Optimal Scheduling for Broadcast Erasure Channels with Energy Harvesting Receivers

Navid Reyhanian*, Behrouz Maham*†, and Chau Yuen‡

*School of Electrical and Computer Engineering, University of Tehran, Tehran, Iran
†School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran
‡Singapore University of Technology and Design, Singapore
Email: n.reyhanian@ut.ac.ir, bmaham@ut.ac.ir, yuenchau@sutd.edu.sg

***Abstract*—In this paper, an optimal scheduling for broadcasting packets to two receivers over erasure channels with feedback is studied. We propose a probabilistic algorithm for packet broadcasting to two receivers, and it is demonstrated that the algorithm is capacity achieving. The probabilistic algorithm is a feedback-based network coding algorithm. By using the probabilistic broadcasting algorithm, we formulate the problem of maximizing the weighted sum of energy harvesting receivers throughputs for any desired number of channel uses. We consider that the harvesting rate of each receiver changes during time slots and is known prior to transmissions. We optimize number broadcasted packets and charging time in order to maximize the weighted sum of throughputs, and then, a packet broadcasting policy is proposed.**

Keywords: Energy harvesting, broadcasting with feedback, erasure channel, receivers throughputs, packet scheduling.

I. INTRODUCTION

Network coding is a technique to increase achievable throughputs of communications networks [1]. The linear network coding has been recently used to achieve the capacity region of a few broadcast Packet Erasure Channels (PEC) with feedback [2], [3]. The capacity region of two-receiver broadcast PECs with feedback and memory is characterized in [4] where the channel state is visible to the source. In contrast to previous works that process data queues sequentially, a probabilistic method is used in [4] to process data queues. Networks with renewable energy supply have attracted lots of attention recently. Renewable energy supply is a way to reduce using up of fossil energy resources and it leads to reduction of green house gases emissions. As a green technology, utilizing transmitter and receivers with energy harvesting capabilities is a solution to reduce non-renewable energy consumption [5], [6], [7], [8]. Energy harvesting receivers are considered in [5] where channel state information is used in order to find an adaptive energy beamforming to supply energy to receivers. The ergodic sum-rate maximization problem is analyzed in [6] by designing the appropriate time slot allocation strategy, covariance matrix of the transmitted energy signal, and covariance matrix of the transmitted information signal at each user. In [7], the transmission time minimization problem in a N users additive white Gaussian noise (AWGN) broadcast channel is studied. The source that broadcasts bits to users is equipped with energy harvesting module and it is assumed that harvested energy is known prior to the transmissions. Broadcasting over energy harvesting nodes is investigated in [8]. Channels between the source and nodes are PECs. The source broadcasts common messages to all nodes and the forward error correction method is used to increase the reliability. They address a trade-off between reliability and throughput, and they propose broadcasting policies. Although they do not use feedback, sending feedbacks from receivers enables the source to track the broadcasted packets. Moreover, it can used for queue and delay management [9], [10].

In this paper, broadcasting packets to two receivers by a source is studied. The channels among the source and receivers are memoryless PECs. We introduce a probabilistic algorithm with feedback for broadcasting packets of receivers and it is demonstrated that the algorithm is capacity achieving. The probabilistic algorithm is a feedback-based network coding algorithm. We use the probabilistic algorithm to broadcast packets to two energy harvesting receivers. We consider that harvesting rate of each receiver changes during time slots and is known prior to the transmissions. According to the number of received packets by each receiver and charging time, receivers throughputs are calculated. The problem of optimizing the number broadcasted packets to receivers and charging time to maximize weighted sum of throughputs for any desired number of channel uses is formulated. Finally, a packet broadcasting policy is proposed.

II. SYSTEM MODEL

Consider one source broadcasts two different messages to two independent receivers. The set of messages belongs to receiver i (Rx_i) is $\mathcal{W}_i$ with 2^{nR_i} packets denoted by W_i, where n and R_i are the number of channel uses and rate of Rx_i, respectively. The source broadcasts a packet to receivers in each channel use. The channel from the source to Rx_i is memoryless. A broadcasted packet is either received at each receiver or erased. The erasure probability of the channel among the source and Rx_i is ϵ_i. Moreover, both channels are erased simultaneously by probability ϵ_{12}. The time interval between channel uses is called broadcasting period. At the end of each broadcasting period, receivers send ACKs if the broadcasted packet is not erased at (reached to) them, and send NACKs when the packet is erased (not reached). Sending an ACK or NACK are considered to be feedback signals. The

This research is partly supported by the Singapore University Technology and Design (grant No. SUTD-ZJU/RES/02/2011). In addition, this work was in part supported by a grant No. CS1393-4-29 from IPM.

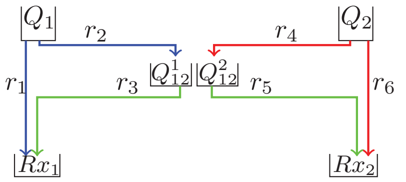


Fig. 1. Networked system of queues. A data queue is chosen by the probability assigned to it. The broadcasted packet from a queue can move on a link according to feedbacks. Flows of packets are listed in Table I.

feedback is considered to be error free. Before transmission, message packets of Rx_i are stored in $\mathcal{Q}_i$, $\forall i \in \{1, 2\}$. For throughput maximizations, receivers are equipped with energy harvesting modules. We assume that the harvesting rates of receivers are different in each time slot. Harvesting rates during time slots are known prior to broadcasting. Energy consumption of energy harvesting receivers has been investigated in [8]. Energy consumption of Rx_i to perform its basic operations like reading packets headers and storing packets reached to Rx_i in each time slot is denoted by E_{M_i}. The amount of lost energy due to battery leakage per time slot in Rx_i is E_{L_i}. The consumed energy to receive a packet by the Rx_i is E_{R_i}. In addition to these energy consumptions, in our system, Rx_i consumes E_{F_i} to send an ACK (NACK) to the source.

III. PROBABILISTIC ALGORITHM FOR BROADCASTING TO TWO RECEIVERS

We use the data queues introduced and used in [2]. When a packet of $\mathcal{Q}_i$, $\forall i \in \{1, 2\}$ is broadcasted and received by the Rx_i, it is removed from the $\mathcal{Q}_i$ and a new received packet is counted by Rx_i. In the networked system of queues given in Fig. 1, removing a packet from the $\mathcal{Q}_i$ and adding it to Rx_i corresponds to moving the packet on link 1 or 6, depending on which queue is chosen for broadcast. A packet is called innovative for Rx_i if a new message packet of that receiver can be decoded from it which is not observed already. If the broadcasted packet from $\mathcal{Q}_i$ is not received to any receiver, it remains in $\mathcal{Q}_i$ and it waits for following packet broadcastings. In the events that the broadcasted packet from $\mathcal{Q}_i$ is not received at Rx_i and it is received at the other receiver, the packet is removed from $\mathcal{Q}_i$ and it is added to $\mathcal{Q}_{12}^i$, which is located at the data queue $\mathcal{Q}_{12}$. In these events, the broadcasted packet move on link 2 or 4, depending on which queue is chosen for broadcast. The data queue $\mathcal{Q}_{12}$ stores misreceived packets of receivers separately in its two queues. Broadcasted packets from $\mathcal{Q}_{12}$ are XORed of misreceived packets stored in $\mathcal{Q}_{12}^1$ and $\mathcal{Q}_{12}^2$. Since Rx_i has already received one of the used packets for making an XORed packet, the XORed packet can be decoded by it. Each receiver obtains one innovative packet from the XORed packet. Therefore, the XORed packet is innovative for both receivers and by using it, the source sends an innovative packet for each receiver in one channel use that increases transmission rates. It is demonstrated in [2] that choosing one packet from each of data queues, $\mathcal{Q}_{12}^1$ and $\mathcal{Q}_{12}^2$, and XORing them is good enough coding to achieve the capacity of memoryless two-receiver broadcast erasure channels with feedback. We use coded packet instead of XORed packet in the rest of the paper. If a coded packet broadcasted from $\mathcal{Q}_{12}$ is received by Rx_i, the used packet for making it chosen from $\mathcal{Q}_{12}^i$ is removed and a new innovative packet is counted. Otherwise, it remains in $\mathcal{Q}_{12}^i$. Packet reception of Rx_i from $\mathcal{Q}_{12}^i$ corresponds to moving the packet on link 3 or 5. Packet movements on links can be seen as data flow during channel uses. The summary of packets movements based on channels realizations are given in Table I. Broadcasted packets from $\mathcal{Q}_i$ and $\mathcal{Q}_{12}$ are innovative for Rx_i. Since the broadcasted packet is chosen from one of the three data queues, the queue that the packet is chosen from can be specified by two bits of the header of the packet. To inform receivers whether the broadcasted packet is innovative for them or not, two bits of the header of the broadcasted packet are used. The third bit of the header of the broadcasted packet to Rx_i determines whether the last correctly received packet by the other receiver has been erased at Rx_i.

To show three data queues, we use $\mathcal{Q}_U$, where $U \in \{1, 2, 12\}$. At each channel use, the output of a data queue $\mathcal{Q}_U$ is selected to broadcast randomly according to the known probability assigned to the each data queue. Since packets are not copied and the queues store innovative packets of receivers separately, all broadcasted packets are innovative for their destinations. As it is seen from Fig. 1, each receiver receives innovative packets from two links. Innovative packets flow for Rx_1 that enters the data queue $\mathcal{Q}_{12}^1$ is equal to the flow leaves data queue $\mathcal{Q}_{12}^1$ due to the fact that $\mathcal{Q}_{12}^1$ is not a sink. With the same argument, the flow enters the data queue $\mathcal{Q}_{12}^2$ is equal to the flow leaves data queue $\mathcal{Q}_{12}^2$. Assume that link k connects the data queue $\mathcal{Q}_i$, $\forall i \in \{1, 2\}$ to $\mathcal{Q}_{12}$. The capacity of link k depends on the probability assigned to the data queue $\mathcal{Q}_i$. Moreover, it depends on the probability of the event that a broadcasted packet from $\mathcal{Q}_i$ is not received by Rx_i and it is received by the other receiver. We denoted this probability by η_k. The capacity of link k is $C_k = P_i \eta_k$ where P_i is the probability by which $\mathcal{Q}_i$ is chosen for broadcast. The passing flow on the link k reaches to the link capacity if packet movements are done for all broadcasted packets from the queue $\mathcal{Q}_i$ that are not received by Rx_i. However, are received by the other receiver. The capacities of links that connect data queues to receivers can be obtained with the same logic. For these links, η_k is the probability of the event that the broadcasted packet from the data queue $\mathcal{Q}_U$ is received by the receiver at end of the link k. Moreover, P_U denotes the probability assigned to $\mathcal{Q}_U$ and the capacity is $C_k = P_U \eta_k$. Links capacities is given in Table I. In the following proposition, we demonstrate that for any rate on the capacity, there exists a set of probabilities assigned to queues by which the probabilistic algorithm achieves the rate.

Proposition 1: The probabilistic algorithm for packet broadcasting to two receivers with feedback is capacity achieving.

Proof: The proof is given in Appendix A. ■

IV. OPTIMAL SCHEDULING FOR BROADCASTING PACKETS OF TWO ENERGY HARVESTING RECEIVERS

In this section, we assume that receivers are supplied via green energy, harvested from environment. Receivers consume energy while they receive packets and send ACKs or NACKs.

TABLE I
INSTRUCTIONS FOR ROUTING BROADCASTED PACKETS

Description of flows		
Flow	Corresponding event	The link capacity
r_1	The broadcasted packet from $\mathcal{Q}_1$ is reached to Rx$_1$.	$(1-\epsilon_1)P_1$
r_2	The broadcasted packet from $\mathcal{Q}_1$ is not reached to Rx$_1$, and reached to Rx$_2$.	$(\epsilon_1-\epsilon_{12})P_1$
r_3	The broadcasted packet from $\mathcal{Q}_{12}$ is reached to Rx$_1$.	$(1-\epsilon_1)P_{12}$
r_4	The broadcasted packet from $\mathcal{Q}_2$ is not reached to Rx$_2$, and reached to Rx$_1$.	$(\epsilon_2-\epsilon_{12})P_2$
r_5	The broadcasted packet from $\mathcal{Q}_{12}$ is reached to Rx$_2$.	$(1-\epsilon_2)P_{12}$
r_6	The broadcasted packet from $\mathcal{Q}_2$ is reached to Rx$_2$.	$(1-\epsilon_2)P_2$

To reduce energy consumption, keeping silence by receivers at the end of channel use is considered as a NACK signal. In contrast to the previous section, the energy management is very important here. Consider that the channel of Rx$_i$ is not erased, but the receiver does not have energy to receive the broadcasted packet. In this case, a number of channel uses are wasted. In the other case, consider that a receiver has received the broadcasted packet, but it does not have sufficient energy to send an ACK and it keeps silent. In this case, similarly, a number of channel uses are wasted due to the fact that when the source does not receive an ACK from a receiver, it considers that the broadcasted packet is erased at that receiver and updates its data queues accordingly. Without loss of generality, we assume that the consumed time for broadcasting and signal transmission is negligible compared to the time needed for harvesting energy. In this section, slotted time is used to measure needed time for charging batteries. Six bits are added to previous header bits to perform the flow control. Their duty is explained in following subsections. As the receivers are equipped with energy harvesting modules and the available energy depends on the harvested energy, there are three reasons for not sending an ACK signal by Rx$_i$

1) The broadcasted packet is erased at Rx$_i$.
2) Rx$_i$ does not have sufficient energy to receive the broadcasted packet.
3) Rx$_i$ does not have sufficient energy to send an ACK signal.

Although the first reason is out of the source and receivers control, the broadcasting scheme can be designed such that the packet reception failure due to the second and third reasons is prevented. To ensure that both receivers have enough energy to receive the broadcasted packet and send an ACK, the source waits a certain number of time slots before broadcasting. We name this period as charging time. In this section, we use the proposed probabilistic capacity achieving algorithm to maximize throughputs. The throughput of Rx$_i$ can be calculated as follows

$$\mathrm{Th_i} = \frac{|\text{Innovative packets of Rx}_i \text{ that it ACKs and receives}|}{\text{Charging time}}, \quad (1)$$

we denote charging time of Rx$_i$ by N_i.

A. Receivers throughputs maximization

In this subsection, we propose an optimization-based method to adjust links flows and determine needed charging time for ψ channel uses such that the weighted sum of receivers throughputs is maximized. By adjusting link flows, we find the maximum number of packets that receivers can receive in ψ channel uses. Rx$_i$ harvests energy with rate $\rho_i(t)$ Joule per second in time slot t. It harvests $\rho_i(t)T$ Joule in time slot t where T is the time slot duration. Harvesting rates profiles are known prior to channel uses. Since available energy in Rx$_i$ is limited and to reduce energy consumption, we develop a scheme in which Rx$_i$ does not consume energy to receive all the reached broadcasted packets. Packets headers inform Rx$_i$ whether to receive reached packets or not. However, both receivers ACK reached packets to them. Remind that to enable the source to send coded packets, Rx$_i$ has to receive broadcasted packets from $\mathcal{Q}_j$ that are erased at their destination, Rx$_j$, where $j \neq i$ and $i, j \in \{1, 2\}$. However, it is not necessary for Rx$_i$ to receive broadcasted packets from $\mathcal{Q}_j$ that are not erased at their destination. Therefore, to apprehend that the reached packet from $\mathcal{Q}_j$ is erased at Rx$_j$ or not, Rx$_i$ stores the reached packet and waits for the header of the next reached packet. Based on the feedbacks, when the source finds that the broadcasted packet from $\mathcal{Q}_i$ is not erased at Rx$_i$ and Rx$_j$, it informs Rx$_j$ to discard the stored packet by the third bit of the header of the next packet reached to Rx$_j$. Corresponding flows to these events are listed in Table II and depicted in Fig. 2. Discarding packets that are not erased at their destinations is a policy for reducing energy consumption of packets reception. If the broadcasted packet from $\mathcal{Q}_1$ is erased at Rx$_1$ and reached to Rx$_2$, by using the forth bit of the header of the next packet that reaches to Rx$_2$, the source informs Rx$_2$ whether or not to receive previous reached packet. By using the following optimization, we find the expected number of the broadcasted packet from $\mathcal{Q}_j$ that Rx$_i$ has to receive. The flow of packets of $\mathcal{Q}_1$ that are received and ACKed by Rx$_2$ is r_2. The flow of packets of $\mathcal{Q}_2$ that are erased at Rx$_2$, and received and ACKed by Rx$_1$ is r_4.

The expected number of broadcasted packets from $\mathcal{Q}_1$ reached to Rx$_1$ is $Z_1(1-\epsilon_1)$ where Z_1 is the number of times $\mathcal{Q}_1$ is chosen in ψ channel uses. The expected number of broadcasted packets from $\mathcal{Q}_{12}$ reached to Rx$_1$ in ψ channel uses is $Z_{12}(1-\epsilon_1)$ where Z_{12} is the number of times $\mathcal{Q}_{12}$ is chosen in ψ channel uses. In addition, the expected number of broadcasted packets from $\mathcal{Q}_2$ reached to Rx$_1$ is $Z_2(1-\epsilon_1)$. The amount of $E_{F_1}Z_1(1-\epsilon_1)$ Joule is consumed by Rx$_1$ to ACK the expected broadcasted packets from $\mathcal{Q}_1$ reached to Rx$_1$. Moreover, $E_{F_2}Z_1(1-\epsilon_2)$ Joule is consumed by Rx$_2$ to ACK the expected broadcasted packets from $\mathcal{Q}_1$. Similarly, $E_{F_1}Z_2(1-\epsilon_1)$ Joule is consumed by Rx$_1$ to ACK the expected broadcasted packets from $\mathcal{Q}_2$ where Z_2 is the number of times $\mathcal{Q}_2$ is chosen in ψ channel uses. In the same way, $E_{F_i}Z_{12}(1-\epsilon_i)$ Joule is consumed by Rx$_i$ to ACK the expected broadcasted packets from $\mathcal{Q}_{12}$. The consumed energy to ACK

TABLE II
FLOWS THAT RECEIVERS DOES NOT CONSUME ENERGY TO RECEIVE THEM

Description of flows	
Flow	Corresponding event
r_7	The broadcasted packets from $\mathcal{Q}_2$ are reached to both receivers. They are not received by Rx_1.
r_8	The broadcasted packets from $\mathcal{Q}_1$ are reached to both receivers. They are not received by Rx_2.

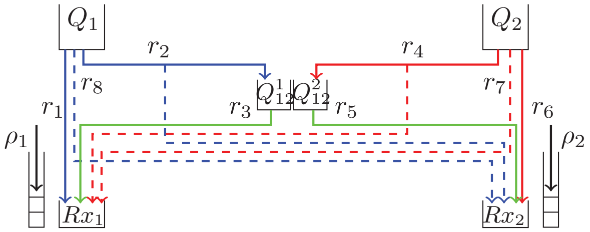


Fig. 2. Networked system of queues. Each dashed link that enters Rx_i, shows the flow of other receiver packets that are not innovative for Rx_i. The flow r_4 enables coding and the Rx_1 does not consume energy to receive r_7. The same argument holds for r_2 and r_8, respectively.

packets by Rx_1 in ψ channel uses is $E_{F_1}(Z_1+Z_2+Z_{12})(1-\epsilon_1)$ Joule and by Rx_2 is $E_{F_2}(Z_1+Z_2+Z_{12})(1-\epsilon_2)$ Joule.

The flows of broadcasted packets from $\mathcal{Q}_1$ and $\mathcal{Q}_2$ that are reached correctly to both receivers are r_8 and r_7, respectively. By using the header bits, Rx_1 does not consume energy to receive r_7. Rx_2 does not consume energy to receive r_8. Flows r_2 and r_4 enable sending coded packets. The total flow that Rx_1 has to receive is $R_1 = r_1 + r_3$ plus the flow of the other receiver packets which is $R'_1 = r_4$. The total flow that Rx_2 has to receive is $R_2 = r_5 + r_6$ plus the flow of the other receiver packets which is $R'_2 = r_2$. Consumed energy by Rx_i to receive flows is $E_{R_i}(R_i + R'_i)$. Regarding above arguments, the consumed energy by Rx_1 is $(E_{M_1} + E_{L_1})N_1 + E_{F_1}(Z_1 + Z_2 + Z_{12})(1-\epsilon_1) + E_{R_1}(R_1 + R'_1)$ Joule. By reversing places of Rx_1 and Rx_2, the consumed energy by Rx_2 is found as $(E_{M_2}+E_{L_2})N_2+E_{F_2}(Z_1+Z_2+Z_{12})(1-\epsilon_2)+E_{R_2}(R_2+R'_2)$ Joule. As the amount of harvested energy by Rx_i is $\rho_i(t)T$ in time slot t, the least number of time slots that are needed to harvest enough energy to perform Rx_i operations is obtained by numerical methods form the following equation

$$\begin{aligned}&(E_{M_i} + E_{L_i})N^\star_i + E_{F_i}(Z_1 + Z_2 + Z_{12})(1-\epsilon_i) + E_{R_i}(R_i\\ &\quad +R'_i) = T\sum_{t=1}^{\lfloor N^\star_i \rfloor} \rho_i(t) + T\rho_i(\lceil N^\star_i \rceil)(N^\star_i - \lfloor N^\star_i \rfloor),\end{aligned} \tag{2}$$

since the number of time slots needed to harvest enough energy can be non-integer and the harvested energy in each time slot is different, the amount of needed harvested energy is written as the right hand side of the (2). When the charging time is less than $N^\star_i$, the Rx_i can not harvest enough energy to receive $R_i + R'_i$ and ACK reached packets. Therefore, the source sets the charging time equal or more than $N^\star_i$. Increasing the charging time increases the harvested energy, but reduces the throughput of the receivers. To ensure that the receivers can receive and ACK packets, and receivers throughputs is not reduced due to the length of the charging period, the source sets $N = \max\{N^\star_1, N^\star_2\}$.

According to (1), the throughput of Rx_i is $\frac{R_i}{N}$. Now that charging time is found, we optimize flows, actions probabilities and charging time to maximize the weighted sum of receivers throughputs, $\mu_1\text{Th}_1 + \mu_2\text{Th}_2$, for all positive constants $\mu_1, \mu_2 \geq 0$ as follows

$$\begin{aligned}\max\ & \mu_1\frac{R_1}{N} + \mu_2\frac{R_2}{N}\\ \text{s.t. } & N = \max\{N^\star_1, N^\star_2\},\\ & (E_{M_1} + E_{L_1})N_1 + E_{F_1}(Z_1 + Z_2 + Z_{12})(1-\epsilon_1) + E_{R_1}\\ & (R_1 + R'_1) = T\sum_{t=1}^{\lfloor N^\star_1 \rfloor} \rho_1(t) + T\rho_1(\lceil N^\star_1 \rceil)(N^\star_1 - \lfloor N^\star_1 \rfloor),\\ & (E_{M_2} + E_{L_2})N_2 + E_{F_2}(Z_1 + Z_2 + Z_{12})(1-\epsilon_2) + E_{R_2}\\ & (R_2 + R'_2) = T\sum_{t=1}^{\lfloor N^\star_2 \rfloor} \rho_2(t) + T\rho_2(\lceil N^\star_2 \rceil)(N^\star_2 - \lfloor N^\star_2 \rfloor),\\ & C_1 = Z_1(1-\epsilon_1), C_2 = Z_1(\epsilon_1 - \epsilon_{12}),\\ & C_3 = Z_{12}(1-\epsilon_1), C_4 = Z_2(\epsilon_2 - \epsilon_{12}),\\ & C_5 = Z_{12}(1-\epsilon_2), C_6 = Z_2(1-\epsilon_2),\\ & 0 \leq r_i \leq C_i,\ \forall i \in \{1, ..., 6\},\\ & r_2 = r_3, r_4 = r_5,\\ & 0 \leq Z_U \leq \psi,\ \forall U \in \{1, 2, 12\},\\ & Z_1 + Z_2 + Z_{12} = \psi, R_1 = r_1 + r_3, R_2 = r_5 + r_6,\\ & R'_1 = r_4, R'_2 = r_2.\end{aligned} \tag{3}$$

where the constraint $0 \leq r_i \leq C_i$ follows from the fact that links flows are positive and below links capacities. The flow observation is captured in the constraint $r_2 = r_3$, $r_4 = r_5$. In other words, the number of innovative packets of Rx_i that enter the data queue $\mathcal{Q}^i_{12}$ is equal to the number of innovative coded packets broadcasted from the data queue Q_{12} to Rx_i. This maximization states a trade-off between the expected number of received packets by receivers and the charging time. The above optimization is non-convex and it can be solved via numerical methods. After optimizing flows of links, it is seen that passing flows on some links do not reach links capacities. When $r^\star_1$ does not achieve C_1, Rx_1 receives $\frac{r^\star_1}{C_1} \times 100$ percent of packets broadcasted from $\mathcal{Q}_1$ reached to Rx_1. The fifth bit of the header is used by the source to inform Rx_1 whether to receive reached packets from $\mathcal{Q}_1$ or not. The fifth bits of the headers of packets broadcasted from $\mathcal{Q}_1$ to Rx_1 are such that $\frac{r^\star_1}{C_1}$ of them has to be received by Rx_1 if they reach it. As Rx_1 receives a broadcasted packet from $\mathcal{Q}_1$, the packet is removed from $\mathcal{Q}_1$ and added to Rx_i. Otherwise, it remains in $\mathcal{Q}_1$. In $\frac{r^\star_2}{C_2}$ of packets broadcasted from $\mathcal{Q}_1$ the forth bits of headers are set such that if they are erased at Rx_1 and reach to Rx_2, they are received by Rx_2. When Rx_2 receives a broadcasted packet from $\mathcal{Q}_1$, it is removed from $\mathcal{Q}_1$, and added to $\mathcal{Q}^1_{12}$ and Rx_2. Otherwise, it remains in $\mathcal{Q}_1$. This is the reason that the flow r_2 enters $\mathcal{Q}^1_{12}$ and Rx_2, simultaneously. In $\frac{r^\star_3}{C_3}$ of packets broadcasted from $\mathcal{Q}_{12}$ the sixth bits of headers are set such that

TABLE III
HEADER CONSTRUCTION OF BROADCASTED PACKETS TO Rx_i

Duty description of header bits of the packets broadcasted to Rx_i	
Bit number	Duty
$1, 2$	Determining the data queue that the packet is broadcasted from.
3	Whether the last reached packet to the other receiver has been erased at Rx_i or not.
4	Whether the broadcasted packet from $\mathcal{Q}_1$, erased at Rx_1, be added to $\mathcal{Q}_{12}^1$ and be received by Rx_2 or not.
5	Whether the broadcasted packet from $\mathcal{Q}_1$ reached to Rx_1, be received by it or not.
6	Whether the broadcasted packet from $\mathcal{Q}_{12}$ reached to Rx_1, be received by it or not.
7	Whether the broadcasted packet from $\mathcal{Q}_2$ erased at Rx_2, be added to $\mathcal{Q}_{12}^2$ and be received by Rx_1 or not.
8	Whether the broadcasted packet from $\mathcal{Q}_{12}$ reached to Rx_2, be received by it or not.
9	Whether the broadcasted packet from $\mathcal{Q}_2$ reached to Rx_2, be received by it or not.

TABLE IV
HEADER BITS USED FOR CONTROLLING LINKS FLOWS

Bit number	4	5	6	7	8	9
Flow number	r_2	r_1	r_3	r_4	r_5	r_6

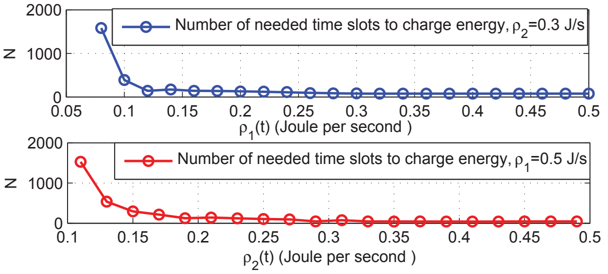


Fig. 3. The needed time slots for charging batteries for 100 channel uses as harvesting rate of one of receivers is constant and the other changes.

if they reach to Rx_1, they are received by Rx_1. As Rx_1 receives a broadcasted packet from $\mathcal{Q}_{12}$, the packet stored in $\mathcal{Q}_{12}^1$ which is used to make a coded packet is removed and added to Rx_1. Otherwise, it remains in $\mathcal{Q}_{12}^1$. For the other links, the same argument holds. Other bits of header are used in the same way by the source to adjust the number of received packets during broadcasting. Summary of header bits duties is given in Table III. The broadcasting policy to achieve the maximized throughputs weighted sum is explained in subsection IV-B based on solution of the proposed optimization. Consider that we want to make the weighted sum of receivers throughputs maximum in ψ channel uses. The needed battery capacities for ψ channel uses are $T \sum_{t=1}^{\lceil N^\star \rceil} \rho_1(t)$ for Rx_1 and $T \sum_{t=1}^{\lceil N^\star \rceil} \rho_2(t)$ for Rx_2, respectively. It is observed that when the harvesting rates are constant, one still can use the proposed optimization for variable harvesting rates as well. In the future work we show that the probabilistic approach can be extended for broadcasting over three receivers.

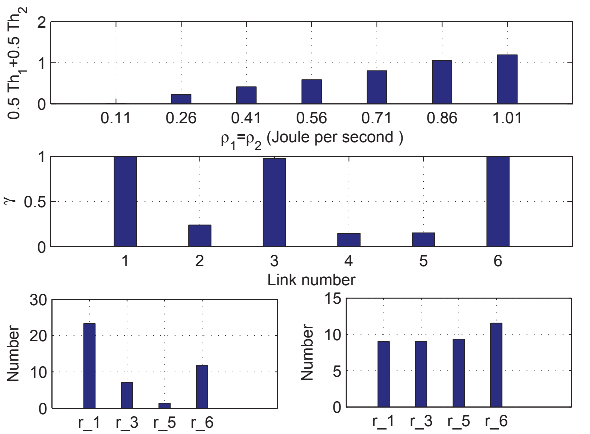
B. Policy for broadcasting packets of receivers

Suppose that the throughputs maximizations with fixed and variable energy harvesting rates are solved. A broadcasting policy is proposed for each case of throughputs maximization, based on the probabilistic algorithm. Consider that the optimal flow on link k is denoted by $r_k^\star$. We define γ_k for link k such that $\gamma_k = \frac{r_k^\star}{C_k}$. Since flows of all links do not always achieve links capacities, the source adjusts headers bits to modify flows enter receivers and Q_{12}. In $\frac{r_1^\star}{C_1}$ of the packets that are broadcasted from $\mathcal{Q}_1$, the fifth headers bits are set such that if packets are reached to Rx_1, they are received as headers are read. Moreover, in $\frac{r_2^\star}{C_2}$ of the packets that are broadcasted from $\mathcal{Q}_1$, the fourth headers bits are set such that if packets are reached to Rx_2 and erased at Rx_1, they are received by Rx_2, and simultaneously, removed from $\mathcal{Q}_1$ and added to $\mathcal{Q}_{12}^1$. The source sets sixth bits of headers of $\frac{r_3^\star}{C_3}$ of the packets broadcasted from $\mathcal{Q}_{12}$ such that if the packets are reached to Rx_1, they are received as headers are read. The same arguments hold for broadcasted packets from data queues $\mathcal{Q}_2$. To adjust the number of received packets by receivers, the source uses solution of optimization problem stated in (3) and seventh to ninth bits of headers.

As the receiver reads the first and second bits of the reached packet header, it finds the link that the packet is coming from. The receiver or the data queue at the end of the link k, receives a passing packet on link k according to the corresponding bit to link k in header of the broadcasted packet. Bits and flows correspondence is given in Table IV. If the corresponding bit to the link k states that the packet has to be received, reception is done and the packet is removed from the data queue that it is broadcasted from. Otherwise, it remains in the data queue chosen for the broadcast. Consequently, either the packet remains in the data queue chosen for broadcast or it is received by the receiver or data queue at the end of link k. As γ_k is found for each link after optimizing flows and actions, header bits of broadcasted packets from data queues are set according to γ_k. The source makes the receiver at the end of link k to receive $\gamma_k \times 100$ percent of packets reached to it from link k by using a bit of the header. The output of data queue $\mathcal{Q}_1$ is chosen Z_1 times, Z_2 times the output of data queue $\mathcal{Q}_2$ and Z_{12} times the output of data queue $\mathcal{Q}_{12}$ are chosen. Broadcasted packets are routed and received according to the received feedbacks and headers bits.

V. SIMULATION RESULTS

In this section, the performances of the proposed throughput maximizations are evaluated through simulations. The amount of consumed energy due to different operations in Rx_1 are $EM_1 = 0.05$J, $EL_1 = 0.02$J, $ER_1 = 0.35$J, and $EF_1 = 0.15$J. The amount of consumed energy due to different operations in



Received packets from links during 100 time slots with fixed harvesting rate ρ$_1$(t)=ρ$_2$(t)=1.01 J/s

Received packets from links during 90 time slots with variable harvesting rate

Fig. 4. The effect of harvesting rates on $0.5\mathrm{Th}_1 + 0.5\mathrm{Th}_2$ maximization for 100 channel uses, the ratio of flow passing on each link to its capacity, the number of packets that can be received via different links for fixed harvesting rate case and variable harvesting rate case are shown, respectively.

Rx$_2$ are $EM_2 = 0.07$ J, $EL_2 = 0.03$ J, $ER_2 = 0.45$ J, and $EF_2 = 0.15$ J. Time slots duration is one second. Channels erasures probabilities are $\epsilon_1 = 0.6$, $\epsilon_2 = 0.5$ and $\epsilon_{12} = 0.1$. In Fig. 3, the effect of harvesting rates of receivers on the charging period for 100 channel uses is investigated. In the first plot of Fig. 3, $\rho_2(t) = 0.5$J/s and $\rho_1(t)$ changes from 0.08J/s to 0.5J/s. In the second plot of Fig. 3, $\rho_1(t) = 0.5$ J/s and $\rho_2(t)$ changes from 0.11 J/s to 0.5 J/s. It is seen that as the harvesting rates of receivers increase, the number of needed time slots to charge batteries reduces.

As the harvesting rates increase, the charging time reduces. When the charging time reduces, the throughput of both receivers are increased. The effect of the harvesting rates on $0.5\mathrm{Th}_1 + 0.5\mathrm{Th}_2$ maximization for 100 channel uses is depicted in the first plot of Fig. 4. The second plot of Fig. 4 shows the ratio of flow passing on each link to its capacity, γ_k, when $\rho_1(t) = \rho_2(t) = 1.01$ J/s. Using these probabilities the broadcasting is done and the weighted sum of throughputs is maximized. In the third plot of Fig. 4, the maximum number of packets that can be received via different links is depicted. The forth plot of Fig. 4 shows maximum number of packets that can be received during 90 time slots via different links when the receiver harvest energy with variable rate in each time slot. The energy harvesting rates in time slots are random and follow a correlated normal distribution with mean 0.5 J/s.

ACKNOWLEDGMENT

The authors would like to thank Dr. Shirin Saeedi Bidokhti from Technische Universität München for her helpful comments.

APPENDIX A
OPTIMALITY OF THE PROBABILISTIC ALGORITHM

In this subsection, we show that the proposed algorithm achieves the capacity region characterized in [2]. The upper bound for achievable rate region for two-user packet erasure broadcast channel with feedback and memory is given in [4]. We simplify the upper bound given in [4] to the memoryless case as follows

$$0 \le x \le 1, \quad 0 \le y \le 1, \tag{4}$$
$$x + y \ge 1, \tag{5}$$
$$R_1 \le (1-\epsilon_1)x, \tag{6}$$
$$R_1 \le (1-\epsilon_{12})(1-y), \tag{7}$$
$$R_2 \le (1-\epsilon_2)y, \tag{8}$$
$$R_2 \le (1-\epsilon_{12})(1-x), \tag{9}$$

We remind the min-cut definition, and then, we bound the rates of receivers by applying the probabilistic algorithm.

Definition 1: Min-cut is a set of links that by cutting them, the source and Rx$_i$ are separated and sum of the links capacities in the min-cut is minimal.

Since links capacities depend on actions probabilities and channel erasure probabilities, we can not compare capacities in general. We consider both cuts on flow of each receiver. Cuts on flow of Rx$_1$ are $\{1,2\}$ and $\{1,3\}$. Cuts on flow of Rx$_2$ are $\{4,6\}$ and $\{5,6\}$. Consequently, rates of receivers are bounded as follows

$$R_1 \le P_1(1-\epsilon_1) + P_1(\epsilon_1 - \epsilon_{12}), \tag{10}$$
$$R_1 \le P_1(1-\epsilon_1) + P_{12}(1-\epsilon_1), \tag{11}$$
$$R_2 \le P_2(1-\epsilon_1) + P_2(\epsilon_2 - \epsilon_{12}), \tag{12}$$
$$R_2 \le P_2(1-\epsilon_1) + P_{12}(1-\epsilon_2). \tag{13}$$

Assume that $P_1 = 1-y$ and $P_2 = 1-x$. By substituting these values for P_1 and P_2 in (10), (11), (12) and (13), it is seen that inequalities (6), (7), (8) and (9) are yielded. As we have $y = 1-P_1$ and $y = 1-P_2$, (4) and (5) hold as probabilities are positive values between zero and one. It is observed that the achievable region and upper bound are the same. In other words, the probabilistic algorithm achieves the capacity region.

REFERENCES

[1] S.-Y. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inform. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[2] L. Georgiadis and L. Tassiulas, "Broadcast erasure channel with feedback-capacity and algorithms," in *Proc. 5th Workshop Netw. Cod. Theory Appl.*, pp. 54âĂŞ–61, June 2009.

[3] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Multiuser broadcast erasure channel with feedback-capacity and algorithms," *IEEE Trans. Inform. Theory*, vol. 59, no. 9, pp. 5779 – 5804, Sept. 2013.

[4] M. Heindlmaier, N. Reyhanian, and S. S. Bidokhti, "Capacity regions of two-receiver broadcast packet erasure channels with feedback and memory," *preprint arXiv:1410.0380*, 2014.

[5] X. Chen, C. Yuen, and Z. Zhang, "Wireless energy and information transfer tradeoff for limited-feedback multiantenna systems with energy beamforming," *IEEE Trans. Veh. Technol.*, vol. 63, no. 1, pp. 407–412, Jan. 2014.

[6] J. Zhang, C. Yuen, and C.-K. Wen, "Large-system analysis of ergodic sum-rate in wireless-powered mimo communication network," in *IEEE SECON Workshops*, pp. 57–61, June 2014.

[7] O. Ozel, J. Yang, and S. Ulukus, "Optimal broadcast scheduling for an energy harvesting rechargeable transmitter with a finite capacity battery," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 2193–2203, Apr. 2012.

[8] C. Kuan, G. Lin, H. Wei, and R. Vannithamby, "Reliable multicast and broadcast mechanisms for energy harvesting devices," *IEEE Trans. Veh. Technol.*, vol. 63, no. 4, pp. 1813 – 1826, Nov. 2014.

[9] X. Li, C.-C. Wang, and X. Lin, "Throughput and delay analysis on uncoded and coded wireless broadcast with hard deadline constraints," in *Proc. IEEE INFOCOM*, pp. 1–5, Mar. 2010.

[10] J. K. Sundararajan, D. Shah, and M. Médard, "ARQ for network coding," in *Proc. IEEE ISIT*, pp. 1651–1655, July 2008.