Supplementary Material

A. Proof of Lemma 1

It is easy to see that all received power in the charging phase must be used up in the sending phase in order to maximize throughput. Thus we have

$$p(T-\beta T)=\rho\beta T,$$
$$\beta=\frac{p}{\rho+p}.$$

Since the transmission rate is $\log(1+\rho)$ at transmission power ρ, the total throughput B can be calculated by $B=\beta T\log(1+\rho)=\frac{p}{p+\rho}T\log(1+\rho)$. The throughput is expressed as a function of ρ, we hence define the throughput function $B(\rho)$ as

$$B(\rho)=\frac{p}{p+\rho}T\log(1+\rho) \tag{3}$$

To find the maximum value of $B(\rho)$, we calculate its first and second order of derivatives as shown below.

$$\begin{aligned}B'(\rho)&=pT(\frac{\log(1+\rho)}{p+\rho})',\\ &=\frac{pT}{(p+\rho)^2}[(1+\frac{p-1}{1+\rho})\frac{1}{\ln 2}-\log(1+\rho)].\end{aligned}$$

Because $\frac{pT}{(p+\rho)^2}>0$, we define function $g(\rho)$ as

$$g(\rho)=B'(\rho)/\frac{pT}{(p+\rho)^2}\quad=(1+\frac{p-1}{1+\rho})\frac{1}{\ln 2}-\log(1+\rho).$$

Obviously, for any given ρ, $g(\rho)$ and $B'(\rho)$ share the same sign and zero point. Because

$$g'(\rho)=-\frac{p-1}{(1+\rho)^2\ln 2}-\frac{1}{(1+\rho)\ln 2}=-\frac{p+\rho}{(1+\rho)^2\ln 2}<0,$$

therefore, $g(\rho)$ monotonically decreases. Hence $B(\rho)$ is convex and its maximum value can be found at point ρ_s, such that $g(\rho_s)=0$. By setting $g(\rho_s)=0$ we have the following derivations:

$$\begin{aligned}(1+\frac{p-1}{1+\rho_s})\frac{1}{\ln 2}&=\log(1+\rho_s),\\ \mathsf{EXP}(1+\frac{p-1}{1+\rho_s})&=1+\rho_s,\\ \frac{p-1}{1+\rho_s}\mathsf{EXP}(\frac{p-1}{1+\rho_s})&=\frac{p-1}{e},\\ \mathcal{W}(\frac{p-1}{e})&=\frac{p-1}{1+\rho_s},\end{aligned}$$

where function $\mathcal{W}(z)$ is called the Lambert W function [24], which has the following property, $\mathcal{W}(z)\mathsf{EXP}(\mathcal{W}(z))=z$. Therefore, we have

$$\rho_s=\frac{p-1}{\mathcal{W}(\frac{p-1}{e})}-1. \tag{4}$$

When $B(\rho)$ is maximized at $\rho=\rho_s$, the corresponding ratio of the sending phase is $\beta=\beta_s=\frac{p}{\rho_s+p}$. Fig. 9 shows an example with $T=10$ and $p=20$ which shows maximum data throughput is obtained at $\rho_s\approx 11.5$. Using a smaller rate ρ_1 or a larger rate ρ_2 will not be optimal.

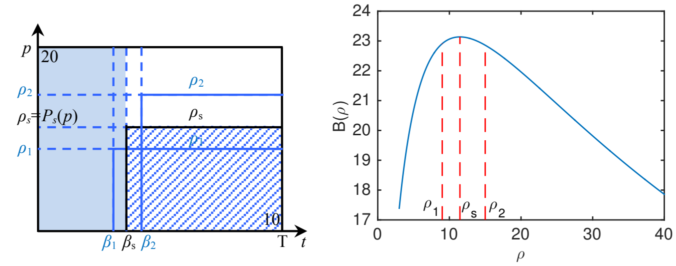

Fig. 9. The shape of function $B(\rho)$ when $T=10$ and $p=20$, which is maximized at $\rho=\rho_s\approx 11.5$ and the length of sending phase is $\beta_s T$.

B. Proof of lemma 2

For any given dividing line $\Omega=(w,b)$ in a $P_s(w,b)$-schedule, either $w=p_i$ for some i then $0\le b\le 1$, or $w\ne p_i$ for any i then $b=0$. See Fig. 4 for an example. In the former case, when the dividing line raises, w stays the same and b grows, *e.g.,* sending phase i increased and charging phase i decreased. In the latter case, when the dividing line raises, w grows and b stays the same, *e.g.,* sending power $P_s(w)$ in all sending phases grows. For both the cases, the energy used in sending increases while energy charged decreases, and hence the remaining energy $R(T)$ reduces monotonically.

C. Proof of Lemma 3

The algorithm works in iteration. E_c is the charging energy and E_s is the sending energy. Initially, we assume dividing line position $(0,0)$, hence E_c stores all energy and E_s stores no energy. Each iteration tests dividing line position $(p_i,0)$ to see whether the charging energy is greater than the sending energy. The testing is in the ascending order of p_i, *e.g.,* $i=T,T-1,\ldots 1$. In an iteration of the **for**, the dividing line raises from $(p_i,0)$ to $(p_{i-1},0)$. A total of $\Delta c=p_i$ energy is less charged and a total of $\Delta s=P_s(p_i)+(P_s(p_{i-1})-P_s(p_i))*l$ energy is increased in sending, where l is the number is sending slots, in which the sending power raises from $P_s(p_i)$ to $P_s(p_{i-1})$. Therefore, if $E_c-\Delta c\ge E_s+\Delta s$, which means the sending energy would be still more than the sending energy after the raise, the **for** continues. Eventually, it exit at **break** in line 6 before testing $(p_0,0)$, because $p_0=P_s^{-1}(\rho_{max})$ and according to the assumption that a full battery will be depleted if all slots transmit at the maximum power ρ_{max}. Suppose the **for** loop exits in iteration i, then $(p_i,0)\le(w_{opt},b_{opt})<(p_{i-1},0)$. Meanwhile, the current E_c-E_s is the remaining energy for $(p_i,0)$. There are two cases for (w_{opt},b_{opt}), *e.g.,* 1) $w^{opt}=p_i$ and $0\le b^{opt}\le 1$, and 2) $p_i<w^{opt}<p_{i-1}$, which depends on whether $E_c-E_s\le p_i+P_s(p_i)$ holds. Fortunately, we can combine the two cases to get general formulas: $b=1-\frac{[(p_i+P_s(p_i))-(E_c-E_s)]^+}{p_i+P_s(p_i)}$, $w=P_s^{-1}(P_s(p_i)+\frac{[E_c-E_s-(p_i+P_s(p_i))]^+}{l})$.

According to Lemma 2, the optimal dividing line (w_{opt},b_{opt}) is unique and E_c-E_s reduces monotonically with the increase of (w,b), so the computed dividing line is optimal.

The **for** loop repeats at most T times, so the time complexity is $O(T)$.

D. Proofs of three optimality properties in Section V

1). Suppose Ω_i is in duration $[\tau_{i-1}, \tau_i)$ and Ω_{i+1} is in duration $[\tau_i, \tau_{i+1})$. First, according to observation 2), the transmission power in all sending phases in $[\tau_{i-1}, \tau_{i+1})$ must be equal unless the energy causality constraint is violated. Second, according to Theorem 1, there must be a single dividing line Ω for all slots in the two durations. If such a shared Ω indeed exists, then it must be in between Ω_i and Ω_{i+1}, since the battery energy consumption keeps the same.

2). Suppose otherwise, the optimal dividing line decreases at time instance τ_i, from Ω_i to Ω_{i+1}, *e.g.,* $\Omega_i > \Omega_{i+1}$. We can apply optimality property 1) to find a shared dividing line Ω for $[\tau_{i-1}, \tau_i)$ and $[\tau_i, \tau_{i+1})$. Switch to use Ω will not violate the energy causality constraint. This is because $\Omega_i > \Omega > \Omega_{i+1}$, some energy originally used before τ_i now saved in battery and used after τ_i. However, according to optimality property 1), the new dividing line transmits more data using the same amount of energy. This is a contradiction.

3). Suppose otherwise, the optimal dividing line increases at time instance t_i, but $R(t_i) > 0$. By optimality property 1), we can increase Ω_i and decrease Ω_{i+1} to improve data transmission using the same amount of energy, *e.g.,* $R(t_{i+1})$ will not be effected. As long as $R(t_i) \geq 0$ after the modification, there is no violation of the energy causality constraint, which is a contradiction.

E. Proof of Theorem 2

In every iteration of the **while** loop, the same problem repeats, that is starting from t_0, find the next optimal changing point and the corresponding dividing line. We therefore need only to show that in the first loop, where $t_0 = 1$, the computed changing time τ_{min} and dividing line Ω_{min} (Line 9) are the first optimal changing point and optimal dividing line respectively, *e.g.,* $\tau_1^{opt} = \tau_{min}$ and $\Omega_1^{opt} = \Omega_{min}$.

Suppose, on the contrary, the first optimal changing point $\tau_1^{opt} \neq \tau_{min}$, we then have the following two cases. (1) $\tau_1^{opt} > \tau_{min}$. Since the **for** loop has already checked in Line 5-6 that no feasible single dividing line can equal the two areas for all $t > \tau_{min}$ including $t = \tau_1^{opt}$. This is impossible. (2) $\tau_1^{opt} < \tau_{min}$. Then, we must have $\Omega_1^{opt} > \Omega_{min}$ according to Observation 1. We claim that in the optimal dividing lines, there is at least one dividing line Ω_i^{opt} in (or partially in) duration $[\tau_1^{opt} + 1, \tau_{min}]$, such that $\Omega_i^{opt} < \Omega_{min}$. Because otherwise, optimal dividing lines in $[t_0, \tau_{min}]$ are all larger than Ω_{min}, contradicting the fact Ω_{min} depletes battery at τ_{min}.

During the execution of `iECP`, the dominate computation are Insert-MaxHeap and Extract-MinHeap operations, both take $O(\log T)$ steps at most. In each **while** loop iteration, one changing point is found by the execution of **for** loop, which has at most T iterations. During these iterations, at most T Insert-MaxHeap operation takes place. Because each powers inserted into the Heap can be extracted no more than once, there are at most T Extrac-MinHeap operations too. Hence, to find one changing point, it needs at most $O(T \log T)$ steps. Since there are no more than T changing points, the finial time complexity is $O(T^2 \log T)$.