Energy Harvesting Communications with Energy and Data Storage Limitations

Burak Varan Aylin Yener
Wireless Communications and Networking Laboratory
Electrical Engineering Department
The Pennsylvania State University, University Park, PA 16802
varan@psu.edu yener@ee.psu.edu

***Abstract*—In this paper, a single user channel is considered with an energy harvesting transmitter that receives its energy and data intermittently. The transmitter is equipped with a finite battery as well as a finite data buffer. The throughput maximization problem with a deadline is solved and the optimal transmission policy is obtained. The optimization problem is shown to yield a directional waterfilling solution with energy pumps. An alternative algorithmic solution is also presented that utilizes the recursive shortest path solution that was shown to be optimal for infinite data buffers in earlier work. Numerical results are provided to demonstrate the throughput performance of optimal policies as well as to assess the impact of the finite buffer on the throughput.**

I. Introduction

Wireless communications devices that harvest their energy intermittently from external sources enable extended network lifetimes and green operations. Energy harvesting communications has been considered in a variety of set ups to date [1]–[9]. Particularly related to this work are references [1]–[3], [6], [10]. In [1], a single user energy harvesting channel is studied with intermittent data arrivals, an infinite capacity battery and and an infinite data buffer at the transmitter. In [2], throughput maximization problem is solved for a single user set up with an infinite backlog of data, an infinite capacity buffer, and a finite capacity battery. Reference [3] provides a directional waterfilling algorithm for identifying the optimal power policy. Reference [10] considers the transmission completion time minimization problem with finite data and energy storage. The communication set up in [10] does not allow dropped packets which sometimes results in an infeasible problem.

In this work, we solve the throughput maximization problem for the energy harvesting single user channel with a transmitter that has limited energy and data storage. The data transmission policies allow the transmitter to drop some of the packets in its data buffer if the future data arrivals are to cause an overflow. We take two approaches to solve the throughput maximization problem. First, we solve the optimization problem by transforming it to a suitable form which results in a new variant of the directional waterfilling algorithm with added energy pumps. We also provide an alternative recursive solution by first assuming an infinite capacity buffer, and then generalizing the result to any buffer capacity. We provide numerical results demonstrating how the data buffer capacity impacts the optimal throughput.

This work was supported in part by the National Science Foundation Grants CNS 09-64364 and CCF 14-22347.

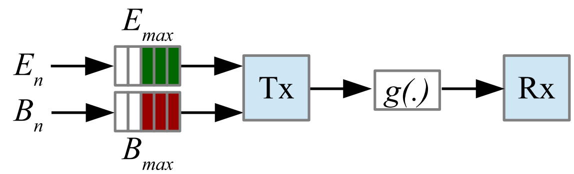

Fig. 1. The energy harvesting single user channel with a finite battery and a finite buffer at the transmitter.

II. System Model

Consider a single user communication system with an energy harvesting transmitter as in Fig. 1. The channel from the transmitter to the receiver has rate $g(p)$ with transmit power p. We assume that $g(p)$ is invertible, differentiable, strictly concave and increasing in p, and satisfies $g(0) = 0$.

Energy is harvested at discrete time instants, and can be expended for transmission immediately or stored for future transmission in an energy storage device, i.e., a battery of capacity E_{max}. If at any point during transmission the transmitter receives more energy than it can store in its battery, then the excess amount of energy is lost. The transmitter also receives its data intermittently over the course of transmission. The transmitter employs a data storage device, i.e., a data buffer of capacity B_{max} to store data for future transmission. Packets that cannot be stored due to the finite capacity of the buffer are dropped. We consider that all arrival times and packet sizes for both energy and data arrivals are known prior to the beginning of transmission, i.e., an offline set up. Although energy and data packets may arrive at the same time, not all energy packets have to be accompanied by a data packet, and vice versa. We refer to the time interval between two consecutive events, i.e., energy or data arrivals, as an epoch. We consider communication with a deadline T, and denote by N the number of epochs by deadline T. We denote the amount of energy and the amount of data that arrive at the beginning of the nth epoch by E_n and B_n, respectively. Without loss of generality, we assume that $E_n \leq E_{max}$ and $B_n \leq B_{max}$ for all $n = 1, 2, \ldots, N$ since the extra amount of energy (data) has to be lost (dropped) due to the finite battery

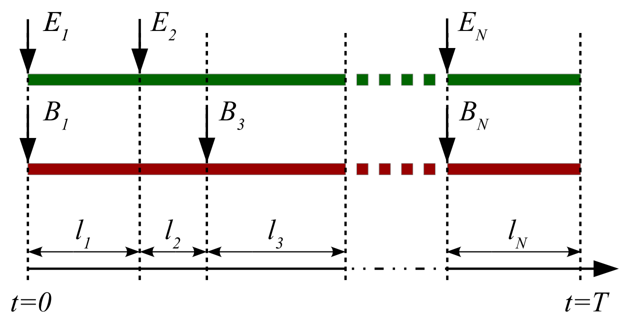

Fig. 2. Energy and data arrivals, and the resulting epoch structure.

(buffer) at the transmitter. We denote the length of the nth epoch by l_n. An example of the epochs is depicted in Fig. 2.

III. Throughput Maximization: A Directional Waterfilling Algorithm

The goal of this work is to find optimal transmission policies for the energy harvesting single user channel which maximize the throughput by deadline T. In general, this results in a continuous optimization problem. It can be shown, however, that [1, Lemma 5] applies to our setting and there always exists an optimal transmission policy under which the transmission power remains constant between any two consecutive arrivals. Thus, we can focus our attention on piecewise linear transmission policies, and solve

$$\max_{\mathbf{p},\mathbf{w},\mathbf{r},\mathbf{d}\geq 0} \sum_{i=1}^{N} l_i r_i \tag{1a}$$

$$\text{s.t.} \sum_{i=1}^{n}(l_i p_i + w_i) \leq \sum_{i=1}^{n} E_i, \tag{1b}$$

$$\sum_{i=1}^{n} E_i - \sum_{i=1}^{n-1}(l_i p_i + w_i) \leq E_{max}, \tag{1c}$$

$$\sum_{i=1}^{n}(l_i r_i + d_i) \leq \sum_{i=1}^{n} B_i, \tag{1d}$$

$$\sum_{i=1}^{n} B_i - \sum_{i=1}^{n-1}(l_i r_i + d_i) \leq B_{max}, \tag{1e}$$

$$r_n \leq g(p_n), \quad \forall n = 1, 2, \ldots, N, \tag{1f}$$

where p_n denotes the transmit power for the nth epoch, r_n denotes the rate achieved in the nth epoch, and w_n and d_n are slack variables that denote the wasted energy and the dropped packets at the transmitter in the nth epoch due to finite E_{max} and B_{max}, respectively. Bold face font denotes vectors, e.g., $\mathbf{p} = (p_n)_{n=1,2,\ldots,N}$. Here, (1b) is the energy causality constraint which ensures that only the harvested amount of energy up to date can be spent for transmission. (1c) is the battery constraint which states that no more than E_{max} units of energy can be stored in the battery. Similar causality and storage constraints are given for the data arrivals in (1d) and (1e) to limit the feasible policies to those which do not transmit data that has not yet arrived at the transmitter, and do not store more than B_{max} bits in the buffer. (1) is a convex problem.

Consider the following modification of (1) where we incorporate (1f) into the objective.

$$\max_{\mathbf{p},\mathbf{w},\mathbf{r},\mathbf{d}\geq 0} \sum_{i=1}^{N} l_i \min\{g(p_i), r_i\} \tag{2a}$$

$$\text{s.t.} \sum_{i=1}^{n}(l_i p_i + w_i) \leq \sum_{i=1}^{n} E_i, \tag{2b}$$

$$\sum_{i=1}^{n} E_i - \sum_{i=1}^{n-1}(l_i p_i + w_i) \leq E_{max}, \tag{2c}$$

$$\sum_{i=1}^{n}(l_i r_i + d_i) \leq \sum_{i=1}^{n} B_i, \tag{2d}$$

$$\sum_{i=1}^{n} B_i - \sum_{i=1}^{n-1}(l_i r_i + d_i) \leq B_{max}, \ \forall n = 1, 2, \ldots, N. \tag{2e}$$

We observe that the transmit power scheduled for an epoch constrains the rate for that epoch via (1f). However, the same effect can be achieved by the minimization in the modified objective of (2). If the solution of (2) schedules $r_n > g(p_n)$ for some n, then the actual achieved rate for epoch n will be $g(p_n)$ due to this minimization, and the total amount of data dropped by the transmitter will be $l_n(r_n - g(p_n)) + d_n$. Note that for this set up, the original problem would schedule a larger d_n and $r_n = g(p_n)$, and achieve the same throughput. Thus, the optima of (1) and (2) are the same, and the optimal solution of (2) can be modified to satisfy (1f) as described above, so that it will also solve (1) optimally.

The modified problem (2) still has a concave objective, and has a feasible region that is separable, i.e., (2b) and (2c) are a function of p_n and w_n only, and (2d) and (2e) are a function of r_n and d_n only. This separation allows us to use block coordinate descent [11, §2.7] to solve (2) iteratively. We use the superscript notation with brackets to denote the iteration index, e.g., $\mathbf{p}^{[k]}$ is the power vector $\mathbf{p}$ found in the kth iteration. In the kth iteration, the updates are given by

$$(\mathbf{p}^{[k]}, \mathbf{w}^{[k]}) = \arg\max_{(\mathbf{p},\mathbf{w})\geq 0} \sum_{i=1}^{N} l_i \min\{g(p_i), r_i^{[k-1]}\} \tag{3a}$$

$$\text{s.t.} \sum_{i=1}^{n}(l_i p_i + w_i) \leq \sum_{i=1}^{n} E_i, \tag{3b}$$

$$\sum_{i=1}^{n} E_i - \sum_{i=1}^{n-1}(l_i p_i + w_i) \leq E_{max}, \tag{3c}$$

$$\forall n = 1, 2, \ldots, N,$$

$$(\mathbf{r}^{[k]}, \mathbf{d}^{[k]}) = \arg\max_{(\mathbf{r},\mathbf{d})\geq 0} \sum_{i=1}^{N} l_i \min\{g(p_i^{[k]}), r_i\} \tag{4a}$$

$$\text{s.t.} \sum_{i=1}^{n}(l_i r_i + d_i) \leq \sum_{i=1}^{n} B_i, \tag{4b}$$

$$\sum_{i=1}^{n} B_i - \sum_{i=1}^{n-1}(l_i r_i + d_i) \leq B_{max}, \quad \text{(4c)}$$
$$\forall n = 1, 2, \ldots, N.$$

The above block coordinate descent algorithm requires a separable feasible set as well as a strictly concave objective [11, §2.7] for convergence to a unique solution. While the feasible set of (2) is separable, the objective is not necessarily strictly concave, but is merely concave. In order to satisfy this requirement, we can add $-\epsilon_1 \|(\mathbf{p}, \mathbf{w}) - (\mathbf{p}^{[k-1]}, \mathbf{w}^{[k-1]})\|^2$ for some $\epsilon_1 > 0$ to the objective of (3) and optimize it over $(\mathbf{p}, \mathbf{w})$. This way, the next solution is always the optimizer that is the closest to the current solution; hence, (3) has a unique optimizer. We can apply the same manipulation on (4) by adding $-\epsilon_2 \|(\mathbf{r}, \mathbf{d}) - (\mathbf{r}^{[k-1]}, \mathbf{d}^{[k-1]})\|^2$ for some $\epsilon_2 > 0$ to the objective, and guarantee convergence as was done in [6]. Here, we leave (3) and (4) as given for brevity, and refer the reader to [12] for a detailed analysis of convergence, stating that the iterative algorithm provided by (3) and (4) converges to the optimal power and data transmission policy for (2) (and thus (1)).

In what follows, we identify the solutions to (3), which allocates optimal power $\mathbf{p}$ and wasted energy $\mathbf{w}$ given $\mathbf{r}$ and $\mathbf{d}$, and (4), which allocates optimal rate $\mathbf{r}$ and dropped packets $\mathbf{d}$ given $\mathbf{p}$ and $\mathbf{w}$.

A. Solution of (3)

By computing the Lagrangian for (3), and differentiating it with respect to p_n, we can obtain the stationarity condition on p_n for all $n = 1, 2, \ldots, N$. In order to simplify the computations, we replace the objective of (3) by $\sum_{i=1}^{N} l_i g(p_i)$, and include $p_n \leq g^{-1}(r_n^{[k-1]})$ as a constraint. We obtain

$$g'(p_n) = \sum_{i=n}^{N} \lambda_i - \sum_{i=n+1}^{N} \mu_i - \kappa_n + \nu_n \quad (5)$$

where $g'(p_n) = \frac{\mathrm{d}g(p_n)}{\mathrm{d}p_n}$, and λ_n, μ_n, ν_n, κ_n are Lagrange multipliers associated with constraints (3b), (3c), $g(p_n) \leq r_n^{[k-1]}$, $p_n \geq 0$, respectively. From the complementary slackness conditions, we have that $\nu_n = 0$ whenever $g(p_n) < r_n^{[k-1]}$. In this case, there is sufficient data allocated by (4) in the previous iteration; thus, the solution is the same as the directional waterfilling solution found in [3] for an infinite backlog of data and $B_{max} = \infty$. That is, we model the epochs as rectangular bins (see Fig. 3) of width l_n, and model the energy arrivals as E_n units of water that are initially filled into the nth bin. The water level in each bin denotes the power allocated for the corresponding epoch. We consider taps between adjacent bins which are right permeable, i.e., they allow water to flow only from past epochs to future epochs due to (3b), and turn off when the amount of water transferred to the next bin is E_{max} due to (3c). An example is shown at the top of Fig. 3.

If $\nu_n > 0$, then we have $g(p_n) = r_n^{[k-1]}$. Notice that $g'(p_n)$ is increasing in ν_n. Assuming an additive white Gaussian noise (AWGN) channel with normalized noise variance and channel gain for our analysis, we get $g(p) = \frac{1}{2}\log(1+p)$ and

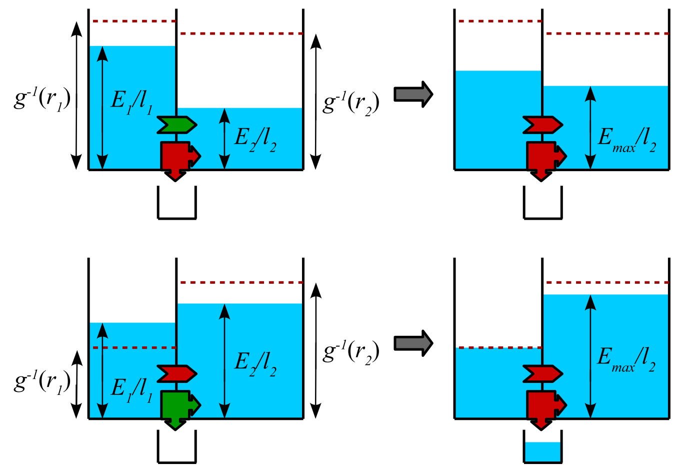

Fig. 3. Directional waterfilling for two set ups with $N = 2$. Top: The tap allows water to flow to the second bin as long as the water level for the first bin is higher, and the second bin is not full. Bottom: The initial water level is higher in the second bin, so the tap is off. However, the maximum power constraint is violated in the first bin, thus, some water is pumped to the second bin until it is full, at which point the excess water is wasted.

$g'(p) = \frac{1/2}{1+p}$. We see that a positive ν_n results in a decrease in p_n until $g(p_n) = r_n^{[k-1]}$. In other words, the data allocated by (4) in the previous iteration results in a maximum power of $g^{-1}(r_n^{[k-1]})$ for the nth epoch. We interpret this phenomenon by introducing water pumps and overflow protection bins to the waterfilling solution. The water pump for the nth epoch is inactive as long as $p_n \leq g^{-1}(r_n^{[k-1]})$. However, if this constraint violated by the initial water levels, or the operation of the right permeable taps, then the water pump is activated. The water pump is responsible for bringing the water level down to $g^{-1}(r_n^{[k-1]})$. In order to do this, it pumps water to the next bin, and when the next bin is full, it pumps water into the overflow protection bin. Here, the transfer of water to an overflow protection bin models a positive w_n, i.e., wasted energy. An example is shown at the bottom of Fig. 3.

This completes the description of identifying the optimal $\mathbf{p}$ and $\mathbf{w}$ for (3) using a directional waterfilling approach with right permeable taps which turn off whenever the next bin has E_{max} units of water, and pump water to the next bin or to an overflow protection bin whenever the water level hits $g^{-1}(r_n^{[k-1]})$.

B. Solution of (4)

Let us generate an equivalent problem to (4) by replacing the objective with $\sum_{i=1}^{N} l_i r_i$, and including the additional constraint $r_n \leq g(p_n^{[k]})$, $n = 1, 2, \ldots, N$. With this modification, (4) becomes a linear program (LP) which can be solved by any LP solver. In order to gain more insights, we propose an analytic solution and prove its optimality inductively.

We start by reiterating that (4) allocates optimal $\mathbf{r}$ and $\mathbf{d}$ given $p_n^{[k]}$ computed by (3) in the same iteration. $p_n^{[k]}$ defines a maximum throughput of $l_n g(p_n^{[k]})$ for each epoch. If the previous optimal solution to (4) has $r_n^{[k-1]} > g(p_n^{[k]})$, then in this iteration, a portion of the rate allocated for the nth epoch must either be transferred to the next epoch, or must be lost in the form of a positive $d_n^{[k]}$ if the amount of data that has been transferred to the next epoch is B_{max}. Consider the first epoch with B_1 units of data available in the data buffer, and a maximum throughput of $l_1 g(p_1^{[k]})$. If $B_1 \leq l_1 g(p_1^{[k]})$, then the optimal solution must allocate $r_1^{[k]} = B_1/l_1$ and $d_1^{[k]} = 0$. Note that $r_1^{[k]} > B_1/l_1$ is infeasible since there is not enough data, and $r_1^{[k]} < B_1/l_1$ is inefficient since $p_1^{[k]}$ is not fully utilized, and the remaining packets result in less free space in the buffer for the next epoch. If $B_1 > l_1 g(p_1^{[k]})$, then the optimal solution must allocate $r_1^{[k]} = g(p_1^{[k]})$ and leave the remaining packets of size $B_1 - l_1 g(p_1^{[k]})$ for the next epoch. Note that $r_1^{[k]} > g(p_1^{[k]})$ is not feasible, and $r_1^{[k]} < g(p_1^{[k]})$ is again inefficient. We can treat the transferred data as additional data that arrives at the beginning of the second epoch as

$$B_2 := \min\{B_2 + B_1 - l_1 g(p_1^{[k]}), B_{max}\}. \quad (6)$$

Clearly $d_1^{[k]} = \max\{B_2 + B_1 - l_1 g(p_1^{[k]}) - B_{max}, 0\}$. The optimality of $r_1^{[k]}$ and $d_1^{[k]}$ has now been established. Then, in the nth epoch ($n = 2, 3, \ldots, N$), we can assume optimality for the $(n-1)$th epoch, apply the same procedure, and update B_{n+1} in the same way as in (6). By induction, the optimality of the procedure given above follows.

IV. Throughput Maximization: An Alternative Algorithm

In this section, we first relax (1) by setting $B_{max} = \infty$, and thus removing (1e). After identifying the optimal policies for this special case, we tackle the general problem with a finite B_{max}. Our solution methodology for both cases is to first propose a recursive solution and then show its optimality. The solution in this section does not require a convex feasible set, thus we set $r_n = g(p_n)$ for all $n = 1, 2, \ldots, N$ to decrease the dimension of the problem, resulting in the removal of (1f).

Reference [10] also proposes a recursive solution for the transmission completion time minimization problem for the same set up. However, feasible solutions are constrained to those which can transmit all the data that arrives at the transmitter. This restriction sometimes causes the problem to be infeasible. For instance, consider a set up with $N = 2$, $B_1 = B_2 = B_{max}$, and $E_1 < l_1 g^{-1}(B_{max}/l_1)$. Here, the data buffer is guaranteed to overflow at the beginning of the second epoch. In this case, the algorithm proposed in [10] declares that no solution exists. Here, since we allow the transmitter to drop some of the packets in its buffer so as to handle such infeasibilities, we can get the throughput maximizing solution.

A. Solution for an Infinite Data Buffer

Here, we set $B_{max} = \infty$ in (1) and solve it. With an infinite capacity buffer, we have $d_n = 0$ for all $n = 1, 2, \ldots, N$ since the transmitter does not need to drop any packets.

Lemma 1: An optimal transmission policy may schedule $w_n > 0$ for some n only if the buffer is empty at the end of the nth epoch, and the battery is full at the beginning of the $(n+1)$th epoch.

Proof: The proof directly follows from the observation that the transmitter should not waste any energy if it has more data to send, or if there is enough room in its battery for energy transfer to future epochs. ■

As a corollary to Lemma 1, we can conclude that the problem in (1) can be separated into subproblems each of which admits an optimal transmission policy with $w_n = 0, \forall n$. Before we elaborate on the separation of the problem, suppose for now that the transmitter has enough data in its buffer throughout the duration of transmission that it never needs to overflow the battery. In this case, we have $w_n = 0$ for all $n = 1, 2, \ldots, N$. We can describe the set of feasible transmission policies as a tunnel as in [2]. However, with data arrivals at the transmitter, the upper wall of the tunnel will be determined not only by (1b), but also by (1d). Let us now replace (1b) and (1d) for all $n = 1, 2, \ldots, N$ by

$$\sum_{i=1}^{n} l_i p_i \leq \min\left\{\sum_{i=1}^{n} E_i, \left(\sum_{i=1}^{n} l_i\right) g^{-1}\left(\frac{\sum_{i=1}^{n} B_i}{\sum_{i=1}^{n} l_i}\right)\right\}. \quad (7)$$

This replacement can be justified as follows. Suppose the optimal transmission powers for the relaxed problem do not change until the beginning of the $(m+1)$th epoch, i.e., $p_1^* = p_2^* = \cdots = p_m^*$. Then, the total energy spent by the end of the mth epoch cannot be greater than the energy harvested by that point, which is ensured by the first term in the minimization on the right hand side of (7). The total energy spent by the end of the mth epoch also should not be greater than $\left(\sum_{i=1}^{m} l_i\right) g^{-1}\left(\frac{\sum_{i=1}^{m} B_i}{\sum_{i=1}^{m} l_i}\right)$ since this is the minimum amount of energy sufficient to depart all the bits that have arrived by the end of the mth epoch, and spending any more energy is inefficient.

Consider the tightest string solution given in [2] which is the optimal solution to (1) with $B_{max} = \infty$, $B_1 = \infty$, and $B_n = 0$ for $n = 2, 3, \ldots, N$. The tightest string analogy here refers to the fact that for optimality, the expended energy curve must be the shortest path between the beginning and the end of the energy tunnel which is defined by an upper wall, i.e., the energy causality constraint (1b), and a lower wall, i.e., the battery constraint (1c). The shortest path is a piecewise linear curve that changes its slope only at energy arrivals. This change in the slope is in the positive direction if the battery is empty just before the energy arrival, and in the negative direction if the battery is full with the new energy arrival. See [2] for a detailed analysis.

Now, consider the tightest string solution for the relaxed problem (1) with an infinite capacity buffer, and an energy tunnel whose lower wall is described by (1c) and upper wall by (7). Note that (1e) is removed with $B_{max} = \infty$. Suppose again that this solution yields power values that are constant up until the beginning of the $(m+1)$th epoch, i.e., $p_1^* = p_2^* = \cdots = p_m^* \triangleq p^*$. Then, either one of (1c) and (7) must be active at the end of the mth epoch. If (7) is active, then increasing p^* will either be infeasible, or inefficient due to not having

Algorithm 1 Recursive algorithm for throughput maximization with $B_{max} = \infty$.

1: Find the minimum $\hat{m}$ such that (8) holds.
2: **if** $\hat{m}$ exists **then**
3: Set $w_{\hat{m}}$ as in (9).
4: Update $N = \hat{m}$ and continue. Rerun the algorithm for epochs $\hat{m}+1, \hat{m}+2, \ldots$
5: **end if**
6: Compute $\bar{p}_n = \min\left\{\frac{\sum_{i=1}^{n} E_i}{\sum_{i=1}^{n} l_i}, g^{-1}\left(\frac{\sum_{i=1}^{n} B_i}{\sum_{i=1}^{n} l_i}\right)\right\}$ for all $n = 1, 2, \ldots, N$.
7: Find $\tilde{m} = \underset{n=1,2,\ldots,N}{\arg\min}\ \bar{p}_n$.
8: Find the minimum m such that $\bar{p}_{\tilde{m}} \sum_{i=1}^{m} l_i < \sum_{i=1}^{m+1} E_i - E_{max}$. If no such m exists, then set $m = \tilde{m}$.
9: Set $p_n^* = \bar{p}_m$ for all $n = 1, 2, \ldots, m$.
10: Update indices as $n := n - m$ for all $n = m+1, m+2, \ldots, N$ and go to Step 1.

enough data to fully utilize the expended energy. Decreasing p^* will be suboptimal as a result of the concavity of $g(\cdot)$. If (1c) is active, then increasing p^* will be suboptimal since $g(\cdot)$ is concave, and decreasing it will be inefficient since some energy will have to be wasted due to finite E_{max}. Therefore, we have optimality for epochs $1, 2, \ldots, m$.

For the remainder of the transmission, we can reindex epochs as $\tilde{n} = n - m, \forall n = m+1, m+2, \ldots, N$, and apply the procedure given above. This way, we can recursively find the optimal power values for all epochs as was done in [1] for an infinite capacity battery at the transmitter.

As a final step, we remove the assumption $w_n = 0$ for all $n = 1, 2, \ldots, N$ by invoking Lemma 1 and noting that the two conditions in Lemma 1 on the battery and the buffer can be satisfied only if the lower wall defined by (1c) exceeds the new upper wall given in (7), i.e., for some epoch $\hat{m}$, we have

$$\sum_{i=1}^{\hat{m}+1} E_i - E_{max} > \left(\sum_{i=1}^{\hat{m}} l_i\right) g^{-1}\left(\frac{\sum_{i=1}^{\hat{m}} B_i}{\sum_{i=1}^{\hat{m}} l_i}\right). \quad (8)$$

If (8) occurs at the end of any epoch $\hat{m}$, then we can let the transmitter lose the excess energy, i.e.,

$$w_{\hat{m}} = \sum_{i=1}^{\hat{m}+1} E_i - E_{max} - \left(\sum_{i=1}^{\hat{m}} l_i\right) g^{-1}\left(\frac{\sum_{i=1}^{\hat{m}} B_i}{\sum_{i=1}^{\hat{m}} l_i}\right) \quad (9)$$

and start a new subproblem by reindexing the epochs as $\tilde{n} = n - \hat{m}, \forall n = \hat{m}+1, \hat{m}+2, \ldots, N$. The overall recursive solution is given in Algorithm 1 where the optimal power values are denoted by p_n^* for all $n = 1, 2, \ldots, N$.

B. Solution for a Finite Data Buffer

We can now solve (1) for the general set up with any B_{max}. A direct implication of the finite buffer capacity is that

$$l_n g(p_n) \leq B_{max} \quad (10)$$

for all $n = 1, 2, \ldots, N$ since the transmitter can start each epoch with at most B_{max} bits in its buffer. Hence,

Algorithm 2 Recursive algorithm for throughput maximization with arbitrary B_{max}.

1: Find the optimal solution to the relaxed problem using Algorithm 1.
2: Find the maximum m such that $p_1^* = p_2^* = \cdots = p_m^*$.
3: Find $\bar{m}$ using (11).
4: **if** $\bar{m}$ exists **then**
5: Set $p_n^\dagger = p_n^*$ for all $n = 1, 2, \ldots, \bar{m}-1$.
6: Set $p_{\bar{m}}^\dagger = g^{-1}\left(\frac{B_{max}}{l_{\bar{m}}}\right)$.
7: **else**
8: Set $p_n^\dagger = p_n^*$ for all $n = 1, 2, \ldots, m$.
9: Set $\bar{m} = m$.
10: **end if**
11: Update $E_{\bar{m}+1}$ using (12).
12: Update indices as $n := n - \bar{m}$ for all $n = \bar{m}+1, \bar{m}+2, \ldots, N$ and go to Step 1.

$p_n \leq g^{-1}(B_{max}/l_n)$ for all $n = 1, 2, \ldots, N$. Note that not all optimal transmission policies have to satisfy this for all $n = 1, 2, \ldots, N$, but there exists at least one which does. We can find the solution in this case in a similar way as we did for $B_{max} = \infty$. Suppose we again have that the optimal powers for the relaxed problem satisfy $p_1^* = p_2^* = \cdots = p_m^* \triangleq p^*$ for some $1 \leq m \leq N$. Let

$$\bar{m} = \min\left\{n : p_n^* \geq g^{-1}\left(\frac{B_{max}}{l_n}\right), 1 \leq n \leq m\right\}. \quad (11)$$

We know that at the end of the $\bar{m}$th epoch, the transmitter has an empty buffer. We also know that the $\bar{m}$th epoch is the first one at the end of which the transmitter runs out of data. Then, no transmission policy can depart more bits than $\sum_{n=1}^{\bar{m}-1} l_n p_n^* + B_{max}$ which can be achieved by the partial transmission policy $\left(p_1^*, p_2^*, \ldots, p_{\bar{m}-1}^*, g^{-1}\left(\frac{B_{max}}{l_{\bar{m}}}\right)\right)$ for epochs $1, 2, \ldots, \bar{m}$. We know that $p^* \geq g^{-1}\left(\frac{B_{max}}{l_{\bar{m}}}\right)$ since we assumed that the tightest string for the relaxed problem would start with constant rate transmission for m epochs, and we had to lower the power in the $\bar{m}$th epoch to make sure no energy was wasted. Any convex combination of p^* and $g^{-1}\left(\frac{B_{max}}{l_{\bar{m}}}\right)$ would result in dropped packets at the beginning of the $\bar{m}$th epoch since we would have to lower p^* for the first $\bar{m}-1$ epochs. Increasing p^* is also suboptimal as a direct consequence of the concavity of the rate function. Therefore, the partial transmission policy is optimal for the first $\bar{m}$ epochs. We can then update $E_{\bar{m}+1}$ as

$$E_{\bar{m}+1} := \left[\sum_{n=1}^{\bar{m}+1} E_n - \sum_{n=1}^{\bar{m}-1} l_n p_n^* - l_{\bar{m}} g^{-1}\left(\frac{B_{max}}{l_{\bar{m}}}\right)\right]^\wedge \quad (12)$$

where $[a]^\wedge = \min\{a, E_{max}\}$, and reindex the remaining epochs as $\tilde{n} = n - \bar{m}, \forall n = \bar{m}+1, \bar{m}+2, \ldots, N$. Finally, we can apply the procedure given above to epochs $\tilde{n} = 1, 2, \ldots, N - \bar{m}$. The recursive solution for the general case is given in Algorithm 2 where the optimal power values are denoted by $p_n^\dagger$ for all $n = 1, 2, \ldots, N$.

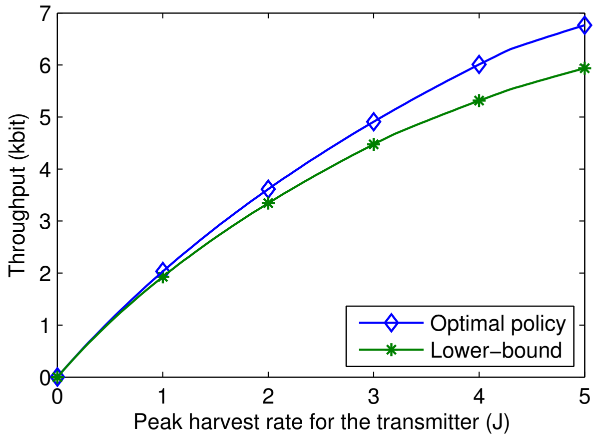

Fig. 4. Throughput values achieved by the optimal transmission policy versus those achieved by the suboptimal naïve policy for varying peak harvest rates.

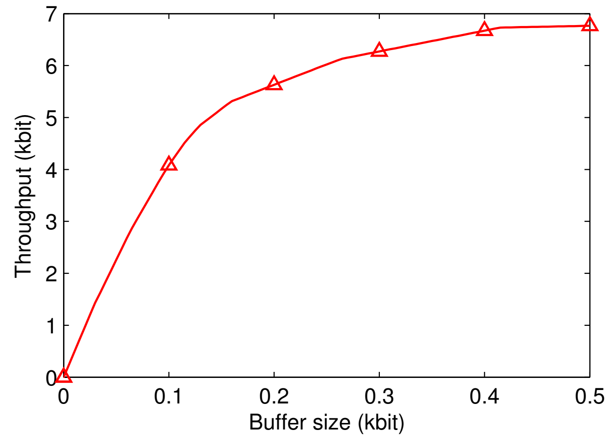

Fig. 5. Throughput by the optimal policy versus buffer size.

V. Numerical Results

For numerical results, we consider an AWGN channel with unit power gain and unit variance noise at the receiver, i.e., $g(p) = \frac{1}{2}\log(1+p)$. The results for the performance comparison simulations are given in Fig. 4. We set $E_{max} = 5$ J, $B_{max} = 0.5$ kbits, and vary the peak energy harvest rate at the transmitter from 0 J to 5 J. The peak data arrival rate is set at 0.5 kbits. The curve labeled "Optimal policy" gives throughput values achieved by our optimal solution. The curve labeled "Lower-bound" gives throughput values achieved by a naïve policy which simply spends E_n units of energy in the nth epoch, and sends $\min\{l_n g(E_n/l_n), B_n\}$ bits of data. We observe that the gap between the two curves widens as the peak energy harvest rate is increased, pointing to the advantage of computing the optimal transmission policy.

Fig. 5 shows the throughput values achieved by the optimal transmission policy in the same set up, except the peak energy harvest rate at the transmitter is fixed at 5 J, and B_{max} is varied from 0 kbit to 0.5 kbits. We observe that the impact of the buffer size (B_{max}) on the throughput is modest as compared to that of the peak energy harvest rate, i.e., lowering the peak energy harvest rate results in a steeper decrease in throughput compared to that caused by decreasing the buffer size. Another observation is that the throughput curve saturates at high buffer size values, meaning that larger buffers do not improve throughput indefinitely whereas high peak energy harvest rates would continue improving the achieved throughput since $g(p)$ is increasing in p, provided that there is a sufficient amount of data arriving at the transmitter. As a final remark, we note that the required buffer size is much lower than the achieved throughput, indicating that a relatively small buffer is sufficient for an energy harvesting transmitter.

VI. Conclusion

In this paper, we have studied an energy harvesting transmitter with a finite battery and a finite buffer. We have formulated the throughput maximization problem for this set up, and solved it by decoupling it into energy and data problems. We have identified a directional waterfilling solution with right permeable taps, water pumps, and overflow protection bins for the energy problem. We have also provided a second characterization of the optimal solution based on recursively applying the shortest path solution in [2]. We have observed that, with this approach, the problem can be divided into subproblems whenever the buffer is depleted. Future work includes characterizing the impact of buffer sizes for multiple energy harvesting node set ups on throughput.

References

[1] J. Yang and S. Ulukus, "Optimal packet scheduling in an energy harvesting communication system," *IEEE Transactions on Communications*, vol. 60, no. 1, pp. 220–230, 2012.

[2] K. Tutuncuoglu and A. Yener, "Optimum transmission policies for battery limited energy harvesting nodes," *IEEE Transactions on Wireless Communications*, vol. 11, no. 3, pp. 1180–1189, 2012.

[3] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, "Transmission with energy harvesting nodes in fading wireless channels: Optimal policies," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1732 –1743, Sep. 2011.

[4] B. Varan, K. Tutuncuoglu, and A. Yener, "Energy harvesting communications with continuous energy arrivals," in *Proceedings of the 2014 Information Theory and Applications Workshop, ITA'14*, Feb. 2014.

[5] C. Huang, R. Zhang, and S. Cui, "Throughput maximization for the Gaussian relay channel with energy harvesting constraints," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 8, pp. 1469–1479, 2013.

[6] K. Tutuncuoglu and A. Yener, "Sum-rate optimal power policies for energy harvesting transmitters in an interference channel," *Journal of Communications and Networks*, vol. 14, no. 2, pp. 151 –161, 2012.

[7] J. Yang and S. Ulukus, "Optimal packet scheduling in a multiple access channel with energy harvesting transmitters," *Journal of Communications and Networks*, vol. 14, no. 2, pp. 140–150, 2012.

[8] J. Yang, O. Ozel, and S. Ulukus, "Broadcasting with an energy harvesting rechargeable transmitter," *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 571–583, 2012.

[9] B. Varan and A. Yener, "Multi-pair and multi-way communications using energy harvesting nodes," in *Proceedings of the 47th Asilomar Conference on Signals, Systems and Computers*, Nov. 2013.

[10] M. Gregori and M. Payaro, "Energy-efficient transmission for wireless energy harvesting nodes," *IEEE Transactions on Wireless Communications*, vol. 12, no. 3, pp. 1244–1254, March 2013.

[11] D. P. Bertsekas, *Nonlinear programming*. Belmont, Mass: Athena Scientific, 1999.

[12] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear Gauss–Seidel method under convex constraints," *Operations Research Letters*, vol. 26, no. 3, pp. 127–136, 2000.