

RESEARCH

Open Access

Optimal offline broadcast scheduling with an energy harvesting transmitter

Hakan Erkal*, F Mehmet Ozelik and Elif Uysal-Biyikoglu

Abstract

We consider an energy harvesting transmitter broadcasting individual data to two receivers. Data packets intended for each user are assumed to arrive at arbitrary but known instants. The goal is to minimize the total transmission time of the packets arriving within a certain time window, using the energy harvested during this time. Energy harvests are also modelled to occur at known discrete instants. An achievable rate region with structural properties satisfied by the two-user additive white Gaussian noise broadcast channel capacity region is assumed. Structural properties of power and rate allocations are established, as well as the uniqueness of the optimal policy. An iterative algorithm, DuOpt, is devised for efficient solution of this offline problem. DuOpt is compared with the sequential unconstrained minimization (SUMT) solution technique on randomly generated problem instances and is observed to solve the problem two orders of magnitude faster on average than SUMT.

Keywords: Packet scheduling; Energy harvesting; AWGN broadcast channel; Energy-efficient scheduling

1 Introduction

The basic offline problem of energy-efficient packet transmission scheduling [1-5] is to assign code rates (consequently, transmission durations) to a set of packets whose arrival times are known beforehand, so that they are all transmitted within a given time window with minimum total energy. The solution needs to strike a trade-off between energy efficiency and delay: lowering energy expenditure per bit of information calls for lowering transmission rates, which causes delay. Recently, the dual problem of minimizing transmission duration of a given amount of data has been formulated with a model where energy gets “harvested” or replenished at certain known instants [6]. The irregularity in the availability of energy introduced additional richness to the problem.

While in former formulations, transmission rate needs to be adapted to the arrival rate of information, in [6], it is adapted to the rate of generation of energy. The practical motivation for this formulation was the case of a communication system that relies on *energy harvesting* from the environment (e.g., through solar, thermal, vibration sources) to supply its battery [7]. When the delay

constraints for the data are relatively relaxed (considering the energy harvesting time scale) and battery size is small, as in the case of a distributed wireless sensor network (consider, for example, a structural monitoring WSN on a building, bridge or aircraft, where most of the data is for periodic network maintenance where battery replacement for nodes is out of the question such that the network has to sustain itself energy obtained from vibrations [8]), it will be reasonable to adapt transmit power and rate to the availability of energy [9]. By conservatively transmitting at low power during low energy harvest periods, the communication node may increase its longevity, such that it will not need to entirely power off until the next recharge. Nodes could thus achieve what is sometimes termed ‘perpetual operation’ by self-sustaining on environmental energy.

The point-to-point problem in [6] was recast for finite energy storage [10] and for a fading wireless channel [11]. The formulation has been extended to a broadcast channel (BC) in [12,13], considering a static pool of data to be sent at the beginning of the schedule. The same BC problem was also studied under a limited battery constraint [14].

The problem in [12] and [13] has recently been reformulated in [15] by relaxing the assumption that data are ready at the beginning of the schedule. The sender

*Correspondence: herkal@ieee.org
Department of Electrical and Electronics Eng., METU, Ankara 06800, Turkey

(transmitter) gets replenished with arbitrary amounts of energy as well as data packets of arbitrary length destined to each user at arbitrary points in time. The rates at which information is transmitted to each user, given a total instantaneous power, are assumed to be selected from an achievable rate region that obeys certain structural properties satisfied by the additive white Gaussian noise (AWGN) BC. This paper, extending the work in [15], shows the uniqueness of the optimal power and rate allocation for the problem and exhibits a fast algorithm for obtaining it.

The choices of power level and the rates to individual receivers across time is called a *schedule*. An optimal scheduling policy is defined to be one that transmits all the bits that have arrived within a certain time window, W , in the minimum possible amount of time $T^{\text{opt}} \geq W$. The policy is allowed to use as many energy harvests as it needs, provided it respects causality (no energy or data are transmitted before it becomes available).

The problem considered in this paper is an *offline* problem, where data arrival and energy harvest instants and amounts are assumed to be known in advance. Although this kind of prior information of data and energy arrivals may not be possible in all practical settings, analysis of the offline optimal solution nevertheless shed light on the nature of a good power/rate policy and boundaries on the best performance. Moreover, in many scenarios, energy harvest processes are predictable (e.g., solar energy harvest rate determined by daily and seasonal irradiation patterns), and data arrivals could also be deterministic depending on session or protocol requirements (e.g., periodic transmissions made by a sensor node). The offline assumption could be a good approximation in the case of predictable processes.

Online formulations have also appeared in the literature. For example, Chen et al. and Gatzianas et al. [16,17] develop online scheduling policies for multihop networks on finite-horizon and infinite horizon problem formulations, respectively. In [18], another finite-horizon throughput maximization problem is investigated. Considering fading conditions, a dynamic programming solution is proposed. Similarly, Ozel et al. [19] also addresses the optimal online solution through stochastic dynamic programming. Online formulations tend to be sensitive to the particulars of the stochastic models assumed for data and energy processes and consequently do not often lead to clean structural results (with the exception of threshold-type results, e.g., [20]) which is another aspect that makes the offline approach attractive.

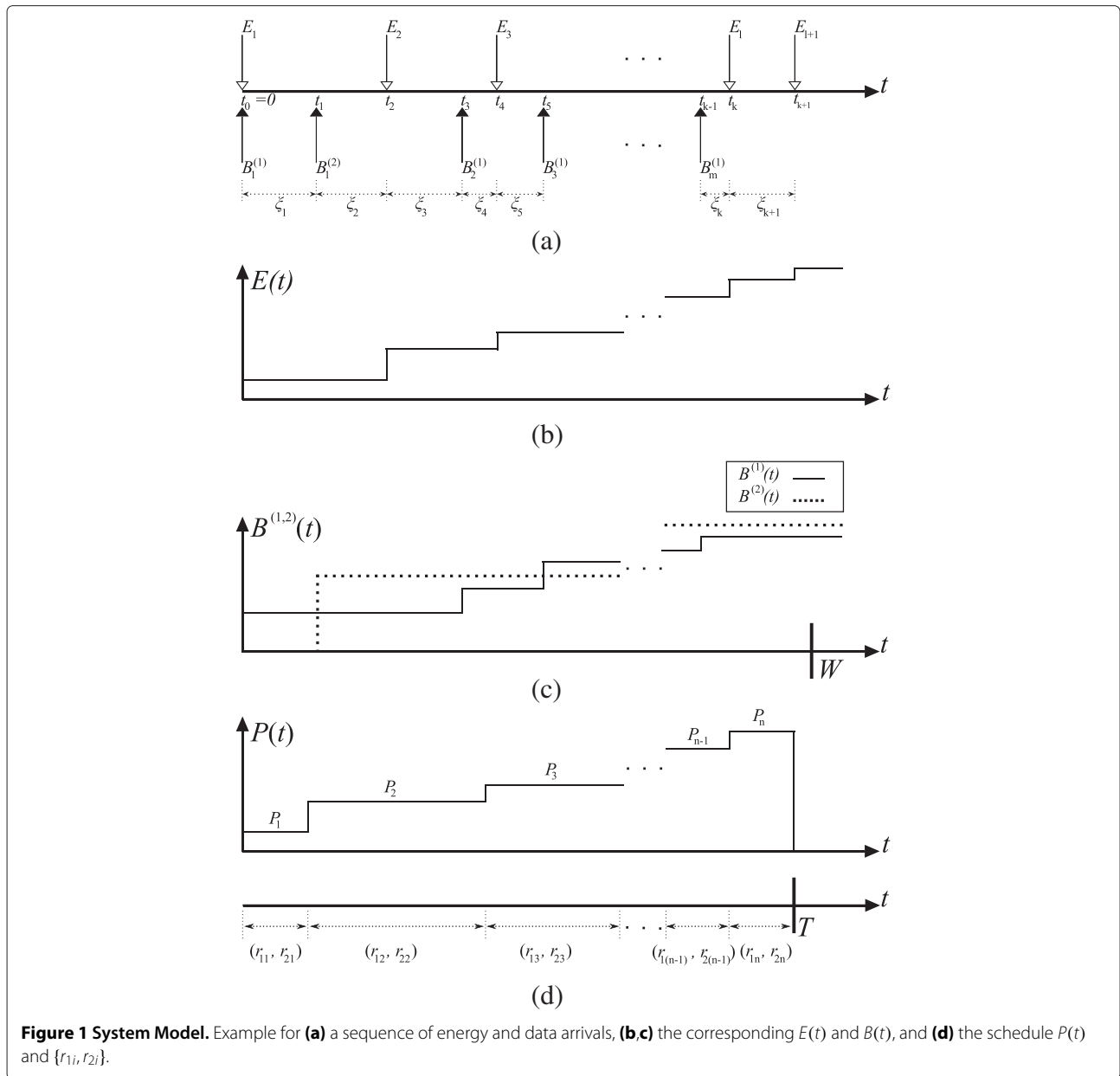
This paper essentially bridges the formulation that considered offline scheduling of *data* that become available at arbitrary points in time [1,2] (where there is no energy harvesting) and the formulation in which chunks of *energy*

becoming available at arbitrary points in time [11,12] on a BC (where data are pooled at the beginning). We employ some of the methods of [2] to strictly address this more general problem. The contribution of this paper can also be described as an extension to the BC problem considered in [6]. Specifically, this paper completes the results of [1,2,11,12] with the following novel contributions: (1) establishment of structural properties of the transmission time minimizing scheduling policy in the energy harvesting BC with packet arrivals, (2) the uniqueness of the optimal policy, (3) an iterative algorithm, *DuOpt*, which is proved to return the optimal schedule when all the data of the user with the weaker channel is pooled, observed on numerical examples to solve the problem in the general case as well, (4) comparison of the computation time of the proposed method against solving the same problem with the well-known *sequential unconstrained minimization technique* (SUMT). It is shown in section 2 that in an optimal policy, transmit power used is constant within each epoch and may only rise from one epoch to the next so that once it starts, the transmitter never lowers its power until it finally goes silent. On the other hand, the transmitter should increase its power only under certain conditions. These conditions, along with other structural properties of power and rate are established in section 3. Next, the uniqueness of the optimal policy is established under the condition that all of the weak user's bits are available at the beginning. In section 5, the DuOpt Algorithm, which returns a feasible schedule carrying the same structural properties that the optimal is shown to have, is described. It is proved that DuOpt returns the optimal schedule when all the data of the user with the weaker channel has been pooled at the beginning of the schedule. Next, through defining an equivalent convex problem, which proves the uniqueness of the optimal solution in the general case, the optimum is shown to be attainable using SUMT in section 6. Finally, studying numerical examples in section 7, DuOpt is observed to be much faster than SUMT in converging to the optimal solution found by the latter, even when the data pooling condition is lifted.

We start by making the problem statement precise in the next section.

2 System model

We consider a BC with one transmitter and two receivers. Arbitrary amounts of energy, $\{E_i < \infty, i = 1, 2, \dots\}$, as well as data intended for each user $\{B_i^{(1)}, B_i^{(2)} < \infty, i = 1, 2, \dots\}$ become available to the sender at arbitrary times t_i . A possible sequence of data and energy arrivals is illustrated in Figure 1. $E(t)$ denotes the total energy that has been *harvested* in $[0, t)$ (regardless of how much of it has been used.) Similarly, $B^{(1)}(t)$ and



$B^{(2)}(t)$ denote the total number of bits destined to the first and second user, respectively, that the sender has obtained in $[0, t)$. The interval between any two sequential arrival events (regardless of energy or data) will be called an inter-arrival epoch. The length of the i th epoch is $\xi_i = t_i - t_{i-1}$.

Arrival times and amounts of energy and bits in the time window $t \in [0, W)$ are known by the sender at $t = 0$. It is assumed that harvested energy and data are stored in infinite capacity buffers and available for use instantaneously as they arrive. It is also assumed that code rate and transmission power decisions can be changed instantaneously. However, codeword block lengths will

be chosen such that each codeword is sent completely within a single epoch (note that starting and ending times of epochs are known ahead of time), so that no arrival event occurs during a codeword. Consequently, the power and rate pair decision will be fixed throughout each codeword.

We are interested in minimizing the total transmission time for packets arriving by a certain time $W < \infty$, so W.L.O.G., set $B^{(i)}(t) = B^{(i)}(W)$ for $t > W$, $i = 1, 2$. A schedule, which is a sequence of power and rate allocations, is feasible if it sends $B^{(1)}(W) < \infty$ bits to the first user and $B^{(2)}(W) < \infty$ to the second user (with a certain level of reliability^a), without violating causality

(at any time, using available energy and data by that time). We are interested in finding among all feasible schedules one with the smallest completion time, T^{opt} .

The structure of the achievable rate region will be based on the two-user AWGN BC. The capacity region of a two-user discrete time AWGN BC with average power constraint P , noise variance σ^2 , where the first user's channel gain ($s_1 > 0$) is larger than the second user's ($s_2 > 0$), consists of rate pairs (r_1, r_2) satisfying:

$$r_1 \leq \frac{1}{2} \log_2 \left(1 + \frac{\alpha s_1 P}{\sigma^2} \right), r_2 \leq \frac{1}{2} \log_2 \left(1 + \frac{(1 - \alpha) s_2 P}{\alpha s_2 P + \sigma^2} \right) \quad (1)$$

where α , ($0 \leq \alpha \leq 1$), denotes the fraction of P used for the first user [21]. Since $s_1 > s_2$, the first user will be referred as the "strong user" and the second as the "weak user". We assume that the relative channel gains of the users stay fixed throughout the scheduling period. This models the case of static channels, or when channel variations are too slow with respect to the scheduling period, or channel variations are small with respect to the difference of path loss between the users, such as when one user is markedly more distant to the transmitter than the other.

Using capacity-achieving codes, the rate pair (r_1, r_2) can be selected from the boundary of the rate region, where inequalities in Equation 1 becomes equalities. Combining these equalities, power can be expressed in terms of the user rates, $P = g(r_1, r_2)$. Moreover, each user's rate can be expressed as a function of P and the other user's rate, such as

$$r_1 = h_1(P, r_2) = \frac{1}{2} \log_2 \left(\frac{s_1(s_2 P + \sigma^2)}{s_2 \sigma^2 2^{2r_2}} - \frac{s_1 - s_2}{s_2} \right) \quad (2)$$

$$r_2 = h_2(P, r_1) = \frac{1}{2} \log_2 \left(\frac{\frac{s_2 P}{\sigma^2} + 1}{\frac{s_2}{s_1} (2^{2r_1} - 1) + 1} \right). \quad (3)$$

The rate functions h_1 and h_2 defined on $\mathfrak{R}^+ \times \mathfrak{R}^+$ satisfy the following properties:

- (i) Nonnegativity: $h_1(P, r) \geq 0, h_2(P, r) \geq 0$.
- (ii) Monotonicity: $h_1(P, r), h_2(P, r)$ are both monotone decreasing in r , and monotone increasing in P .
- (iii) Concavity: $h_1(P, r), h_2(P, r)$ are strictly concave in $P > 0$ and $r > 0$: $\frac{\partial^2 h_i(P, r)}{\partial P^2} < 0, \frac{\partial^2 h_i(P, r)}{\partial r^2} < 0$, for $i \in 1, 2$.
- (iv) $\frac{\partial^2 h_1(P, r)}{\partial r \partial P} \geq 0, \frac{\partial^2 h_1(P, r)}{\partial P \partial r} \leq 0$.
- (v) $\frac{\partial^2 h_2(P, r)}{\partial r \partial P} = 0, \frac{\partial^2 h_2(P, r)}{\partial P \partial r} = 0$.

The results in the rest of the paper will be valid for any rate function satisfying (i,ii,iii,iv,v), which are also satisfied by the AWGN BC [12].

It is established in the following lemma that one can limit attention to schedules that do not change their power and rate allocations within epochs. This is a simple and well-known consequence of our choice of concave rate functions, as has been proved previously in [12]: when there are no constraints, an even (equal across time) allocation is optimal. By definition, there are no data or energy arrivals during an epoch, hence an even allocation during an epoch is always possible. Given any schedule that has variation in an epoch, it can simply be modified to one with even rate, which will only improve the policy. In fact, proof of the claim is identical with the one stated and proved in Lemma 2 of [12]; however, we summarize it here for clarity.

Lemma 1. *In an optimal schedule, the power and rate pair remain constant within each epoch, except for the epoch during which the schedule ends.*

Proof. During an epoch, there are no energy or data arrivals, hence a constant rate and power allocation is feasible. By concavity of the functions $h_1(P, r)$ and $h_2(P, r)$, and Jensen's inequality, equalizing r_2 across time minimizes the power needed to maintain a rate r_1 in time, i.e., if $r_1 = \alpha h_1(P_1, r_{21}) + (1 - \alpha) h_1(P_2, r_{22})$ and $r_1 = h_1(P, \alpha r_{21} + (1 - \alpha) r_{22})$, then $P \leq \alpha P_1 + (1 - \alpha) P_2$. Further details are given in [12]. As any allocation that is claimed to be optimal can be modified and can only be improved to obtain a constant rate and constant power across each epoch, one can limit attention to policies that has constant allocations within epochs. The only exception to this is the last epoch used by the schedule. In the last epoch, when the schedule ends, the power will drop to zero. ■

Accordingly, let P_i be the total transmit power and r_{ji} be the rate assigned to the j th user during epoch i . Let P_{ji} represent the power assigned to j th user during epoch i . We are now ready to state the problem in terms of power and rate allocations to epochs, more precisely, an assignment of total power and the strong user's rate to each epoch (the weak user's rate is then determined by $r_{2i} = h_2(P_i, r_{1i})$. A final technical assumption will be useful in stating the problem: we shall assume that there is some $k^{\text{up}} < \infty$ such that there is at least one feasible schedule that ends within the first k^{up} epochs. In other words, k^{up} is an upper bound for both the number of epochs and the number of harvests to be considered (please see Figure 1a). If $k^* + 1$ denotes the number of epochs used by an optimal schedule, then $k^* + 1 \leq k^{\text{up}}$.

Problem 1. Transmission time minimization of data arriving at arbitrary points on an energy harvesting BC:

Minimize: $T = T(\{P_i, r_{1i}\}_{1 \leq i \leq k^{\text{up}}})$

subject to: $P_i \geq 0, 0 \leq r_{1i} \leq h_1(P_i, 0), r_{2i} = h_2(P_i, r_{1i})$

$$\sum_{i=1}^k P_i \xi_i \leq E(t_k)$$

$$\sum_{i=1}^{k^*} P_i \xi_i + P_{(k^*+1)}(T - \sum_{i=1}^{k^*} \xi_i) \leq E(T) \quad (4)$$

$$\sum_{i=1}^k r_{1i} \xi_i \leq B^{(1)}(t_k), \sum_{i=1}^k r_{2i} \xi_i \leq B^{(2)}(t_k) \quad (5)$$

for $k = 1, 2, \dots, k^* = \max\{i : \sum_{j=1}^i \xi_j < T\}$

$$\sum_{i=1}^{k^*} r_{1i} \xi_i + r_{1(k^*+1)}(T - \sum_{i=1}^{k^*} \xi_i) = B^{(1)}(W) \quad (5a)$$

$$\sum_{i=1}^{k^*} r_{2i} \xi_i + r_{2(k^*+1)}(T - \sum_{i=1}^{k^*} \xi_i) = B^{(2)}(W). \quad (5b)$$

We will refer to Equations 4 and 5 as energy and data causality constraints, respectively, as these ensure no energy is consumed and no bit is transmitted before becoming available. In addition, when the k th inequality in Equation 4 holds with equality, we shall say that k th energy constraint is active. Similarly, equality case in Equation 5 will be referred as a data constraint being active. Finally, the feasibility constraints (Equations 5a and 5b) ensure all the data bits destined to each user are transmitted.

In the next section, we investigate structural properties that any optimal schedule has to satisfy.

3 Structure of an optimal policy

Lemma 1 recorded that in an optimal schedule, power can only change upon a data arrival or energy harvest. The next result states that when power changes, it can only increase. The key to the proof is that more “bits per Joule” can be sent by evenly distributing energy across a time interval (i.e., maintaining a constant power level, which is a consequence of the convexity properties of our rate functions.) If an even distribution of power requires transferring energy or bits to the latter epoch, it can always be done; hence, total transmit power never decreases in time, but power may increase in time because even distribution of power may result in unmet causality constraints. We state these results in Lemma 2.

Lemma 2. Consider an optimal schedule that ends during epoch $k^* + 1$. Power is non-decreasing with epoch index, i.e, $P_i \leq P_{i+1}$ for $i = 1, 2, \dots, k^*$.

Proof. See Appendix. ■

As stated in Lemma 2, power cannot decrease, yet may rise in time like a staircase function formed bands of constant power that last for several epochs (see Figure 1d). In the next lemma, we note a necessary condition for such a rise to occur in an optimal policy.

Lemma 3. In an optimal policy, power can only rise at t_i (end of epoch i) if at least one of the conditions below holds:

- a. Energy constraint is active at point t_i . (i.e., the i th energy constraint is active)
- b. The data constraints for both users are active at point t_i . (i.e., the set of constraints in Equation 5)
- c. The weak user’s data constraint is active and data arrival to the weak user occurs at time t_i .

Proof. See Appendix. ■

The next set of results illustrate the structure of rate allocation in conjunction with the power allocation in an optimal policy.

Corollary 1. In an optimal policy,

1. If power increases upon a data arrival for the weak user, all available data destined to the weak user have been transmitted by this point in time.
2. If power rises upon a data arrival for the strong user, all available bits have been sent by this event.
3. If power increases upon an energy harvest, all energy available at the beginning of the former constant power band has been consumed by this energy harvest.

Proof.

1. Suppose that power increases upon a bit arrival for the weak user occurring at t_i . As there is no energy constraint at t_i , bringing power levels closer does not contradict with the energy causality in this case. This implies that conditions (b) or (c) stated in Lemma 3 must hold. However, we know that there is a data arrival for the weak user at t_i ; so, if (b) were true, then (c) would be true as well. Therefore, condition(c) holds in either case.
2. Suppose that power increases upon a bit arrival for the first user. With similar reasoning to part 1, condition (a) of Lemma 3 cannot hold. As there is also no data arrival for the weak user, condition (b) must be satisfied.

3. As there is no data arrival at the time when power increases, the only possibility that power increases upon an energy harvest is condition (a) of Lemma 3. ■

In the rest, some properties will be proved under the assumption that weak user data constraint would not be active. This assumption is guaranteed if all the weak user data is ready at the beginning. We shall abbreviate this condition as follows:

Definition 1. *Weak user full buffer condition (WUFBC) is said to be satisfied whenever all of the data of the weak user is available at the beginning of transmission. That is, $B^{(2)}(W) = B^{(2)}(0)$.*

The following lemma states an important feature of the strong user's rate policy under WUFBC.

Lemma 4. *Consider two consecutive epochs i and $i + 1$ of a given schedule, ending at t_i and t_{i+1} by definition, and suppose WUFBC holds for the problem instance. The following is necessary for the rate and power allocation to these two epochs of the given schedule to be locally optimal: The strong user's rate is constant throughout $[t_{i-1}, t_i)$, and $[t_i, t_{i+1})$. Furthermore, the rate may jump up at $t = t_i$ (staying constant otherwise) if at least one of the below is true:*

1. There is data arrival to the strong user at $t = t_i$, and all the data that arrived before t_i has been transmitted by t_i .
2. An energy harvest occurs at $t = t_i$, and all of the power has been used for the strong user during epoch i .

Proof. Suppose that $r_{1i} \leq r_{1(i+1)}$. Since WUFBC is assumed, weak user's data constraint would never be active and one can find a better schedule by bringing the rates of the strong user closer by Lemma 6 of [12]^b. Therefore, the strong user's rate cannot decrease. However, the strong user's rate may increase because it would violate either bit or energy causality to transfer some strong user bits from epoch $i + 1$ to i . First, it is against *bit causality* to transfer strong user bits from epoch $i + 1$ to epoch i , if the first condition holds. Secondly, if the second condition is satisfied, we cannot bring strong user's rates in the two consecutive epochs closer to each other as doing so would violate *energy causality*. ■

We establish the uniqueness of the solution of Problem 1 under WUFBC in the next section.

4 Uniqueness of the optimum schedule under WUFBC

In the following lemma, we note that an optimal schedule uses all energy harvested by the time the schedule ends completely.

Lemma 5. *The energy consumed by an optimal schedule that ends at T^{opt} is equal to $E(T^{opt})$.*

Proof. Note that, as $E(T^{opt})$ is the total energy that has been harvested by T^{opt} , it is impossible for the schedule to have consumed more than $E(T^{opt})$. Hence it suffices to show that the energy used by an optimal schedule cannot be less than $E(T^{opt})$. Consider a schedule that completes the transmission of all data at T^{opt} by consuming less energy than harvested and has leftover energy in its energy buffer^c at T^{opt} . The remaining energy in the buffer could have been used in the last epoch to strictly reduce the transmission completion time. This contradicts the optimality of T^{opt} . Hence, we have proved that the energy used cannot be less than $E(T^{opt})$. This in turn implies that the energy used must be exactly equal to $E(T^{opt})$. ■

Next, we show the uniqueness of the optimal schedule under WUFBC.

Theorem 1. *There is a unique optimum schedule under WUFBC, i.e., a unique power-rate allocation achieving T^{opt} .*

Proof. See Appendix. ■

5 The DuOpt Algorithm

The problem in [12] which is a special case of Problem 1, where both users' data is available at the beginning, was shown to be solved in [12] by the FlowRight Algorithm [22]. Along similar lines, we develop an algorithm that we call DuOpt for solving Problem 1 in its general form. As a matter of fact, DuOpt simply reduces to FlowRight when the given problem instance has all the data arriving at $t = 0$. Similarly to FlowRight, DuOpt starts with any feasible schedule and reduces the transmission completion time iteratively. Let the number of epochs and the transmission completion time of the initial schedule be k^{up} and T^{up} , respectively. In each iteration, DuOpt sequentially updates rates and powers of two consecutive epochs at a time, i.e., epochs $(1, 2)$, $(2, 3)$, ..., until all epochs are updated. Then, starting from the first epoch pair, DuOpt continues with the next iteration. DuOpt stops after N iterations such that $N = \min\{n : T^{n-1} - T^n \leq \epsilon, i = 1, \dots, k^n, j = 1, 2\}$, where $T^n \leq T^{up}$ is the transmission completion time, $k^n \leq k^{up}$ is the number of epochs used at the end of n th iteration, and ϵ is a predefined threshold.

Hereafter, we will briefly outline the local optimizations over epoch pairs. In Theorem 2, it will be shown that local optimizations can only improve the schedule. We will also prove that under WUFBC, successive iterations strictly improves the schedule unless it is optimal.

5.1 Local optimizations

Let E_i^n denote the energy used during the i th epoch and b_{ji}^n denote the number of bits transmitted to the j th user during epoch i at the end of n th iteration. Suppose that DuOpt is at the n th iteration and running a local optimization over epoch pair $(i, i + 1)$. The values of b_{jz}^n and E_z^n , $\forall z \in \{1, 2, \dots, i - 1\}$ have already been found by previous local optimizations. At the end of this optimization, E_i^n and b_{ji}^n will be determined; E_{i+1}^{n-1} , E_{i+2}^{n-1} , and $b_{j(i+1)}^{n-1}$ will be reset to new values that conserve total energy consumption and data transmission in these epochs. The goal of the local optimization is surely to minimize the total transmission completion time of all the packet arrivals. Hence, it is logical to minimize the transmission time in the local optimization problem, which results in a gap^d if transmission ends before the end of $(i + 1)$ th epoch. This gap is used in the next local optimization to further reduce the transmission time via transferring bits or energy between epochs $(i + 1)$ and $(i + 2)$; hence, a new gap occurs at the end of the next local optimization. This new gap propagates to the end of the transmission resulting a reduction in the total transmission completion time [12]. However, in some cases, an epoch long gap occurs, and this gap is useless for the next local optimization, i.e., energy or data transfer between epochs in the next local optimization is impossible because of constraints. In that case, it is better to just spread the data out till the end of the second epoch in the local problem and minimize the energy consumption so that the excess energy can be used to further reduce the transmission time in the next local optimization. This leads to two different local optimization functions: *time minimization* and *energy minimization*. These functions both support the global objective in different ways. Time minimization aims to find the minimum amount of time,

$T_{(i,i+1)}^n$, to transmit $b_{j(i,i+1)}^n = b_{ji}^{n-1} + b_{j(i+1)}^{n-1}$ bits to each user using the energy available in epoch pair $(i, i + 1)$, i.e., $E_{(i,i+1)}^{n-1} = E_i^{n-1} + E_{i+1}^{n-1}$. On the other hand, energy minimization aims to find the minimum energy, $E_{(i,i+1)}^n$, to transmit $b_{j(i,i+1)}^n$ bits to each user in two epoch durations, i.e., $\xi_i + \xi_{i+1}$, and excess energy, $E_i^{n-1} + E_{i+1}^{n-1} - E_{(i,i+1)}^n$, is transferred to the $(i + 2)$ th epoch in order to conserve energy. Both of the optimizations respect energy and bit causalities, i.e., $E_i^n \leq E(t_i) - \sum_{s=1}^{i-1} E_s^n$ and $b_{ji}^n \leq B^{(j)}(t_i) - \sum_{s=1}^{i-1} b_{js}^n$, $j \in \{1, 2\}$. For details of the local optimization, see Appendix.

Suppose that all the feasible packets have been transmitted until the end of the i th epoch and there are still packets to arrive after t_i . Then, further minimization of transmission completion time of sequential epochs before t_i will be suboptimal. On the other hand, we can minimize the energy consumption until t_i and use the excess energy to minimize the transmission completion time. Therefore, utilization of energy minimization for local optimizations in Problem 1 is very crucial. If it is guaranteed that current schedule uses at least the same amount of energy as optimal schedule until t_i , DuOpt uses the energy minimization function up to i th epoch pair and the time minimization function for the rest. In order to determine when to switch from energy minimization to time minimization, a *Flag* is placed at the i th epoch pair. Initially, the Flag is set to zero, and DuOpt starts with performing time minimization on epoch pairs. During the n th iteration, if all the feasible bits are transmitted by the i th epoch for $\exists i \in \{1, 2, \dots, k^n\}$, then the Flag is set to i (Flag $< i$). In the following iterations, energy minimization function is used up to the i th epoch pair. Figure 2 illustrates the Flag usage and the pseudo-code in Algorithm 1 outlines the DuOpt Algorithm.

Theorem 2. *The following statements hold:*

1. *Successive iterations of DuOpt can only improve the schedule.*
2. *DuOpt stops and returns a schedule with $\{r_{1i}^\infty, r_{2i}^\infty\}$.*

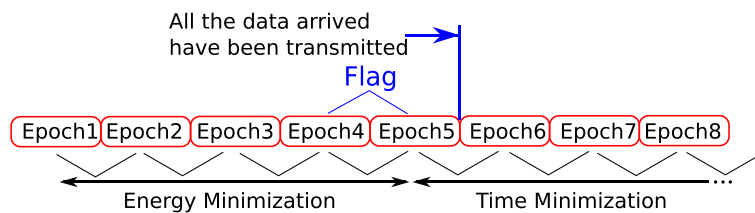


Figure 2 Illustration of DuOpt Flag. Illustration of the Flag and local optimizations, where all the feasible bits have been transmitted until the end of the fifth epoch; hence, a Flag is set to the fifth epoch pair, i.e., Equations 4 and 5. Energy minimization is performed upto epoch pair with the Flag, and time minimization is performed for the rest.

Algorithm 1 DuOpt Algorithm

```

1: Initialize();
2:  $n \leftarrow 0$ ,  $\text{Flag} \leftarrow 0$ ,  $T^0 \leftarrow T^{\text{up}}$ 
3: repeat
4:    $n++$ 
5:   for  $i = 1$  to  $(k^n - 1)$  do
6:      $e_{i,\max}^n \leftarrow E(t_i) - \sum_{m=1}^{i-1} e_m^n$ 
7:      $b_{1i,\max}^n \leftarrow B^{(1)}(t_i) - \sum_{m=1}^{i-1} b_{1m}^n$ 
8:      $b_{2i,\max}^n \leftarrow B^{(2)}(t_i) - \sum_{m=1}^{i-1} b_{2m}^n$ 
9:      $b_1^n \leftarrow b_{1i}^{n-1} + b_{1(i+1)}^{n-1}$ 
10:     $b_2^n \leftarrow b_{2i}^{n-1} + b_{2(i+1)}^{n-1}$ 
11:    if  $i \leq \text{Flag}$  then
12:       $[b_{1i}^n, b_{1(i+1)}^{n-1}, b_{2i}^n, b_{2(i+1)}^{n-1}, E_i^n, E_{i+1}^{n-1}, E_{i+2}^{n-1}]$ 
        = Minimize_Energy( $E_i^{n-1}, E_{i+1}^{n-1}, E_{i+2}^{n-1}$ ,
         $b_1^n, b_2^n, e_{i,\max}^n, b_{1i,\max}^n, b_{2i,\max}^n$ )
13:    else
14:       $[b_{1i}^n, b_{1(i+1)}^{n-1}, b_{2i}^n, b_{2(i+1)}^{n-1}, E_i^n, E_{i+1}^{n-1}]$ 
        = Minimize_Time( $E_i^{n-1}, E_{i+1}^{n-1}, b_1^n, b_2^n$ ,
         $e_{i,\max}^n, b_{1i,\max}^n, b_{2i,\max}^n$ )
15:    end if
16:    if  $b_{1i,\max}^n == b_{1i}^n$  &&  $b_{2i,\max}^n == b_{2i}^n$  &&  $\text{Flag} < i$ 
        &&  $i < k^n - 1$  then
17:       $\text{Flag} = i$ 
18:    end if
19:  end for
20:  Calculate_{T} (&T^n) {Calculate current transmission completion time.}
21: until  $T^n == T^{n-1}$ 

```

Proof.

- Suppose that DuOpt is running its n th iteration. After the local optimization on the i th epoch pair, we obtain $\{(r_{1i}^n, r_{2i}^n), E_i^n\}$ and reset the values of $\{(r_{1(i+1)}^{n-1}, r_{2(i+1)}^{n-1}), E_{i+1}^{n-1}, E_{i+2}^{n-1}\}$. If the Flag is not placed before the i th epoch pair, i.e., $\text{Flag} \geq i$, then the aim of the local optimization will be energy minimization. Following the local optimization on the i th epoch pair, the excess energy will be transferred to E_{i+2}^{n-1} . In the next local optimization, this excess energy is either further transferred or is used to reduce the transmit time. On the other hand, if $\text{Flag} < i$, then the aim of the local optimization on the i th epoch pair will be time minimization. After the local optimization the transmission completion time of the bits in epochs, $(i, i+1)$ will either be equal to or before the end of the epoch $(i+1)$. That is, a gap may occur within i th epoch pair. In the next local optimization, this gap would propagate to the $(i+2)$ th epoch [12]. During the n th iteration of DuOpt, if a gap occurs or excess energy is transferred during local optimizations, then the gap (or the excess energy,

respectively) will propagate to the last epoch pair resulting in an ultimate reduction the transmission completion time at the end of the iteration, i.e., $T(r_{1i}^n, r_{2i}^n) < T(r_{1i}^{n-1}, r_{2i}^{n-1})$. If neither excess energy nor a gap occurs during local optimizations, then transmission completion time cannot be decreased and DuOpt will stop by definition.

Both local optimizations are in favor of the next local optimization. Therefore, if in either one of the local optimizations, a gap occurs or excess energy is transferred, then it would propagate till the last epoch pair, and finally the transmission completion time would decrease at the end of n th iteration, i.e., $T(r_{1i}^n, r_{2i}^n) < T(r_{1i}^{n-1}, r_{2i}^{n-1})$. If neither excess energy nor gap occurs during local optimizations, then transmission completion time would not be decreased and DuOpt would stop.

- In Part 1, we have shown that transmission completion time, $T(r_{1i}^n, r_{2i}^n)$, is strictly decreasing in each iteration; meanwhile, it is bounded below by T^{OPT} . Therefore, the iterations of DuOpt stop and return a schedule, $\{r_{1i}^\infty, r_{2i}^\infty\}$. ■

5.1.1 Optimality of the DuOpt Algorithm under WUFBC

Theorem 3. Under WUFBC, the schedule returned by DuOpt is optimal, i.e., $T(\{r_{1i}^\infty, r_{2i}^\infty\}) = T^{\text{opt}}$.

Proof. See Appendix.

6 SUMT Algorithm

Theorem 1 states that the solution of Problem 1 is unique under WUFBC. In order to generalize the uniqueness result, we shall introduce the following equivalent problem:

Problem 2. Energy Consumption Minimization of an Energy Harvesting Transmitter with Data and Energy Arriving at Arbitrary Points on a BC:

$$\text{Minimize : } E = \sum_{i=1}^{k^*} g(r_{1i}, r_{2i}) \xi_i + g(r_{1(k^*+1)}, r_{2(k^*+1)}) \times (T - \sum_{i=1}^{k^*} \xi_i)$$

$$\text{subject to : } \forall k \in \{1, \dots, k^*\}, k^* = \max\{i : \sum_{j=1}^i \xi_j \leq T\}$$

$$r_{1k} \geq 0, r_{1(k^*+1)} \geq 0, r_{2k} \geq 0, r_{2(k^*+1)} \geq 0$$

$$\sum_{i=1}^k g(r_{1i}, r_{2i}) \xi_i \leq E(t_k)$$

$$\begin{aligned}
 \sum_{i=1}^k r_{1i}\xi_i &\leq B^{(1)}(t_k), \quad \sum_{i=1}^k r_{2i}\xi_i \leq B^{(2)}(t_k) \\
 \sum_{i=1}^{k^*} r_{1i}\xi_i + r_{1(k^*+1)}(T - \sum_{i=1}^{k^*} \xi_i) &= B^{(1)}(W) \\
 \sum_{i=1}^{k^*} r_{2i}\xi_i + r_{2(k^*+1)}(T - \sum_{i=1}^{k^*} \xi_i) &= B^{(2)}(W)
 \end{aligned} \tag{6}$$

The objective function of Problem 2 is strictly convex as it is a weighted sum of $g(r_{1i}, r_{2i})$'s, $i \in \{1, 2, \dots, k^* + 1\}$ each of which are strictly convex and increasing [23]. The set of constraints (6) are also convex by the same reasoning, and the rest of the constraints are linear. Hence, Problem 2 is a strictly convex problem, with a unique solution.

The power function $P = g(r_1, r_2)$ can be rearranged to obtain an energy function $E = \frac{1}{T}g(\frac{B_1}{T}, \frac{B_2}{T})$, where E is the energy consumption, T is the transmission deadline and B_1 and B_2 are the total data to be transmitted. For constant B_1 and B_2 , this energy function turns out to be strictly decreasing with the transmission deadline in AWGN BC. Using this fact, it can be proved by induction that energy consumption of a multi-epoch optimal schedule is a strictly decreasing function of transmission deadline. Hence, there is a unique transmission completion time by which an optimal policy consumes a given amount of energy, and vice versa. Now, suppose that schedules S^1 and S^2 are optimal solutions of Problems 1 and 2, defined with the deadline T , respectively. Both schedules must consume the same amount of energy since the opposite claim would contradict the optimality of the schedules: S^2 cannot use more energy by T than S^1 , by definition. Suppose less energy is used, implying that S^1 has used some extra energy to deliver the same amount of data as S^2 . However, this extra energy could be used in the last epoch to decrease the completion time by a nonzero amount, which would contradict optimality. Therefore, S^1 and S^2 are two schedules completing the transmission of the same amount of data at the same time by consuming same amount of energy. Hence, the solution of Problem 1 having a transmission completion time of T^{opt} , is also a solution to Problem 2 with transmission deadline T^{opt} . Combining this result with the uniqueness of the solution of Problem 2, we conclude that the solution of Problem 1 is also unique [24]. Next, we consider the solution of Problem 2.

Gradient descent and Newton's method are two commonly used techniques for solving linear equality constrained convex problems. Problem 2 is strictly convex but has both linear equality and convex inequality constraints. Interior-point methods are utilized for solving inequality constrained convex problems [23]. Specifically for this problem, we use the *sequential unconstrained*

minimization technique (SUMT) which has been proposed by Fiacco and McCormick [25]. SUMT Algorithm converts a constrained convex problem into an unconstrained problem by adding penalty terms into objective function.

The main objective of the SUMT method is to minimize $F(\mathbf{r}) = E(\mathbf{r}) + \mu P(\mathbf{r})$, where $E(\mathbf{r})$ is the objective function of Problem 2, μ is the penalty parameter, and $P(\mathbf{r})$ is the penalty function. $P(\mathbf{r})$ forces the solution to satisfy all the constraints, and μ is used to control the effect of $P(\mathbf{r})$. Selecting a sufficiently large μ , we can reach the optimal solution with an arbitrarily small error. However, choosing a very large μ might also have some computational disadvantages, and choosing a very small μ might also lead to a slow convergence or early termination [26]. Hence, an arbitrary small penalty parameter, i.e., $\mu = 1$, is selected at the beginning of algorithm and doubled after each iteration of the SUMT Algorithm until a predefined maximum value, i.e., $\mu_{\max} = 10^{10}$, is reached. Starting from an initial rate vector, SUMT Algorithm updates the rate vector, \mathbf{r} , in each iteration by utilizing the generalized Newton's method and then increase the penalty parameter. The algorithm terminates if either μ reaches the maximum value or $P(\mathbf{r})$ is smaller than a predefined threshold ϵ_S . Algorithm 2 represents the pseudo code of the SUMT Algorithm.

Algorithm 2 SUMT Algorithm

- 1: $r_{1i} \leftarrow \frac{B^{(1)}(T)}{T}, r_{2i} \leftarrow \frac{B^{(2)}(T)}{T}, \mu \leftarrow 1$
 - 2: **repeat**
 - 3: $\mathbf{r} \leftarrow \text{NewtonsMethod}(\mathbf{r})$
 - 4: $\mu \leftarrow 2\mu$
 - 5: **until** $\mu \geq \mu_{\max} \vee P(\mathbf{r}) \leq \epsilon_S$
-

So far, we have seen that Problem 2, which minimizes the energy consumption given a transmission deadline, can be solved by SUMT Algorithm, and Problem 2 is equivalent of Problem 1 by means of solutions. As total energy consumption of an optimal schedule is a strictly decreasing function of total transmission time [24], we can find lower and upper bounds on the transmission completion time of Problem 1 using the SUMT Algorithm. Then, we iteratively narrow down the distance between the bounds and finally obtain the solution of Problem 1 within some error, i.e., $\epsilon > 0$. A pseudo code of this algorithm is given in Algorithm 3.

7 Convergence rate and examples

The main difference between the two algorithms described above for solving Problem 1 is the parameter update procedure. DuOpt acts like a block coordinate descent technique (while, strictly speaking, it is not) and

Algorithm 3 Time minimization with SUMT Algorithm

```

1:  $N \leftarrow \operatorname{argmin}_{k \in \{1, 2, \dots, k^{\text{up}}\}} (B^{(1)}(t_{k^{\text{up}}}) = B^{(1)}(t_k) \vee B^{(2)}(t_{k^{\text{up}}}) = B^{(2)}(t_k))$ 
2: repeat
3:    $N \leftarrow N + 1$ 
4:    $E_{\min} \leftarrow \text{SUMT\_Algorithm}(t_N)$ 
5: until  $E_{\min} \leq E(t_N)$ 
6:  $T^{\min} \leftarrow t_{N-1}$ ,  $T^{\max} \leftarrow t_N$ 
7: repeat
8:    $T \leftarrow (T^{\min} + T^{\max})/2$ 
9:    $E_{\min} \leftarrow \text{SUMT\_Algorithm}(T)$ 
10: if  $E_{\min} < E(T)$  then
11:    $T^{\max} \leftarrow T$ 
12: else
13:    $T^{\min} \leftarrow T$ 
14: end if
15: until  $T^{\max} - T^{\min} \leq \epsilon$ 
16:  $T_{\min} \leftarrow T$ 
    
```

optimizes only a portion of parameters (rates and powers in consecutive epochs) at a time; whereas SUMT updates all parameters at the same time.

In the SUMT Algorithm, after finding an upper and a lower bound on the transmission completion time,

using bisection method that the distance between the lower and the upper bound is halved in each iteration. The average number of iterations is at most $\lceil \log_2(\frac{\bar{\xi}}{\epsilon}) \rceil$, where $\bar{\xi}$ is the average epoch length. Hence, the number of iterations does not depend on the epoch count and has a complexity of $O(1)$. In each iteration, Newton's method, which scales well the problem size (number of epochs) [23], is performed. In every step of Newton's method a $2k \times 2k$ Hessian matrix is constructed and its inverse is calculated. Computational complexity of matrix inversion with Gaussian elimination is polynomial with $O(n^3)$, yet it can be as low as $O(n^2 \log(n))$ theoretically, which is the lower bound on matrix multiplication. Since the average iteration count for SUMT and the step count of Newton's method, does not change with epoch count, the complexity of SUMT Algorithm, which is determined by complexity of Newton's step, is polynomial. We have simulated DuOpt and SUMT Algorithms on randomly generated problem instances with different lengths and observed that simulation time for both algorithms increase quadratically as the number of epochs increases. The termination time of DuOpt Algorithm is almost two orders of magnitude shorter than the SUMT Algorithm, as the number of epochs is varied (see Figure 3).

We should also compare the memory requirements of algorithms. In each step of DuOpt, only two consecutive epochs are considered; therefore, memory requirement of

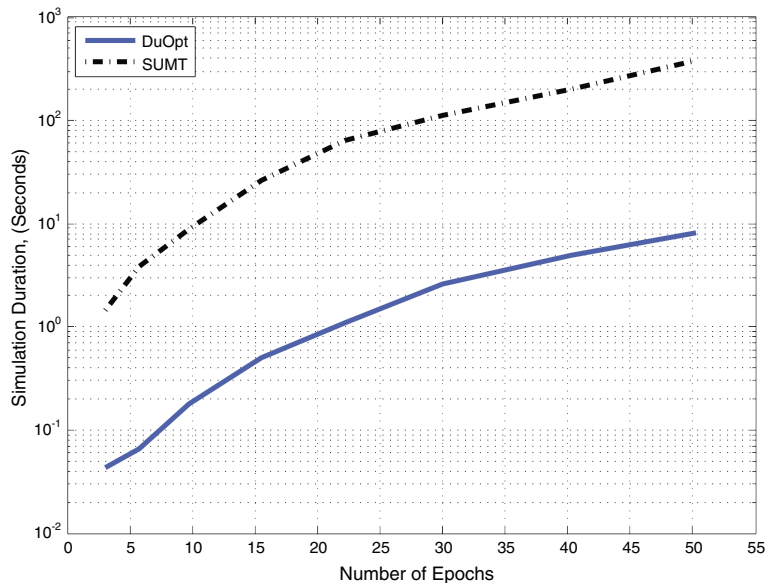


Figure 3 Comparison of DuOpt and SUMT Algorithms. Average computation time (log scale) versus number of epochs for the optimal solutions of Problem 1 with DuOpt and SUMT Algorithms: Solid blue line is the average computation time of the optimal schedule with DuOpt and black dashed line represents the computation time of SUMT Algorithm. A two-user AWGN BC with 1-KHz bandwidth and noise spectral density of $N_0 = 10^{-12}$ W/Hz is considered. Path loss factors on the links of the strong and the weak user are assumed to be $s_1 = 70$ and $s_2 = 75$ dB, respectively. There is a probability of 0.25 that no energy harvest occurs at the start of an epoch. With probability 0.75 energy harvest amounts (in W) are chosen from a Pareto distribution with parameters $b = 2$ and $\alpha = 2$. Similarly, there is a probability of 0.25 that no data arrival occurs; otherwise, data arrival amounts (in Kbits) are selected from a Pareto distribution with parameters $b = 4$, $\alpha = 2$ and then rounded to the nearest integer.

an iteration is constant across a varying number of epochs. On the other hand, SUMT constructs a Hessian matrix in each iteration and the size of this matrix increases quadratically with increasing number of epochs. The only drawback of DuOpt seems to be the termination process. In SUMT Algorithm, the ultimate distance of the optimal schedule is known at the beginning of each iteration, and this distance is halved at each iteration. However, transmission time reduction of DuOpt Algorithm can be erratic and may induce an early termination. We have observed that controlling total power levels before termination is very useful in DuOpt Algorithm. If the total power level decreases greater than a predefined threshold, i.e., an ϵ of average power level, then we should not let the algorithm terminate.

Next, we will illustrate and discuss some properties of the optimal schedule on two numerical examples. In both examples, we consider a two-user AWGN BC with 1-Khz bandwidth with noise spectral density $N_0 = 10^{-12}$ W/Hz. Path losses from the sender to the strong and the weak user are assumed to be 70 and 75 dB, respectively. First, we will consider a WUFBC such that 25 Kbits of data to be transmitted to the weak user is ready at the beginning of the schedule and 8, 25, 12, 20, and 15 Kbits of data to be transmitted to the strong user arrives at

respective times 0, 2, 4, 8, and 10 s. Energy harvesting profile is assumed to be 3, 10, 4, 7, 13, 3, 5, 8, 6, and 12 J at respective times 0, 3, 5, 8, 9, 10, 11, 13, 15, and 17 s. The main objective in this example is to transmit the given data arrivals as soon as possible. Under these circumstances, we run both DuOpt and SUMT Algorithms on a 3-GHz Core2Duo processor and obtain the optimal schedule (See Figure 4) in 0.3 and 8.7 s, respectively. The optimal schedule finalizes the transmission at $t = 12.903$ s with a total power allocation vector 1.000, 1.000, 2.800, and 5.710 W and the strong user power allocation vector of 0.150, 0.708, 0.708, and 1.399 W with durations 2, 1, 5, and 4.903 s.

The following example illustrates a general scenario, i.e., transmitter has energy harvest and data arrivals for both users after the beginning of the schedule: 15, 12, and 8 Kbits of data to be transmitted to the strong user arrives at times 0, 5, and 8 s; and 2, 6, and 12 Kbits of the weak user data arrives at 0, 2, and 5 s. Energy harvesting profile is assumed to be 1, 2, 1, 2, 2, 1, 2 J at times 0, 2, 5, 7, 8, 10, and 12 s. The main objective in this example is to transmit the given data arrivals as soon as possible. Under these circumstances, we run both DuOpt and SUMT Algorithms on a 3-GHz Core2Duo processor and obtain the optimal schedule (See Figure 5) in

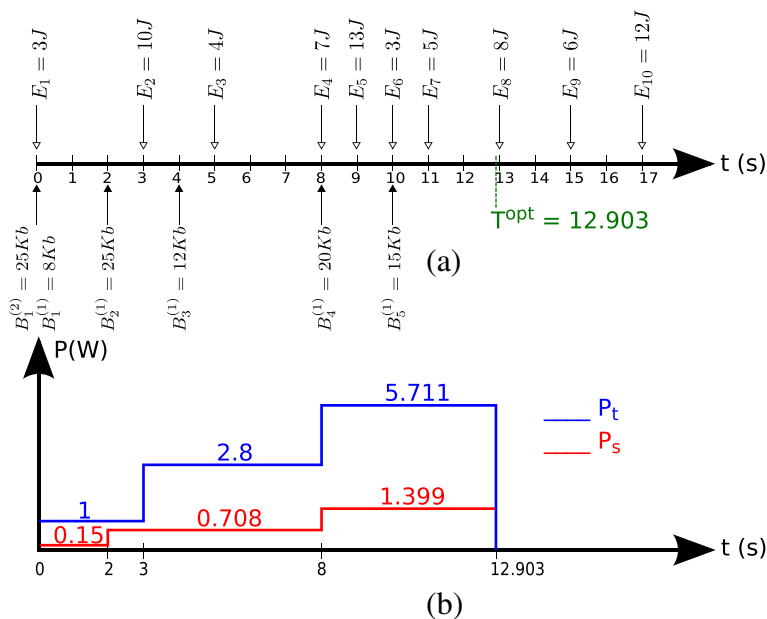


Figure 4 Numerical Example 1. A numerical example of optimal schedule for Problem 1. Top figure (a) illustrates the energy harvest and bit arrival sequences and the transmission completion time: 3, 10, 4, 7, 13, 3, 5, 8, 6, and 12 J of energy is harvested at respective times 0, 3, 5, 8, 9, 10, 11, 13, 15, and 17 s; and 8, 25, 12, 20, and 15 Kbits of the strong user data arrive at respective times 0, 2, 4, 8, and 10 s whereas 25 Kbits of the weak user data is ready at the beginning. Optimal schedule finalized at $t = 12.903$ s by transmitting 80 and 25 Kbits to the the strong and the weak user, respectively. Bottom figure (b) depicts the strong user power level and the total power level during transmission. Blue line represents the total transmitter power level, P_t , whereas the red line represents the power reserved for transmission to the strong user, P_s , during transmission. Total power levels and the strong user power levels are 1.000, 1.000, 2.800, and 5.710 W and 0.150, 0.708, 0.708, and 1.399 W with durations 2, 1, 5, and 4.903 s.

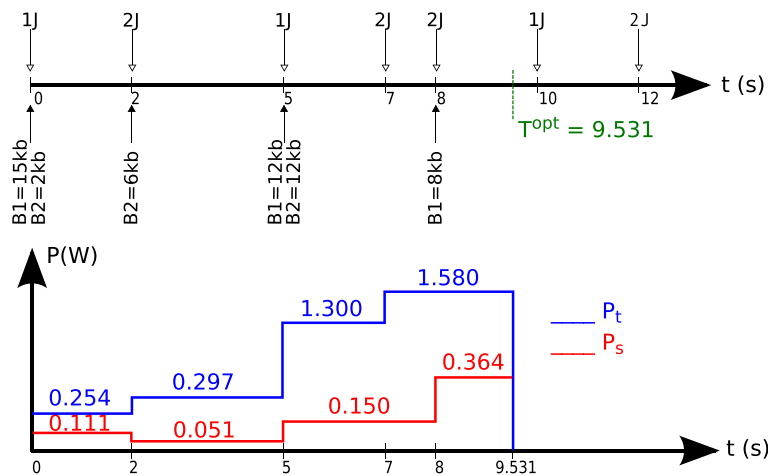


Figure 5 Numerical Example 2. A numerical example of optimal schedule for Problem 1. Top figure illustrates the energy harvest and bit arrival sequences and the transmission completion time: 1, 2, 1, 2, 2, 1, and 2 J of energy is harvested at times 0, 2, 5, 7, 8, 10, and 12 and 15, 12, and 8 Kbits of the strong user data arrives at 0, 5, and 8 s whereas 2, 6, and 12 Kbits of the weak user data arrives at 0, 2, and 5 s. Optimal schedule finalize at $t = 9.531$ s by transmitting 35 and 20 Kbits to the strong and the weak users, respectively. Bottom figure depicts the strong user power level and the total power level during transmission. Blue line represents the total transmitter power level, P_t , whereas the red line represents the power reserved for transmission to the strong user, P_s , during transmission. Total power levels and the strong user power levels are 0.254, 0.297, 1.300, 1.580, and 1.580 and 0.111, 0.051, 0.150, 0.150, and 0.364 W with durations 2, 3, 2, 1, and 1.531 s, respectively.

0.5 and 4.3 s, respectively. The optimal schedule finalize the transmission at $t = 9.531$ s with a total power allocation vector 0.254, 0.297, 1.300, 1.580, and 1.580 W and the strong user power allocation vector of 0.111, 0.051, 0.150, 0.150, and 0.364 W with durations 2, 3, 2, 1, and 1.531 s.

As stated in Lemma 2, the total power is non-decreasing in both examples, and it increases only if at least one of the conditions in Lemma 3 occurs, i.e., in the first example energy constraint is active at $t = 3$ and both energy and the strong user data constraints are active at $t = 8$, whereas in the second example the weak user data constraint is active at $t = 2$, both the weak and the strong user data constraint is active at $t = 5$ and finally energy constraint is active at $t = 7$. We should also note that, as stated in Lemma 4, the strong user power is non-decreasing in the first example, yet it increases at $t = 2$ due to the strong user data constraint and at $t = 8$ s due to both energy and the strong user data constraints. However, in the second example, both the weak and the strong user powers decrease and increase during transmission due to packet arrivals. The weak user power in the second example decrease at $t = 8$ in order to preserve the total transmit power at a constant level. On the other hand, the strong user power decreases at $t = 2$ while the total power level increases. In fact, when only the weak user constraint is active, at $t = 2$ in the second example, it is optimal to preserve the sum of rates (not the sum of powers) at a constant level [24].

8 Conclusions

With the new advances in energy harvesting technologies, optimization of communication systems that depend on renewable energy resources has emerged as an important problem. A large body of recent research effort in the field has focused on transmission scheduling policies.

This paper continued the work on to the solution of the previous formulation of the problem stated in [12] and [13]. In particular, it aimed to find the power and rate allocation policy in a broadcast channel with two users, that minimizes the total transmission completion time of data that becomes available at arbitrary points in time, with energy that is harvested at arbitrary points in time. It should be noted that this line of work relies on offline problem formulations where exact knowledge of data and energy arrival events is assumed. The approach is useful for obtaining structures and guidelines as well as benchmarks for online problems which may be more applicable in practical transmission scenarios.

In this study, structural properties about the solution of the general optimal offline broadcast packet scheduling policy for an energy harvesting broadcast channel have been established. The uniqueness of the optimal policy has been shown. An iterative algorithm, DuOpt, that returns a feasible schedule is devised and the algorithm is shown to possess the same structural properties with the optimal. Moreover, DuOpt was shown to obtain the unique optimal policy under the WUFBC. By devising a convex problem that is shown to be equivalent to

the general problem, an alternative iterative algorithm that utilizes the SUMT is proposed. SUMT converges to the optimal transmission completion time without the WUFB condition assumption. However, it is observed on numerical examples that DuOpt iterations converges on average two orders of magnitude faster and DuOpt returns the same schedule as the one found by SUMT. This makes us believe that DuOpt converges to the optimal solution even in lieu of the WUFB condition, and it achieves this much faster than standard optimization methods.

Endnotes

^a The achievable rate regions will be implicitly assumed to correspond to a certain constant tolerable error probability respecting which it is possible to transmit a finite number of bits with a finite amount of energy per bit.

^b Lemma 6 of [12] originates from the observation in [13] (See Lemma 4 and Corollary 1 in) that there is a cut-off level for the total power, below which no power is assigned to the weak user.

^c The term “energy buffer” refers to an abstraction of an energy storage unit such as a rechargeable battery. A realistic battery would have imperfections: not all energy harvested can be retrieved, and there is a finite capacity for storage. On the other hand, the energy buffer abstraction can store an arbitrary amount E_i at time t_i , all of which can be used after t_i . This abstraction could model the *net* energy, taking into account any leakage or other imperfections, that becomes *available* at t_i after the battery is charged between t_{i-1} and t_i .

^d A gap is a time period with zero power allocation.

Appendix

Proof of Lemma 2

The claim is that power is non-decreasing with epoch number i . Equivalently, power is non-decreasing in time. To show this, we will argue that given a schedule in which power decreases at some time t_i , this schedule can only be improved by equating the power levels before and after t_i . Consider a time interval (τ_1, τ_2) , so that power is constant at $P_1 > 0$ during (τ_1, t_i) , and at $P_2 < P_1$ during (t_i, τ_2) . As illustrated in Figure 6, let $t = \tau_2 - \tau_1$, and the lengths of the constant-power slots be βt and $(1 - \beta)t$, where $0 < \beta < 1$ in order to have two epochs exactly. Denote the rate pairs in the first and second slots as (r_{11}, r_{21}) and (r_{12}, r_{22}) , respectively.

We will show that keeping the total consumed energy constant, and transferring some amount of energy ΔE from the first slot to the second such that power levels are reallocated closer to each other, the sender can transmit at least the same number of bits within the same duration. Let us denote the average rate of the weak user as $\bar{r}_2 \triangleq$

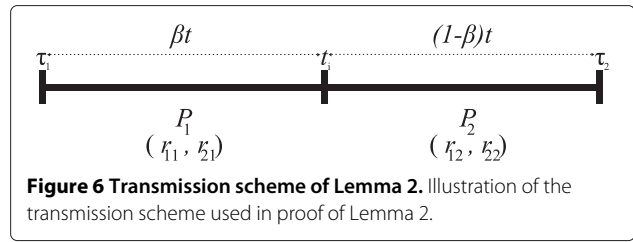


Figure 6 Transmission scheme of Lemma 2. Illustration of the transmission scheme used in proof of Lemma 2.

$\beta r_{21} + (1 - \beta)r_{22}$. Provided $r_{21} > 0$, the sender could transfer some energy and some of user 2's bits from the first epoch to the second while keeping user 1's rates r_{11} and r_{12} constant. As energy and bits are simply being deferred for later use, this operation does not violate feasibility. Specifically, let

$$P'_1 = P_1 - (1 - \beta)\Delta P, P'_2 = P_2 + \beta\Delta P, \quad (7)$$

such that the new power allocation to the slots (P'_1, P'_2) is satisfying $P_2 < P'_2 \leq P'_1 < P_1$. With this new allocation, the weak user's rate in the first slot is $h_2(P'_1, r_{11}) < h_2(P_1, r_{11}) = r_{21} > 0$. Its new average rate over the duration of t is:

$$\begin{aligned} \bar{r}'_2 &= h_2(P'_1, r_{11})\beta + h_2(P'_2, r_{12})(1 - \beta) \\ &\geq h_2(P_1, r_{11})\beta + h_2(P_2, r_{12})(1 - \beta) \\ &= \bar{r}_2. \end{aligned} \quad (8)$$

This is shown by straightforward application of the properties listed in section 2.

Equation 8 follows from the fact that

$$\begin{aligned} &h_2(P'_1, r_{11})\beta + h_2(P'_2, r_{12})(1 - \beta) \\ &- h_2(P_1, r_{11})\beta - h_2(P_2, r_{12})(1 - \beta) \geq 0 \end{aligned} \quad (9)$$

for all $\beta \in (0, 1)$ with equality achieved at $\beta = 0, 1$.

$$\begin{aligned} f(\beta) &= h_2(P_1 - (1 - \beta)\Delta P, r_{11})\beta \\ &+ h_2(P_2 + \beta\Delta P, r_{12})(1 - \beta) \\ &- h_2(P_1, r_{11})\beta - h_2(P_2, r_{12})(1 - \beta) \end{aligned} \quad (10)$$

The concavity of $f(\beta)$ in β combined with the fact that $f(\beta)$ is zero for β values $\{0, 1\}$ implies that Equation 9 holds.

The first- and second-order derivatives of f with respect to β are the following^e

$$\begin{aligned} \frac{\partial f}{\partial \beta} &= h_2(P_1 - (1 - \beta)\Delta P, r_{11}) - h_2(P_2 + \beta\Delta P, r_{12}) \\ &+ \beta \{h_{2x}(P_1 - (1 - \beta)\Delta P, r_{11})(\Delta P)\} \\ &+ (1 - \beta) \{h_{2x}(P_2 + \beta\Delta P, r_{12})(\Delta P)\} \\ &- h_2(P_1, r_{11}) + h_2(P_2, r_{12}) \end{aligned} \quad (11)$$

$$\begin{aligned}
 \frac{\partial^2 f}{\partial \beta^2} &= 2 \underbrace{(h_{2x}(P_1 - (1 - \beta)\Delta P, r_{11})(\Delta P) - h_{2x}(P_2 + \beta\Delta P, r_{12})(\Delta P))}_{\leq 0} \\
 &+ \beta \left\{ \underbrace{h_{2xx}(P_1 - (1 - \beta)\Delta P, r_{11})(\Delta P)^2}_{\leq 0} \right\} \\
 &+ (1 - \beta) \left\{ \underbrace{h_{2xx}(P_2 + \beta\Delta P, r_{12})(\Delta P)^2}_{\leq 0} \right\} \\
 &\leq 0
 \end{aligned} \tag{12}$$

In the remaining case which is $r_{21} = 0$, we know that $r_{11} > 0$ must hold (as $P_1 > 0$). In this case, the allocation can similarly be improved by bringing power levels closer and transferring some of the first user's bits to the right, while keeping the rate allocation of the weak user unchanged. Let $\bar{r}_1 \triangleq \beta r_{11} + (1 - \beta)r_{12}$ be the average rate of the strong user over the duration t . After the reallocation, the average rate of the strong user becomes

$$\begin{aligned}
 \bar{r}_1 &= h_1(P'_1, r_{21})\beta + h_1(P'_2, r_{22})(1 - \beta) \\
 &\geq h_1(P_1, r_{21})\beta + h_1(P_2, r_{22})(1 - \beta) \\
 &= \bar{r}_1.
 \end{aligned} \tag{13}$$

Equation 13 follows from the fact that

$$\begin{aligned}
 &h_1(P'_1, r_{21})\beta + h_1(P'_2, r_{22})(1 - \beta) - h_1(P_1, r_{21}) \\
 &\times \beta - h_1(P_2, r_{22})(1 - \beta) \geq 0
 \end{aligned} \tag{14}$$

for all $\beta \in \{0, 1\}$ with equality achieved at $\beta = 0, 1$.

$$\begin{aligned}
 q(\beta) &= h_1(P_1 - (1 - \beta)\Delta P, r_{21})\beta - h_1(P_2, r_{22})(1 - \beta) \\
 &+ h_1(P_2 + \beta\Delta P, r_{22})(1 - \beta) - h_1(P_1, r_{21})\beta.
 \end{aligned}$$

We can show that Equation 14 holds by proving $q(\beta)$ is concave in β .

The first- and second-order derivatives of q with respect to β are the following^f:

$$\begin{aligned}
 \frac{\partial q}{\partial \beta} &= h_1(P_1 - (1 - \beta)\Delta P, r_{21}) - h_1(P_2 + \beta\Delta P, r_{22}) \\
 &+ \beta \{h_{1x}(P_1 - (1 - \beta)\Delta P, r_{21})(\Delta P)\} - h_1(P_1, r_{21}) \\
 &+ (1 - \beta) \{h_{1x}(P_2 + \beta\Delta P, r_{22})(\Delta P)\} + h_1(P_2, r_{22})
 \end{aligned}$$

$$\begin{aligned}
 \frac{\partial^2 q}{\partial \beta^2} &= 2 \underbrace{(h_{1x}(P_1 - (1 - \beta)\Delta P, r_{21})(\Delta P) - h_{1x}(P_2 + \beta\Delta P, r_{22})(\Delta P))}_{\leq 0} \\
 &+ \beta \left\{ \underbrace{h_{1xx}(P_1 - (1 - \beta)\Delta P, r_{21})(\Delta P)^2}_{\leq 0} \right\} \\
 &+ (1 - \beta) \left\{ \underbrace{h_{1xx}(P_2 + \beta\Delta P, r_{22})(\Delta P)^2}_{\leq 0} \right\} \\
 &\leq 0
 \end{aligned} \tag{15}$$

According to the properties listed in section 2, Equation 15 always holds if $r_{21} \geq r_{22}$. Hence q is concave in β , if $r_{21} = 0$. We conclude that a policy that contains a drop in power level is sub-optimal. ■

Proof of Lemma 3

To reach contradiction, suppose that power increases at time t_i ($P_{i+1} > P_i$). We will show that if none of the conditions (a), (b), or (c) hold, then it is possible to improve the schedule by transferring some energy from the $(i + 1)$ th epoch to the i th. Assuming the i th epoch length is βt and the $(i + 1)$ th is $(1 - \beta)t$, after bringing power levels closer,

$$P'_i = P_i + \beta\Delta P, P'_{i+1} = P_{i+1} - (1 - \beta)\Delta P. \tag{16}$$

Observe that if we treat P_i as P_2 and P_{i+1} as P_1 , then Equation 16 becomes identical with Equation 7. This implies that at least the same number of bits could be transmitted to the weak user, if we can bring power levels closer while keeping the strong user's rates constant. In addition to this, the allocation could also be improved by bringing power levels while keeping the weak user's rates constant in case $r_{2(i)} \geq r_{2(i+1)}$. Consequently, Equations 8 and 13 hold.

It is straightforward that we cannot bring power levels any closer when condition (a) holds, due to the energy causality constraint. Secondly, it also doesn't yield a better schedule, if we can not transfer data from the latter epoch to the former (condition (b)). When it is possible to transfer some positive amount of energy from the $(i + 1)$ th epoch to the i th, as shown in Equation 8, we can always improve allocation while keeping the rates of the strong user the same. Although bringing power levels closer while keeping the rates of the strong user the same is not feasible in case weak user's data constraint is active, we may still improve allocation as proved in Equation 13. Nevertheless, this time we require $r_{2(i)} \geq r_{2(i+1)}$. As rate of the weak user can only rise upon a data arrival for the weak user in case the weak user's data constraint is active, condition (c) describes the last case that we may not improve allocation by bringing power levels closer.

We have thus shown that this set of three conditions contains all the cases in which power can rise, if none of these hold, then power cannot rise. It is straightforward to show that this set cannot be further reduced by finding counterexamples for the claim that if any one of (a), (b), or (c) is satisfied, then power may rise at time t_i . ■

Proof of Theorem 1

Suppose that there are two distinct optimal schedules, S^A and S^B , which have equal power and rate assignments until t_{s-1} and differ for the first time at epoch s . Consider that the corresponding power allocation vectors, \mathbf{P}^A and \mathbf{P}^B , also differ at epoch s such that $P_i^A = P_i^B, \forall i \in \{1, 2, \dots, s-1\}$ and $P_s^A < P_s^B$. First, assume that \mathbf{P}^A remains constant after epoch s , i.e., $P_i^A = P_s^A, \forall i > s$. By definition, both schedules end at T^{opt} . The total energy consumption of S^A would be less than that of S^B by T^{opt} , i.e., $(\sum_{i=1}^{k^*} P_i^A \xi_i + (T^{\text{opt}} - t_{k^*+1})P_{k^*+1}^A) < (\sum_{i=1}^{k^*} P_i^B \xi_i + (T^{\text{opt}} - t_{k^*+1})P_{k^*+1}^B)$, which contradicts Lemma 5. Hence, the total transmit power of S^A cannot remain constant after t_s . Since total transmit power is non-decreasing (See Lemma 2), it should increase after epoch s and before the end of transmission, i.e., $P_u^A < P_{u+1}^A, \exists u \in \{s, s+1, \dots, k^*\}$. Since there are no data arrivals for the weak user, the increase in total transmit power is either due to energy constraint being met or due to *all* the packets arrived by the time t_u having been transmitted (cf. conditions (a) or (c) in Lemma 3). As $\sum_{i=s}^u P_i^A \xi_i < \sum_{i=s}^u P_i^B \xi_i$, S^A has not consumed all the available energy at the end of epoch u . Hence, S^A must have transmitted all the bits arrived until t_u , which means that S^A has transmitted at least the same number of bits to both users while consuming less energy than S^B between t_0 and t_u , which contradicts the optimality of S^B . Therefore, if there are two distinct optimal schedules, S^A and S^B , their power allocation vectors cannot be different, i.e., $\mathbf{P}^A = \mathbf{P}^B$.

Now, consider two rate pair vectors, \mathbf{R}^A and \mathbf{R}^B , where $(r_{1i}^A, r_{2i}^A) = (r_{1i}^B, r_{2i}^B), \forall i \in \{1, 2, \dots, s-1\}$ and $r_{1s}^A < r_{1s}^B$. Let the strong user's rate in $S^A, \{r_{1j}^A\}$ stay constant after t_{s-1} . By

Lemma 4, rate of the strong user cannot decrease; hence, the rate of the strong user in S^B would be larger than that of S^A after epoch s , i.e., $r_{1(j+1)}^A = r_{1s}^A < r_{1s}^B \leq r_{1j}^B, \forall j \in \{s, s+1, \dots, k-1\}$. Since both schedules end transmission at the same time, S^A transmits fewer bits to the strong user than S^B does, which contradicts the fact that optimal schedule transmits all the packet arrivals by the end of transmission. Therefore, the rate of the strong user in S^A cannot stay constant after epoch s . Now, suppose that rate of the strong user in S^A increases at the end of epoch u , i.e., $r_{1u}^A < r_{1(u+1)}^A, \exists u \in \{s, s+1, \dots, k^*\}$. This increase cannot be due to (1) in Lemma 4 because S^B has transmitted more bits to the strong user by t_u , i.e., $\sum_{i=1}^u r_{1i}^A \xi_i < \sum_{i=1}^u r_{1i}^B \xi_i$. Moreover, this increase cannot be due to (2) in Lemma 4 since rate of the weak user in S^A is greater than zero in epoch u , i.e., $r_{2u}^A = h_2(P_u, r_{1u}^A) > h_2(P_u, r_{1u}^B) \geq 0$. Hence rate of strong user in S^A cannot increase after epoch s . Finally, rate of the strong user in S^A cannot *decrease* (See Lemma 4) as this would also contradict optimality. Hence, there cannot be two optimal schedules with different rate pair vectors.

As both the power allocation vector and the rate pair vector of an optimal schedule are unique, we conclude that the optimal schedule is *unique* under WUFBC. ■

Details of the local optimization

Consider the local optimization problem given in Figure 7, where B_{ij} is the data arrival for the i th user and E_j is the energy harvest at the beginning of j th epoch for $i, j \in \{1, 2\}$. T_1 is the length of the first epoch and T_2 is the transmit duration in the second epoch. Let P_{ij} and r_{ij} be the power and rate assigned to the i th user during j th epoch after optimization. The energy and data causality constraints for the local optimization problem are as follows:

$$E_1 \geq P_{11} \cdot T_1$$

$$B_{11} \geq r_{11} \cdot T_1$$

$$B_{21} \geq r_{21} \cdot T_1.$$

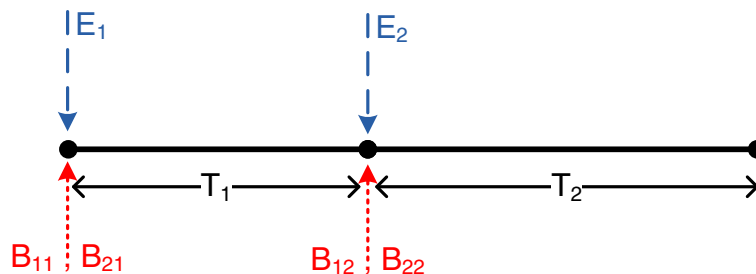


Figure 7 Two-epoch problem. Illustration of the local optimization problem with two epochs. B_{ij} represents the data arrival for the j th user at the beginning of the i th epoch. Similarly, E_j represents the energy harvest at the beginning of the j th epoch.

The structure of the solution changes if either one of the constraints satisfied with equality. Since there are three different constraints, after optimization, one will encounter one of the $2^3 = 8$ results. We have studied all eight cases and derived solutions to each one of them for both energy minimization and time minimization functions. In each case, the solution can be calculated analytically for energy minimization or it can be found iteratively for time minimization functions. Before starting the optimization, if one already knows which constraints should be satisfied with equality, then the result could be obtained solving just that case. Otherwise, one should compute the results for each case and then select the best one⁸ that respects energy and bit causalities. Next, we will analyze one of the cases and present the algorithms for both energy and time minimization functions. The analyses of the remaining cases can be done in a similar fashion [24].

Local optimization when only strong user data constraint is active

One of the possible structures of the local optimal solution is when only the strong user data constraint is active. In this case, total transmit power level in local optimal solution should be constant (See Lemma 3). Under this condition, a possible illustration of optimal power allocation is depicted in Figure 8. Since the total power should stay constant during transmission, we should have $P_{11} + P_{21} = P_{12} + P_{22}$. Moreover, we should have $P_{11} \leq P_{12}$ since only the strong user bit causality is met.

For a given transmission completion time $T_1 + T_2$, the optimum schedule that minimize energy consumption can be found as follows:

$$P_{11} = \frac{\sigma^2}{s_1} \left(2^{\frac{2B_{11}}{T_1}} - 1 \right), \quad P_{12} = \frac{\sigma^2}{s_1} \left(2^{\frac{2B_{12}}{T_2}} - 1 \right) \quad (17)$$

$$P_{22} = P_{11} + P_{21} - P_{12} \quad (18)$$

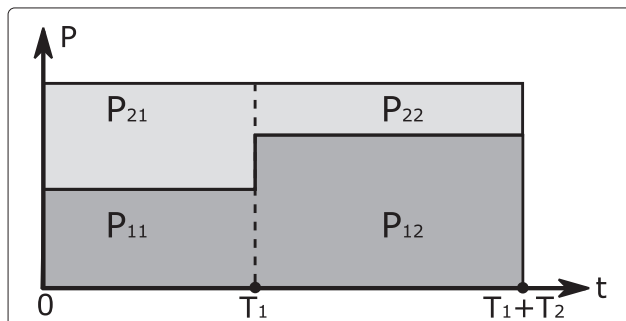


Figure 8 A solution for the two-epoch problem. Optimal power allocation for the strong and the weak users if only the strong user data causality constraint is active. Dark-shaded levels represent the strong user's power levels, whereas the light-shaded ones represent the weak user's power levels. The total transmit power stays constant.

$$B_2 = B_{21} + B_{22} = \frac{T_1}{2} \log_2 \left(1 + \frac{P_{21}s_2}{P_{11}s_2 + \sigma^2} \right) + \frac{T_2}{2} \log_2 \left(1 + \frac{P_{22}s_2}{P_{12}s_2 + \sigma^2} \right). \quad (19)$$

From Equations 18 and 19, we derive

$$P_{21} = \frac{1}{s_2} \left[2^{2B_2(P_{11}s_2 + \sigma^2)T_1(P_{12}s_2 + \sigma^2)T_2} \right]^{\frac{1}{T_1+T_2}} - \left(P_{11} + \frac{\sigma^2}{s_2} \right). \quad (20)$$

Then, total energy consumed in two epochs is calculated by

$$E_{min} = (P_{11} + P_{21})(T_1 + T_2). \quad (21)$$

Minimum energy to transmit $B_1 = B_{11} + B_{12}$ and $B_2 = B_{21} + B_{22}$ bits to the users in two epochs, E_{min} , can be calculated by setting T_2 as the length of the second epoch in Equation 21.

Substituting Equations 17 and 18 into Equation 19 and arranging terms, we obtain

$$B_2 = \frac{T_1}{2} \log_2 \left(\frac{\frac{E_1+E_2}{T_1+T_2} + \frac{\sigma^2}{s_2}}{\frac{\sigma^2}{s_1} \left(2^{\frac{2B_{11}}{T_1}} - 1 \right) + \frac{\sigma^2}{s_2}} \right) + \frac{T_2}{2} \log_2 \left(\frac{\frac{E_1+E_2}{T_1+T_2} + \frac{\sigma^2}{s_2}}{\frac{\sigma^2}{s_1} \left(2^{\frac{2B_{12}}{T_2}} - 1 \right) + \frac{\sigma^2}{s_2}} \right) = \frac{T_1 + T_2}{2} \log_2 \left(\frac{E_1 + E_2}{T_1 + T_2} + \frac{\sigma^2}{s_2} \right) - \frac{T_1}{2} \log_2 \left(\frac{\sigma^2}{s_1} \left(2^{\frac{2B_{11}}{T_1}} - 1 \right) + \frac{\sigma^2}{s_2} \right) - \frac{T_2}{2} \log_2 \left(\frac{\sigma^2}{s_1} \left(2^{\frac{2B_{12}}{T_2}} - 1 \right) + \frac{\sigma^2}{s_2} \right)$$

The first and second order derivatives of B_2 with respect to T_2 are as follows:

$$\frac{\partial B_2}{\partial T_2} = \frac{1}{2} \log_2 \left(\frac{E_1 + E_2}{T_1 + T_2} + \frac{\sigma^2}{s_2} \right) - \frac{1}{2 \ln(2) \left(1 + \frac{\sigma^2}{s_2} \frac{T_1+T_2}{E_1+E_2} \right)} - \frac{1}{2} \log_2 \left(\frac{\sigma^2}{s_1} 2^{\frac{2B_{12}}{T_2}} + \sigma^2 \frac{s_1 - s_2}{s_1 s_2} \right) + \frac{B_{12}}{T_2} \frac{1}{1 + \frac{s_1 - s_2}{s_2} 2^{-\frac{2B_{12}}{T_2}}}$$

$$\frac{\partial^2 B_2}{\partial T_2^2} = - \frac{1}{2 \ln(2)(T_1 + T_2) \left(1 + \frac{\sigma^2 T_1 + T_2}{s_2 E_1 + E_2}\right)^2} - \frac{2 \ln(2) B_{12}^2 \frac{s_1 - s_2}{s_2} 2^{-\frac{2B_{12}}{T_2}}}{T_2^3 \left(1 + \frac{s_1 - s_2}{s_2} 2^{-\frac{2B_{12}}{T_2}}\right)^2} < 0 \quad (22)$$

As shown in Equation 22, second derivative of B_2 is always negative for $s_1 > s_2$, which implies that B_2 is a strictly concave function of T_2 . As T_2 goes to infinity, B_2 is as follows:

$$\lim_{T_2 \rightarrow \infty} B_2 = - \frac{T_1}{2} \log_2 \left(\frac{s_2}{s_1} (2^{\frac{2B_{11}}{T_1}} - 1) + 1 \right) - \frac{s_2 B_{12}}{s_1} + \frac{(E_1 + E_2)s_2}{2 \ln(2)\sigma^2} > -\infty$$

Since B_2 is a strictly concave function of T_2 and goes to a finite number as T_2 goes to infinity, B_2 is an increasing strictly concave function of T_2 and there is a unique B_2 for each value of T_2 .

In time minimization, we have $E_1, E_2, T_1, B_2 = B_{21} + B_{22}, B_{21}, B_{22}; s_1$ and s_2 are constant terms and T_2 is transmission time within the second epoch which is to be minimized. Since B_2 is an increasing concave function of T_2 , we iteratively find T_2 that sends exactly B_2 bits to weak user by using bisection method. Minimum time to transmit $B_1 = B_{11} + B_{12}$ and B_2 bits to the users in these two epochs can be calculated by $T_{\min}^{(2)} = T_1 + T_2$. Algorithm 4 presents a pseudo-code of time minimization algorithm for the case that only the strong user bit causality event occurs.

Algorithm 4 Local time minimization algorithm when only the strong user data constraint is active

- 1: $T_{\max} \leftarrow T_2, T_{\min} \leftarrow 0$
 - 2: **repeat**
 - 3: $\hat{T} \leftarrow (T_{\max} + T_{\min})/2$
 - 4: $\hat{B}_2 \leftarrow \frac{T_1}{2} \log_2 \left(\frac{\frac{E_1 + E_2}{T_1 + \hat{T}} + \frac{\sigma^2}{s_2}}{\frac{\sigma^2}{s_1} (2^{\frac{2B_{11}}{T_1}} - 1) + \frac{\sigma^2}{s_2}} \right) + \frac{\hat{T}}{2} \log_2 \left(\frac{\frac{E_1 + E_2}{T_1 + \hat{T}} + \frac{\sigma^2}{s_2}}{\frac{\sigma^2}{s_1} (2^{\frac{2B_{12}}{\hat{T}}} - 1) + \frac{\sigma^2}{s_2}} \right)$
 - 5: **if** $\hat{B}_2 < B_2$ **then**
 - 6: $T_{\min} \leftarrow \hat{T}$
 - 7: **else**
 - 8: $T_{\max} \leftarrow \hat{T}$
 - 9: **end if**
 - 10: **until** $\hat{B}_2 == B_2$
 - 11: $T_{\min}^{(2)} \leftarrow \hat{T} + T_1$
-

Proof of Theorem 3

Suppose that DuOpt stopped and returned a schedule $\{r_{1i}^\infty, r_{2i}^\infty\} \triangleq S^{Du}$, with completion time $T(\{r_{1i}^\infty, r_{2i}^\infty\}) \triangleq T^{Du}$. Let S^{opt} be the unique optimal schedule with transmission completion time T^{opt} . We will now prove that $S^{Du} = S^{\text{opt}}$. Let us suppose $S^{Du} \neq S^{\text{opt}}$, then these schedules have to differ in either the power allocation or rate allocation (or both). First, suppose $P_i^{Du} = P_i^{\text{opt}}, i \in \{1, 2, \dots, s-1\}$ and $P_s^{Du} \neq P_s^{\text{opt}}$ for some s . We will show that this case is impossible. There are two possible cases for epoch s : (i) $P_s^{Du} > P_s^{\text{opt}}$ and (ii) $P_s^{Du} < P_s^{\text{opt}}$. Let us begin with the first case.

- (i) We assumed $P_s^{Du} > P_s^{\text{opt}}$. If P^{opt} stays constant after epoch s till the end of transmission, this would mean that S^{Du} consumes more energy than S^{opt} until T^{opt} , which would contradict the fact that optimal schedule consumes all the harvested energy till the end of transmission. Therefore, the power of the optimal schedule must increase at the end of epoch $(s+m)$ for some $m \geq 0$ before the end of transmission. As S^{Du} has been able to use more energy than the optimal schedule until t_{s+m} , the optimal schedule cannot have run into an energy constraint at t_{s+m} , hence the rise in the power can only be due to a data constraint at t_{s+m} , i.e., all the bits arrived have been transmitted by the optimal schedule until t_{s+m} . In order to contradict the assumption that $P_s^{Du} > P_s^{\text{opt}}$, we will now analyze the rate assignments for both schedules. First, let us focus on the case that both schedules use exactly the same rates for the strong user up to t_{s+m} , i.e., $r_{1i}^{Du} = r_{1i}^{\text{opt}}, \forall i \in \{s, \dots, s+m\}$. As we have shown above, S^{opt} should have transmitted all the bits available until t_{s+m} . However, if we compare the weak user bits transmitted by both schedules until t_{s+m} , we observe that S^{Du} transmits more bits to the weak user than S^{opt} does, because $\sum_{i=1}^{s+m} (r_{2i}^{Du} - r_{2i}^{\text{opt}}) \xi_i = \sum_{i=s}^{s+m} (r_{2i}^{Du} - r_{2i}^{\text{opt}}) \xi_i = \sum_{i=s}^{s+m} (h_2(P_i^{Du}, r_{1i}^{Du}) - h_2(P_i^{\text{opt}}, r_{1i}^{\text{opt}})) \xi_i > 0$. On the other hand, DuOpt respects *bit causality*, i.e., DuOpt does not transmit bits that have not arrived yet, so we reach contradiction. That is, rates cannot stay constant up to t_{s+m} , i.e., there is some $k \in \{1, 2, \dots, s+m-1\}$ such that $r_{1i}^{Du} = r_{1i}^{\text{opt}}$ for $i < k$ and $r_{1k}^{Du} \neq r_{1k}^{\text{opt}}$. But we shall now show that this is not possible. First consider the case that $r_{1k}^{Du} < r_{1k}^{\text{opt}}$. From Lemma 4, the strong user's rate cannot decrease under WUFBC. If $r_{1i}^{Du} = r_{1k}^{Du}, i \in \{k, \dots, s+m\}$, then S^{Du} transmits more bits to the weak user than S^{opt} does by t_{s+m} , i.e., $\sum_{i=1}^{s+m} (r_{2i}^{Du} - r_{2i}^{\text{opt}}) \xi_i = \sum_{i=k+1}^{s+m} (r_{2i}^{Du} - r_{2i}^{\text{opt}}) \xi_i = \sum_{i=k+1}^{s+m} (h_2(P_i^{Du}, r_{1i}^{Du}) - h_2(P_i^{\text{opt}}, r_{1i}^{\text{opt}})) \xi_i > 0$. However, at the end of the $(s+m)$ th epoch, S^{Du}

cannot send more bits to weak user because S^{opt} should have transmitted all the weak user bits. Therefore, r_1^{Du} should increase before t_{s+m} , i.e., at the end of epoch $k+n$, where $0 < n < (s+m-k)$. We have $r_{1(k+n)}^{\text{Du}} < r_{1(k+n+1)}^{\text{Du}}$; hence, either one of the two conditions in Lemma 4 must hold. Since $\sum_{i=1}^{k+n} (r_{1i}^{\text{opt}} - r_{1i}^{\text{Du}}) \xi_i > 0$, until t_{k+n} , S^{opt} has transmitted more bits to the strong user than S^{Du} does; therefore, all the strong user's bits arrived have not been transmitted by S^{Du} at the end of epoch $(k+n)$. Also, $r_{2(k+n)}^{\text{Du}} = h_2(P_{k+n}^{\text{Du}}, r_{1(k+n)}^{\text{Du}}) > h_2(P_{k+n}^{\text{opt}}, r_{1(k+n)}^{\text{opt}}) \geq 0$. Hence neither of the two conditions in Lemma 4 holds and strong user's rate cannot increase at t_{k+n} , which implies $r_{1k}^{\text{Du}} \neq r_{1k}^{\text{opt}}$. Thus, we are left with the case $r_{1k}^{\text{Du}} > r_{1k}^{\text{opt}}$. If $r_{1i}^{\text{opt}} = r_{1k}^{\text{opt}}, i \in \{k, \dots, s+m\}$, then $\sum_{i=1}^{s+m} (r_{1i}^{\text{Du}} - r_{1i}^{\text{opt}}) \xi_i > 0$, which contradicts the fact that DuOpt respects bit feasibility. Hence, strong user's rate in S^{opt} cannot remain constant after epoch k . Then we should have $r_{1i}^{\text{opt}} = r_{1k}^{\text{opt}}, i \in \{k, \dots, k+n\}$ and $r_{1(k+n)}^{\text{opt}} < r_{1(k+n+1)}^{\text{opt}}$. Since there is an increase in the strong user rate, at least one of the conditions in Lemma 4 should hold at t_{k+n} . However, we have $\sum_{i=1}^{k+n} (P_i^{\text{Du}} - P_i^{\text{opt}}) \xi_i > 0$ and $\sum_{i=1}^{k+n} (r_{1i}^{\text{Du}} - r_{1i}^{\text{opt}}) \xi_i > 0$, which tells us that neither one of the conditions in Lemma 4 holds, which implies that this final case is also not possible. Hence, we conclude that the case $P_s^{\text{Du}} > P_s^{\text{opt}}$ is not possible.

- (ii) Now consider the case $P_s^{\text{Du}} < P_s^{\text{opt}}$. We will prove that this case is also not possible by following a similar method to the one in case (i). First, suppose that the power of S^{Du} increases after s th epoch. This increase cannot be due to an energy constraint since S^{opt} consumes more energy than S^{Du} does until the increase in power. Hence, it should be due to data constraint and under WUFBC both user data constraints should be active. That is, S^{Du} transmits

all the feasible data until the increase in power. This implies that while consuming less energy, S^{Du} transmits at least the same number of bits than S^{opt} does, which contradicts the optimality of S^{opt} . Thus, power of S^{Du} cannot increase after epoch s . Also, it cannot decrease in time; otherwise, a local optimization results in either a gap or excess energy that propagates till the end of the schedule and transmission duration decreases. Therefore, the power of S^{Du} should stay constant after epoch s until T^{opt} . Now, we will analyze the rate assignments for both schedules. Let the transmission of S^{opt} end in epoch $(s+m)$ for $m \geq 0$ and suppose that $r_{1i}^{\text{Du}} = r_{1i}^{\text{opt}} \forall i < k, 0 < k < (s+m)$. At the k th epoch, there are three possible cases: $r_{1k}^{\text{Du}} > r_{1k}^{\text{opt}}$, $r_{1k}^{\text{Du}} < r_{1k}^{\text{opt}}$, and $r_{1k}^{\text{Du}} = r_{1k}^{\text{opt}}$. We will first consider the case $r_{1k}^{\text{Du}} > r_{1k}^{\text{opt}}$ and prove that this is not possible. Let $r_{1k}^{\text{Du}} > r_{1k}^{\text{opt}}$ and consider the rate of the strong user in S^{opt} after k th epoch. It cannot stay constant until T^{opt} , because it contradicts the fact that S^{opt} transmits all the feasible bits before T^{opt} , i.e., $\sum_{i=1}^{s+m} (r_{1i}^{\text{Du}} - r_{1i}^{\text{opt}}) \xi_i > 0$. Since the strong user's rate in S^{opt} cannot decrease by Lemma 4, it should increase at the end of epoch $(k+n)$ for $0 \leq n < (s+m-k)$, i.e., $r_{1(k+n)}^{\text{opt}} < r_{1(k+n+1)}^{\text{opt}}$. However, we have $\sum_{i=1}^{k+n} (r_{1i}^{\text{Du}} - r_{1i}^{\text{opt}}) \xi_i > 0$ and $r_{2(k+n)}^{\text{opt}} = h_2(P_{k+n}^{\text{opt}}, r_{1(k+n)}^{\text{opt}}) > h_2(P_{k+n}^{\text{Du}}, r_{1(k+n)}^{\text{Du}}) \geq 0$ which implies that none of the conditions in Lemma 4 holds and the strong user's rate in S^{Du} cannot increase after epoch k . Hence, we conclude that $r_{1k}^{\text{Du}} \neq r_{1k}^{\text{opt}}$. Now we consider the case $r_{1k}^{\text{Du}} < r_{1k}^{\text{opt}}$. Suppose that the strong user's rate in S^{opt} increase at epoch $(k+n)$ for $0 \leq n < (s+m-k)$. This increase in strong user's rate requires that at least one of the conditions in Lemma 4 should hold. However, we have $\sum_{i=1}^{k+n} (r_{1i}^{\text{Du}} - r_{1i}^{\text{opt}}) \xi_i > 0$ and

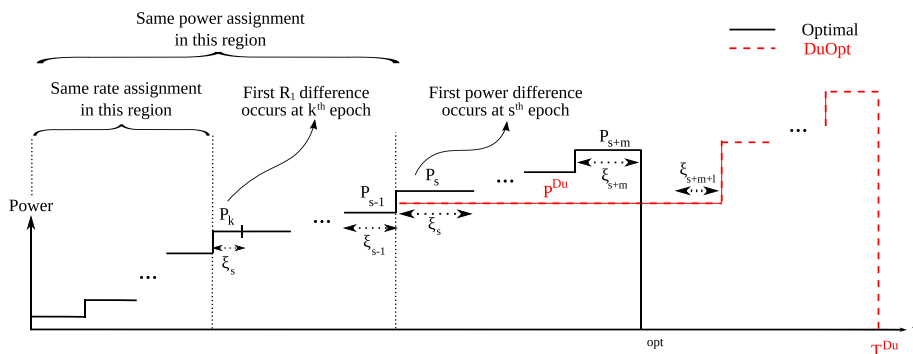


Figure 9 Comparison of the optimal schedule and the final schedule of DuOpt Algorithm. Illustration of the final case in the proof of Theorem 3.

$r_{2(k+n)}^{\text{opt}} = h_2(P_{k+n}^{\text{opt}}, r_{1(k+n)}^{\text{opt}}) > h_2(P_{k+n}^{\text{Du}}, r_{1(k+n)}^{\text{Du}}) \geq 0$, so the strong user's rate in S^{Du} cannot increase, i.e., $r_{1k}^{\text{Du}} \neq r_{1k}^{\text{opt}}$. Since the strong user's rate in S^{opt} cannot decrease by Lemma 4, it should stay constant until T^{opt} .

Thus far, we have shown that if S^{Du} is different than S^{opt} , then S^{Du} cannot have higher power level than S^{opt} until T^{opt} . Moreover, if power level of S^{Du} becomes lower than that of S^{opt} , then it should stay constant until T^{opt} , and if the strong user's rate of S^{Du} becomes lower than that of S^{opt} , then it should stay constant until T^{opt} . These results are shown in the general case in Figure 9.

Now, let \tilde{b}_2^{Du} and \tilde{b}_2^{opt} be the number of bits transmitted to the weak user till $t_{s+m+l-1}$ by S^{Du} and till T^{opt} by S^{opt} , respectively. Then, we have

$$\begin{aligned}
 \tilde{b}_2^{\text{Du}} - \tilde{b}_2^{\text{opt}} &= \sum_{i=1}^{s+m+l} \xi_i h_2(P_i^{\text{Du}}, r_{1s}^{\text{Du}}) - \sum_{i=1}^{s+m+l} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}) \\
 &= \sum_{i=k}^{s-1} \xi_i \underbrace{(h_2(P_i^{\text{Du}}, r_{1s}^{\text{Du}}) - h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}))}_{>0} \\
 &\quad + \left(\sum_{i=s}^{s+m+l} \xi_i \right) h_2(P_s^{\text{Du}}, r_{1s}^{\text{Du}}) - \sum_{i=s}^{s+m} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}) \\
 &> \left(\sum_{i=s}^{s+m+l} \xi_i \right) h_2(P_s^{\text{Du}}, r_{1s}^{\text{Du}}) - \sum_{i=s}^{s+m} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}) \\
 &\geq \left(\sum_{i=s}^{s+m+l} \xi_i \right) h_2 \left(\sum_{i=s}^{s+m} \frac{\xi_i}{\sum_{i=s}^{s+m+l} \xi_i} P_i^{\text{opt}}, r_{1s}^{\text{Du}} \right) \\
 &\quad - \sum_{i=s}^{s+m} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}) \\
 &> \left(\sum_{i=s}^{s+m+l} \xi_i \right) \left(\sum_{i=s}^{s+m} \frac{\xi_i}{\sum_{i=s}^{s+m+l} \xi_i} h_2(P_i^{\text{opt}}, r_{1s}^{\text{Du}}) \right) \\
 &\quad - \sum_{i=s}^{s+m} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}) \\
 &= \sum_{i=s}^{s+m} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{Du}}) - \sum_{i=s}^{s+m} \xi_i h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}) \\
 &= \sum_{i=s}^{s+m} \xi_i \underbrace{(h_2(P_i^{\text{opt}}, r_{1s}^{\text{Du}}) - h_2(P_i^{\text{opt}}, r_{1s}^{\text{opt}}))}_{>0} \\
 &> 0
 \end{aligned}$$

(23)

From Equation 23, S^{Du} transmits more bits to the weak user than S^{opt} does, then this final case also cannot happen. Therefore, we conclude that the schedule returned by DuOpt cannot be different than the unique optimal schedule, i.e., $S^{\text{Du}} = S^{\text{opt}}$. ■

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This work was supported by TUBITAK under grant 110E252. A preliminary version of some of these results was presented at ISIT 2011 St. Petersburg, Russia, Aug 2011 [15].

Received: 2 December 2012 Accepted: 30 June 2013

Published: 18 July 2013

References

1. E Uysal-Biyikoglu, B Prabhakar, AE Gamal, Energy-efficient packet transmission over a wireless link. *IEEE Trans. Netw.* **10**, 487–499 (2002)
2. E Uysal-Biyikoglu, AE Gamal, in *Proc. IEEE Intl. Symposium on Information Theory*. Energy-efficient packet transmission over a multi-access channel (Lausanne, Switzerland, 30 June–5 July 2002), p. 153
3. RA Berry, RG Gallager, Communication over fading channels with delay constraints. *IEEE Trans. on Inf. Theory*. **48**, 1135–1149 (2002)
4. P Nuggehalli, V Srinivashan, RR Rao, in *Proc. IEEE INFOCOM*, vol. 3. Delay constrained energy efficient transmission strategies for wireless devices (IEEE New York, 23–27 June 2002), pp. 1765–1772
5. MA Zafer, E Modiano, A calculus approach to energy-efficient data transmission with quality of service constraints. *IEEE Trans. Netw.* **17**, 898–911 (2009)
6. J Yang, S Ulukus, Optimal packet scheduling in an energy harvesting communication system. *IEEE Trans. Commun.* **60**, 220–230 (2012)
7. M Gorlatova, T Sharma, D Shrestha, E Xu, J Chen, A Skolnik, D Piao, P Kinget, J Kymissis, D Rubenstein, G Zussman, in *2010 7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*. Prototyping Energy Harvesting Active Networked Tags (EnHANTs) with MICA2 Motes (IEEE Boston, Massachusetts, 21–25 June 2010), pp. 1–3
8. JM Dilhac, *Presentation of Chist-ERA Project "SMARTER"*, (Brussels, Belgium). <http://www.chistera.eu/projects/smarter>
9. M Gorlatova, A Wallwater, G Zussman, in *2011 IEEE Proceedings on INFOCOM*. Networking low-power energy harvesting devices: Measurements and algorithms (IEEE Shanghai, China, 10–15 April 2011), pp. 1602–1610
10. K Tutuncuoglu, A Yener, Optimum transmission policies for battery limited energy harvesting nodes. *IEEE Trans. Wireless Commun.* **11**, 1180–1189 (2012)
11. O Ozel, K Tutuncuoglu, J Yang, S Ulukus, A Yener, Transmission with energy harvesting nodes in fading wireless channels: Optimal policies. *IEEE J. Selected Areas Commun.* **29**, 1732–1743 (2011)
12. MA Antepi, E Uysal-Biyikoglu, H Erkal, Optimal packet scheduling on an energy harvesting broadcast link. *IEEE J. Selected Areas Commun.* **29**, 1721–1731 (2011)
13. J Yang, O Ozel, S Ulukus, Broadcasting with an energy harvesting rechargeable transmitter. *IEEE Trans. Wireless Commun.* **11**, 571–583 (2012)
14. O Ozel, J Yang, S Ulukus, in *International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*. Broadcasting with a battery limited energy harvesting rechargeable transmitter (IEEE Princeton, NJ, May 2011), pp. 205–212
15. FM Ozelcelik, H Erkal, E Uysal-Biyikoglu, in *Proc. IEEE Intl. Symposium on Information Theory*. Optimal offline packet scheduling on an energy harvesting broadcast link (Saint Petersburg, Russia, 31 July–5 August 2011, 2011), pp. 2886–2890
16. S Chen, P Sinha, NB Shroff, C Joo, in *2011 Proc. IEEE INFOCOM*. Finite-horizon energy allocation and routing scheme in rechargeable sensor networks (IEEE Shanghai, China, 10–15 Apr 2011), pp. 2273–2281
17. M Gatzianas, L Georgiadis, L Tassioulas, Control of wireless networks with rechargeable batteries. *IEEE Trans. Commun.* **9**, 581–593 (2010)

18. CK Ho, R Zhang, in *Proc. IEEE Intl. Symposium on Information Theory*. Optimal energy allocation for wireless communications powered by energy harvesters (IEEE Austin, Texas, 13–18 June 2010, 2010), pp. 2368–2372
19. O Ozel, J Yang, S Ulukus, K Tutuncuoglu, in *Information Sciences and Systems (CISS)*. A Yener, Adaptive transmission policies for energy harvesting wireless nodes in fading channels (IEEE Baltimore, MD, March 2011), pp. 1–6
20. V Sharma, U Mukherji, V Joseph, S Gupta, Optimal energy management policies for energy harvesting sensor nodes. *Wireless Commun., IEEE Trans.* **9**(4), 1326–1336 (2010)
21. TM Cover, JA Thomas, *Elements of Information Theory*. (Wiley, New York, 1991). Wiley Series in Telecommunications
22. E Uysal-Biyikoglu, AE Gamal, On adaptive transmission for energy efficiency in wireless data networks. *IEEE Trans. Inf Theory.* **50**, 3081–3094 (2004)
23. S Boyd, L Vandenberghe, *Convex Optimization*. (Cambridge University, New York, 2004)
24. H Erkal, *Optimization of energy harvesting wireless communication systems Master's thesis*. (Middle East Technical University, Dept. of Electrical and Electronics Engineering, Ankara, Turkey, 2011)
25. AV Fiacco, GP McCormick, *Nonlinear Programming; Sequential Unconstrained Minimization Techniques*. (Wiley, New York, 1968)
26. MS Bazaraa, HD Sherali, CM Shetty, *Nonlinear Programming: Theory and Algorithms*. (Wiley-Interscience, Hoboken, 2006)

doi:10.1186/1687-1499-2013-197

Cite this article as: Erkal et al.: Optimal offline broadcast scheduling with an energy harvesting transmitter. *EURASIP Journal on Wireless Communications and Networking* 2013 **2013**:197.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
