$$

If $\sum_{i=1}^K \triangle s_i < E_{(G,\text{total})}$, then $\lambda = 0$. It implies that $\overline{\sigma}_i > 0, \forall i$. Thus, $s_i^* + \triangle s_i = \overline{P}_i, \forall i$. It is obtained that such a set of $\{s_i^* + \triangle s_i\}$ is an optimal solution to (22). Due to the uniqueness mentioned above, $\overline{s}_i^* = \overline{P}_i \ge s_i^*, \forall i$.

On the other hand, assume $\sum_{i=1}^K \triangle s_i = E_{(G,\text{total})}$. Let

$$E_g = \{k | \triangle s_k > 0, 1 \le k \le K\}, \overline{k}_g = \max\{k | k \in E_g\}$$
$$\text{and } \underline{k}_g = \min\{k | k \in E_g\},$$
$$\Lambda_1 = \left\{ i_1 | \lambda < \sum_{k=i_1}^K \lambda_k, \overline{k}_g < i_1 \le K \right\} \text{ and}$$
$$\Lambda_0 = \left\{ i_0 | \lambda > \sum_{k=i_0}^K \lambda_k, 1 \le i_0 < \underline{k}_g \right\}, \text{ and}$$

$\overline{i}_1 = \max\{i_1 | i_1 \in \Lambda_1\}$ and $\underline{i}_0 = \min\{i_0 | i_0 \in \Lambda_0\}$. According to six cases of $\{k\}$, the optimal dual variables are constructed as follows:

1) If $\overline{i}_1 < k \le K$, where $\overline{i}_1 < K$, let $\tilde{\lambda}_k = \lambda_k, \tilde{\nu}_k = \nu_k$ and $\tilde{\mu}_k = \mu_k$. Note that $\{\lambda_k, \nu_k, \mu_k\}$ have the same meaning as those in the proof of Proposition 1, *i.e.*, they are the optimal dual variables to (11).
2) If $k = \overline{i}_1$, let $\tilde{\lambda}_k = \lambda - \sum_{i=k+1}^K \lambda_i, \tilde{\nu}_k = (\sum_{i=k}^K \lambda_i - \lambda) + \nu_k$ and $\tilde{\mu}_k = \mu_k$. Note that the special summation of $\sum_{i=K+1}^K \lambda_i$ is defined as zero.
3) If $\overline{k}_g < k < \overline{i}_1$, let $\tilde{\lambda}_k = 0, \tilde{\nu}_k = \nu_k + (\sum_{i=k}^K \lambda_i - \lambda)$ and $\tilde{\mu}_k = \mu_k$.
4) If $\underline{k}_g \le k \le \overline{k}_g$, let $\tilde{\lambda}_k = 0, \tilde{\nu}_k = \overline{\phi}_k$ and $\tilde{\mu}_k = \underline{\phi}_k$. Note that it is seen that, if $\underline{k}_g \le k \le \overline{k}_g$, there does not exist the case of $s_k > 0$ and $\sum_{i=k}^K \lambda_i + \nu_k < \lambda$ holding. Also, note that $\{\lambda, \overline{\sigma}_k, \underline{\sigma}_k\}$ are the mentioned optimal dual variables in the KKT conditions of (24).
5) If $\underline{i}_0 \le k < \underline{k}_g$, it is seen that $s_k^* = 0$. Thus, if $\underline{i}_0 \le k < \underline{k}_g$, let $\tilde{\lambda}_k = 0, \tilde{\nu}_k = \nu_k$ and $\tilde{\mu}_k = (\lambda - \sum_{i=k}^K \lambda_i) + \mu_k$.
6) If $1 \le k < \underline{i}_0$, let $\tilde{\lambda}_k = \lambda_k, \tilde{\nu}_k = \nu_k + (\sum_{i=k+1}^K \lambda_i - \lambda)$ and $\tilde{\mu}_k = \mu_k$.

Note that the cases mentioned above have been enumerated. Further, in more detail, it is easily seen that, if $1 \le k < \underline{k}_g - 1$, there does not exist the case of $s_k > 0$ and $\sum_{i=k}^K \lambda_i + \nu_k < \lambda$ holding, and that every $\tilde{\lambda}_k, \tilde{\nu}_k$ and $\tilde{\mu}_k$ are all non-negative. For the optimization problem (22), $\{s_k^* + \triangle s_i\}_{k=1}^K$, as optimization variables, and $\{\tilde{\lambda}_k, \tilde{\nu}_k, \tilde{\mu}_k\}_{i=1}^K$, as the corresponding dual variables, satisfy the KKT conditions of (22). The $\{\tilde{\lambda}_k, \tilde{\nu}_k, \tilde{\mu}_k\}$ correspond to the kth sum power, peak power and non-negative power constraint of (22), respectively, for any k. At the same time, the General Constraint Qualification (refer to (3.71) of Theorem 3.8 in [24]) of the problem holds. Together with the mentioned uniqueness of the optimal solution, $\{\overline{s}_k^* = s_k^* + \triangle s_k\}_{k=1}^K$ is the optimal solution to (22).

Therefore, the conclusion of Lemma 1 has been proved to be true, from the meaning of $\{\triangle s_k\}_{k=1}^K$. ■

Similarly, we prove the following lemma.

Lemma 2: let the solution to (22) be denoted by $\{\overline{s}_k^*\}_{k=1}^K$, and the solution to (11) be denoted by $\{s_k^*\}_{k=1}^K$. Then $\sum_{k=1}^K(\overline{s}_k^* - s_k^*) \leq E_{(G,\text{total})}$.

Proof: According to Lemma 1 and its proof, $\overline{s}_k^* = s_k^* + \triangle s_k, \forall k$, where $\{\triangle s_k\}_{k=1}^K$ is the optimal solution to (23). Thus, $\sum_{k=1}^K \triangle s_k \leq E_{(G,\text{total})}$. Then, $\sum_{k=1}^K(\overline{s}_k^* - s_k^*) \leq E_{(G,\text{total})}$, from $\{\triangle s_k = \overline{s}_k^* - s_k^*\}$.

Therefore, the conclusion of Lemma 2 is true. ■

Proposition 2: Utilizing HPA1 twice can form a newer algorithm to compute the optimal exact solution to the target problem (2) with finite computation. In addition, the newer algorithm most efficiently utilizes the harvested energy.

Proof: First, assume $\{s_{H,i}^*, s_{G,i}^*\}_{i=1}^K$ to be an optimal solution to problem (2). Let $s_i = s_{H,i}^* + s_{G,i}^*, \forall i$. Thus, $\{s_i\}_{i=1}^K$ is a feasible solution to problem (22). The optimal value of problem (2) is not greater than that of problem (22) because (22) is more general than (2). Conversely, for $\{s_i^*\}_{i=1}^K$, as the optimal solution to problem (11), and $\{\overline{s}_i^*\}_{i=1}^K$, as the optimal solution to problem (22), we have $\overline{s}_i^* - s_i^* \geq 0, \forall i$, stemming from Lemma 1. It implies, with Lemmas 1–2, that $\{s_{H,i}, s_{G,i}\}_{i=1}^K$ is a feasible solution of problem (2), as $s_{H,i}$ is assigned by s_i^* and $s_{G,i}$ is assigned by $\overline{s}_i^* - s_i^*$. The objective function value of the problem (2) is equal to that of (22), with respect to the aforementioned $\{s_{H,i}, s_{G,i}\}$ and $\{\overline{s}_i^*\}$. Then, the former value corresponding to a feasible point is equal to the latter value corresponding to the optimal point. It leads to the maximum objective function value of the problem (2) is not less than that of (22), together with the previously stated fact that the maximum objective function value of the problem (2) is not greater than that of (22). Therefore, the maximum objective function value of the problem (2) is equal to that of (22).

Second, to compute an optimal solution to problem (2), we need to compute the $\{\overline{s}_i^*\}_{i=1}^K$, as the optimal solution to problem (22) and the $\{s_i^*\}_{i=1}^K$, as the optimal solution to problem (11). Further, $s_{H,i}$ is assigned by s_i^* and $s_{G,i}$ is assigned by $\overline{s}_i^* - s_i^*, \forall i$. Therefore, the obtained $\{s_{H,i}, s_{G,i}\}$ is an optimal solution to problem (2). The mentioned procedure to the obtained $\{s_{H,i}, s_{G,i}\}$ forms a newer algorithm, named by HPA2, which utilize HPA1 twice to obtain $\{s_{H,i}, s_{G,i}\}$ respectively. In addition, $\{s_i^*\}_{i=1}^K$, as the optimal solution to problem (11), implies that HPA2 can utilize the harvested energy, most efficiently. Since the two sets of optimal solutions can all be computed by HPA2(K), with finite computation, Proposition 2 is thus proven to be true. ■

To conveniently refer to the computation of an optimal solution to problem (2) mentioned in Proposition 2, the newer algorithm HPA2 is presented at the end of this paper. Its equivalent recursive version, labelled as Algorithm 2 or HPA2-R, is also presented at the end. HPA2 is concise, but the recursive HPA2-R can offer more details for rapid computation of the maximum throughput in (2).

V. Numerical Examples and Complexity Analysis

Two subsections, numerical examples and computational complexity analysis, are presented. At the beginning, we use the first example to account for the procedures of HPA2-R, and then the second example to compare with the PD-IPM which has been previously regarded as an efficient optimization algorithm with great promise ([25] and references therein). The second part of this section discusses computational complexity of the proposed algorithm, and arrives at the conclusion of its being polynomial complexity with low degree. Via exploiting the structure of the proposed problem, the proposed algorithm shows significant efficiency.

A. Numerical Examples

We assume that there are three epochs, each with the unit weight ($w_i = 1, i = 1, 2, 3$). Also, the following examples perform the logarithm operation with the default base 2. Besides the previous assumptions that the height of the blue (dark) stair bars at the bottom layer denotes the fading gains and the allocated harvested power is illustrated by the height of the green (light) bars at the middle layer if there are the three layers, the grid power is illustrated by the height of the brown bars at the top layer if there are the three layers.

Example 1: Suppose the fading profile for the three epochs is $a_1 = 1, a_2 = 2$ and $a_3 = 3$. At the beginning of each epoch, unit energy is harvested ($E_{\text{in}}(i) = 1, i = 1, 2, 3$). Also, the upper bound constraint of the hybrid power is $\overline{P}_i = 1 + |3(i-2)| = [4, 1, 4], i = 1, 2, 3$, and the entire power sum from the power grid is $E_{(G,\text{total})} = 5$.

First, the heights of the bottom layer bars which denote the fading gains are shown in Fig. 5(a). They are $\{\frac{1}{a_1} = 1, \frac{1}{a_2} = \frac{1}{2}, \frac{1}{a_3} = \frac{1}{3}\}$. Second, epoch 1 is first scanned to output HPA2-R(1)= $\{s_{H,1} = 1, s_{G,1} = 3\}$ as shown in Fig. 5(b). Now we move to epoch 2 and output HPA2-R(2)= $\{s_{H,1} = 1, s_{G,1} = 3; s_{H,2} = 1, s_{G,2} = 0\}$ as shown in Fig. 5(c). Finally, At epoch 3, we have $\{s_{H,1} = \frac{2}{3}, s_{G,1} = \frac{15}{6}; s_{H,2} = 1, s_{G,2} = 0; s_{H,3} = \frac{4}{3}, s_{G,3} = \frac{15}{6}\}$ by algorithm HPA2-R. Thus, HPA2-R(3) outputs the completed solution, as shown in Fig. 5(d).

Example 2: PD-IPM is chosen for the purpose of comparison due to its competitiveness in computing the solutions to the convex optimization problems.

Fig. 6 is used to show the difference between PD-IPM and HPA2-R for the maximum throughput problems, through sweeping of the $K = 5, 10, 15, \ldots, 50$. Channel gains are generated randomly using random variables with the standard Gaussian distribution. For convenience, $\{E_{\text{in}}(k) = 6, \forall k\}$. The sum power constraint of the power grid $E_{(G,\text{total})} = K$, and the peak power constraints $\{\overline{P}_k = k, \forall k\}$. A group of different weights are also generated randomly. The chosen parameters mentioned above are assigned to both of the algorithms with the identical values for comparability. In this figure, the circle markers and the cross markers represent the results of HPA2-R and PD-IPM respectively. The obtained throughput is shown in

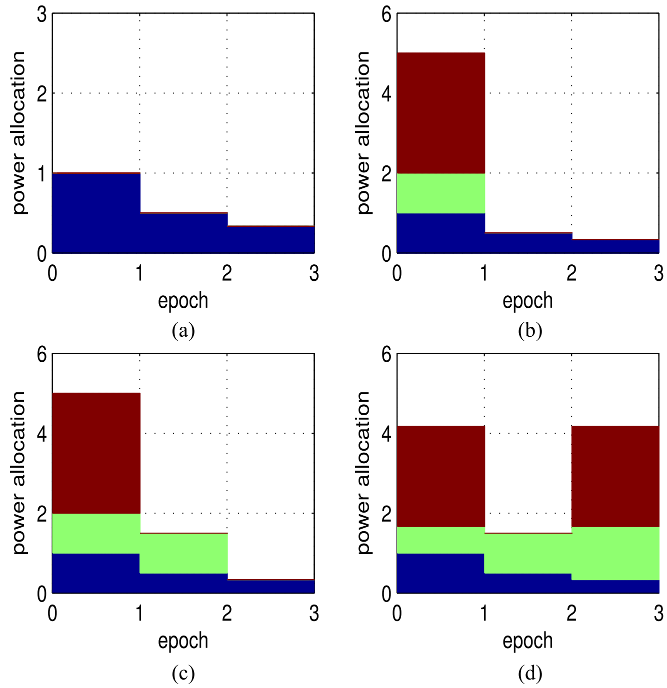


Fig. 5. Procedures to solve Example 1: (a) $\{\frac{1}{a_1}=1, \frac{1}{a_2}=\frac{1}{2}, \frac{1}{a_3}=\frac{1}{3}\}$; (b) HPA2-R(1)= $\{s_{H,1}=1, s_{G,1}=3\}$; (c) HPA2-R(2)= $\{s_{H,1}=1, s_{G,1}=3; s_{H,2}=1, s_{G,2}=0\}$; (d) HPA2-R(3)= $\{s_{H,1}=\frac{2}{3}, s_{G,1}=\frac{15}{6}; s_{H,2}=1, s_{G,2}=0; s_{H,3}=\frac{4}{3}, s_{G,3}=\frac{15}{6}\}$.

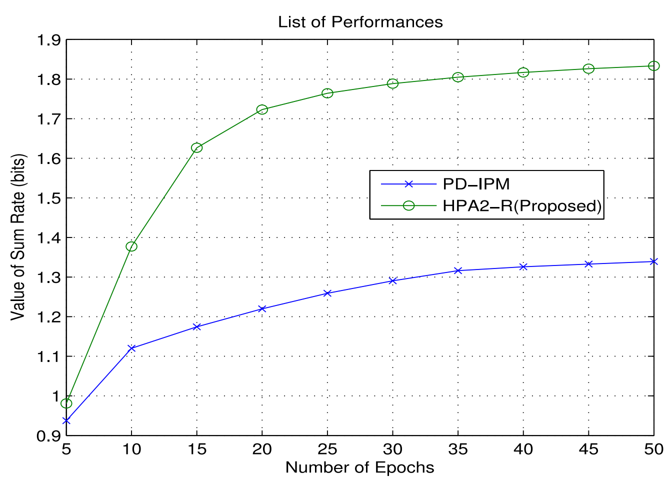


Fig. 6. Weighted sum-rates (Unit: bits) of HPA2-R and PD-IPM, as a function of the number of epochs.

Fig. 6, where the obtained throughput by PD-IPM is the result through the same number of operations as that by HPA2-R for fairness. Further, since PD-IPM only can compute an ϵ solution, not an optimal solution, even using more operations (refer to [25]), the just mentioned same number is naturally chosen as the number of the operations through which, HPA2-R can obtain the exact solution. Note the mentioned operations in this paper mean basic arithmetic or logical operations. For the performance of PD-IPM looking better, PD-IPM is allowed to use a slightly more amount of computation with choosing an appropriate initial point. Our proposed algorithms do not need setting any initial points, as a burden. The proposed HPA2-R uses recursion. Significant throughput enhancement can be observed by using the proposed algorithm. It also shows that as the number of the epochs increases, the throughput or the weighted sum-rate increases. In addition, PD-IPM has more and more variables as the number of epochs becomes larger and larger, even through it has utilized more iterations. This point comes from a well known fact that PD-IPM is not guaranteed to be convergent without additional assumption(s).

B. Computational Complexity Analysis

To compute the optimal solution, it is seen that the proposed algorithms can lead to utilizing GWFPP $K+1$ times, *i.e.*, $O(K^2) + \sum_{k=1}^{K} O(k^2) = O(K^3)$ fundamental operations (refer to [18]). According to the corresponding complexity analysis result in [18], a polynomial can be constructed, which plays a role of an upper bound for computational time or complexity. This concrete upper bound numerical value can, for given K, provide the more accurate information, the number for operations, than the big O theory and it can be taken here as $K(K+2)(3K+13)$, since $[4K^2+7K+\sum_{k=1}^{K}(4k^2+7k)]\times 2+2K \le K(K+2)(3K+13)$ from the statements of the proposed algorithms. However, PD-IPM needs a polynomial computational complexity level: $O(K^{3.5})\log(1/\epsilon)$ (refer to [25], [26]), to compute an ϵ solution, where the ϵ solution is not an optimal but approximate solution. Furthermore, this big O complexity level, used by PD-IPM, cannot offer a concrete upper bound for its computation time, even for a given ϵ of an ϵ solution. Hence, utilizing PD-IPM here cannot guarantee to output the ϵ solution with a concrete number of operations. On top of that, it is important to note that PD-IPM cannot guarantee to be convergent for the general convex optimization problems, including this proposed problem. These two well known points, *i.e.*, the number of operations and the issue of convergence for PD-IPM, lead to clear distinction of the proposed method over PD-IPM method. It is not necessary to measure the convergence rate of the proposed algorithms since these algorithms can compute the exact solution with a finite amount of computation rapidly.

In plain language, HPA2-R or HPA2 result in $K(K+2)(3K+13)$ basic operations, at most, to compute *an exact (optimal) solution*, while PD-IPM has a computational complexity level of $O(K^{3.5})\log(1/\epsilon)$ for computing *an ϵ solution*. It is seen that an algorithm with the low degree polynomial that can provide a concrete upper bound number of operations, with the exact solution, performs better than others. Our proposed algorithm is superior in this sense.

As a side note, according to optimization theory and methods, and computational complexity theory, it is seen that an optimization algorithm is excellent if it computes the exact solution with a polynomial computational complexity. Such a computation is the advantage on exactness and efficiency. To the best of the authors' knowledge, for the proposed problem, existing standard convex optimization algorithms, including the efficient PD-IPM which shows great promise [25], do not guarantee to own such an advantage. Maybe it is because these algorithms often utilize the derivatives or the gradients iteratively to approximate the solution, unlike ours. As a result, not only does the proposed algorithm have a polynomial computational complexity, but also this polynomial has the rather low degree: 3, for the exact solution.

VI. Conclusion

In this general model of the optimal power allocation for the advanced wireless communications in which the energy harvesting and the smart power grid coexist with the peak power constraints, we proposed recursive algorithms to solve the radio resource allocation problems with these complicated constraints. Due to the additional peak power constraints, a greater challenge to compute the exact solution to the proposed hybrid problem is presented, against the cases without the peak power constraints. This point is also reflected in Lemma 1.

As a starting point, we reviewed the proposed Geometric Water Filling with individual Peak Power constraints (GWFPP) to solve the optimal power allocation problem with sum power constraints. Successively, GWFPP with Switching was proposed as a functional block. Then, for solving the proposed problem only with the energy harvesting, HPA1, utilizing GWFPP with Switching, was investigated. Finally, HPA1 was extended to solve the target problem for the hybrid energy harvesting and power grid coexisting system. The extended algorithm is referred to as HPA2 or HPA2-R, while HPA2-R emphasizes recursiveness of the algorithm. Numerical results and computational complexity analysis are presented to illustrate the efficiency of the proposed algorithm HPA2 and HPA2-R.

Appendix

Proof of Proposition 1: Mathematical induction is carried out with respect to the index K, appearing in both problem (11) and the algorithm: HPA1. As $K = 1$, it is easy to see the conclusion of Proposition 1 holds. As $L < K$ and L is a natural number, assume that HPA1 (L) can compute the optimal exact solution to problem (11) where the final epoch index is L, within finite loops. The following is to prove that the conclusion of Proposition 1 holds, where the final epoch index is K.

On one hand, if $E_{\text{in}} \geq \overline{P}_K$, then let $s_K^* = \overline{P}_K$, from (14) in HPA1. According to the assumption of the induction, for $L = K-1$, there are $\{s_i^*\}_{i=1}^{K-1} = \text{HPA1}(K-1)$ and dual variables: $\{\lambda_i^{(1)}, \nu_i^{(1)}, \mu_i^{(1)}\}_{i=1}^{K-1}$ such that the KKT conditions of the problem which is solved by HPA1$(K-1)$ hold, as,

$$\begin{cases} \dfrac{1}{\frac{1}{a_i w_i} + \frac{s_i^*}{w_i}} = \sum_{k=1}^{K-1} \lambda_k^{(1)} + \nu_i^{(1)} - \mu_i^{(1)}, \text{ as } 1 \leq i \leq K-1; \\ \mu_i^{(1)} s_i^* = 0, s_i^* \geq 0, \mu_i^{(1)} \geq 0, \quad \forall i; \\ \nu_i^{(1)}(s_i^* - \overline{P}_i) = 0, s_i^* \leq \overline{P}_i, \nu_i^{(1)} \geq 0, \quad \forall i; \\ \lambda_i^{(1)} \left(\sum_{k=1}^{i} s_k^* - \sum_{k=1}^{i} E_{\text{in}}(k) \right) = 0, \\ \sum_{k=1}^{i} s_k^* \leq \sum_{k=1}^{i} E_{\text{in}}(k), \quad \lambda_i^{(1)} \geq 0, \quad \forall i. \end{cases}$$

Let $\lambda_K^{(1)} = 0, \mu_K^{(1)} = 0$, and

$$\nu_K^{(1)} = \frac{1}{\frac{1}{a_K w_K} + \frac{s_K^*}{w_K}}$$

Then for $\{s_i^*\}_{i=1}^K = \text{HPA1}(K)$ in which, $\{s_i^*\}_{i=1}^{K-1} = \text{HPA}$ $1(K-1)$, there are the dual variables: $\{\lambda_i^{(1)}, \nu_i^{(1)}, \mu_i^{(1)}\}_{i=1}^K$ mentioned above such that the KKT conditions of (11) hold. The constraint qualification of (11) holds, too. Thus, $\{s_i^*\}_{i=1}^K$ is the optimal solution to (11). *On the other hand*, if $E_{\text{in}} < \overline{P}_K$, according to (16), (17) and (18) in HPA1, it is seen that there is the natural number n, which satisfies the condition of $1 \leq n \leq K$, such that (18) in HPA1 holds. Note that the HPA1 determines the harvested energy at current epoch not to be used for its previous epochs. Hence, the following derivative is implied. First, for the aforementioned n, and $\{s_i^*\}_{i=n}^K$, as a part of the output of HPA1(K) which is shown at the final line of HPA1, there are the dual variables $\{\lambda_i^{(1)}, \nu_i^{(1)}, \mu_i^{(1)}\}_{i=n}^K$ such that they satisfy the following KKT conditions:

$$\begin{cases} \dfrac{1}{\frac{1}{a_i w_i} + \frac{s_i^*}{w_i}} = \sum_{k=i}^{K} \lambda_k^{(1)} + \nu_i^{(1)} - \mu_i^{(1)}, \quad \text{as } n \leq i \leq K; \\ \mu_i^{(1)} s_i^* = 0, s_i^* \geq 0, \mu_i^{(1)} \geq 0, \quad \forall i; \\ \nu_i^{(1)}(s_i^* - \overline{P}_i) = 0, s_i^* \leq \overline{P}_i, \nu_i^{(1)} \geq 0, \quad \forall i; \\ \lambda_i^{(1)} \left[\sum_{k=n}^{i} s_k^* - \left(\sum_{k=1}^{i} E_{\text{in}}(k) - \sum_{k=1}^{n-1} s_k^* \right) \right] = 0, \\ \sum_{k=n}^{i} s_k^* \leq \sum_{k=1}^{i} E_{\text{in}}(k) - \sum_{k=1}^{n-1} s_k^*, \\ \quad \lambda_K^{(1)} \geq 0, \lambda_i^{(1)} = 0, \text{ as } i \neq K. \end{cases}$$

As a supplement, to meet and understand the requirement of deeper strictness, construction of the important dual variables, $\{\lambda_K^{(1)}, \nu_K^{(1)}, \mu_K^{(1)}\}$, in the above KKT conditions is concisely stated as follows. Let

$$\lambda_K^{(1)} = \frac{1}{\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}^*}{w_{k^*}}}.$$

If $\overline{s}_K^*$ at the second line of (20) takes the value $\overline{P}_K$, let

$$\nu_K^{(1)} = \frac{1}{\frac{1}{a_K w_K} + \frac{s_K^*}{w_K}} - \lambda_K^{(1)} \geq 0,$$

and $\mu_K^{(1)} = 0$. Else, if $\overline{s}_K^*$ takes the value s_K^* at the final line and $s_K^* = 0$, let

$$\mu_K^{(1)} = \lambda_K^{(1)} - \frac{1}{\frac{1}{a_K w_K} + \frac{s_K^*}{w_K}} \geq 0,$$

and $\nu_K^{(1)} = 0$. Further, if $\overline{s}_K^*$ takes the value s_K^* at the final line and $\overline{P}_K > s_K^* > 0$, let

$$\mu_K^{(1)} = \nu_K^{(1)} = 0.$$

Similarly, so can all the rest of dual variables be constructed. Second, for the part, $\{s_i^*\}_{i=1}^{n-1}$, of the output of HPA1(K) which is shown at the final line of HPA1, without loss of generality, assume $n \geq 2$. According to the inductive hypothesis, for

$\{s_i^*\}_{i=1}^{n-1}$, there are the dual variables $\{\lambda_i^{(2)}, \nu_i^{(2)}, \mu_i^{(2)}\}_{i=1}^{n-1}$ such that they satisfy the following KKT conditions:

$$\begin{cases} \frac{1}{\frac{1}{a_i w_i} + \frac{s_i^*}{w_i}} = \sum_{k=i}^{K} \lambda_k^{(2)} + \nu_i^{(2)} - \mu_i^{(2)}, \quad \text{as } 1 \le i \le n-1; \\ \mu_i^{(2)} s_i^* = 0, s_i^* \ge 0, \mu_i^{(2)} \ge 0, \quad \forall i; \\ \nu_i^{(2)}(s_i^* - \overline{P}_i) = 0, s_i^* \le \overline{P}_i, \nu_i^{(2)} \ge 0, \quad \forall i; \\ \lambda_i^{(2)} \left(\sum_{k=1}^{i} s_k^* - \sum_{k=1}^{i} E_{\text{in}}(k) \right) = 0, \\ \sum_{k=1}^{i} s_k^* \le \sum_{k=1}^{i} E_{\text{in}}(k), \lambda_i^{(2)} \ge 0, \quad \forall i. \end{cases}$$

According to the transition of HPA1(K) between the two parts, it is seen that $\lambda_{n-1}^{(2)} \ge \sum_{k=n}^{K} \lambda_k^{(1)}$. Thus, for the output of HPA1(K): $\{s_i^*\}_{i=1}^{K}$, the dual variables $\{\lambda_i, \nu_i, \mu_i\}_{i=1}^{K}$ are constructed as follows.

$$\begin{cases} \lambda_k &= \lambda_k^{(1)}, \text{ as } n \le k \le K, \\ \lambda_{n-1} &= \lambda_{n-1}^{(2)} - \sum_{k=n}^{K} \lambda_k^{(1)} \ge 0, \\ \lambda_k &= \lambda_k^{(2)}, \text{ as } 1 \le k \le n-1; \\ \mu_k &= \mu_k^{(1)} \text{ as } n \le k \le K, \\ \mu_k &= \mu_k^{(2)} \text{ as } 1 \le k \le n-1; \\ \nu_k &= \nu_k^{(1)} \text{ as } n \le k \le K, \\ \nu_k &= \nu_k^{(2)} \text{ as } 1 \le k \le n-1. \end{cases}$$

Thus, the constructed dual variables of $\{\lambda_i, \mu_i, \nu_i\}_{i=1}^{K}$ and the output of $\{s_i^*\}_{i=1}^{K}$ satisfy the KKT conditions of problem (11). At the same time, the constraint qualification of problem (11) also holds. Therefore, from the mathematical induction, $\{s_i^*\}_{i=1}^{K}$ outputted by HPA1(K) is the exact optimal solution to problem (11) with finite computation, for any natural number K.

Therefore, Proposition 1 is proved.

Acknowledgment

The authors are grateful to the anonymous reviewers and the editor for their valuable comments and suggestions to improve the quality of the article.

References

[1] Z. G. Wan, Y. Tan, and C. Yuen, “Review on energy harvesting and energy management for sustainable wireless sensor networks,” in *Proc. IEEE 13th Int. Conf. Commun. Technol.*, 2011, pp. 362–367.

[2] M. Ku, W. Li, Y. Chen, and K. Liu, “Advances in energy harvesting communications: Past, present, and future challenges,” *IEEE Commun. Surv. Tuts.*, vol. 18, no. 2, pp. 1384–1412, 2016, doi: 10.1109/COMST.2015.2497324.

[3] C. Ho and R. Zhang, “Optimal energy allocation for wireless communications with energy harvesting constraints,” *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4808–4818, Sep. 2012.

[4] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, “Transmission with energy harvesting nodes in fading wireless channels: Optimal policies,” *IEEE J. Sel. Areas Commun.*, vol. 29, no. 8, pp. 1732–1743, Jun. 2011.

[5] F. Amirnavaei and M. Dong, “Online power control optimization for wireless transmission with energy harvesting and storage,” *IEEE Trans. Wireless Commun.*, vol. 15, no. 7, pp. 4888–4901, Jul. 2016.

[6] J. Lei, R. Yates, and L. Greenstein, “A generic model for optimizing single-hop transmission policy of replenishable sensors,” *IEEE Trans. Wireless Commun.*, vol. 8, no. 2, pp. 547–551, Feb. 2009.

[7] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, “Optimal energy management policies for energy harvesting sensor nodes,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 4, pp. 1326–1336, Apr. 2010.

[8] S. Mao, M. H. Cheung, and V. W. S. Wong, “An optimal energy allocation algorithm for energy harvesting wireless sensor networks,” in *Proc. IEEE Int. Conf. Commun.*, 2012, pp. 265–270.

[9] M. Gorlatova, A. Wallwater, and G. Zussman, “Networking low-power energy harvesting devices: Measurements and algorithms,” *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1853–1865, Sep. 2013.

[10] X. Fafoutis and N. Dragoni, “Analytical comparison of MAC schemes for energy harvesting—Wireless sensor networks,” in *Proc. Int. Conf. Netw. Sensing Syst.*, 2012, pp. 1–6.

[11] Z. A. Eu and H. P. Tan, “Adaptive opportunistic routing protocol for energy harvesting wireless sensor networks,” in *Proc. IEEE Int. Conf. Commun.*, 2012, pp. 318–322.

[12] D. Hasenfratz, A. Meier, C. Moser, J. J. Chen, and L. Thiele, “Analysis, comparison, and optimization of routing protocols for energy harvesting wireless sensor networks,” in *Proc. IEEE Int. Conf. Sensor Netw., Ubiquitous, Trustworthy Comput.*, 2010, pp. 19–26.

[13] K. B. Huang, “Throughput of wireless networks powered by energy harvesting,” in *2011 Conf. Rec. 45th AsilomarConf. Signal, Syst. Comput. (Asilomar)*, 2011, pp. 8–12.

[14] S. Peng and C. Low, “Throughput optimal energy neutral management for energy harvesting wireless sensor networks,” in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2012, pp. 2347–2351.

[15] M. Gatzianas, L. Georgiadis, and L. Tassiulas, “Control of wireless networks with rechargeable batteries,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 2, pp. 581–593, Mar. 2010.

[16] S. Wei, W. Guan, and K. Liu, “Power scheduling for energy harvesting wireless communications with battery capacity constraint,” *IEEE Trans. Wireless Commun.*, vol. 14, no. 8, pp. 4640–4653, Aug. 2015.

[17] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Nashua, NH, USA: Athena Scientific, 1999.

[18] P. He, L. Zhao, S. Zhou, and Z. Niu, “Water-filling: A geometric approach and its application to solve generalized radio resource allocation problems,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 7, pp. 3637–3647, Jul. 2013.

[19] P. He, L. Zhao, S. Zhou, and Z. Niu, “Recursive water-filling for wireless links with energy harvesting transmitters,” *IEEE Trans. Veh. Technol.*, vol. 63, no. 3, pp. 1232–1241, Mar. 2014.

[20] J. Gong, S. Zhou, and Z. Niu, “Optimal power allocation for energy harvesting and power grid coexisting wireless communication systems,” *IEEE Trans. Commun.*, vol. 61, no. 6, pp. 3040–3049, Mar. 2013.

[21] P. He, L. Zhao, S. Zhou, and Z. Niu, “Optimal recursive power allocation for energy harvesting nodes with multiple antennas,” *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4525–4636, Oct. 2015.

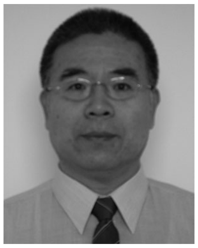
[22] L. S. Pontryagin, *Mathematical Theory of Optimal Processes*. Toronto, ON, Canada: CRC, 1987.

[23] X. Zhou, C. Ho, and R. Zhang, “Wireless power meets energy harvesting: a joint energy allocation approach in OFDM-based system,” *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3481–3491, Oct. 2016.

[24] M. Avriel, *Nonlinear Programming: Analysis and Methods*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1976.

[25] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[26] L. Zhang, Y. Xin, Y.-C. Liang, and H. V. Poor, “Cognitive multiple access channels: Optimal power allocation for weighted sum rate maximization,” *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2754–2762, Sep. 2009.



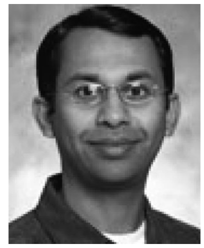
Peter He (S'15–M'15) received the M.A.Sc. degree in electrical engineering from McMaster University, Hamilton, ON, Canada, in 2009, and the Ph.D. degree in electrical engineering from Ryerson University, Toronto, ON Canada, in 2015. At present, his postdoctoral research interests include the union of large-scale optimization, information theory, communications, and smart power grid. During his Ph.D. studies, he received Graduate Research Excellence Award from the Department of Electrical and Computer Engineering, Ryerson University and Ontario Graduate Scholarshipawards. He received Natural Sciences and Engineering Research Council Postdoctoral Fellowship in 2016. He received the Best Paper Award of the 2013 International Conference on Wireless Communications and Signal Processing.

Lian Zhao (S'99–M'03–SM'06) received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, in 2002.

She joined the Electrical and Computer Engineering Department, Ryerson University, Toronto, ON, Canada, as an Assistant Professor in 2003, an Associate Professor in 2007, and then a Professor in 2014. She has served as the Symposium Cochair for IEEE Global Communications Conference 2013 Communication Theory Symposium; the Web-Conference Cochair for IEEE Toronto International Conference Science and Technology for Humanity 2009; a Guest Editor for the *International Journal on Communication Networks and Distributed Systems*, special issue on Cognitive Radio Networks in 2011; the IEEE Ryerson Student Branch Counselor since 2005; an Associate Chair for Graduate Studies at ELCE Department from 2013 to 2015. She has been an Associate Editor for IEEE Transaction on Vehicular Technology since 2013; she has also served as a Technical Program Committee Member for numerous IEEE flagship conferences, and reviewers for IEEE Transaction journals and for various Natural Sciences and Engineering Research Council proposals. Her research interests include the areas of wireless communications, radio resource management, power control, cognitive radio, and cooperative communications.

Dr. Zhao received the 2016 Best Land Transportation Paper Award from the IEEE Vehicular Technology Society; 2015 TOP 15 Editor for her outstanding contributions to IEEE Transaction on Vehicular Technology; the Best Paper Award (with her student) from 2013 International Conference on Wireless Communications and Signal Processing; Best Student Paper Award (with her student) from Chinacom in 2011; Canada Foundation for Innovation New Opportunity Research Award in 2005; Ryerson Faculty Merit Award in 2005 and 2007; Faculty Research Excellence Award from the ELCE Department of Ryerson University in 2010, 2012, and 2014.

She is a Licensed Professional Engineer in Ontario.



Bala Venkatesh (M'95–SM'08) received the Ph.D. degree in electrical engineering from Anna University, Chennai, India, in 2000. He is an Academic Director of the Centre for Urban Energy and a Professor in the Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON, Canada. His research interests include power system analysis, economics, and optimization.