Counterexamples on the Monotonicity of Delay Optimal Strategies for Energy Harvesting Transmitters

Borna Sayedana⬤ and Aditya Mahajan⬤

Abstract—We consider cross-layer design of delay optimal transmission strategies for energy harvesting transmitters where the data and energy arrival processes are stochastic. Using Markov decision theory, we show that the value function is weakly increasing in the queue state and weakly decreasing in the battery state. It is natural to expect that the delay optimal policy should be weakly increasing in the queue and battery states. We show via counterexamples that this is not the case. In fact, we show that for some sample scenarios the delay optimal policy may perform 5–13% better than the best *monotone* policy.

Index Terms—Energy harvesting transmitters, Markov decision processes, monotone policy, power-delay trade-off.

I. INTRODUCTION

LATENCY is an important consideration in many Internet of Things (IoT) applications which provide real-time and/or critical services. Often IoT devices are battery powered and harvest energy from the environment. In such situations, intelligent transmission strategies are needed to mitigate the unreliability of available energy and provide low-latency services.

In this letter, we investigate the cross-layer design of delay optimal transmission strategies for energy harvesting transmitters when both the data arrival and the energy arrival processes are stochastic. Our motivation is to characterize qualitative properties of optimal transmission policies for such model. For example, in queuing theory, it is often possible to establish that the optimal policy is monotone increasing in the queue length [1], [2]. Such a property, in addition to being intuitively satisfying, simplifies the search and implementation of the optimal strategies. Such monotonicity properties are also known to hold for cross-layer design of communication systems when energy is always available at the transmitter [3]. So it is natural to ask if such qualitative properties hold for energy harvesting transmitters.

Partial answers to this question for throughput optimal policies for energy harvesting transmitters are provided in [4]–[9]. Under the assumptions of backlogged traffic or deterministic data arrival process or deterministic energy arrival process,
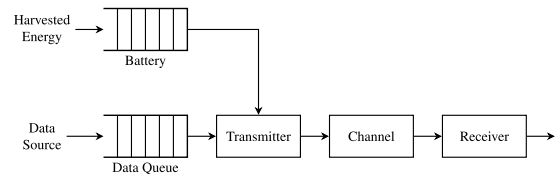
Fig. 1. Model of a transmitter with energy-harvester.

these papers show that the optimal policy is weakly increasing in the queue state and/or weakly increasing in the battery state. There are other papers that investigate the structure of delay or throughput optimal policies under the assumption of a deterministic energy arrival process [10]–[12].

There are some papers which investigate the problem of delay optimization for energy harvesting transmitters [12]–[14], but they don't characterize the structure of delay-optimal policies rather provide numerical solutions or propose low-complexity heuristic policies or only establish structural properties of value functions.

We show that the delay optimal policy for energy harvesting communication systems is not necessarily monotone in battery or queue state. This is in contrast to the monotonicity of delay optimal policies when energy is always available [3] or throughput optimal policies for energy harvesting models [4]–[9]. We present counterexamples to show that the delay optimal policy need not be weakly increasing in queue or battery state. Furthermore, for some sample scenarios, the performance of the optimal policy is about 5–13% better than that of the best monotone policy. These counterexamples continue to hold for i.i.d. fading channels as well.

Notation: Uppercase letters (e.g., E, N, etc.) represent random variables; the corresponding lowercase letters (e.g., e, n, etc.) represent their realizations. Cursive letters (e.g., $\mathcal{L}$, $\mathcal{B}$, etc.) represent sets. The sets of real, positive integers, and non-negative integers are denoted by $\mathbb{R}$, $\mathbb{Z}_{>0}$, and $\mathbb{Z}_{\geq 0}$ respectively. The notation $[a]_L$ is a short hand for $\min\{a, \bar{L}\}$.

II. MODEL AND PROBLEM FORMULATION

Consider a discrete-time communication system shown in Fig 1. A source generates bursty data packets that have to be transmitted to a receiver by an energy-harvesting transmitter. The transmitter has finite buffer where the data packets are queued and a finite capacity battery where the harvested energy is stored.

At the beginning of a slot, the transmitter picks some data packets from the queue, encodes them, and transmits the

encoded symbol. Transmitting a symbol requires energy that depends on the number of encoded packets in the symbol. At the end of the slot, the system incurs a delay penalty that depends on the number of packets remaining in the queue.

Time slots are indexed by $k \in \mathbb{Z}_{\geq 0}$. The length of the buffer is denoted by L and the size of the battery by B; $\mathcal{L}$ and $\mathcal{B}$ denote the sets $\{0, 1, \ldots, L\}$ and $\{0, 1, \ldots, B\}$, respectively. Other variables are as follows:

- $N_k \in \mathcal{L}$: the number of packets in the queue at the beginning of slot k.
- $A_k \in \mathcal{L}$: the number of packets that arrive during slot k.
- $S_k \in \mathcal{B}$: the energy stored in the battery at the beginning of slot k.
- $E_k \in \mathcal{B}$: the energy that is harvested during slot k.
- U_k: the number of packets transmitted during slot k. The feasible choices of U_k are denoted by $\mathcal{U}(N_k, S_k)$ where

$$\mathcal{U}(n, s) := \{u \in \mathcal{L} : u \leq n \text{ and } p(u) \leq s\},$$

where $p(u)$ denotes the amount of power needed to transmit u packets.[1] We assume that $p : \mathcal{L} \to \mathbb{R}_{\geq 0}$ is a strictly convex and increasing function with $p(0) = 0$.

The dynamics of the data queue and the battery are

$$N_{k+1} = [N_k - U_k + A_k]_L \text{ and } S_{k+1} = [S_k - p(U_k) + E_k]_B.$$

Packets that are not transmitted during slot k incur a delay penalty $d(N_k - U_k)$, where $d : \mathcal{L} \to \mathbb{R}_{\geq 0}$ is a convex and increasing function with $d(0) = 0$.

The data arrival process $\{A_k\}_{k \geq 0}$ is i.i.d. with pmf (probability mass function) P_A. The energy arrival process $\{E_k\}_{k \geq 0}$ is i.i.d. with pmf P_E and is also independent of $\{A_k\}_{k \geq 0}$.

The number U_k of packets to transmit are chosen according to a scheduling policy $f := \{f_k\}_{k \geq 0}$, where

$$U_k = f_k(N_k, S_k), \quad U_k \in \mathcal{U}(N_k, S_k).$$

The performance of a scheduling policy f is given by

$$J(f) := \mathbb{E}^f \left[\sum_{k=0}^{\infty} \beta^k d(N_k - U_k) \mid N_0 = 0, S_0 = 0 \right], \quad (1)$$

where $\beta \in (0, 1)$ denotes the discount factor and the expectation is taken with respect to the joint measure on the system variables induced by the choice of f.

We are interested in the following optimization problem.

Problem 1: Given the buffer length L, battery size B, power cost $p(\cdot)$, delay cost $d(\cdot)$, pmf P_A of the arrival process, pmf P_E of the energy arrival process, and the discount factor β, choose a feasible scheduling policy f to minimize the performance $J(f)$ given by (1).

III. DYNAMIC PROGRAMMING DECOMPOSITION

The system described above can be modeled as an infinite horizon time homogeneous Markov decision process (MDP) [15]. Since the state and action spaces are finite, standard results from Markov decision theory imply that there

exists an optimal policy which is time homogeneous and is given by the solution of a dynamic program. To succinctly write the dynamic program, we define the following Bellman operator: Define the operator $\mathscr{B} : [\mathcal{L} \times \mathcal{B} \to \mathbb{R}] \to [\mathcal{L} \times \mathcal{B} \to \mathbb{R}]$ that maps any $V : \mathcal{L} \times \mathcal{B} \to \mathbb{R}$ to

$$[\mathscr{B} V](n, s) = \min_{u \in \mathcal{U}(n, s)} \left\{ d(n - u) + \beta \mathbb{E}[V([n - u + A]_L, [s - p(u) + E]_B)] \right\}, \quad (2)$$

where A and E are independent random variables with pmfs P_A and P_E. Then, an optimal policy for the infinite horizon MDP is given as follows [15].

Theorem 1: Let $V^* : \mathcal{L} \times \mathcal{B} \to \mathbb{R}$ denote the unique fixed point of the following equation:

$$V(n, s) = [\mathscr{B} V](n, s), \quad \forall (n, s) \in \mathcal{L} \times \mathcal{B}. \quad (3)$$

Furthermore, let f^* be such that $f^*(n, s)$ attains the minimum in the right hand side of (3). Then, the time homogeneous policy $f^{*,\infty} = (f^*, f^*, \ldots)$ is optimal for Problem 1.

The dynamic program described in (3) can be solved using standard algorithms such as value iteration, policy iteration, or linear programming algorithms [15].

A. Properties of the Value Function

Let $\mathcal{M}$ denote the family of the functions $V : \mathcal{L} \times \mathcal{B} \to \mathbb{R}$ such that for any $s \in \mathcal{B}$, $V(n, s)$ is weakly increasing in n and for any $n \in \mathcal{L}$, $V(n, s)$ is weakly decreasing in s. Furthermore, let $\mathcal{F}_s$ denote the family of functions $f : \mathcal{L} \times \mathcal{B} \to \mathcal{U}$ such that for any $n \in \mathcal{L}$, $f(n, s)$ is weakly increasing in s. Similarly, let $\mathcal{F}_n$ be family of functions $f : \mathcal{L} \times \mathcal{B} \to \mathcal{U}$, such that for any $s \in \mathcal{B}$, $f(n, s)$ is weakly increasing in n.

Proposition 1: The optimal value function $V^* \in \mathcal{M}$.

The proof is presented in the Appendix. Proposition 1 says the optimal cost weakly increases with the queue state and weakly decreases with the battery state. Thus, it is better to have less packets in the queue and it is better to have more energy in the battery. Such a result is intuitively appealing.

One might argue that it should be the case that the optimal policy is weakly increasing in state of the queue, and weakly increasing in the available energy in the battery. In particular, if it is optimal to transmit u packets when the queue state is n, then (for the same battery state) the optimal number of packets to transmit at any queue state larger than n should be at least u. Similarly, if it is optimal to transmit u packets when the battery state is s, then (for the same queue state) the optimal number of packets to transmit at any battery state larger than s should be at least u. In the next section, we present counterexamples that show both of these properties do not hold. The code for all the results is available at [16].

IV. COUNTEREXAMPLES ON THE MONOTONICITY OF OPTIMAL POLICIES

A. On the Monotonicity in Queue State

Consider the communication system with a band-limited AWGN channel where $\mathcal{L} = 5$, $\mathcal{B} = 5$, $\beta = 0.99$, $N_0 = 2.0$,

---

[1] In our examples, we model the channel as a band-limited AWGN channel with bandwidth W and noise level N_0. The capacity of such a channel when transmitting at power level P is $W \log_2(1 + P/(N_0 W))$. Therefore, for such channels we assume $p(u) = \lfloor N_0 W (2^{u/W} - 1) \rfloor$.

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	1	1	1	1
2	0	1	1	1	1	1
3	0	1	1	1	1	1
4	0	1	1	1	1	1
5	0	0	0	0	1	1

(a) The optimal policy

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	1	1	1	1
2	0	1	1	1	1	1
3	0	1	1	1	1	1
4	0	1	1	1	1	1
5	0	1	1	1	1	1

(b) The best monotone policy

Fig. 2. The optimal and the best monotone policies for the example of Section IV-A.

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	1	1	1	1
2	0	1	1	1	2	2
3	0	1	1	1	2	2
4	0	1	1	1	2	2
5	0	1	1	0	2	2

(a) The optimal policy

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	1	1	1	1
2	0	1	1	1	2	2
3	0	1	1	1	2	2
4	0	1	1	1	2	2
5	0	1	1	1	2	2

(b) The best monotone policy

Fig. 3. The optimal and the best monotone policies for the example of Section IV-A.

$W = 1.75$ (thus, $p(u) = \lfloor 3.5 \cdot (2^{(u/1.75)} - 1) \rfloor$), $d(q) = q$, data arrival distribution $P_A = \text{Geom}(0.9)$, and energy arrival distribution $P_E = \text{Geom}(0.89)$, where both the pmfs are truncated and normalized with domain equal to 5.

The optimal policy for this system (obtained by policy iteration [15]) is shown in Fig. 2(a), where the rows correspond to the current queue length and the columns correspond to the current energy level. Note that the policy is not weakly increasing in queue state (i.e., $f^* \notin \mathcal{F}_n$). For instance, $f(5, 3) < f(4, 3)$.

Given that the optimal policy is not monotone, one might wonder how much do we lose if we use a monotone policy instead of the optimal policy. To characterize this, we define the best queue-monotone policy as:

$$f_n^\circ = \arg \min_{f \in \mathcal{F}_n} \left\{ \max_{(n,s) \in \mathcal{L} \times \mathcal{B}} \left| V(n, s) - V^*(n, s) \right| \right\}$$

and let V_n° denote the corresponding value function.

The best monotone policy cannot be obtained using dynamic programming and one has to resort to a brute force search over all monotone policies. For the model described above, there are 86400 monotone policies.[2] The best monotone policy obtained by searching over these is shown is Fig. 2(b). The worst case difference between the two value functions is given by

$$\alpha_n = \max_{(n,s) \in \mathcal{L} \times \mathcal{B}} \left\{ \frac{V^\circ(n,s) - V^* n(n,s)}{V^*(n,s)} \right\} = 0.1186.$$

Thus, for this counterexample, the best queue-monotone policy performs 11.86% worse than the optimal policy.

We also compare the performance of the optimal policy with the greedy policy, which is a heuristic policy that transmits the maximum number of packets in each state. The greedy policy is monotone so we expect $\alpha_n^{\text{greedy}} \geq \alpha_n$. In this particular example, we find that $\alpha_n^{\text{greedy}} = 0.8609$. Thus, the greedy policy performs 86.09% worse than the optimal policy.

B. On the Monotonicity in Battery State

Consider the communication system described in Section IV-A but with the data arrival distribution $P_A = [0.33, 0.67, 0, 0, 0]$ and the energy arrival distribution $P_E = [0.05, 0.90, 0.05, 0, 0]$.

The optimal policy (obtained using policy iteration [15]) is shown in Fig. 3(a). Note that the policy is not weakly

---

---

increasing in the battery state (i.e., $f^* \notin \mathcal{F}_s$). In particular, we have that $f^*(5, 2) > f^*(5, 3)$.

Given that optimal policy is not monotone, the previous question arises again that how much do we lose if we use a monotone policy instead of the optimal policy. To characterize this, we define the best battery-monotone policy as:

$$f_s^\circ = \arg \min_{f \in \mathcal{F}_s} \left\{ \max_{(n,s) \in \mathcal{L} \times \mathcal{B}} \left| V(n, s) - V^*(n, s) \right| \right\}$$

and let V_s° denote the corresponding value function.

As before, we find the best monotone policy by a brute force search over all 303750 monotone battery-policies.[2] The resultant policy is shown in Fig. 3(b).

The worst case difference between the two value functions is given by

$$\alpha_s = \max_{(n,s) \in \mathcal{L} \times \mathcal{B}} \left\{ \frac{V^\circ(n,s) - V^* s(n,s)}{V^*(n,s)} \right\} = 0.0560.$$

Thus, for this counterexample, the best battery-monotone policy performs 5.60% worse than the optimal policy. Note that for this example, the best monotone policy is a greedy policy, hence the performance of the greedy policy is same as that of the best monotone policy.

V. COUNTEREXAMPLES FOR FADING CHANNELS

A. Channel Model With i.i.d. Fading

Consider the model in Section II where the channel has i.i.d. fading. In particular, let $H_k \in \mathcal{H}$ denote the channel state at time k and $g(H_k)$, where $g : \mathcal{H} \to \mathbb{R}_{>0}$, denotes the attenuation at state H_k. Thus, the power needed to transmit u packets when the channel is in state h is given by $p(u)/g(h)$. We assume that $\{H_k\}_{k \geq 0}$ is an i.i.d. process with pmf P_H that is independent of the data and energy arrival processes $\{A_k\}_{k \geq 0}$ and $\{E_k\}_{k \geq 0}$.

B. On the Monotonicity in Queue State

Consider the model in Section IV-A with $N_0 = 1$, $W = 1.75$, and an i.i.d. fading channel where $\mathcal{H} = \{1, 2\}$, $g(\cdot) = \{0.7, 0.8\}$ and $P_H = [0.4, 0.6]$. The optimal policy for this model (obtained using policy iteration) is shown in Fig. 4(a)–4(b). Note that for all h, the optimal policy in not monotone in the queue length.

In this case, there are $(4320) \times (1296) \times (362) \approx 10^8$ monotone policies. Therefore, a brute force search to find the best monotone policy is not possible. We choose a heuristic monotone policy f_n° which differs from f^* only at the following

$$
\begin{array}{c}
\begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\end{array}
$$

(a) $f^*(\cdot, \cdot, h = 1)$

$$
\begin{array}{c}
\begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 2 & 2 & 2 & 2 \\
0 & 1 & 2 & 2 & 2 & 2 \\
0 & 1 & 2 & 2 & 2 & 2 \\
0 & 0 & 0 & 0 & 2 & 2
\end{bmatrix}
\end{array}
$$

(b) $f^*(\cdot, \cdot, h = 2)$

$$
\begin{array}{c}
\begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 0 & 2 & 2
\end{bmatrix}
\end{array}
$$

(c) $f^*(\cdot, \cdot, h = 3)$

$$
\begin{array}{c}
\begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 0 & 2 & 2
\end{bmatrix}
\end{array}
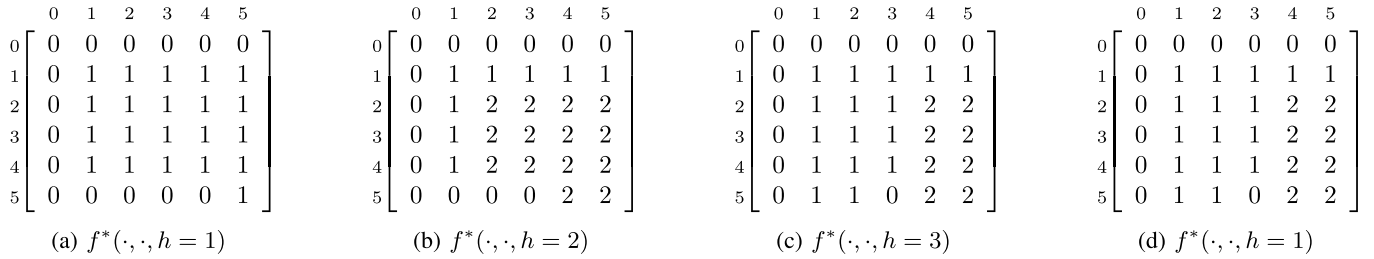$$

(d) $f^*(\cdot, \cdot, h = 1)$

Fig. 4. The optimal policy for the examples of Section V-B shown in subfigures (a)–(b) and Section V-C shown in subfigures (c)–(d).

points: $f_n^\circ(5, s, 1) = 1$, for $s \in \{1, 2, 3, 4\}$, $f_n^\circ(5, 1, 2) = 1$, and $f_n^\circ(5, s, 2) = 2$, for $s \in \{2, 3\}$. The policy f_n° may be thought of as the queue-monotone policy that is closest to f^*. Let V_n° denote the corresponding value function. The worst case difference between the two value functions is given by

$$
\alpha_n = \max_{(n, s, h) \in \mathcal{L} \times \mathcal{B} \times \mathcal{H}} \frac{\left| V^*(n, s, h) - V_n^\circ(n, s, h) \right|}{\left| V^*(n, s, h) \right|} = 0.1344.
$$

Thus, the heuristically chosen queue-monotone policy performs 13.44% worse than the optimal policy. We also compare the optimal policy with the greedy policy and find that $\alpha_n^{\text{greedy}} = 0.8005$. Thus, the greedy policy performs 80.05% worse than the optimal policy.

C. On the Monotonicity in the Battery State

Consider the model in Section IV-B with $N_0 = 1.55$, $W = 1.75$, and an i.i.d. fading channel where $\mathcal{H} = \{1, 2\}$, $g(\cdot) = \{0.75, 0.80\}$, and $P_H = [0.3, 0.7]$. The optimal policy for this model (obtained using policy iteration) is shown in Fig. 4(c)–4(d). Note that for $h \in \{1, 2\}$, the optimal policy is not monotone in the battery state.

In this case, there are $(629856) \times (30375019) \approx 10^{10}$ monotone policies. Therefore, a brute force search is not possible. As before, we choose a heuristic policy f_s° which is the battery-monotone policy that is closest to f^*. In particular, f_s° differs from f^* only at two points: $f_s^\circ(5, 3, 1) = 1$ and $f_s^\circ(5, 3, 2) = 1$. Let V_s° denote the corresponding value function. The worst case difference between the two value functions is given by

$$
\alpha_s = \max_{(n, s, h) \in \mathcal{L} \times \mathcal{B} \times \mathcal{H}} \frac{\left| V^*(n, s, h) - V_s^\circ(n, s, h) \right|}{\left| V^*(n, s, h) \right|} = 0.0560.
$$

Thus, the heuristically chosen battery-monotone policy performs 5.60% worse than the optimal policy. Note that for this example, the best monotone policy is a greedy policy, hence the performance of the greedy policy is same as that of the best monotone policy.

VI. CONCLUSION

In this letter, we consider delay optimal strategies in cross layer design with energy harvesting transmitter. We show that the value function is weakly increasing in the queue state and weakly decreasing in the battery state. We show via counterexamples that the optimal policy is not monotone in queue length nor in the available energy in the battery.

A. Discussion About the Counterexamples

One might ask why the optimal policy is not monotone in the above model. The standard argument in MDPs to establish monotonicity of the optimal policies is to show that the value-action function is submodular in the state and action. The value-action function is given by

$$
H(n, s, u) = d(n - u) + \beta \mathbb{E}[V([n - u + A]_L, [s - p(u) + E]_B)]
$$

A sufficient condition for the optimal policy to be weakly increasing in the queue length is:
(S1) for every $s \in \mathcal{B}$, $H(n, s, u)$ is submodular in (n, u).
Note that since $d(\cdot)$ is convex, $d(n - u)$ is submodular in (n, u). Thus, a sufficient condition for (S1) to hold is:
(S2) for all $s \in \mathcal{B}$, $\mathbb{E}[V([n - u + A]_L, [s - p(u) + E]_B)]$ is submodular in (n, u).
Since submodularity is preserved under addition, a sufficient condition for (S2) to hold is:
(S3) for all $s \in \mathcal{B}$, $V(n - u, s - p(u))$ is submodular in (n, u).

By a similar argument, it can be shown that a sufficient condition for the optimal policy to be weakly increasing in battery state is:
(S4) for all $n \in \mathcal{L}$, $V(n - u, s - p(u))$ is submodular in (s, u).
We have not been able to identify sufficient conditions under which (S3) or (S4) hold. Note that if the data were backlogged, then we do not need to keep track of the queue state; thus, the value function is just a function of the battery state. In such a scenario, (S4) simplifies to $V(s - p(u))$ is submodular in (s, u). Since $p(\cdot)$ is convex, it can be shown that convexity of $V(s)$ is sufficient to establish submodularity of $V(s - p(u))$. This is the essence of the argument given in [6], [7].

Similarly, if the transmitter had a steady supply of energy, then we do not need to keep track of the battery state; thus, the value function is just a function of the queue state. In such a scenario, (S3) simplifies to $V(n - u)$ is submodular in (n, u). It can be shown that convexity of the $V(n)$ is sufficient to establish submodularity of $V(n - u)$. This is the essence of the argument given in [3].

In our model, data is not backlogged and energy is intermittent. As a result, we have two queues—the data queue and the energy queue—which have coupled dynamics. This coupling makes it difficult to identify conditions under which $V(n - u, s - p(u))$ will be submodular in (n, u) or (s, u).

B. Implication of the Results

In general, there are two benefits if one can establish that the optimal policy is monotone. The first advantage is that monotone policies are easier to implement. In particular, one needs a $(L + 1) \times (B + 1)$-dimensional look-up table to implement a general transmission policy (similar to the matrices shown in Figs. 2 and 3). In contrast, one only needs to store the thresholds boundaries of the decision regions (which can be stored in a sparse matrix) to implement a queue- or battery-monotone policy. Our counterexamples show that such a simpler implementation will result in a loss of optimality in energy-harvesting systems.

The second advantage is that if we know that the optimal policy is monotone, we can search for them efficiently using monotone value iteration and monotone policy iteration [15]. Our counterexamples show that these more efficient algorithms cannot be used in energy-harvesting systems.

One might want to restrict to monotone policies for the sake of implementation simplicity. However, if the system does not satisfy properties (S3) and (S4) mentioned in the previous section, then dynamic programming cannot be used to find the best monotone policy. Thus, one has to resort to a brute force search, which suffers from the curse of dimensionality.

APPENDIX

A. Monotonicity of Bellman Operator

Lemma 1: Given $V : \mathcal{L} \times \mathcal{B} \to \mathbb{R}$, define $H : \mathcal{L} \times \mathcal{B} \times \mathcal{U} \to \mathbb{R}$:

$$H(n, s, u) = d(n - u) + \beta \mathbb{E}[V([n - u + A]_L, [s - p(u) + E]_B)].$$

If $V \in \mathcal{M}$, then for all $n \in \mathcal{L}$, $s \in \mathcal{B}$, and $u \in \mathcal{U}(n, s)$:

1) $H(n, s, u) \leq H([n + 1]_L, s, u)$.
2) $H(n, s, n) \leq H([n + 1]_L, s, [n + 1]_L)$.
3) $H(n, [s + 1]_B, u) \leq H(n, s, u)$.

Consequently, $\mathcal{B} V \in \mathcal{M}$.

Proof: The properties of H follow from the monotonicity of $d(\cdot)$ and the fact that monotonicity is preserved under expectations. The details are omitted due to lack of space.

To prove that $W = \mathcal{B} V \in \mathcal{M}$, we consider any $n \in \mathcal{L}$ and $s \in \mathcal{B}$ and let $f(n, s)$ denote a policy that achieves the minimum in the definition of $\mathcal{B} V$. There are two cases: $f(n + 1, s) \neq n + 1$ and $f(n + 1, s) = n + 1$.

1) Suppose $u^* = f(n + 1, s) \neq n + 1$. Then, it must be the case that $u^* \in \mathcal{U}(n, s)$. Thus,

$$W(n + 1, s) = H(n + 1, s, u^*) \overset{(a)}{\geq} H(n, s, u^*)$$
$$\geq \min_{u \in \mathcal{U}(n,s)} H(n, s, u) = W(n, s),$$

where (a) follows from Property 1.

2) Suppose $u^* = f(n + 1, s) = n + 1$. Then, it must be the case that $p(n + 1) \leq s$ and, therefore, $p(n) \leq s$. Hence $n \in \mathcal{U}(n, s)$. Thus,

$$W(n + 1, s) = H(n + 1, s, n + 1) \overset{(b)}{\geq} H(n, s, n)$$
$$\geq \min_{u \in \mathcal{U}(n,s)} H(n, s, u) = W(n, s),$$

where (b) follows from Property 2.

As a result of both of these cases, we get that

$$W(n, s) \leq W(n + 1, s). \tag{4}$$

Now let $u^* = f(n, s)$, recall that $\mathcal{U}(n, s) \subseteq \mathcal{U}(n, s + 1)$ then $u^* \in \mathcal{U}(n, s + 1)$ thus

$$W(n, s) = H(n, s, u^*) \overset{(c)}{\geq} H(n, s + 1, u^*)$$
$$\overset{(d)}{\geq} \min_{u \in \mathcal{U}(n,s+1)} H(n, s + 1, u) = W(n, s + 1), \tag{5}$$

Where (c) follows from Property 3 and (d) follows from the fact that $u^* \in \mathcal{U}(n, s + 1)$.

From (4) and (5) we infer $W \in \mathcal{M}$. ∎

B. Proof of Proposition 1

Arbitrarily initialize $V^{(0)} \in \mathcal{M}$ and for $n \in \mathbb{Z}_{>0}$, recursively define $V^{(n+1)} = \mathcal{B} V^{(n)}$. Since $V^{(0)} \in \mathcal{M}$, Lemma 1 implies that $V^{(n)} \in \mathcal{M}$, for all $n \in \mathbb{Z}_{>0}$. Since monotonicity is preserved under the limit, we have that $\lim_{n \to \infty} V_0^{(n)} \in \mathcal{M}$. By [15], $\lim_{n \to \infty} V_0^{(n)} = V$. Hence, $V \in \mathcal{M}$.

REFERENCES

[1] S. Stidham, Jr., and R. R. Weber, "Monotonic and insensitive optimal policies for control of queues with undiscounted costs," *Oper. Res.*, vol. 37, no. 4, pp. 611–625, 1989.

[2] E. Gallisch, "On monotone optimal policies in a queueing model of M/G/1 type with controllable service time distribution," *Adv. Appl. Probab.*, vol. 11, no. 4, pp. 870–887, 1979.

[3] R. A. Berry, "Power and delay trade-offs in fading channels," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2000.

[4] M. Zafer and E. Modiano, "Optimal rate control for delay-constrained data transmission over a wireless channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4020–4039, Sep. 2008.

[5] I. Ahmed, K. T. Phan, and T. Le-Ngoc, "Optimal stochastic power control for energy harvesting systems with delay constraints," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3512–3527, Dec. 2016.

[6] A. Sinha and P. Chaporkar, "Optimal power allocation for a renewable energy source," in *Proc. IEEE Nat. Conf. Commun.*, Kharagpur, India, 2012, pp. 1–5.

[7] S. Mao, M. H. Cheung, and V. W. S. Wong, "Joint energy allocation for sensing and transmission in rechargeable wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 6, pp. 2862–2875, Jul. 2014.

[8] M. Kashef and A. Ephremides, "Optimal packet scheduling for energy harvesting sources on time varying wireless channels," *J. Commun. Netw.*, vol. 14, no. 2, pp. 121–129, 2012.

[9] D. Shaviv and A. Özgür, "Online power control for block i.i.d. energy harvesting channels," *IEEE Trans. Inf. Theory*, vol. 64, no. 8, pp. 5920–5937, Aug. 2018.

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

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

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

[13] I. Fawaz, M. Sarkiss, and P. Ciblat, "Optimal resource scheduling for energy harvesting communications under strict delay constraint," in *Proc. IEEE Int. Conf. Commun.*, Kansas City, MO, USA, May 2018, pp. 1–6.

[14] N. Sharma, N. Mastronarde, and J. Chakareski, "Structural properties of optimal transmission policies for delay-sensitive energy harvesting wireless sensors," in *Proc. IEEE Int. Conf. Commun.*, Kansas City, MO, USA, May 2018, pp. 1–7.

[15] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* New York, NY, USA: Wiley, 2014.

[16] B. Sayedana and A. Mahajan. (2020). *Counterexamples on the Monotonicity of Delay Optimal Transmission Policies in Energy Harvesting Communication Systems [Source Code].* [Online]. Available: https://doi.org/10.24433/CO.9177