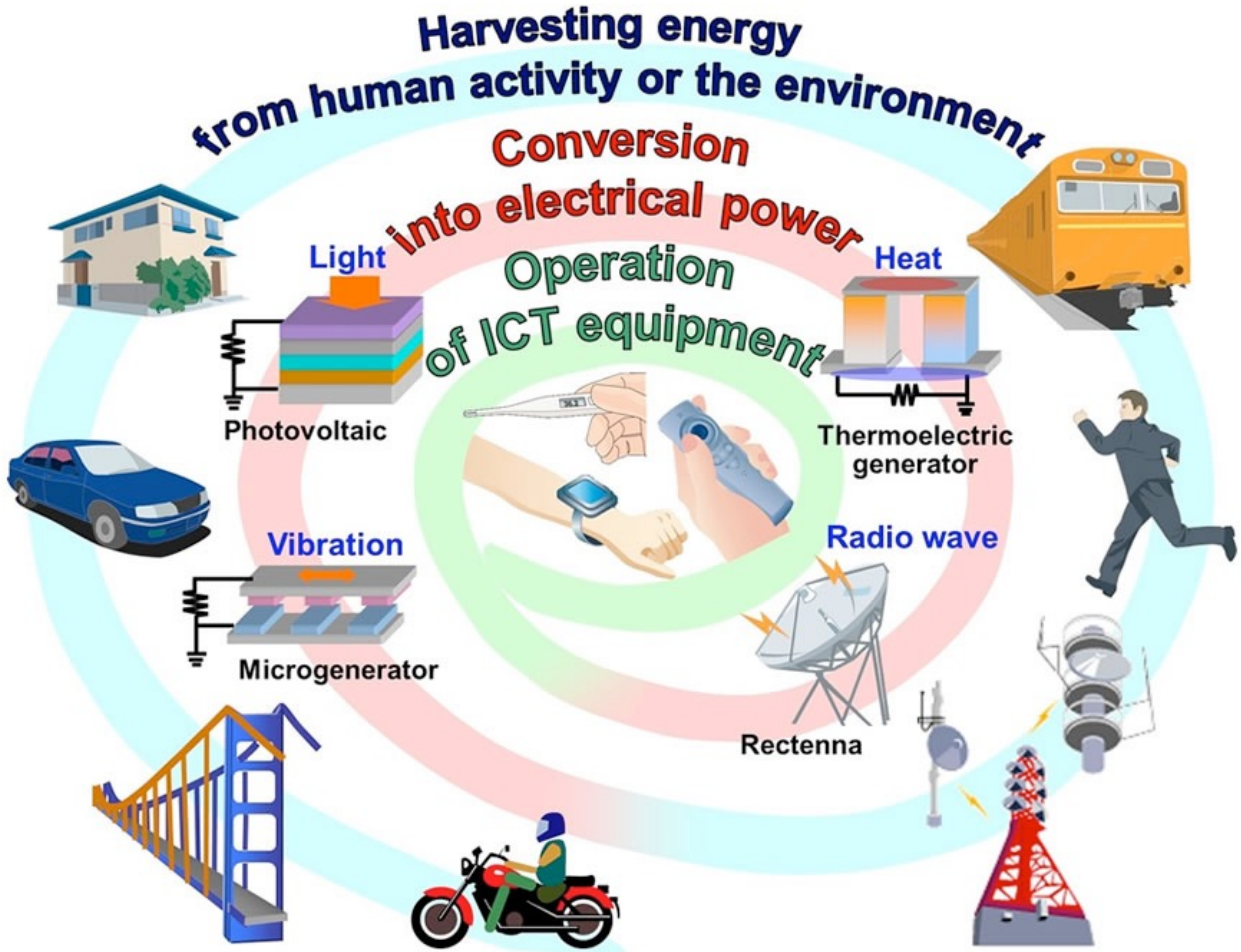
Algorithm Design for TxRx Energy Harvesting Communication System

Rahul Vaze

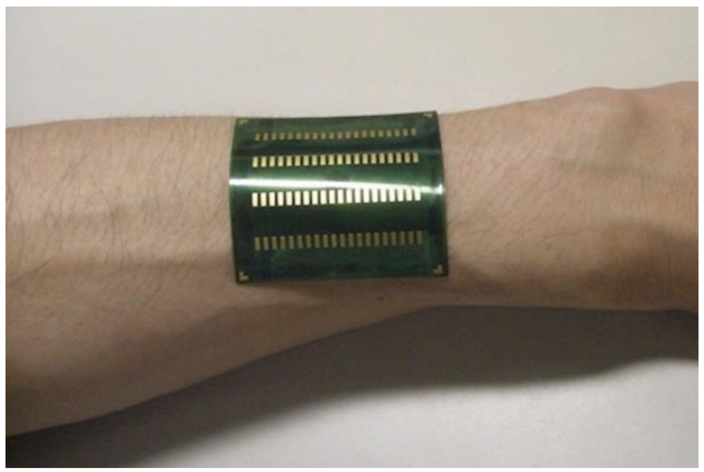
tifr

Sidhartha Satpathi   Rushil Nagda

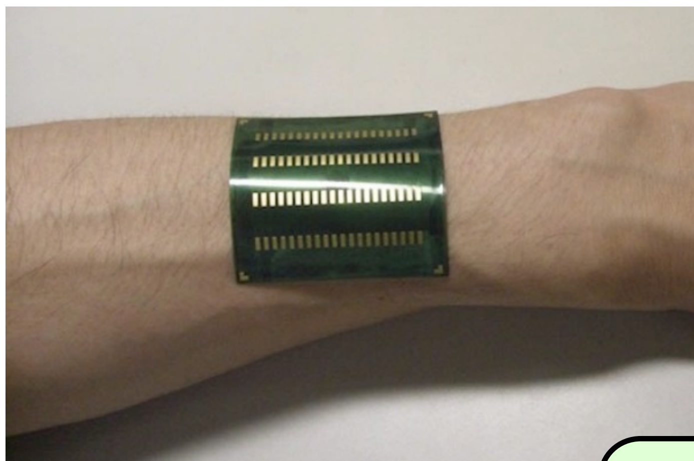
Energy Harvesting Paradigm



Reality



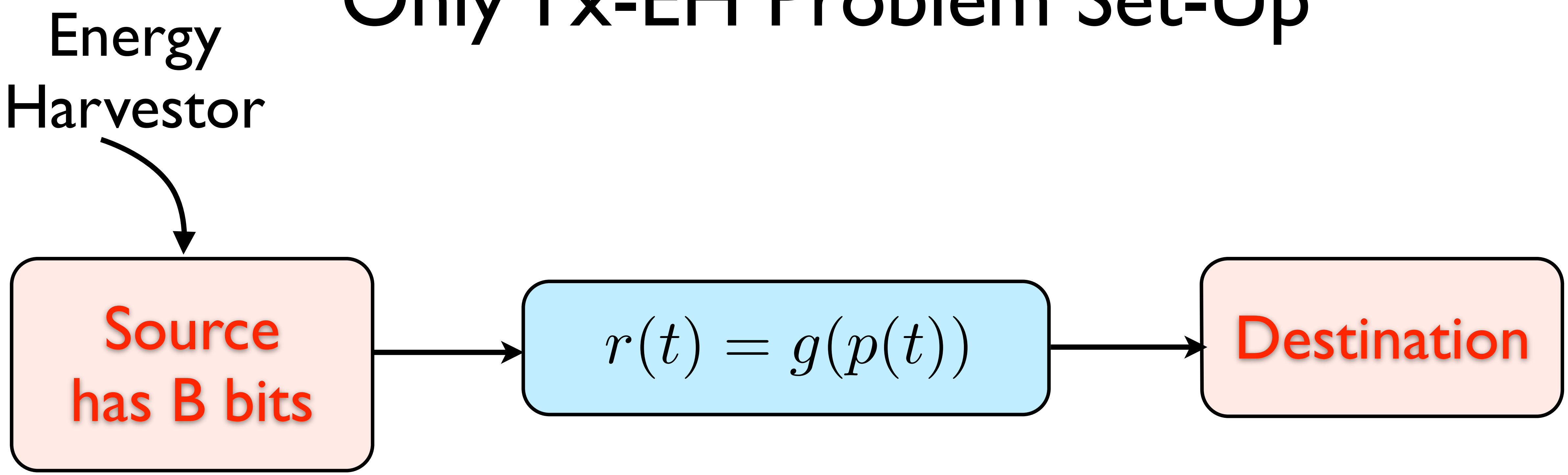
Reality



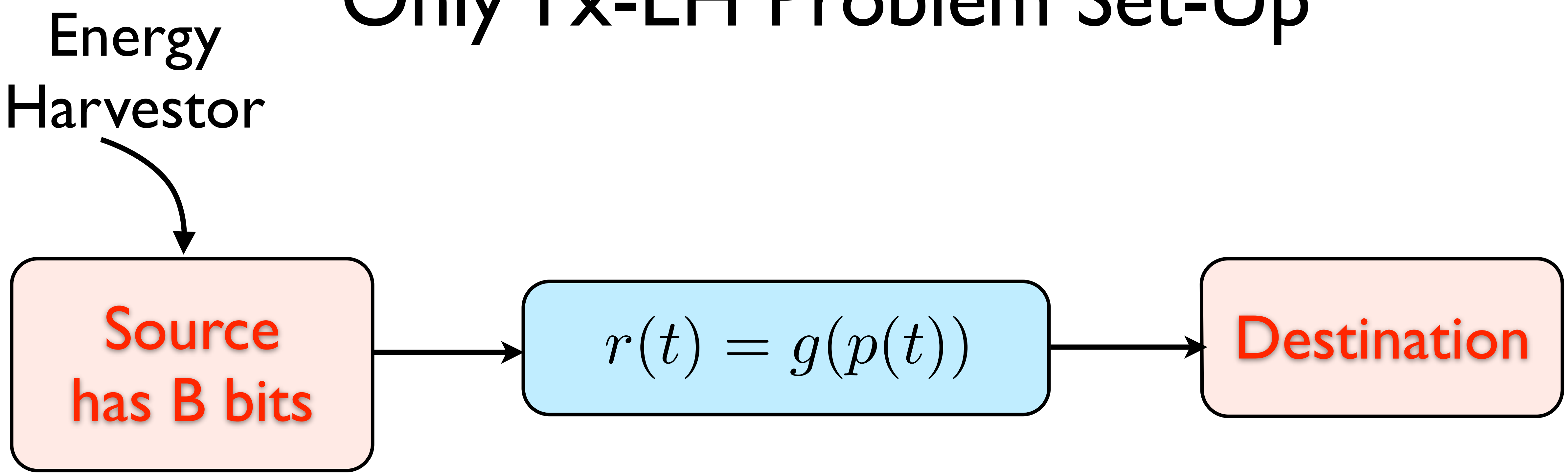
Energy Arrivals

- Arbitrary
- Not drawn from a distribution

Only Tx-EH Problem Set-Up



Only Tx-EH Problem Set-Up



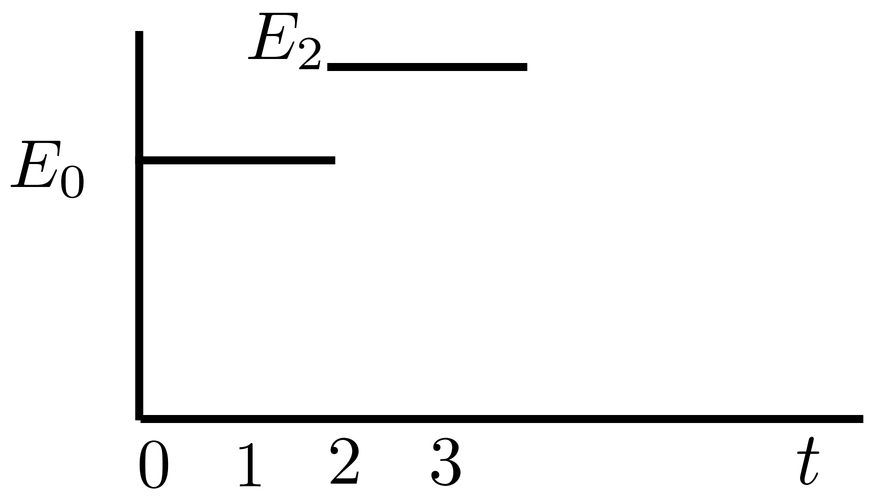
$B(t)$ Bits sent until time t

$E(t)$ Energy used up until time t

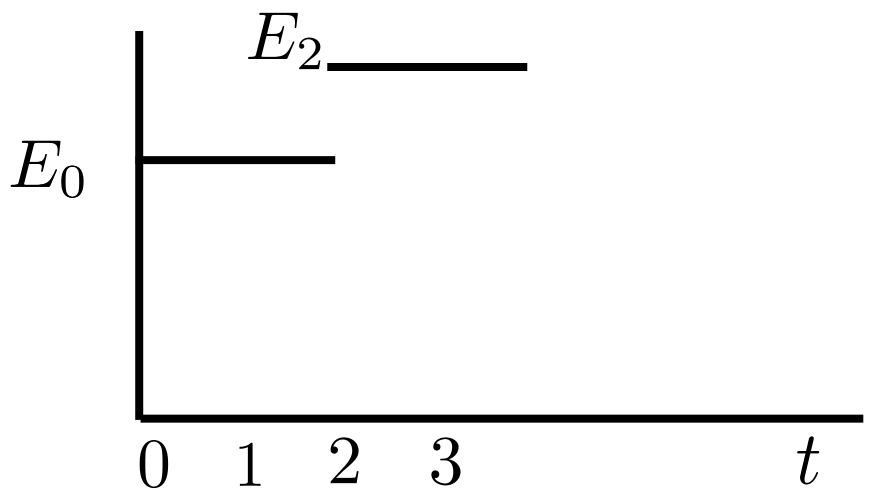
Find an Online Algorithm

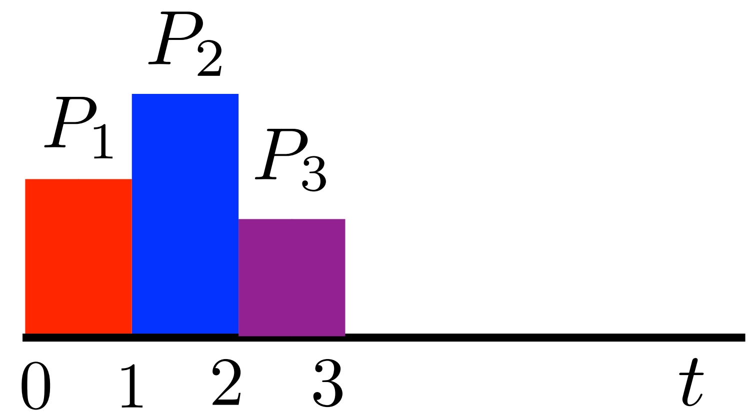
$$\boxed{T^\star = \min_{B(T)=B,\, E(t) \leq \sum_{i, i \leq t} E_i} T}$$

Example

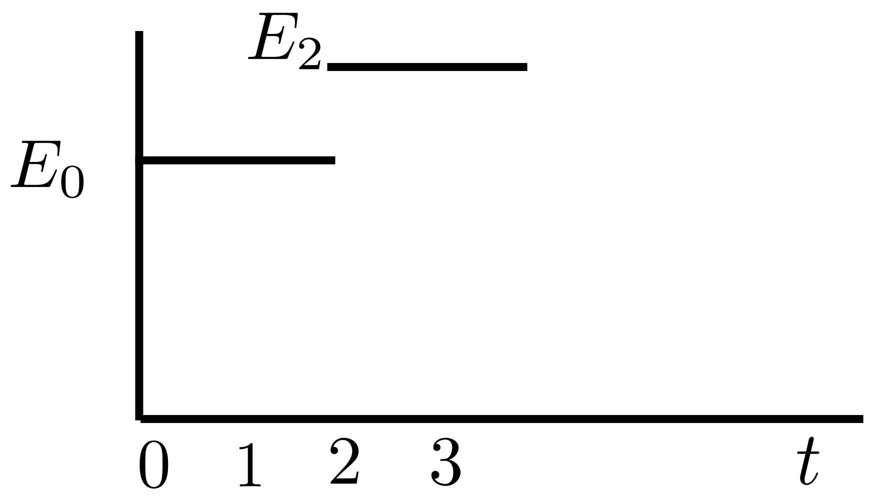


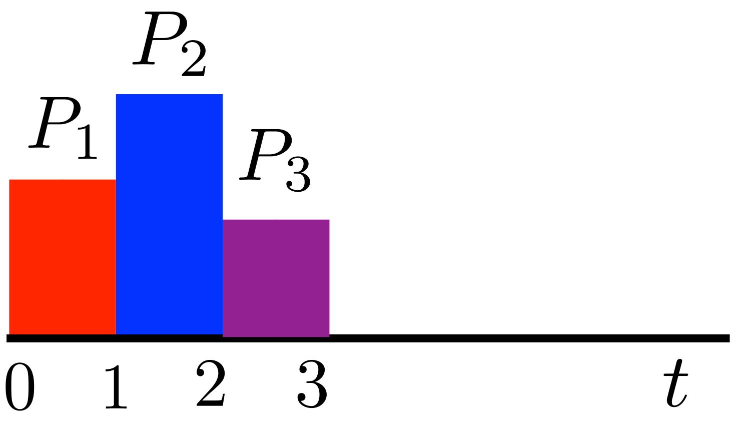
Example





Example

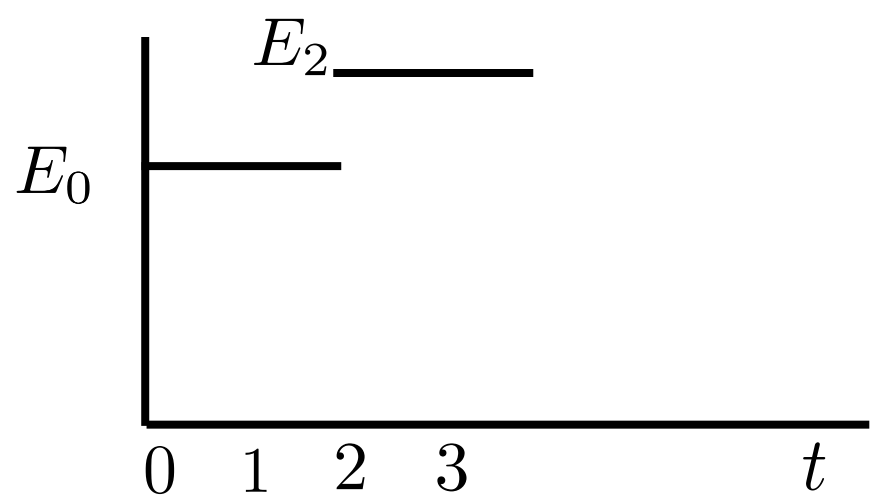


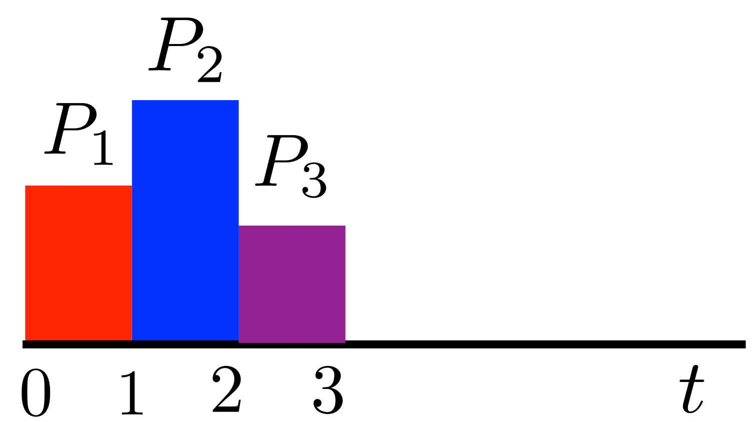


$$P_1 + P_2 \leq E_0$$

$$(E_0 - P_1 + P_2) + P_3 \leq E_2$$

Example





$$P_1 + P_2 \leq E_0$$

$$(E_0 - P_1 + P_2) + P_3 \leq E_2$$

Under these constraint maximize throughput without knowing the future energy arrivals

Equivalent Problem

Equivalent Problem

Total Distance D_0

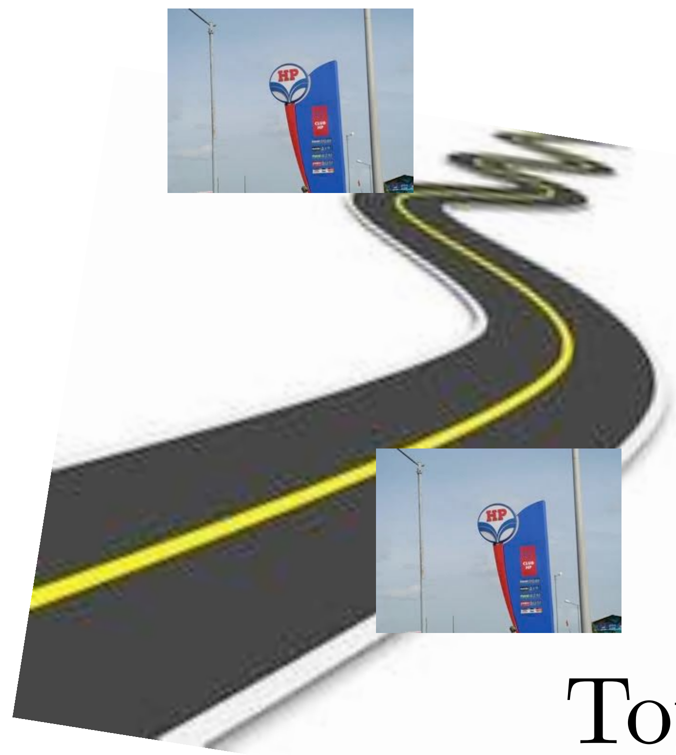
Equivalent Problem

Total Distance D_0

$D = t\log(1+v)$

Petrol use $= v * t$

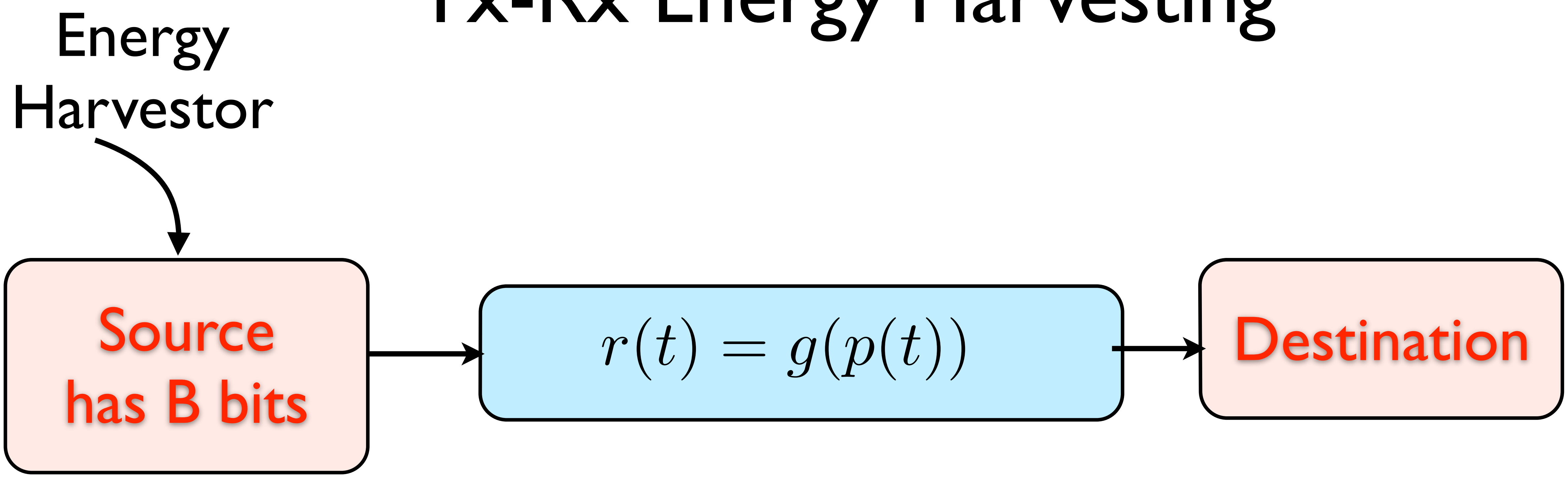
Equivalent Problem



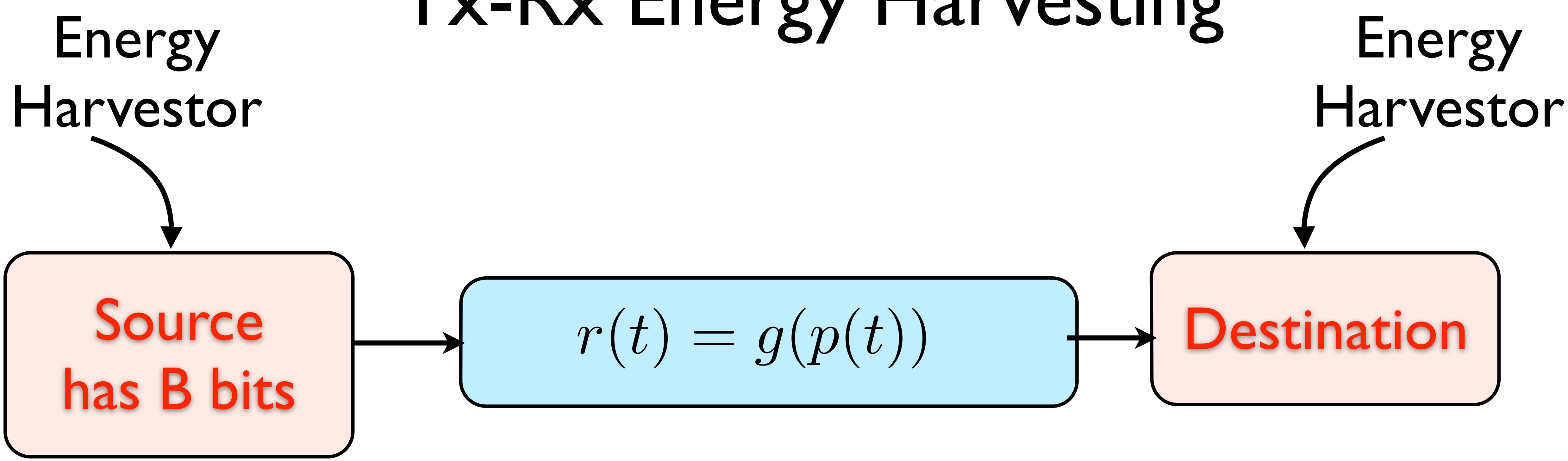
$$\text{Total Distance } D_0$$
$$D = t\log(1+v)$$
$$\text{Petrol use} = v * t$$

- Initial Petrol Available
- Next Petrol Station Distance and Availability Unknown
- Minimize the time to destination

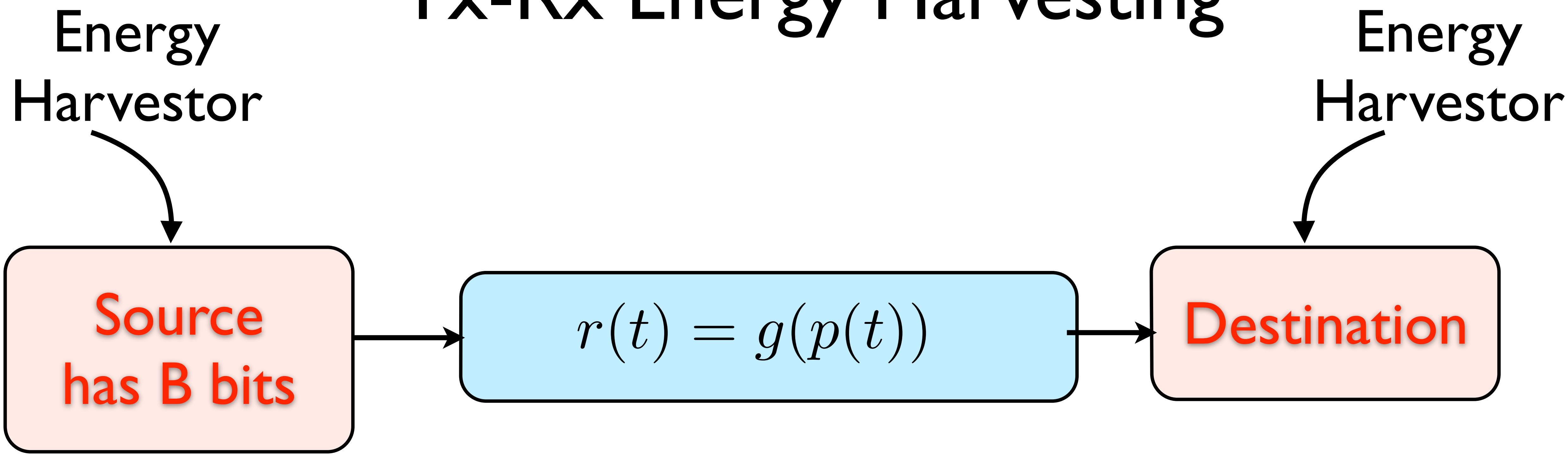
Tx-Rx Energy Harvesting



Tx-Rx Energy Harvesting

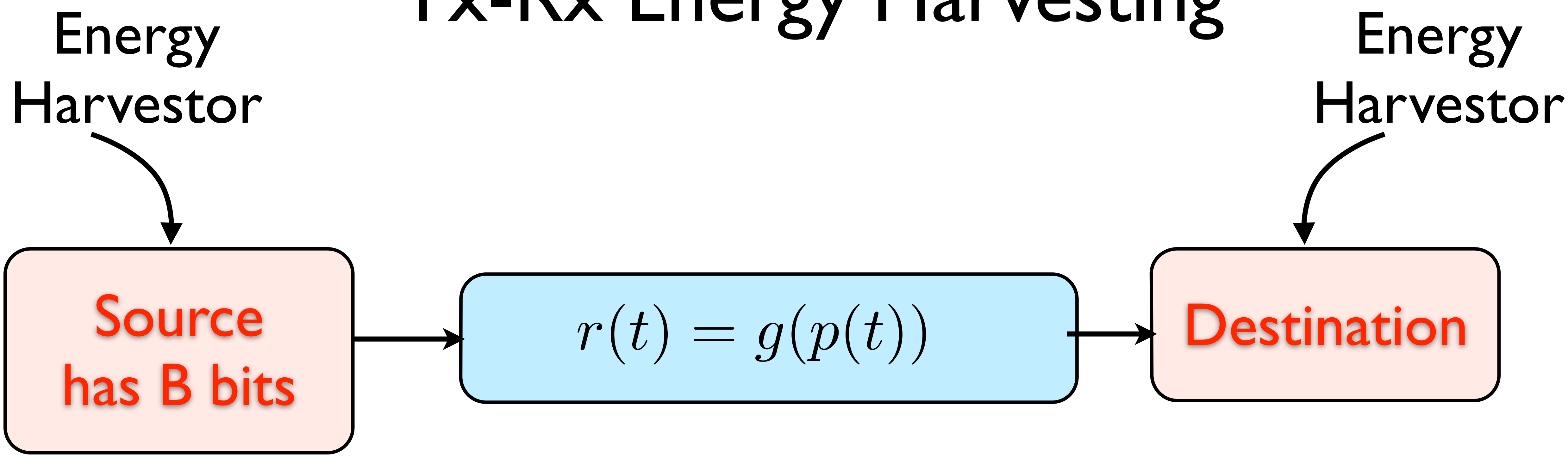


Tx-Rx Energy Harvesting



$U(t), C(t)$ Energy used at time t at Tx, Rx

Tx-Rx Energy Harvesting

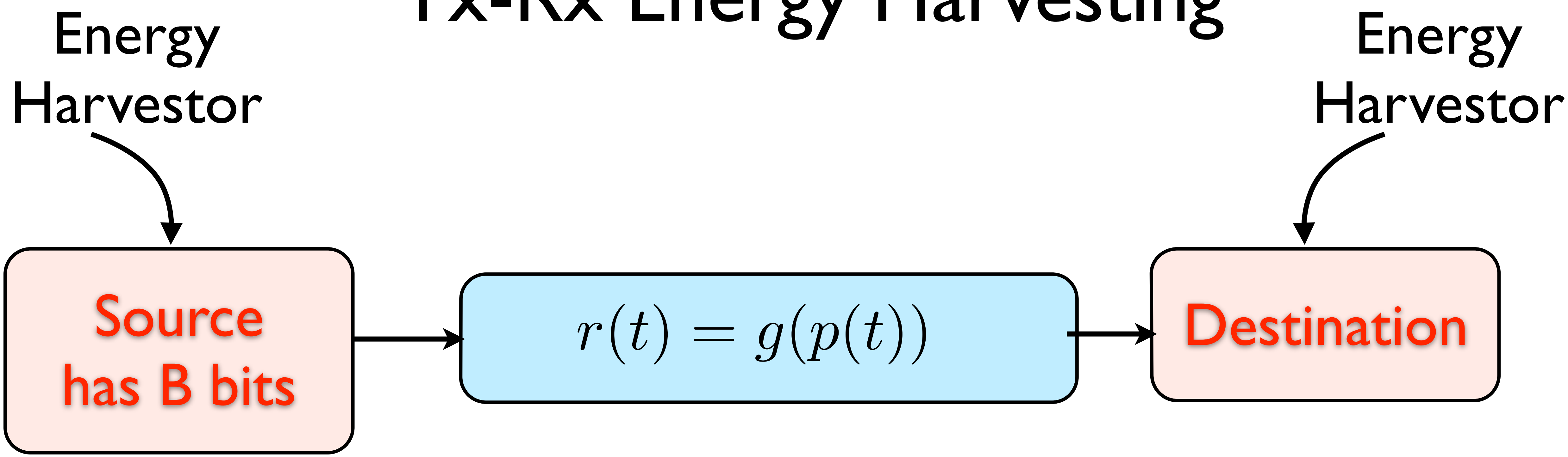


$U(t), C(t)$ Energy used at time t at Tx, Rx

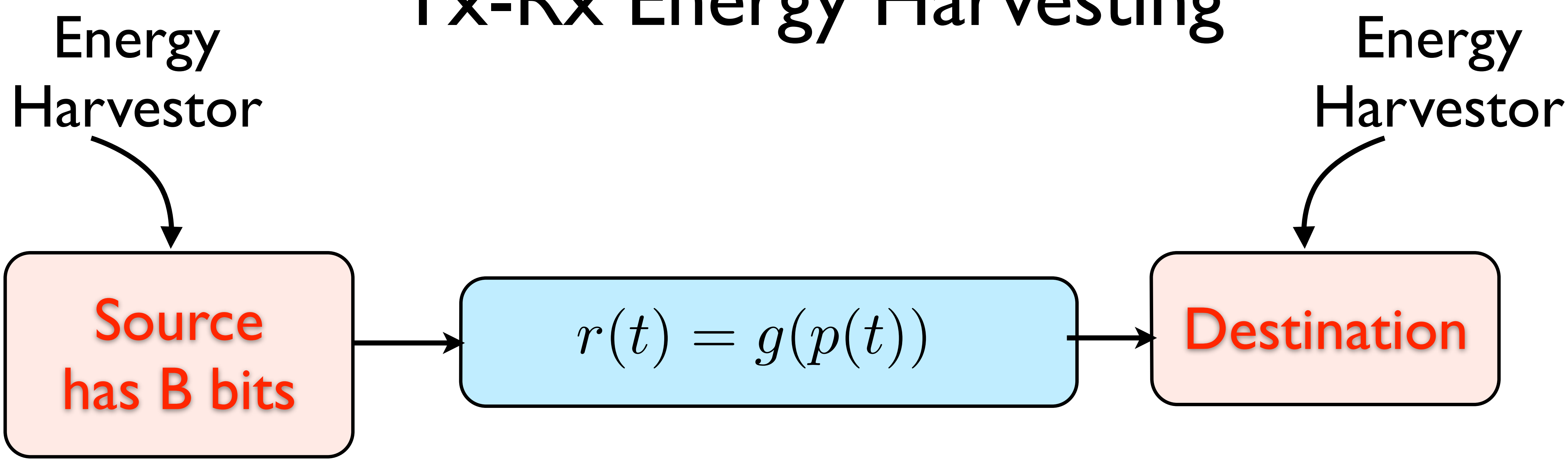
Find an Online Algorithm

$$T^{\star} = \min_{B(T)=B, U(t) \leq E(t), C(t) \leq \Gamma(t)} T$$

Tx-Rx Energy Harvesting



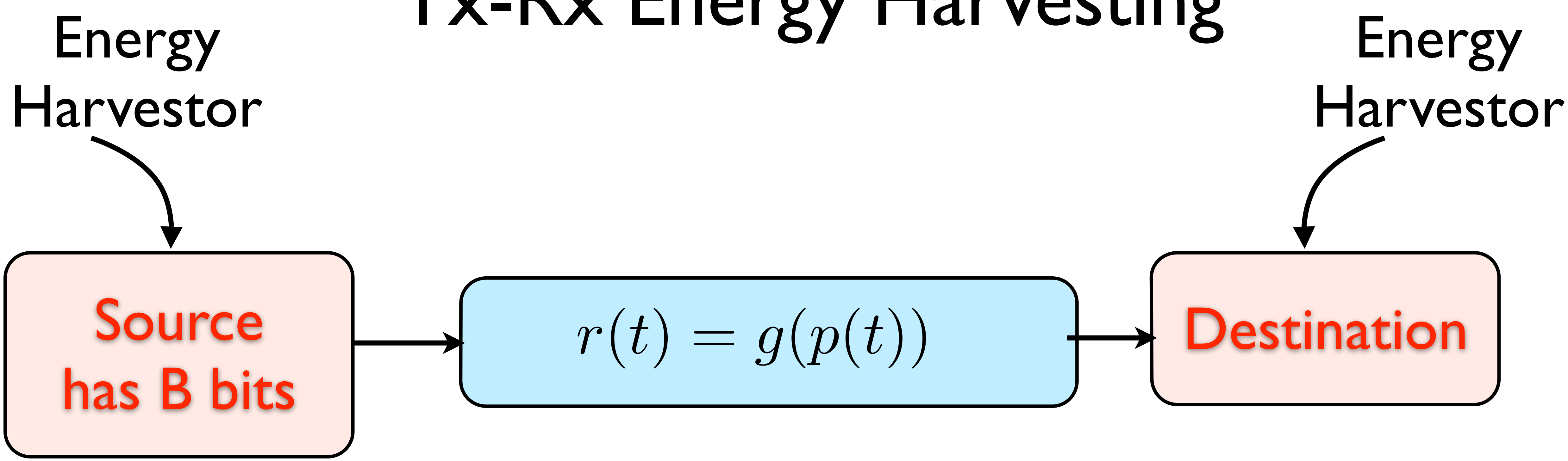
Tx-Rx Energy Harvesting



Receiver consumes fixed power to stay ON, say P_r

- Rx decision is binary, either its ON or OFF

Tx-Rx Energy Harvesting



Receiver consumes fixed power to stay ON, say P_r
- Rx decision is binary, either its ON or OFF

Thus the total receiver energy constraints
the total time for which it is ON

$$\Gamma \leq \frac{R}{P_r}$$

Equivalent Problem

Equivalent Problem

Total Distance D_0

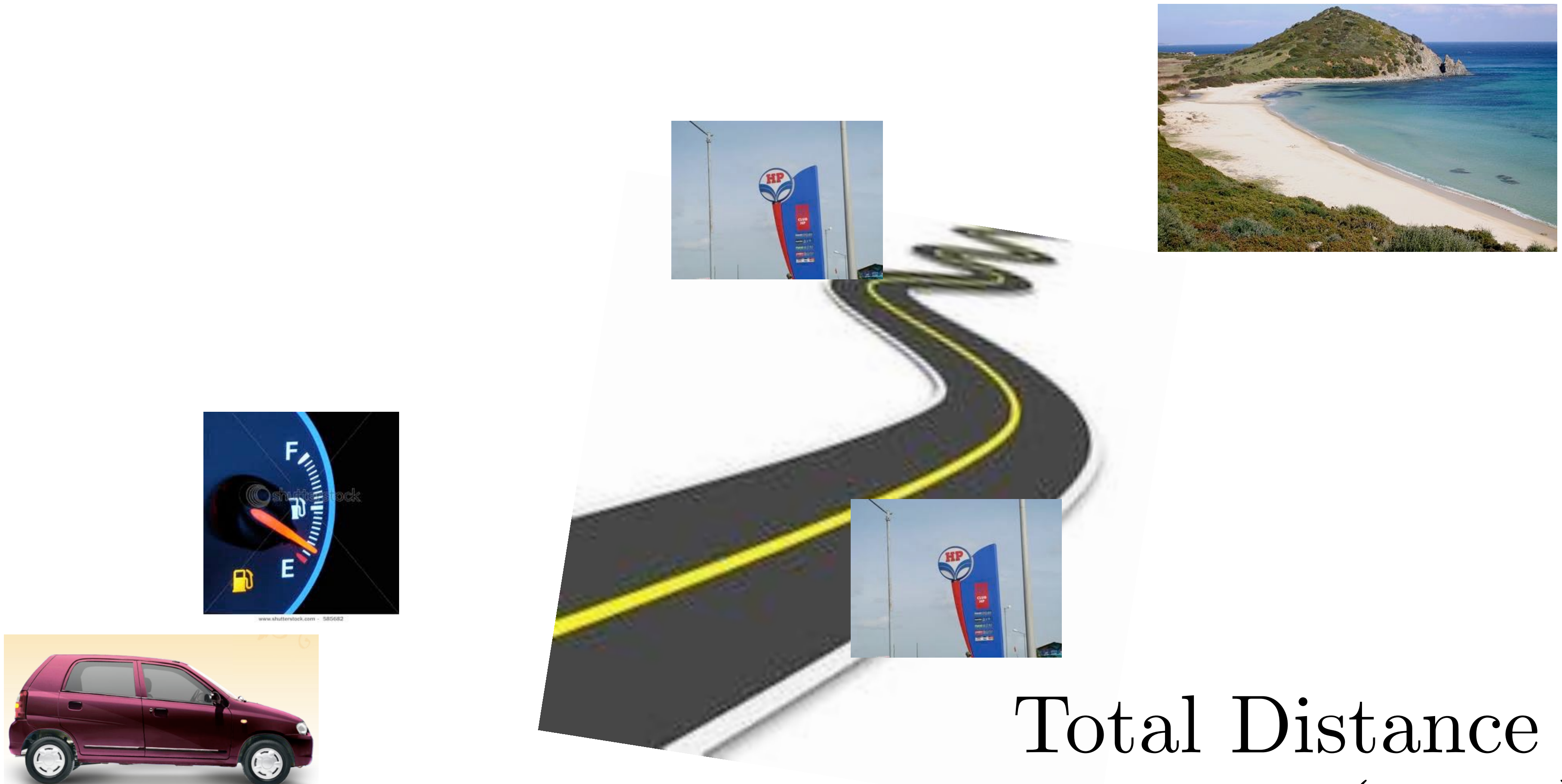
Equivalent Problem

Total Distance D_0

$D = t\log(1+v)$

Petrol use $= v * t$

Equivalent Problem



Total Distance D_0

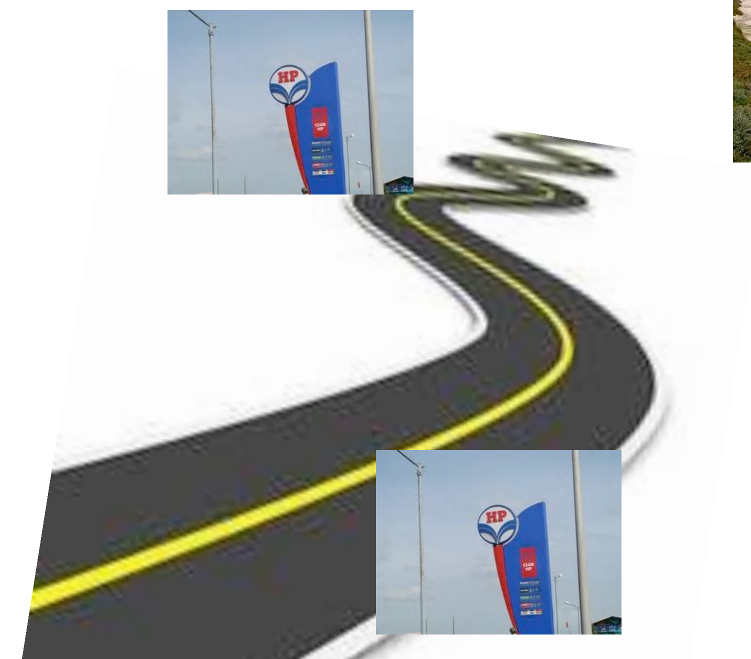
$D = t\log(1+v)$

Petrol use $= v * t$

- Initial Petrol Available
- Next Petrol Station Distance and Availability Unknown
- Minimize the time to destination

(with max time of actual run < Rx ON-time)

Equivalent Problem



Total Distance D_0

$D = t \log(1 + v)$

Petrol use $= v * t$

- Initial Petrol Available
- Next Petrol Station Distance and Availability Unknown
- Minimize the time to destination

(with max time of actual run < Rx ON-time) Challenging !

Prior Work - EH

Prior Work - EH

Only Tx

Offline Algorithms

[Ulukus, Yener et al]

- AWGN
- Fading
- MAC,BC,Intf.

Prior Work - EH

	Only Tx	Tx-Rx
Offline Algorithms	[Ulukus, Yener et al] • AWGN • Fading • MAC,BC,Intf.	

Prior Work - EH

	Only Tx	Tx-Rx
Offline Algorithms	[Ulukus, Yener et al] • AWGN • Fading • MAC,BC,Intf.	Nothing

Prior Work - EH

	Only Tx	Tx-Rx
Offline Algorithms	[Ulukus, Yener et al] • AWGN • Fading • MAC,BC,Intf.	Nothing
Online Algorithms	[Vaze'13] • AWGN • Fading • MAC	Nothing

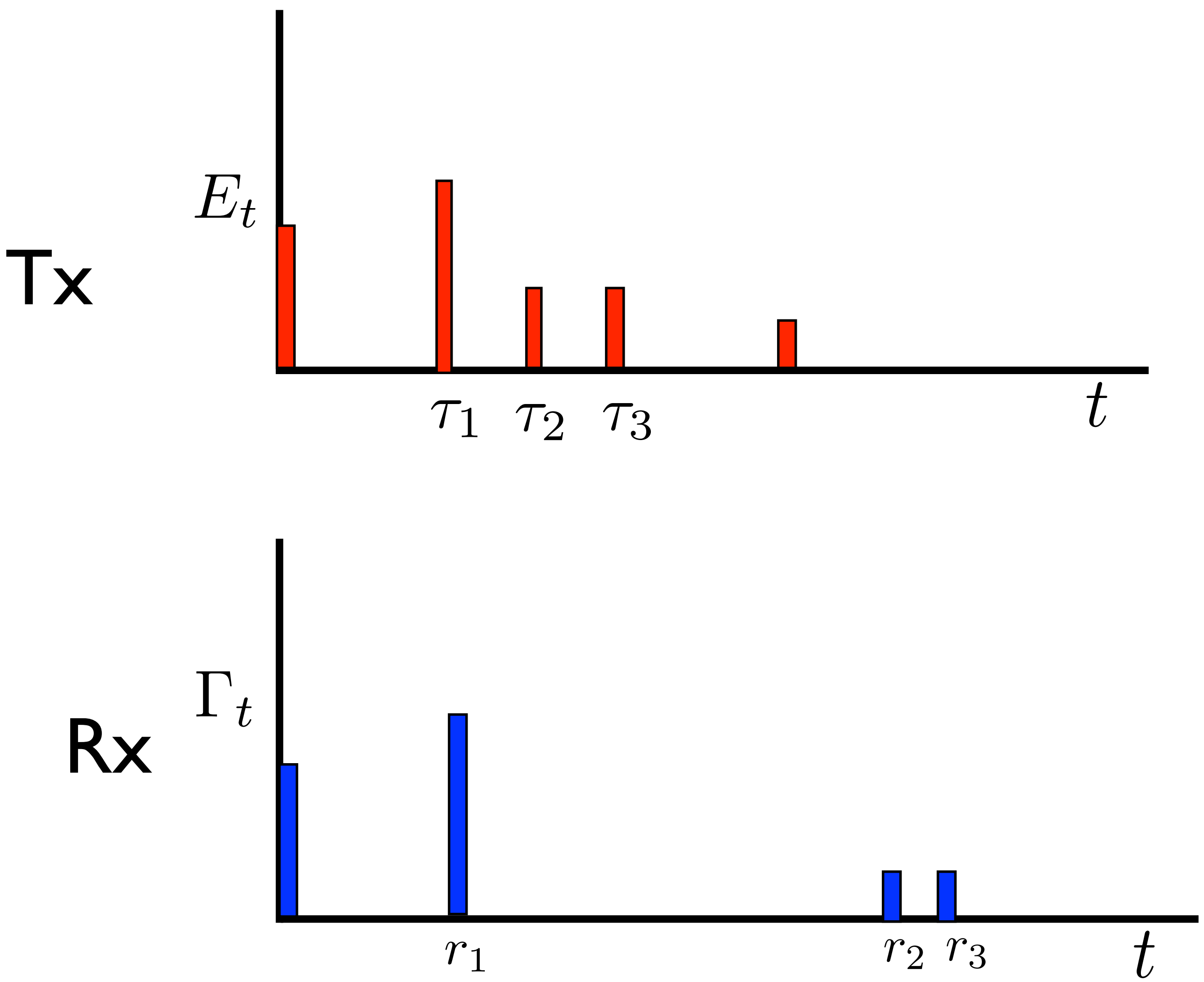
Prior Work - EH

	Only Tx	Tx-Rx
Offline Algorithms	[Ulukus, Yener et al] • AWGN • Fading • MAC,BC,Intf.	Nothing
Online Algorithms	[Vaze'13] • AWGN • Fading • MAC	Nothing
		Cap. Approx. [DoshiVaze'14]

Dr. know
it all

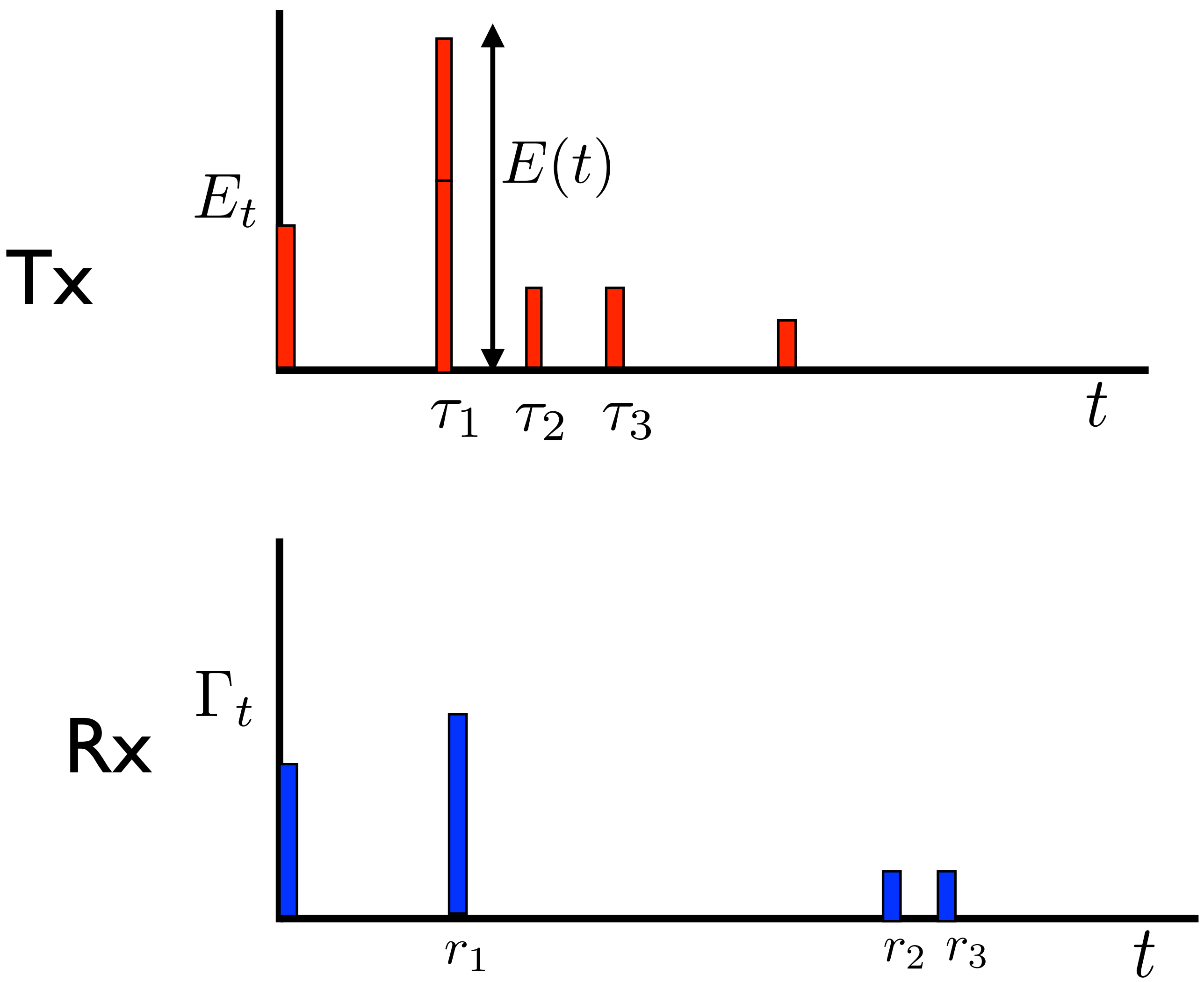
Offline Algorithm

everything known in future - arrivals epochs and amounts



Offline Algorithm

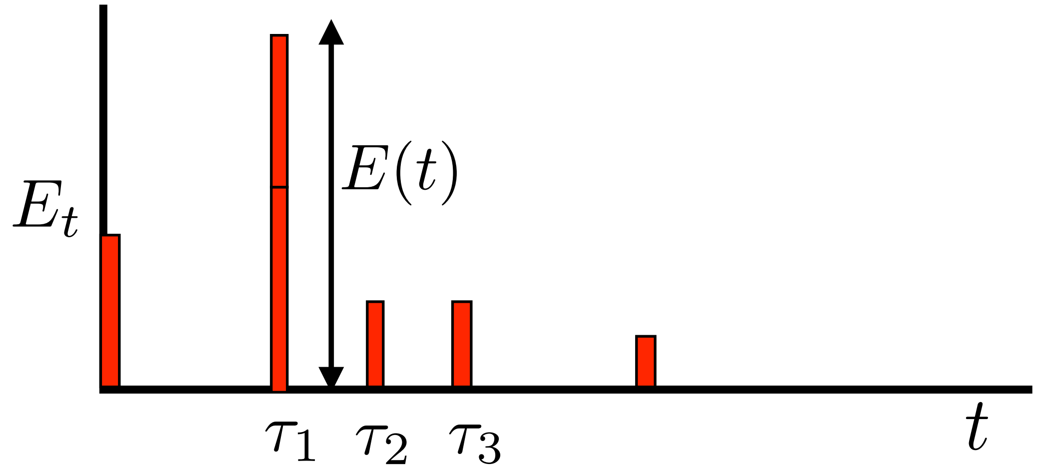
everything known in future - arrivals epochs and amounts



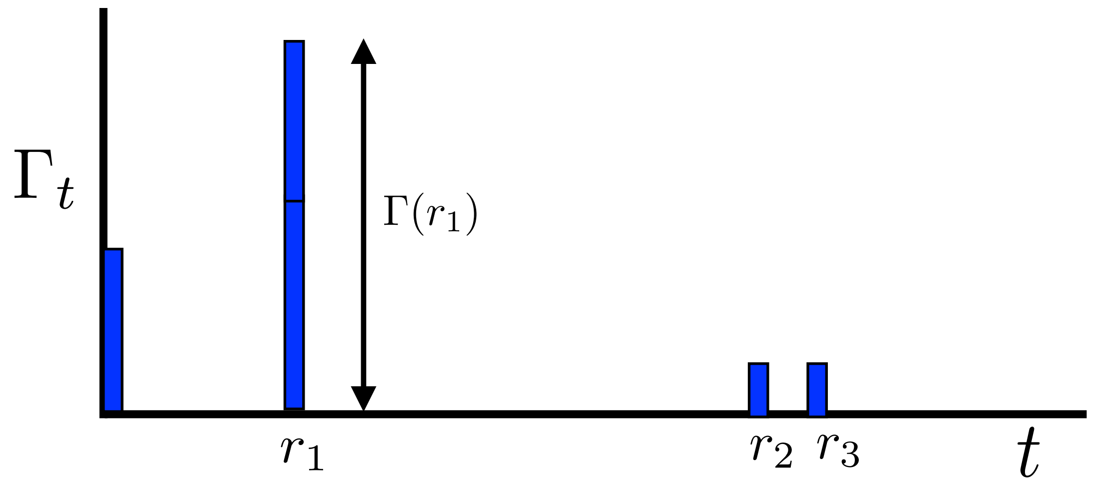
Offline Algorithm

everything known in future - arrivals epochs and amounts

Tx

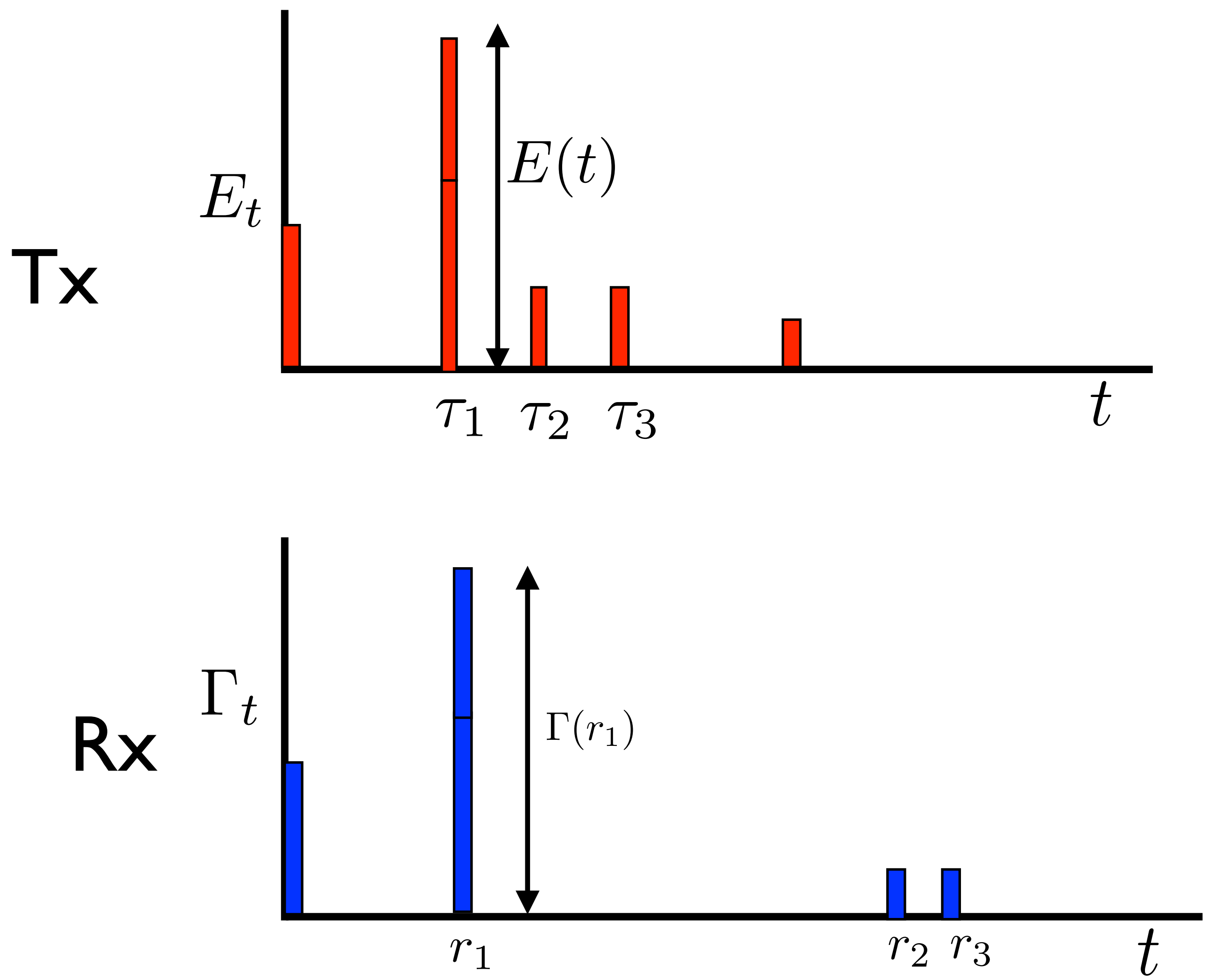


Rx



Offline Algorithm

everything known in future - arrivals epochs and amounts

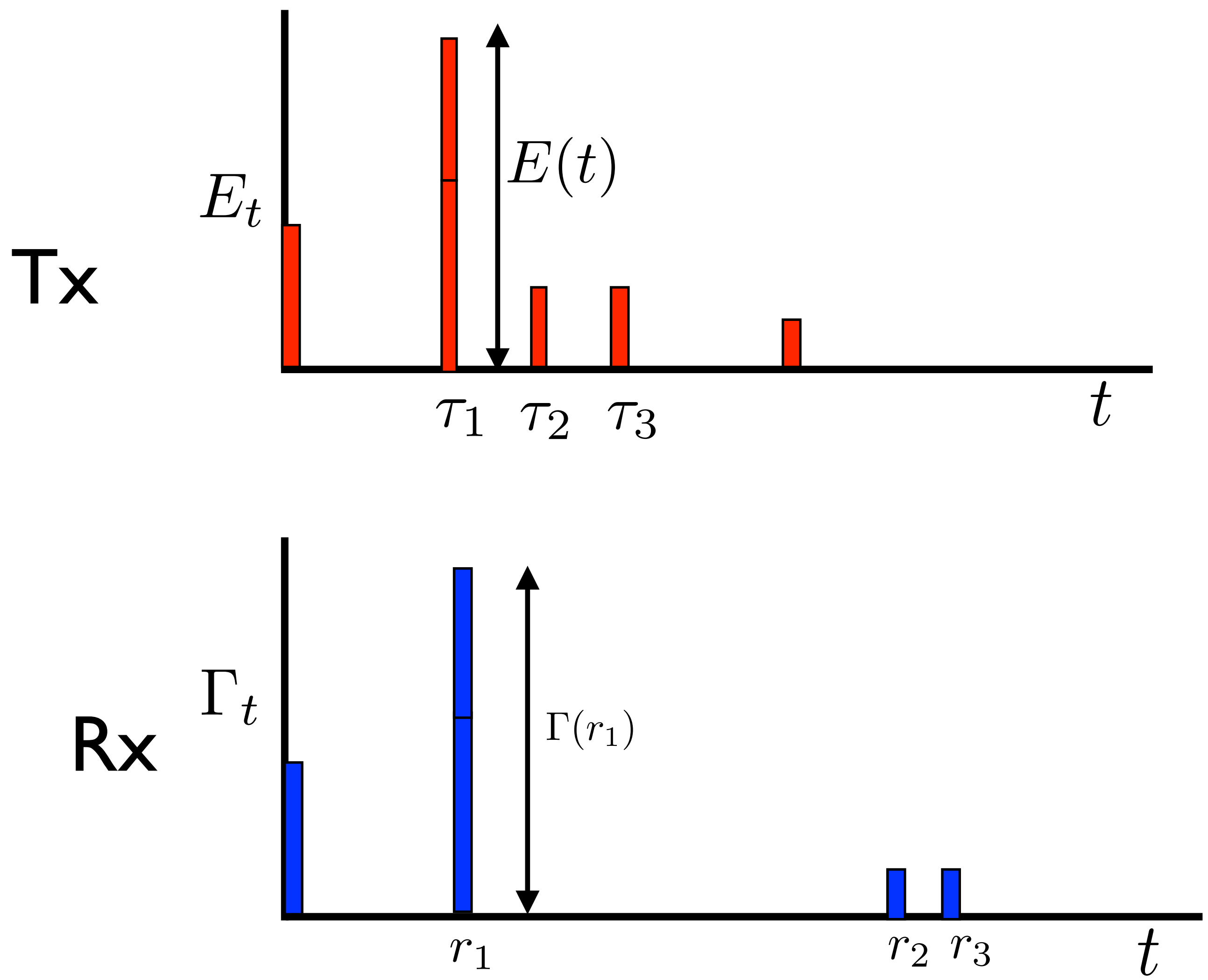


$U(t), C(t)$

Energy used until time t at Tx, Rx

Offline Algorithm

everything known in future - arrivals epochs and amounts



$$U(t), C(t)$$

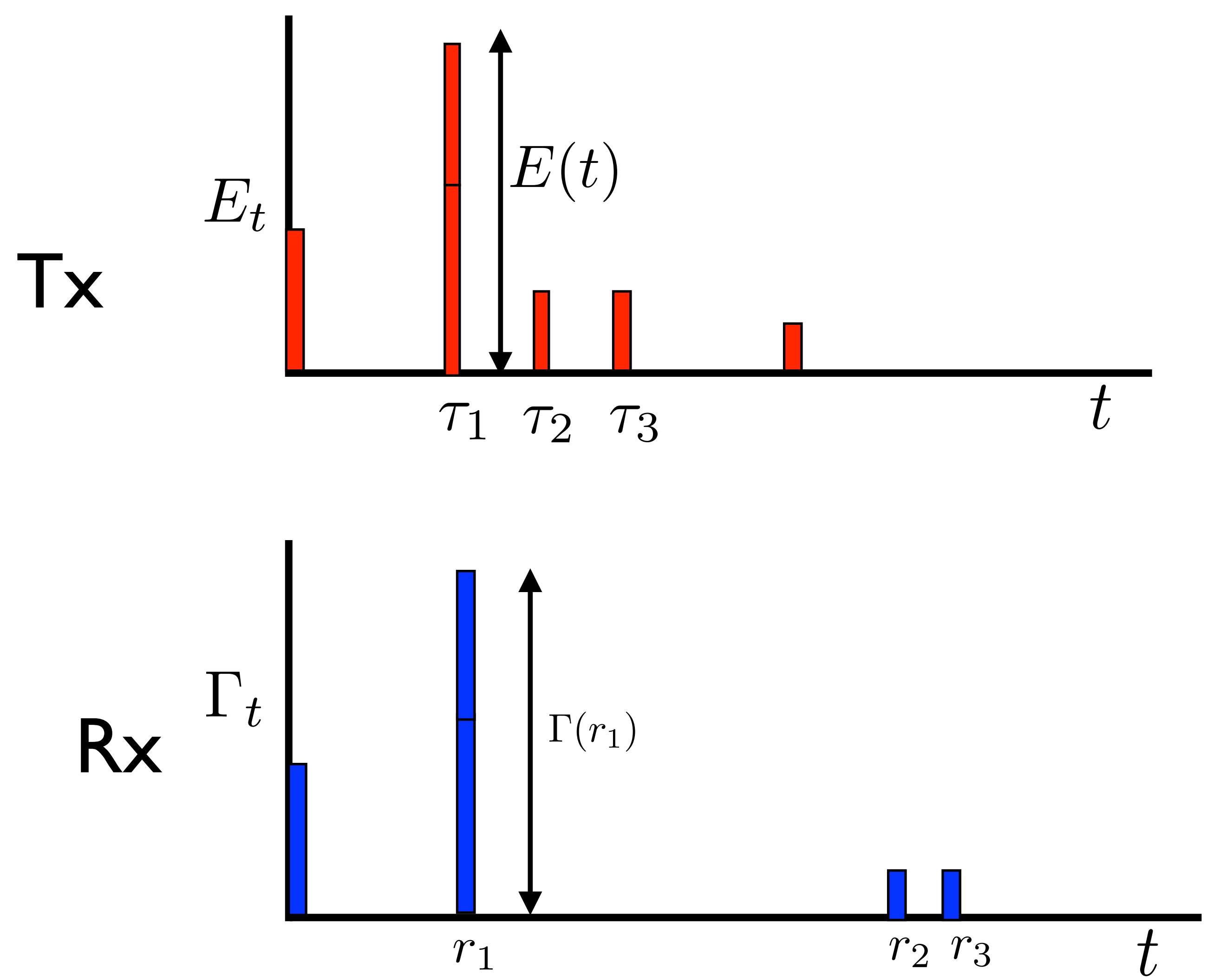
Energy used until time t at Tx, Rx

bits with power p(t)

$$r(t) = g(p(t))$$

Offline Algorithm

everything known in future - arrivals epochs and amounts



$$U(t), C(t)$$

Energy used until time t at Tx, Rx

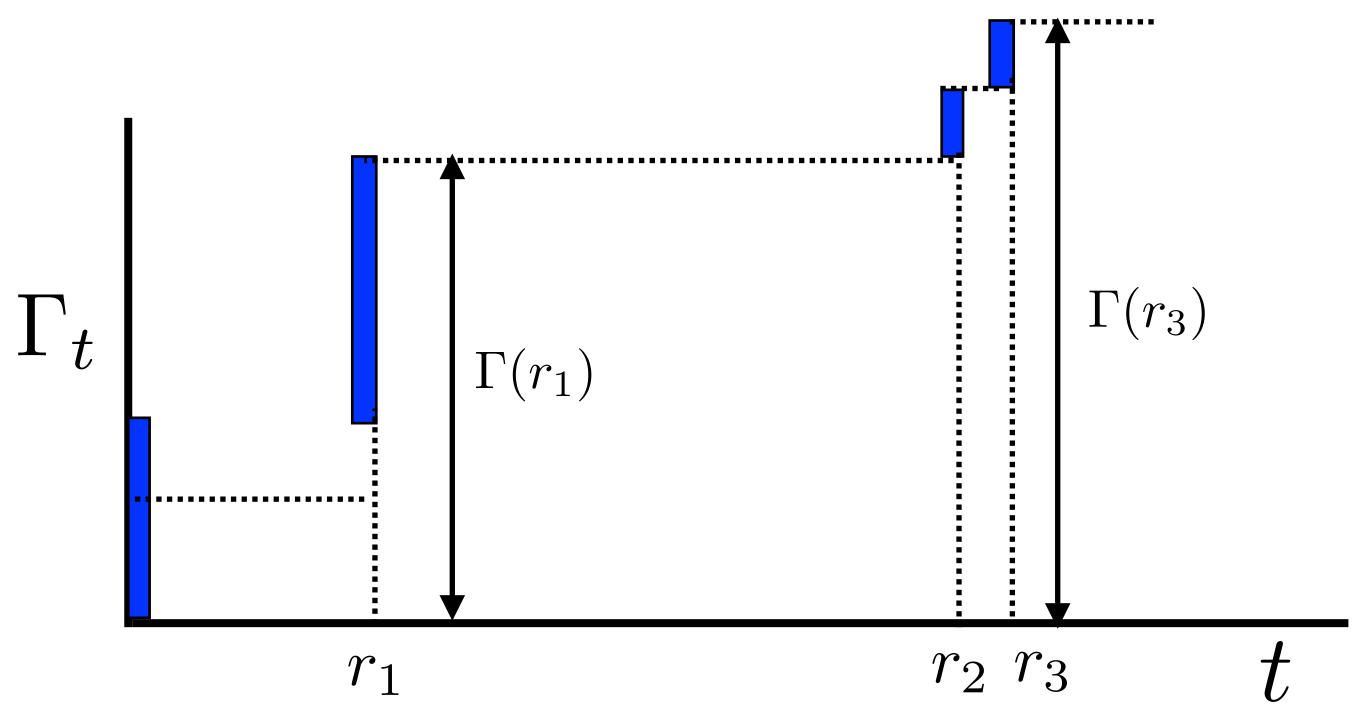
bits with power p(t)

$$r(t) = g(p(t))$$

$$T^\star = \min_{B(T)=B, U(t)\leq E(t), C(t)\leq \Gamma(t)} T$$

Offline Algorithm- Deconstructed

Rx

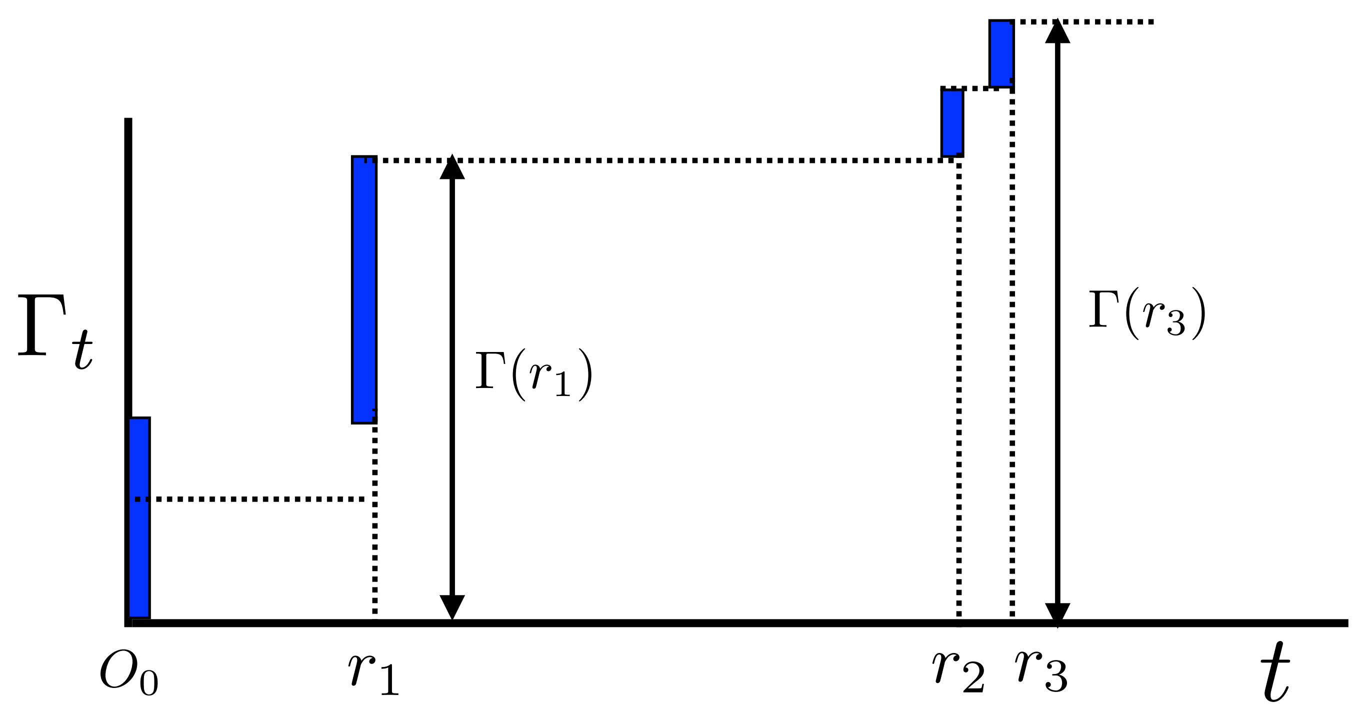


Look at receiver energy arrival instant r_i

O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Offline Algorithm- Deconstructed

Rx

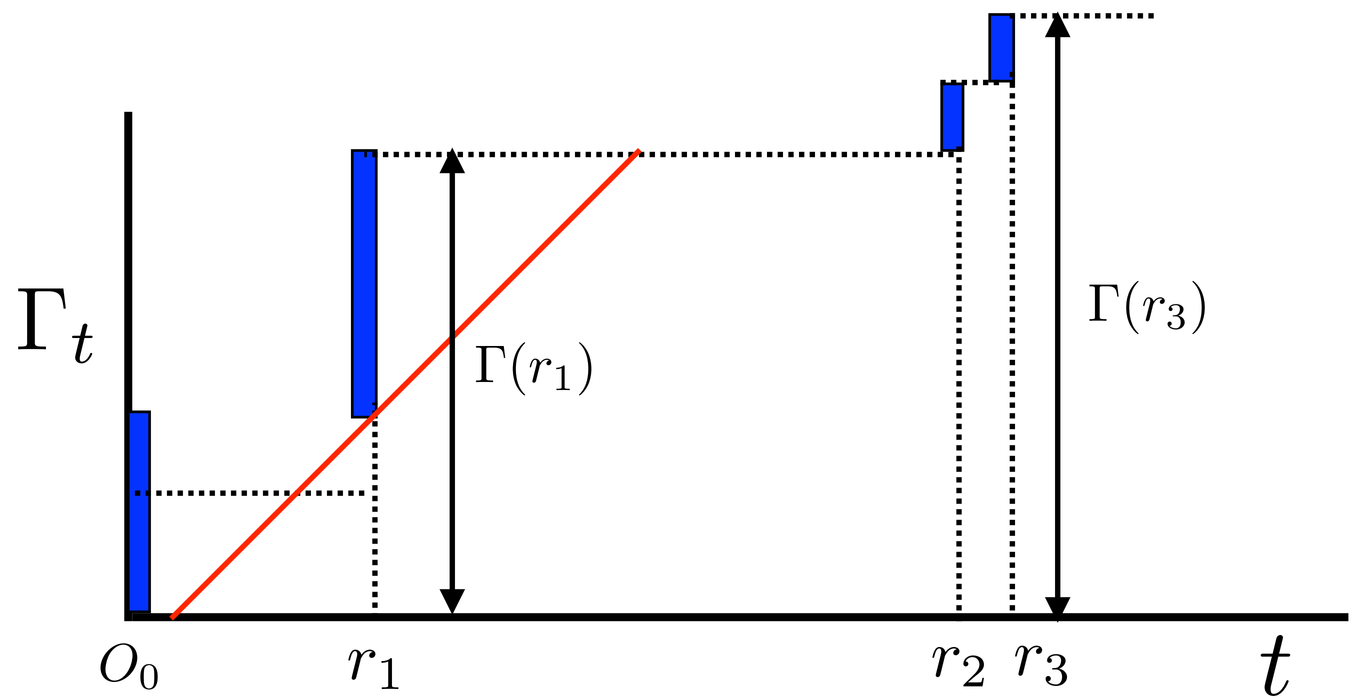


Look at receiver energy arrival instant r_i

O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Offline Algorithm- Deconstructed

Rx

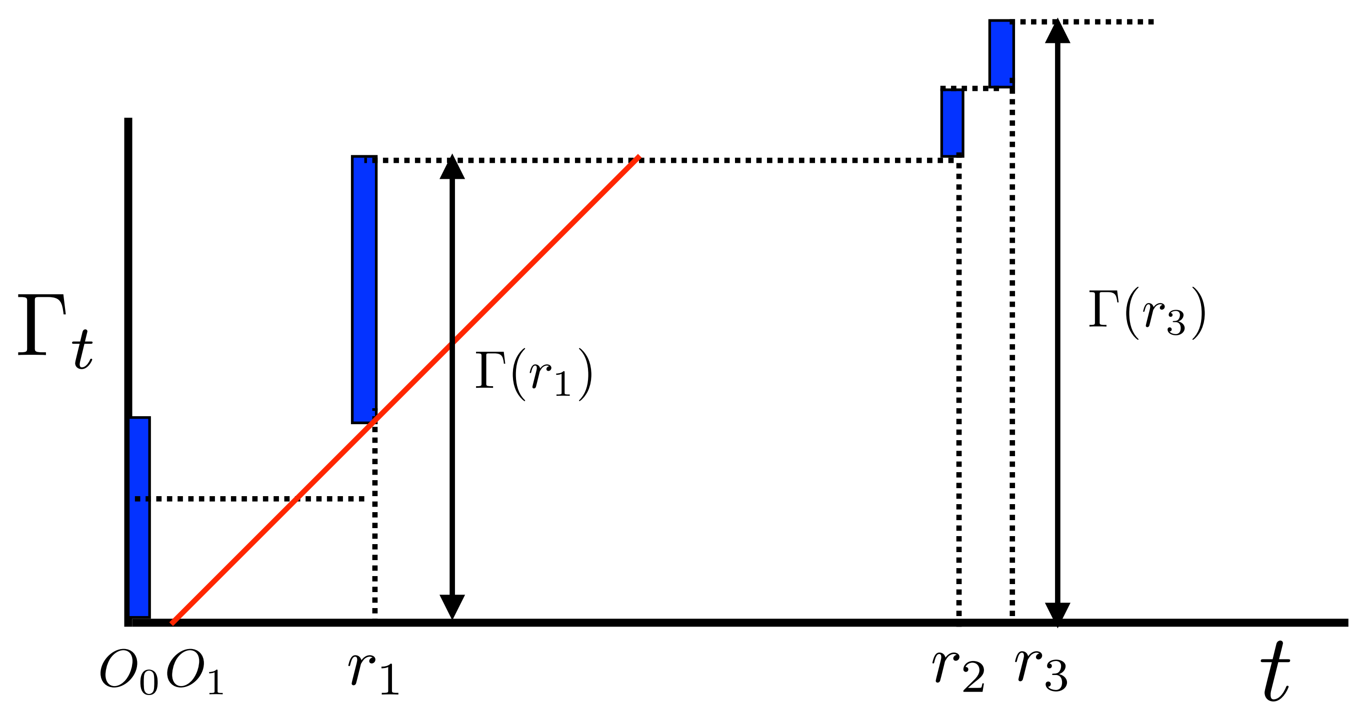


Look at receiver energy arrival instant r_i

O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Offline Algorithm- Deconstructed

Rx

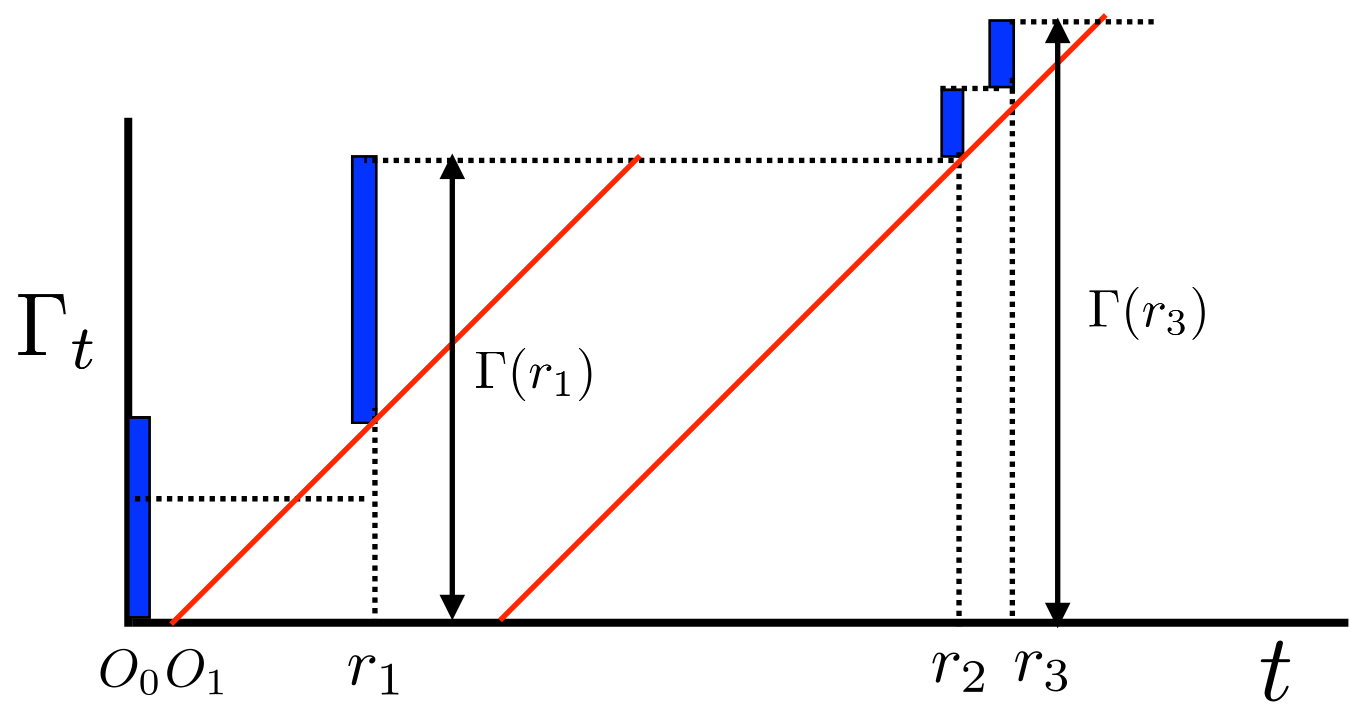


Look at receiver energy arrival instant r_i

O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Offline Algorithm- Deconstructed

Rx

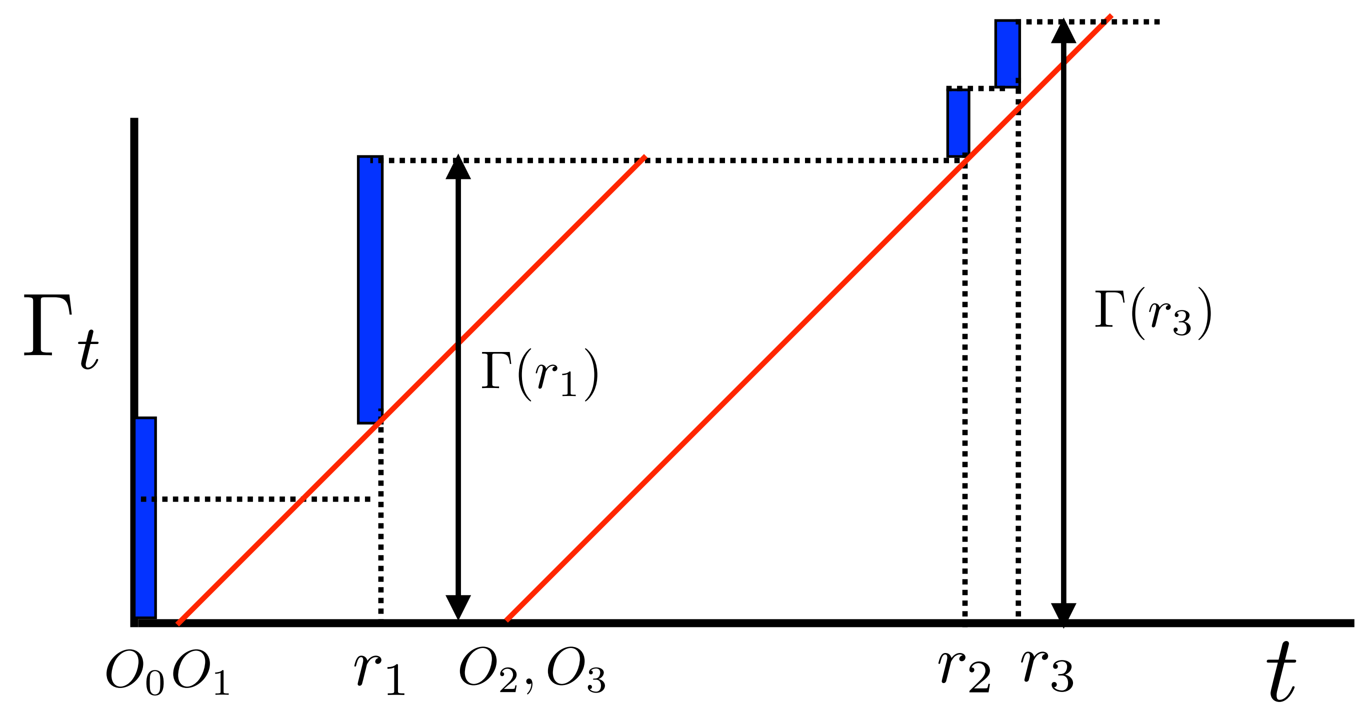


Look at receiver energy arrival instant r_i

O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Offline Algorithm- Deconstructed

Rx

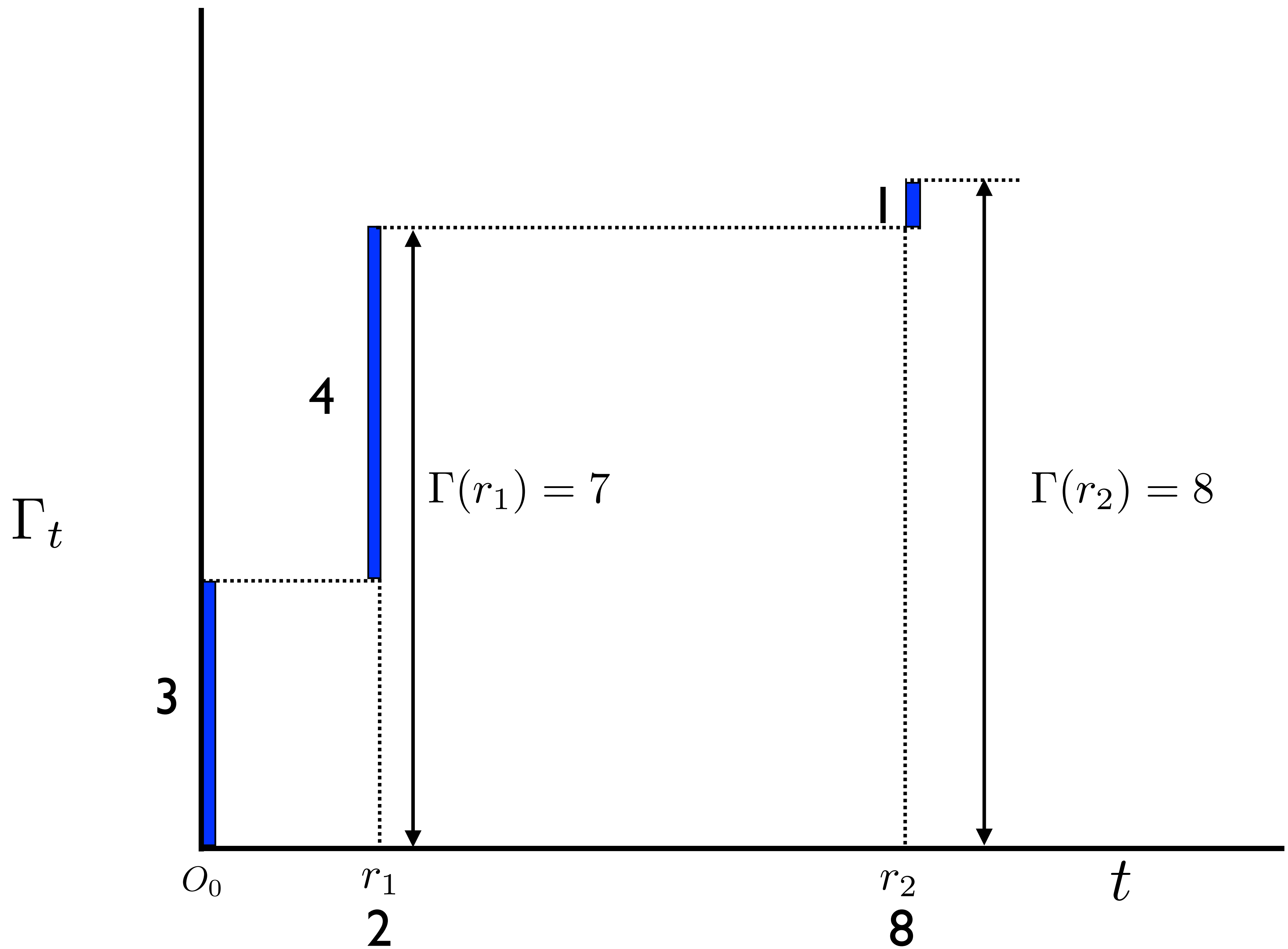


Look at receiver energy arrival instant r_i

O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

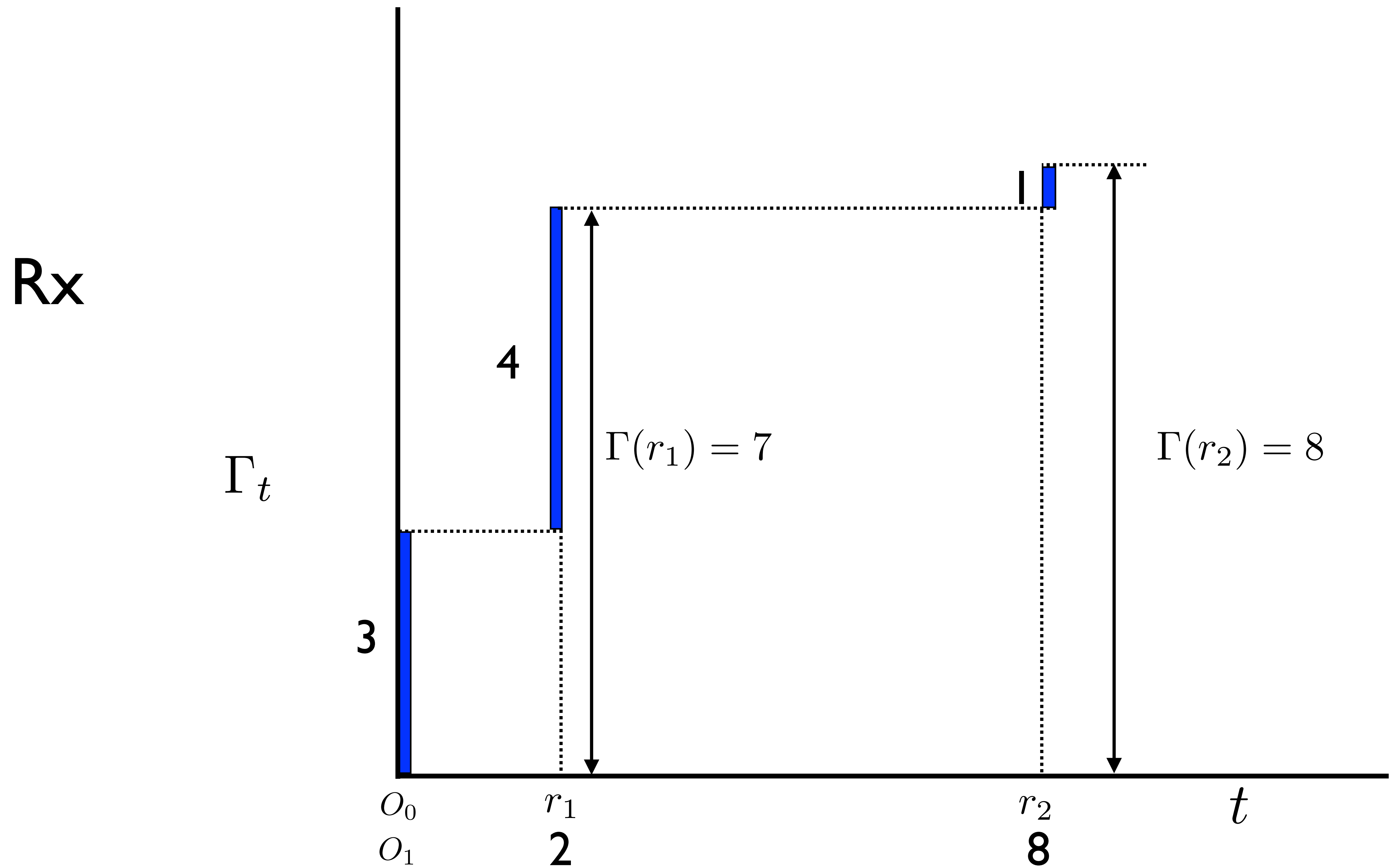
Example for O's

Rx



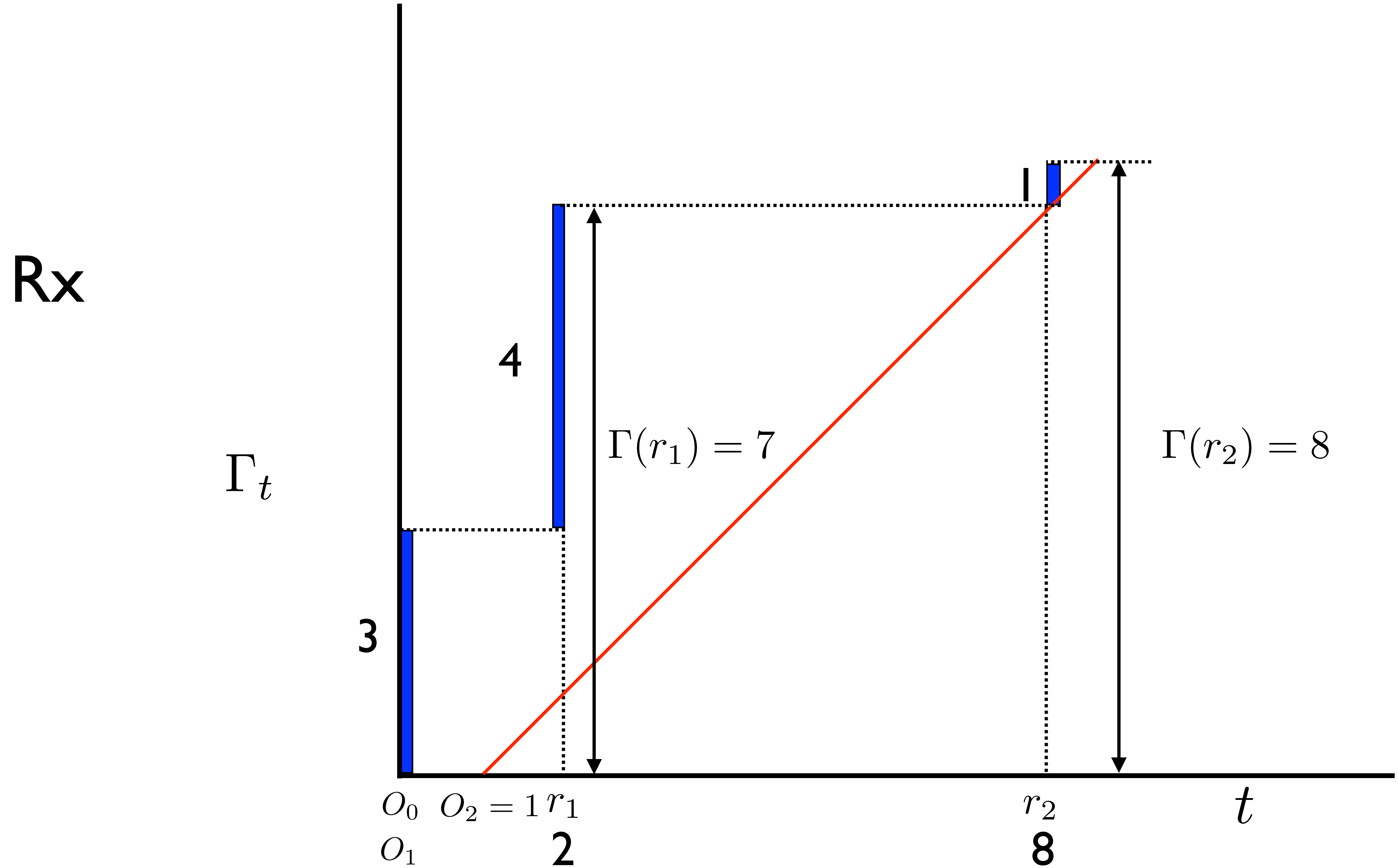
O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Example for O's



O$_i$ be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O$_i$ to O$_i$ + $\Gamma(r_i)$

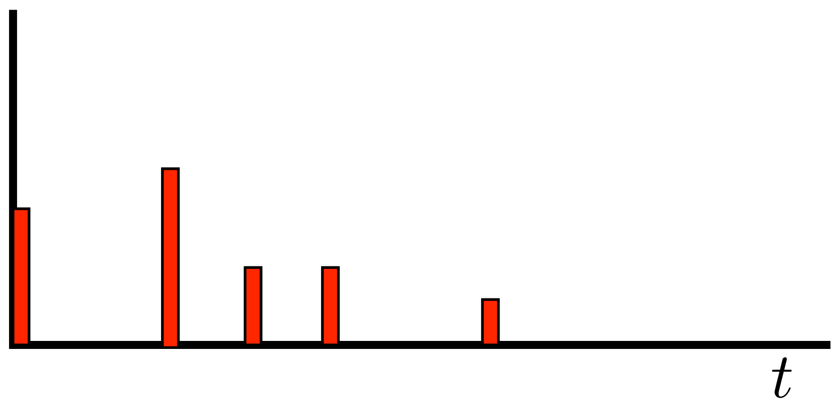
Example for O's



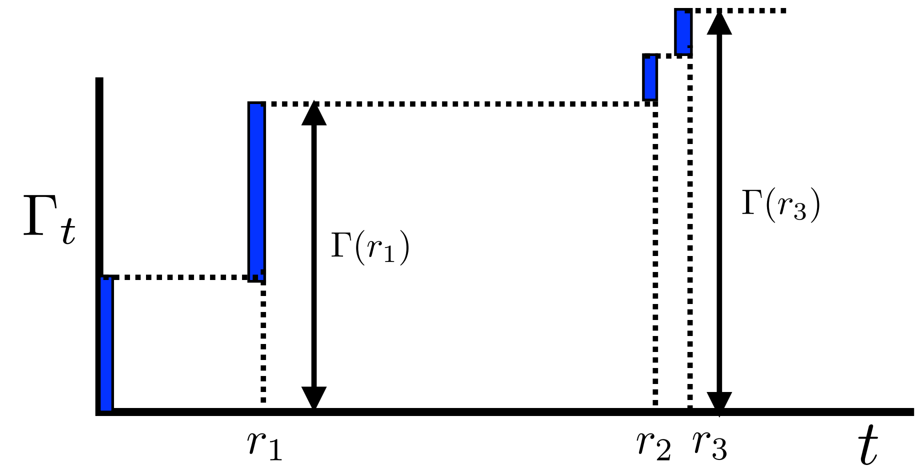
O_i be the earliest time instant such that the receiver can be kept *on* continuously, without any break, from time O_i to $O_i + \Gamma(r_i)$

Offline Algorithm for simpler problem

Tx



count all Tx energy



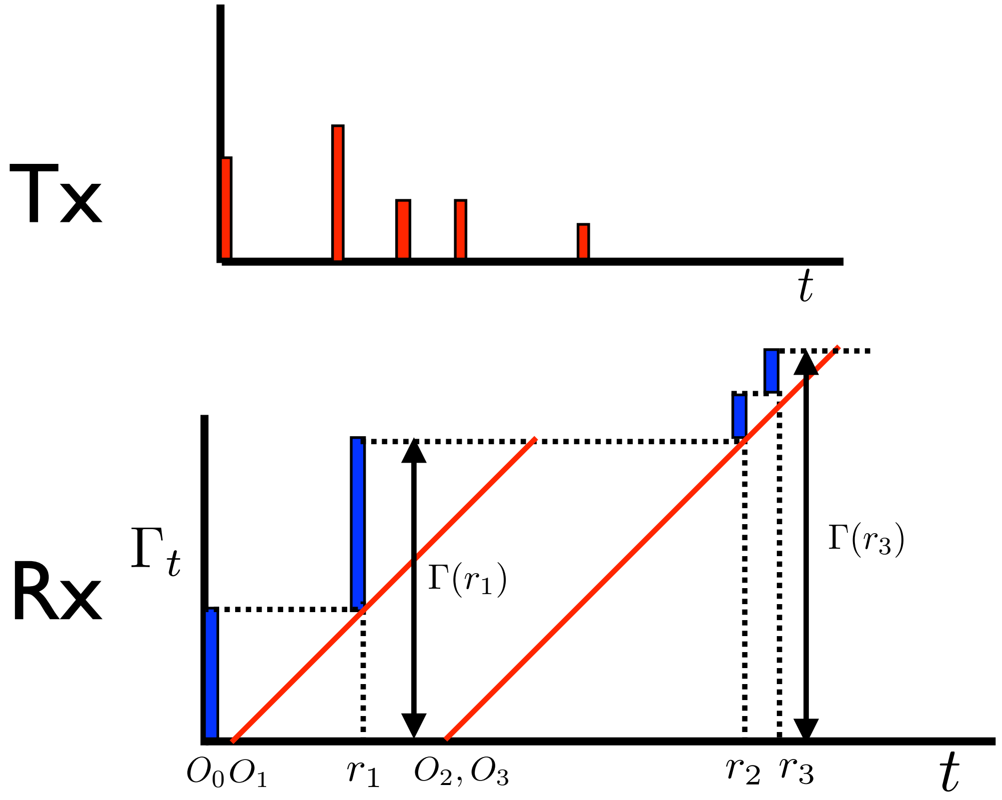
Rx

Earliest time such that Rx-time available till then is sufficient to transmit B bits eventually from Tx

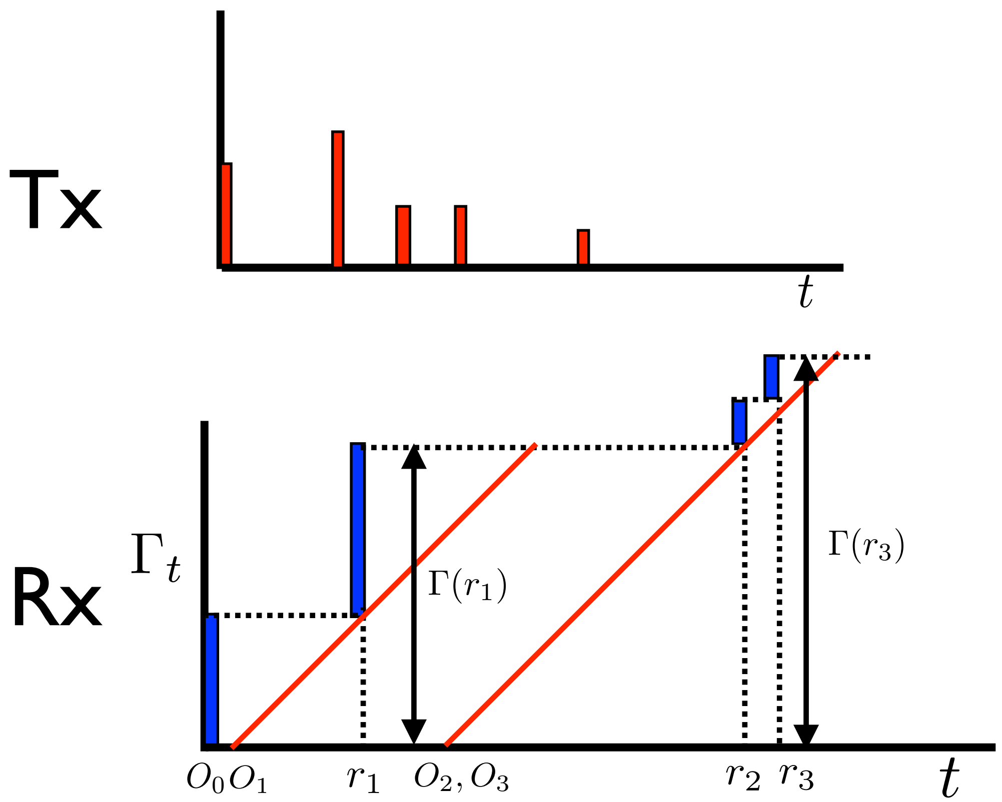
$$i_0 = \min \left\{ i : \lim_{t \to \infty} \Gamma(r_i) g \left(\frac{E(t)}{\Gamma(r_i)} \right) \geq B \right\}$$

Claim 1: If problem is feasible then $i_0 < \infty$

Offline Algorithm for simpler problem



Offline Algorithm for simpler problem

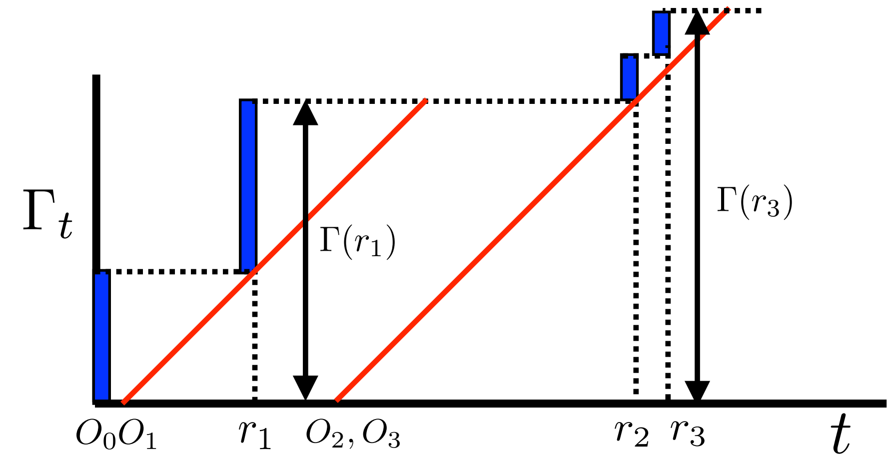


For $i \geq i_0$

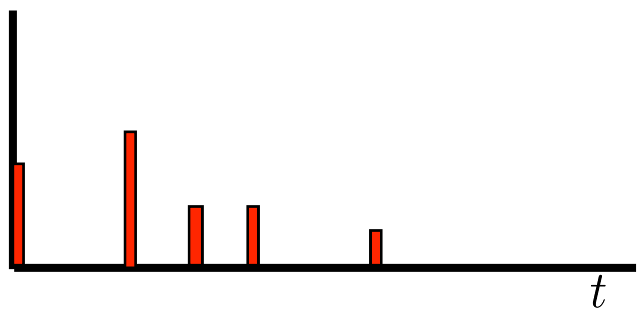
OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

OPT_i Optimal offline solution for OFF_i

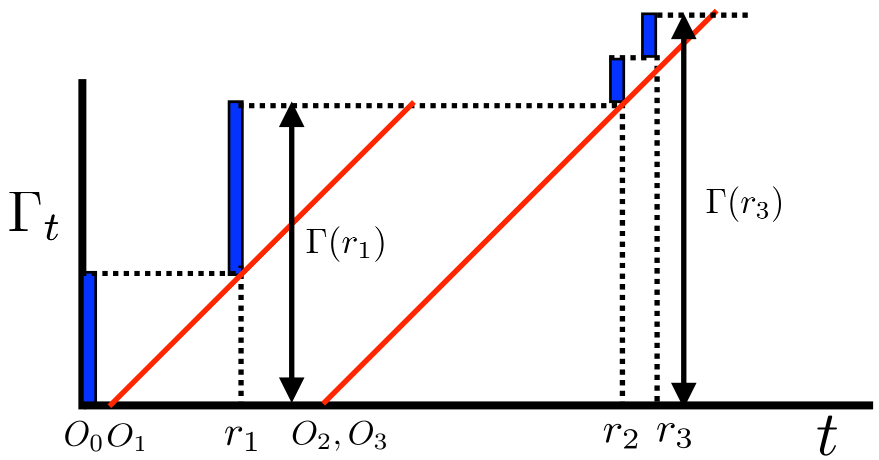
Offline Algorithm for simpler problem



Tx



Rx

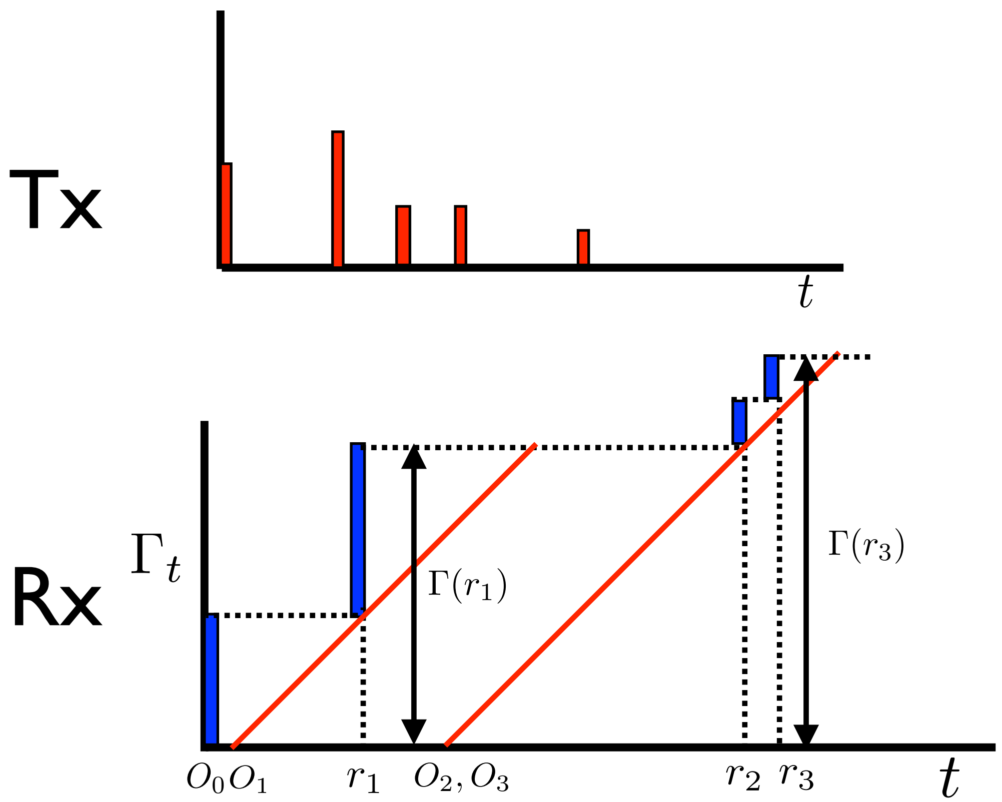


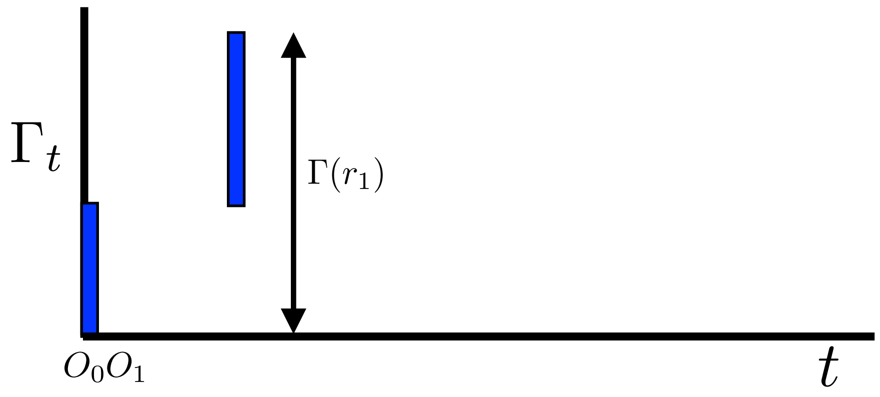
For $i \geq i_0$

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

OPT_i Optimal offline solution for OFF_i

Offline Algorithm for simpler problem



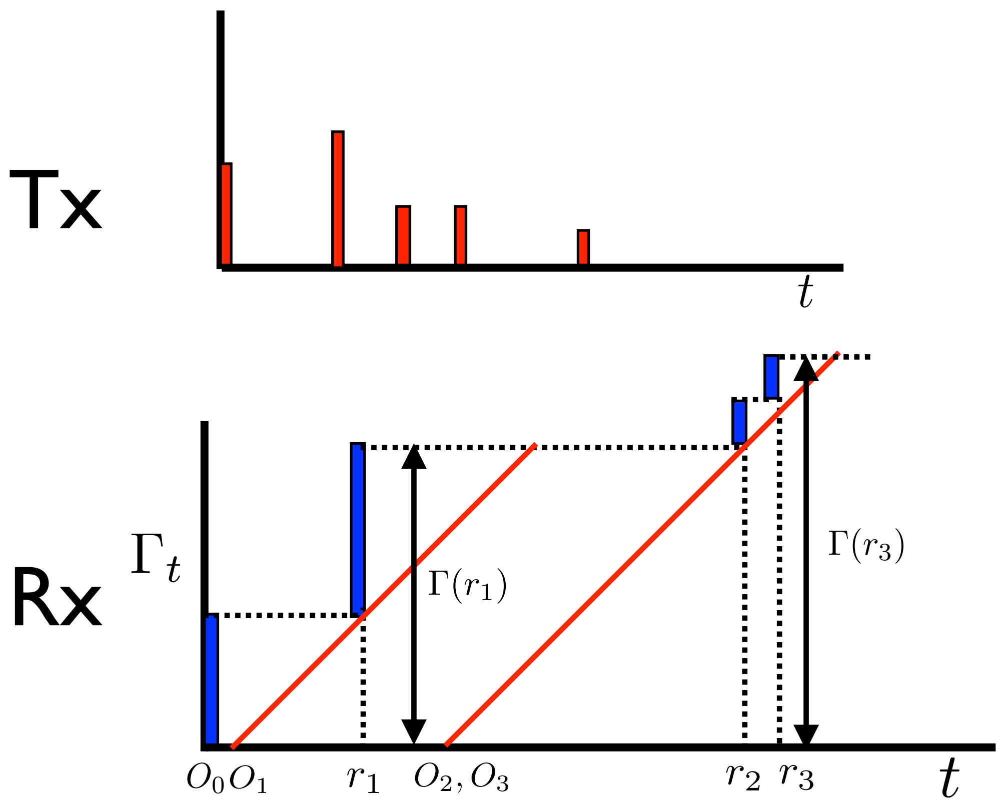


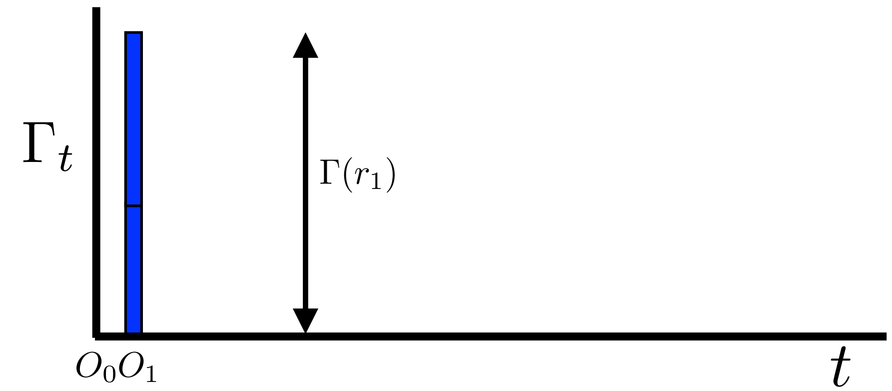
For $i \geq i_0$

OFF$_i$ Starting time O$_i$, only one receiver energy harvest of $\Gamma(r_i)$

OPT$_i$ Optimal offline solution for OFF$_i$

Offline Algorithm for simpler problem





For $i \geq i_0$

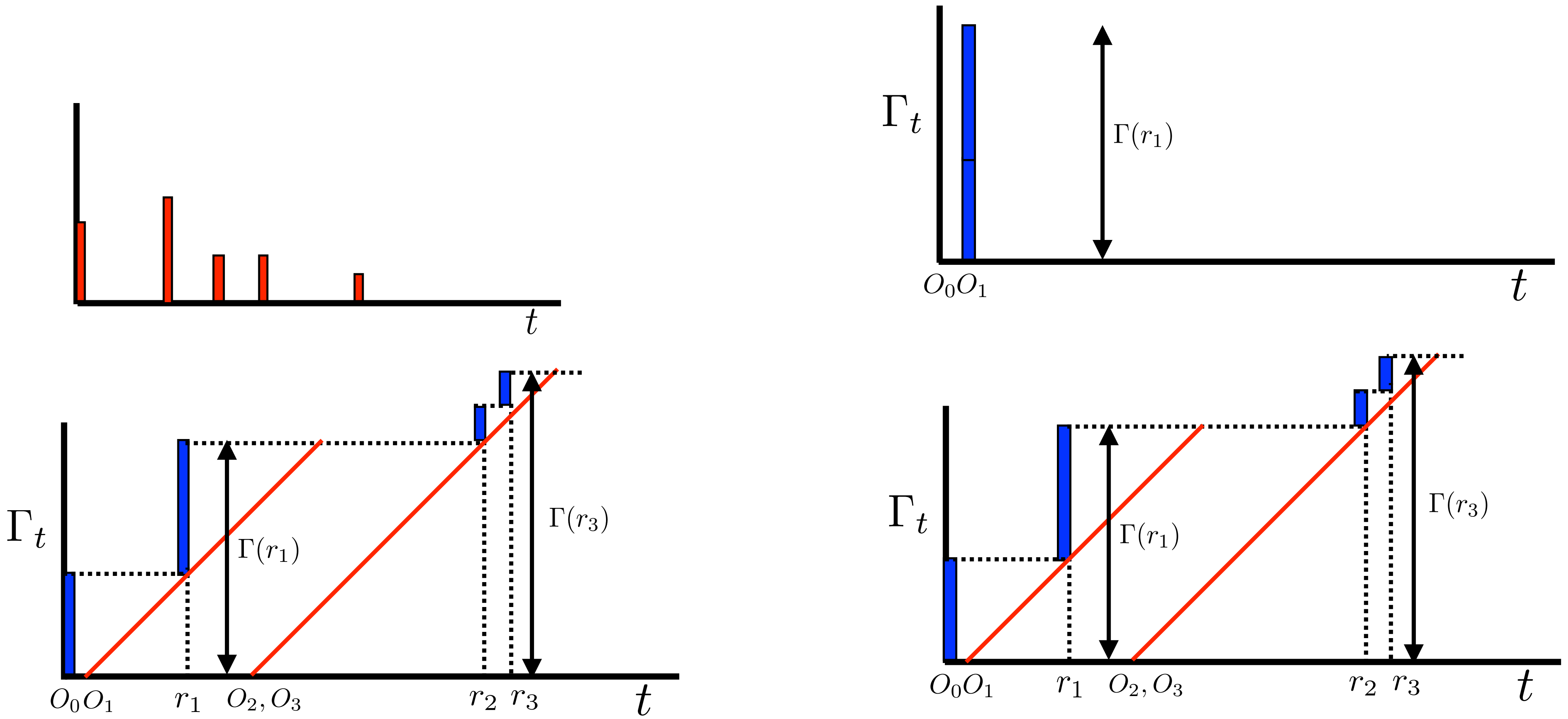
OFF$_i$ Starting time O$_i$, only one receiver energy harvest of $\Gamma(r_i)$

OPT$_i$ Optimal offline solution for OFF$_i$

Offline Algorithm for simpler problem

Tx

Rx

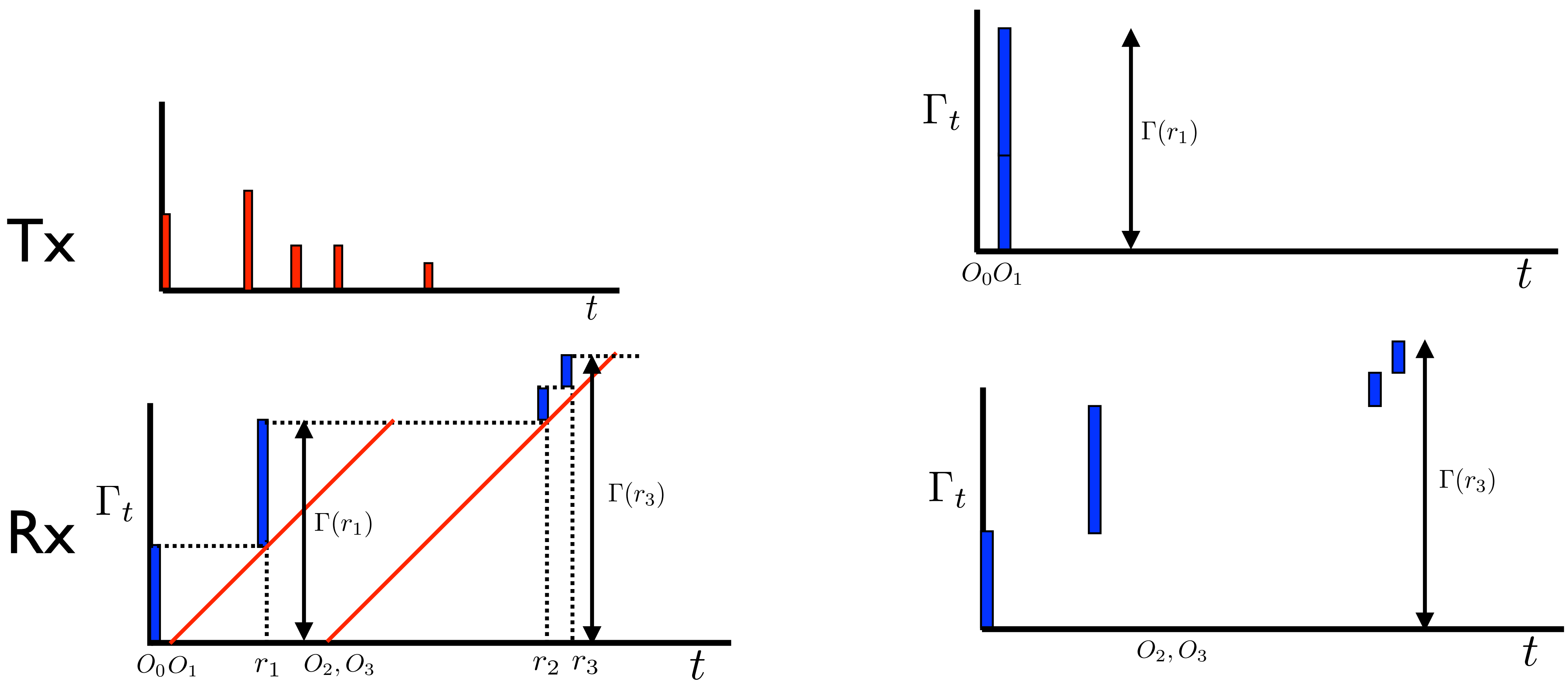


For $i \geq i_0$

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

OPT_i Optimal offline solution for OFF_i

Offline Algorithm for simpler problem



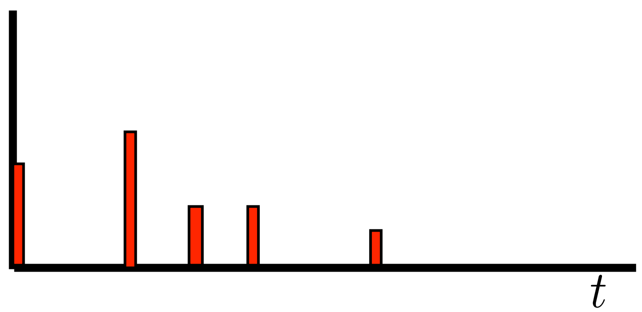
For $i \geq i_0$

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

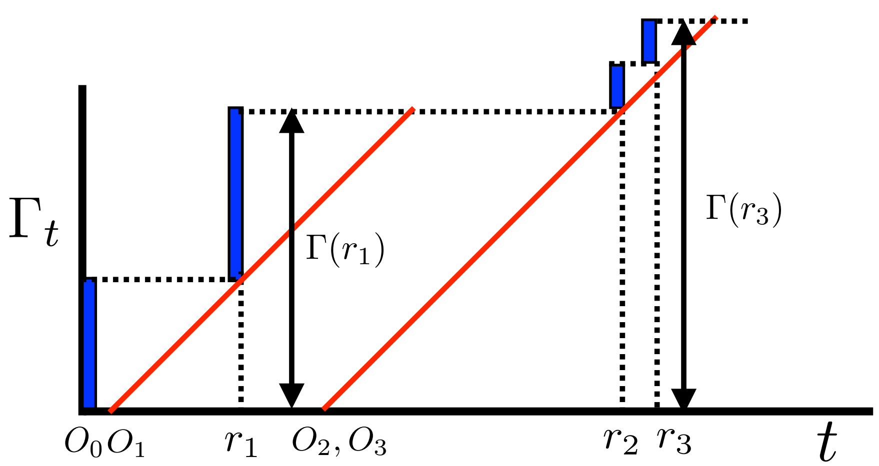
OPT_i Optimal offline solution for OFF_i

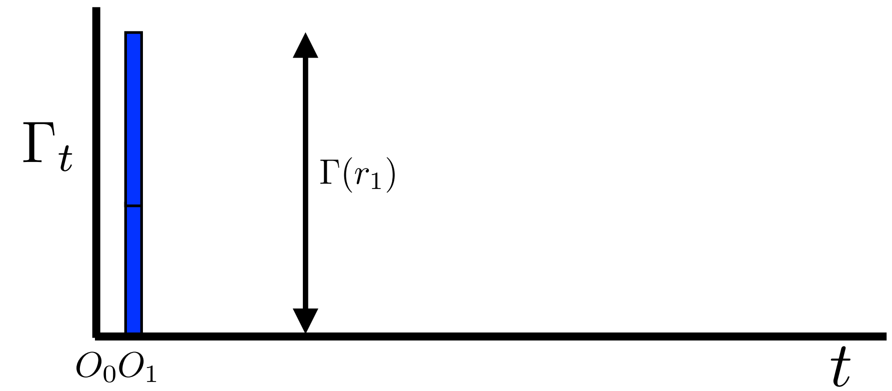
Offline Algorithm for simpler problem

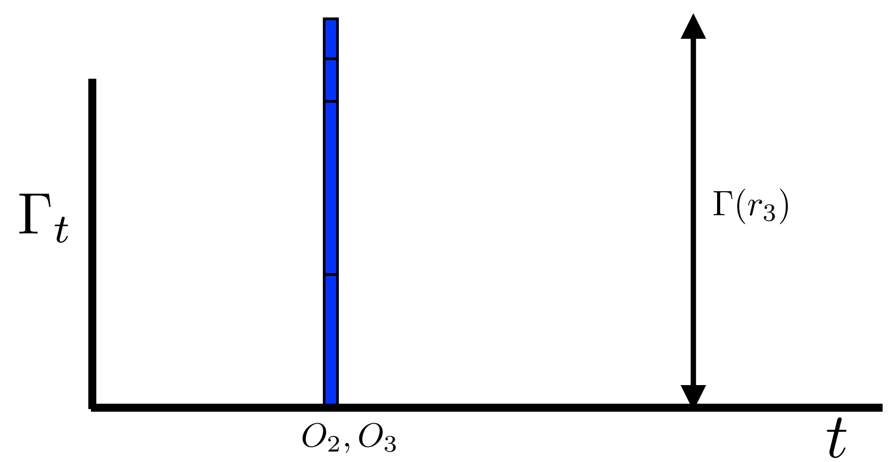
Tx



Rx





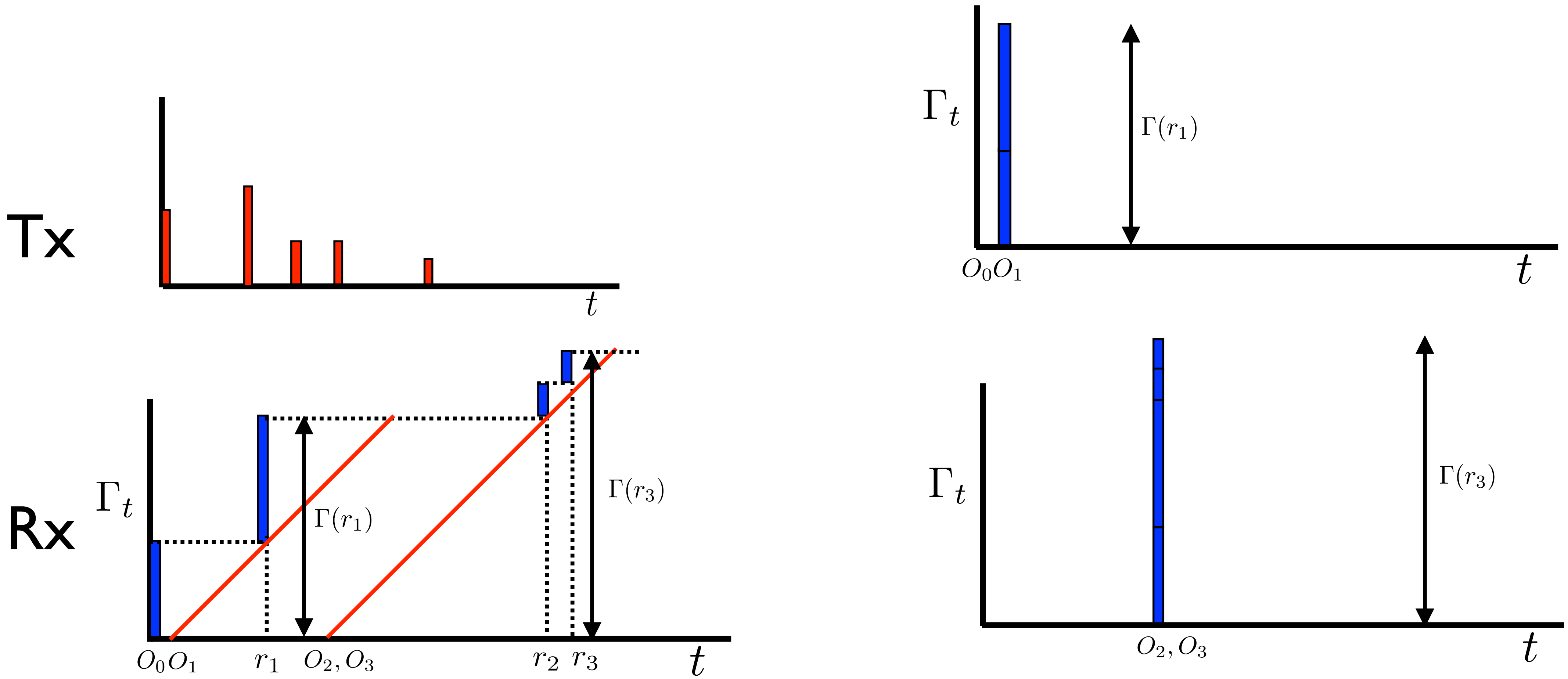


For $i \geq i_0$

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

OPT_i Optimal offline solution for OFF_i

Offline Algorithm for simpler problem



For $i \geq i_0$

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

OPT_i Optimal offline solution for OFF_i

Claim 2: If problem is feasible then $\exists\, i, \mathsf{OPT}_i = \mathsf{OPT}$

Idea for Claim 2

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\ i, \mathsf{OPT}_i = \mathsf{OPT}$

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\ i, \mathsf{OPT}_i = \mathsf{OPT}$

Contradiction : Assume none of OPT_i are optimal

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\ i, \mathsf{OPT}_i = \mathsf{OPT}$

Contradiction : Assume none of OPT_i are optimal

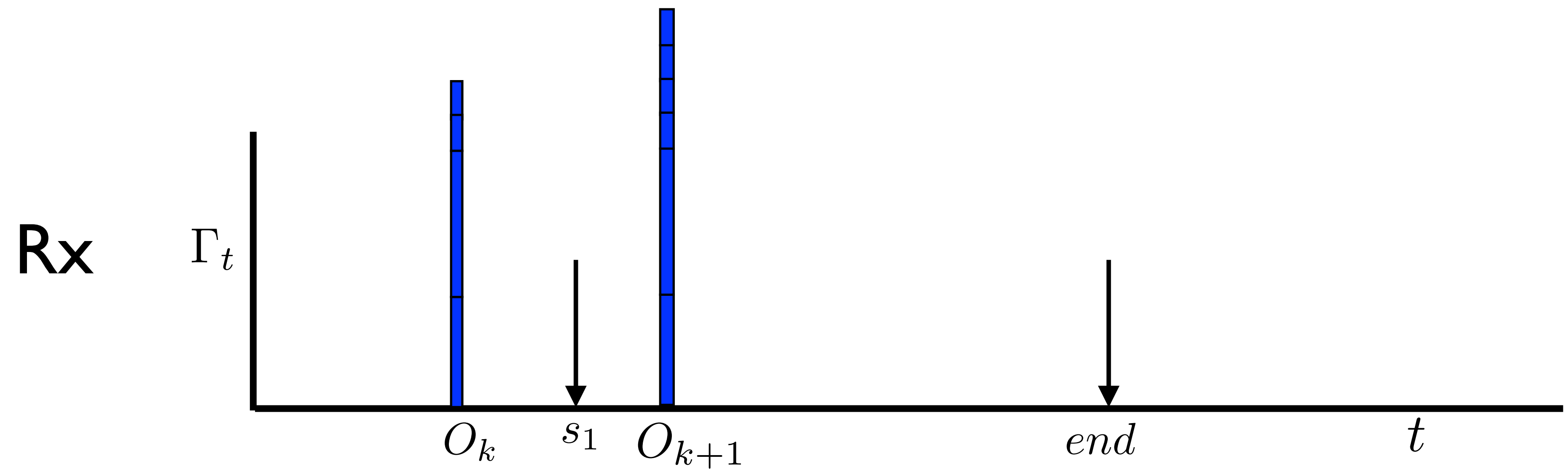
Let OPT start at s1 $O_k \leq s_1 \leq O_{k+1}$

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\ i, \mathsf{OPT}_i = \mathsf{OPT}$

Contradiction : Assume none of OPT_i are optimal

Let OPT start at s1 $O_k \leq s_1 \leq O_{k+1}$

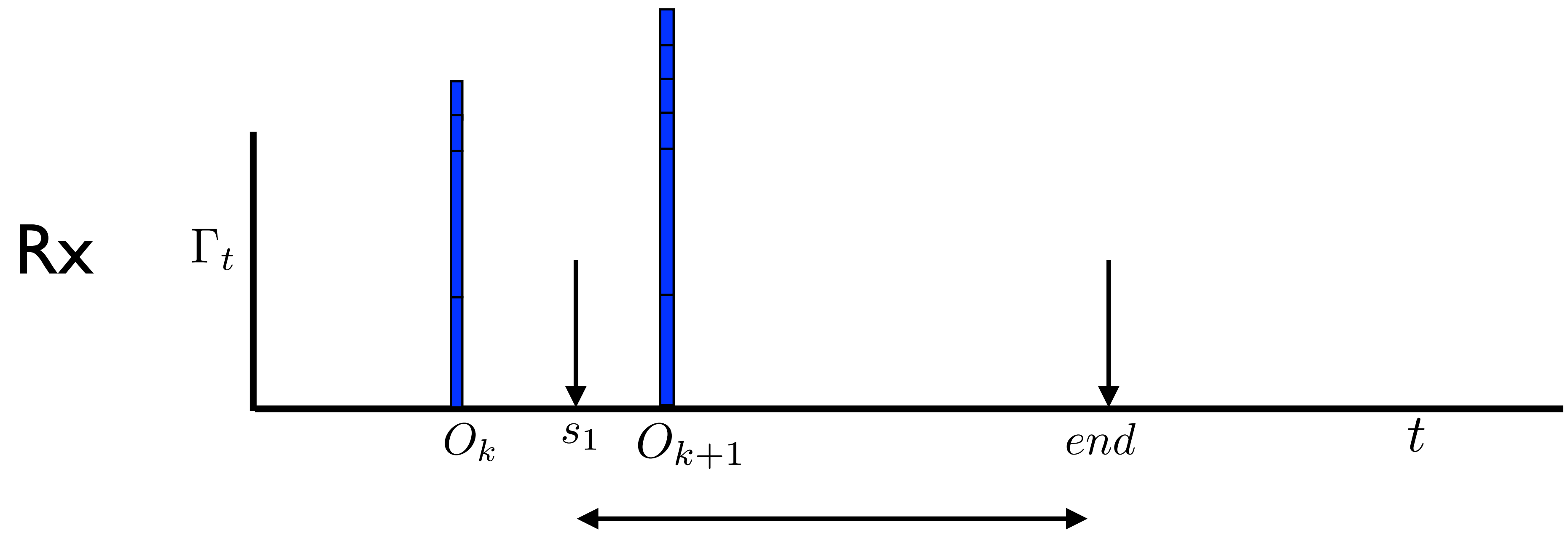


Idea for Claim 2

Claim 2: If problem is feasible then $\exists\, i, \mathrm{OPT}_i = \mathrm{OPT}$

Contradiction : Assume none of OPT_i are optimal

Let OPT start at s1 $\quad O_k \le s_1 \le O_{k+1}$



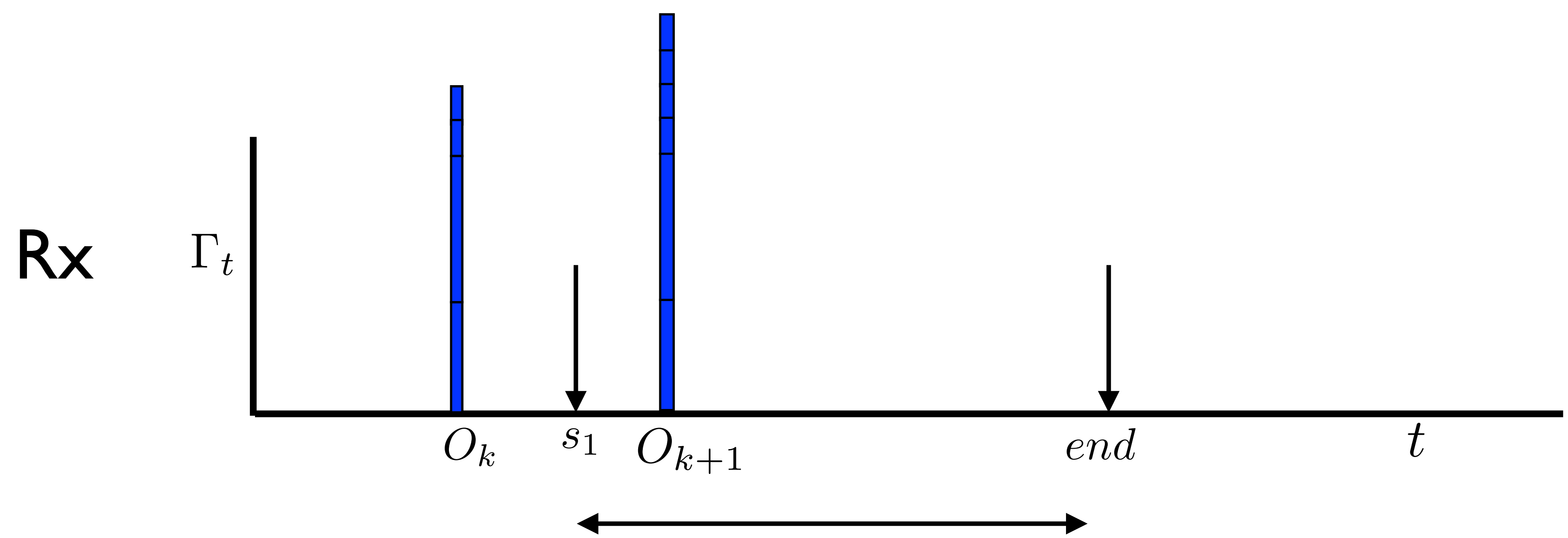
is less than or equal to $\Gamma(r_k)$ $\quad$ by defn. of O_{k+1}

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\, i, \text{OPT}_i = \text{OPT}$

Contradiction : Assume none of OPT_i are optimal

Let OPT start at s_1 $\quad O_k \leq s_1 \leq O_{k+1}$



is less than or equal to $\Gamma(r_k)$ by defn. of O_{k+1}

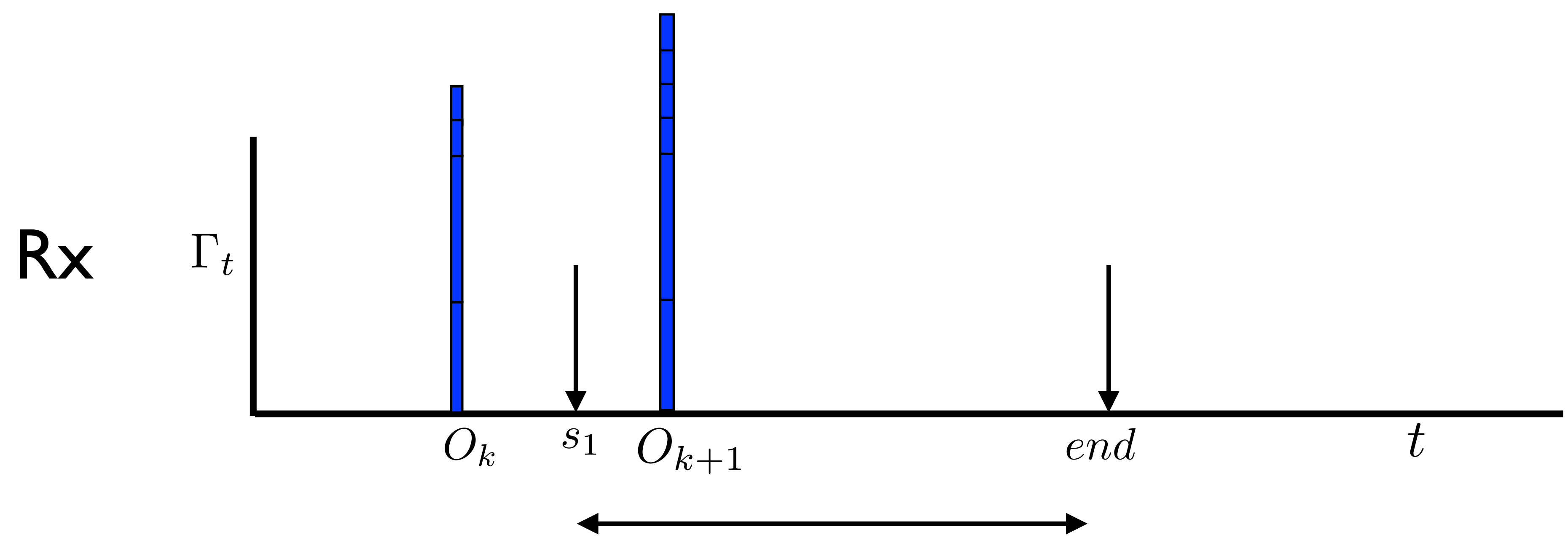
Since OPT starts after O_k and uses less than $\Gamma(r_k)$ time, it is feasible to OFF_k

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\, i, \mathrm{OPT}_i = \mathrm{OPT}$

Contradiction : Assume none of OPT_i are optimal

Let OPT start at s_1 $\quad O_k \leq s_1 \leq O_{k+1}$



is less than or equal to $\Gamma(r_k)$ by defn. of O_{k+1}

Since OPT starts after O_k and uses less than $\Gamma(r_k)$ time, it is feasible to OFF_k

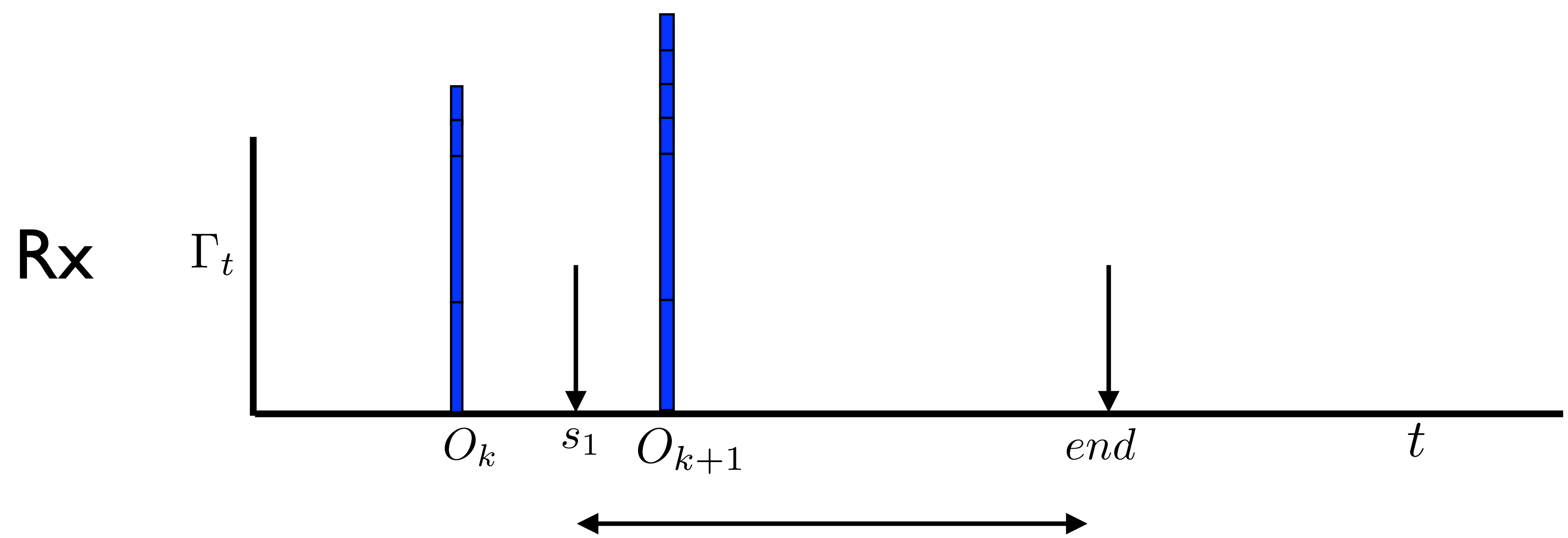
$$T(\mathrm{OPT}_k) \leq T(\mathrm{OPT})$$

Idea for Claim 2

Claim 2: If problem is feasible then $\exists\, i, \mathsf{OPT}_i = \mathsf{OPT}$

Contradiction : Assume none of OPT_i are optimal

Let OPT start at s_1 $\quad O_k \leq s_1 \leq O_{k+1}$



is less than or equal to $\Gamma(r_k)$ by defn. of O_{k+1}

Since OPT starts after O_k and uses less than $\Gamma(r_k)$ time, it is feasible to OFF_k

$$T(\mathsf{OPT}_k) \leq T(\mathsf{OPT})$$

But since none of OPT_i are optimal to OFF $\quad T(\mathsf{OPT}_k) > T(\mathsf{OPT})$

Idea for Claim 2

Idea for Claim 2

So OPT is one among OPT_i

Idea for Claim 2

So OPT is one among OPT_i which one ?

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

Idea for Claim 2

So OPT is one among OPT_i which one ?

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

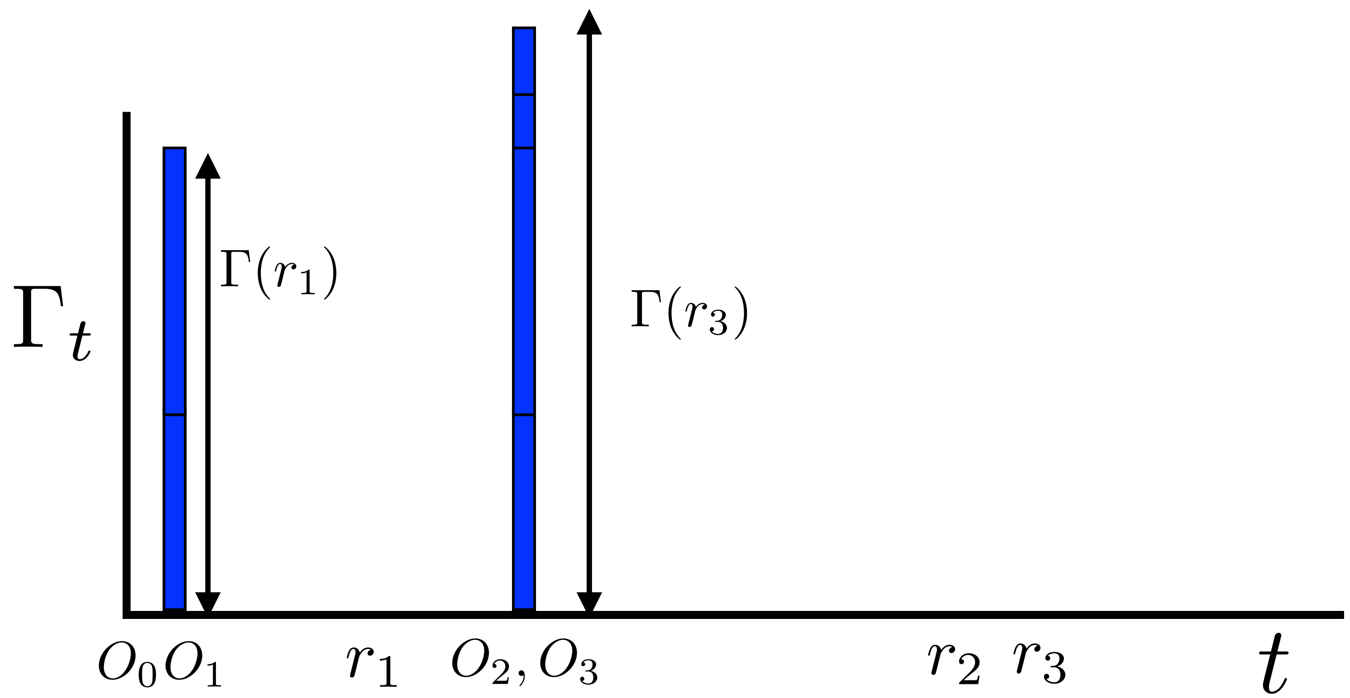
optimal: min *i* for which OPT_i starts before O_{i+1}

Idea for Claim 2

So OPT is one among OPT_i which one ?

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

optimal: min *i* for which OPT_i starts before O_{i+1}

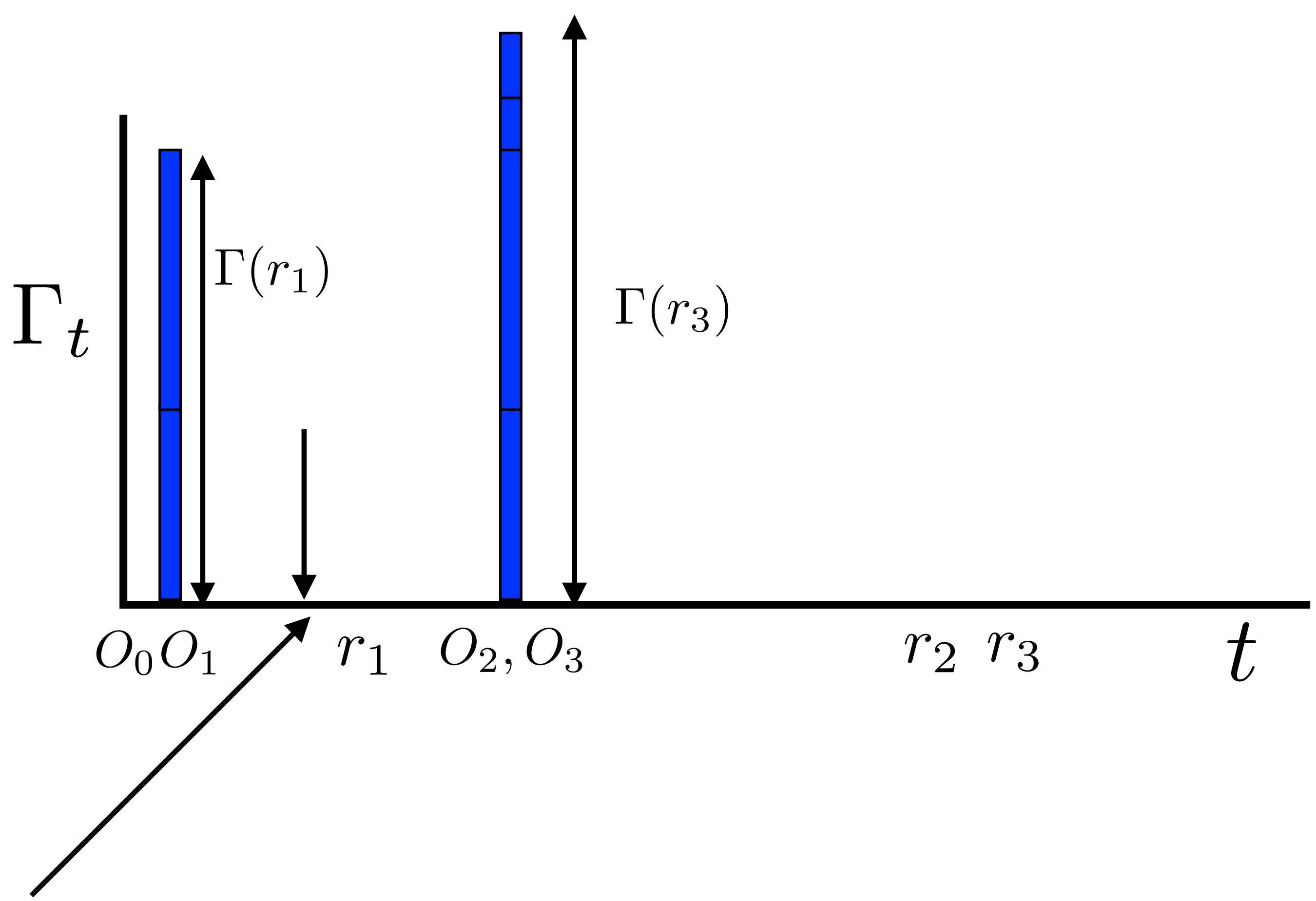


Idea for Claim 2

So OPT is one among OPT_i which one ?

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

optimal: min *i* for which OPT_i starts before O_{i+1}



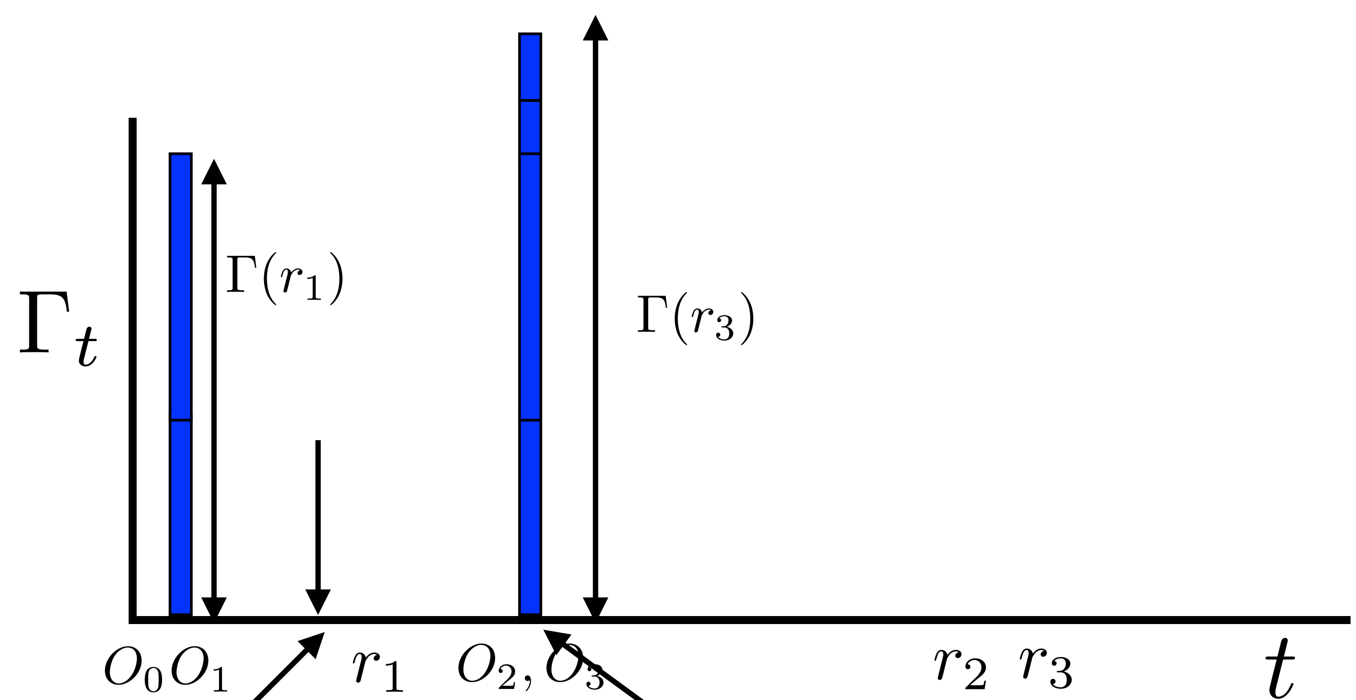
If OPT_1 starts here, then its OPT

Idea for Claim 2

So OPT is one among OPT_i which one ?

OFF_i Starting time O_i, only one receiver energy harvest of $\Gamma(r_i)$

optimal: min *i* for which OPT_i starts before O_{i+1}



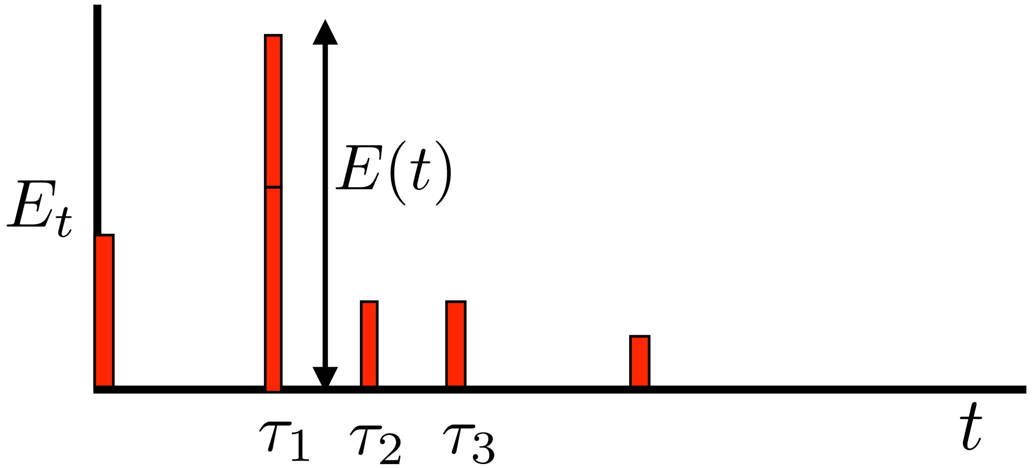
Otherwise if OPT_2 starts here, then its OPT

If OPT_1 starts here, then its OPT

Solving Offline Algorithm with 1 Rx arrival

everything known in future - arrivals epochs and amounts

Tx



$$U(t), C(t)$$

Energy used until time t at Tx, Rx

bits with power p(t)

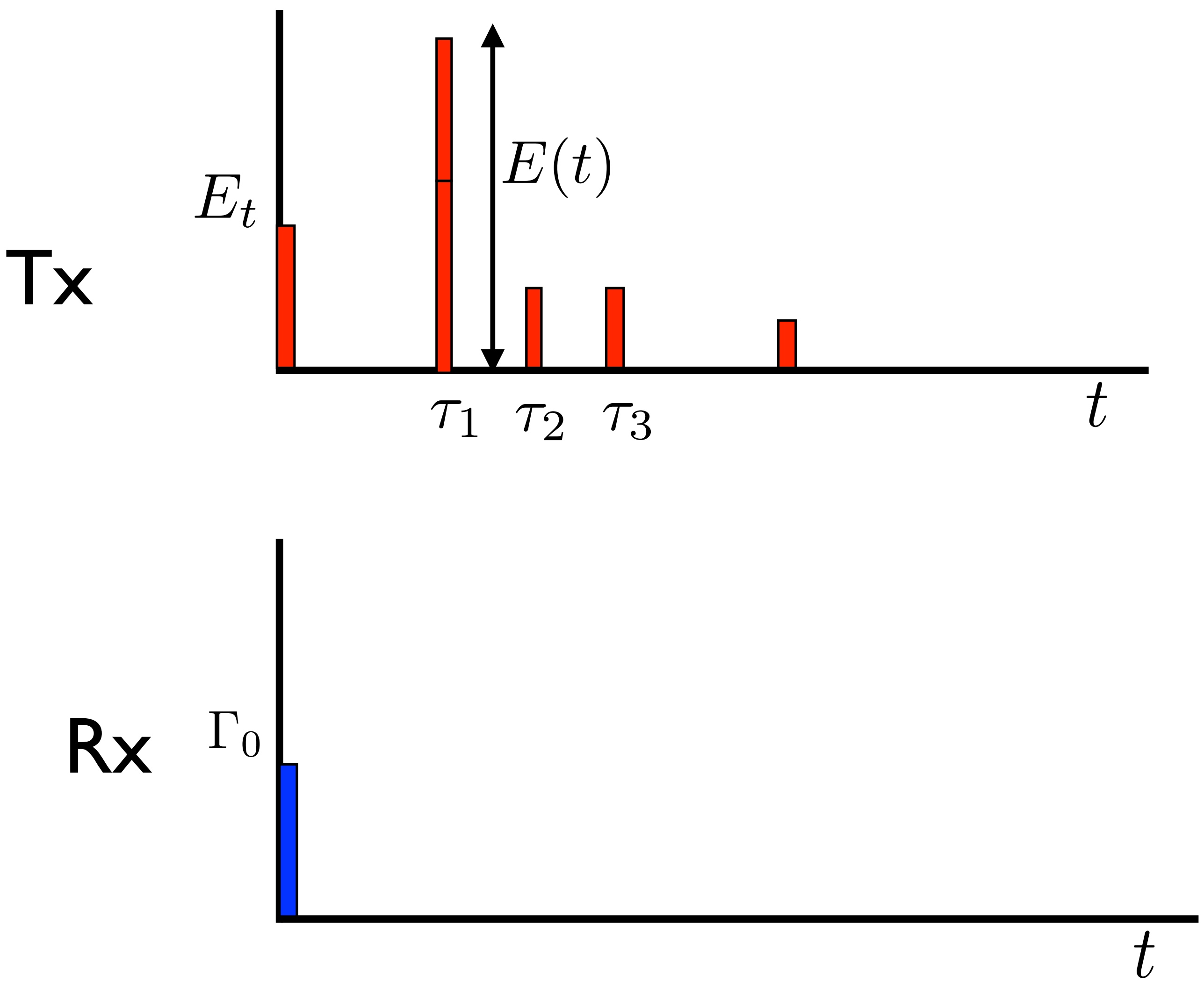
$$r(t) = g(p(t))$$

Rx

$$T^\star = \min_{B(T)=B, U(t)\leq E(t), C(t)\leq \Gamma_0} T$$

Solving Offline Algorithm with 1 Rx arrival

everything known in future - arrivals epochs and amounts



$$U(t), C(t)$$

Energy used until time t at Tx, Rx

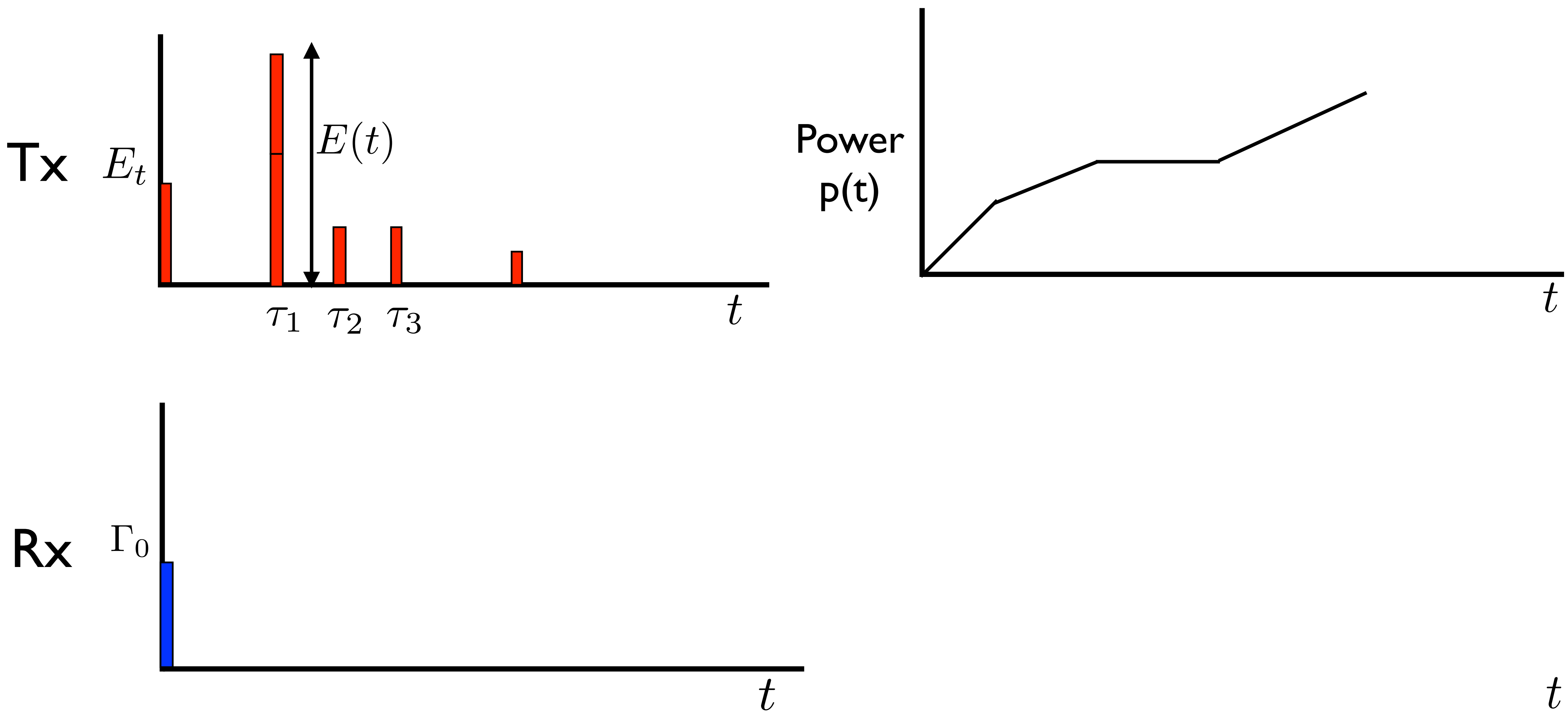
bits with power p(t)

$$r(t) = g(p(t))$$

$$T^\star = \min_{B(T)=B, U(t)\leq E(t), C(t)\leq \Gamma_0} T$$

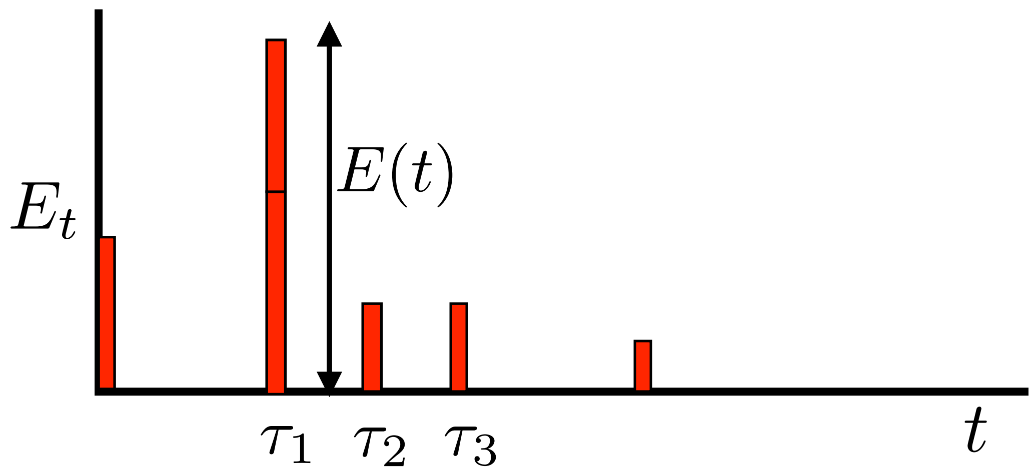
Important Properties

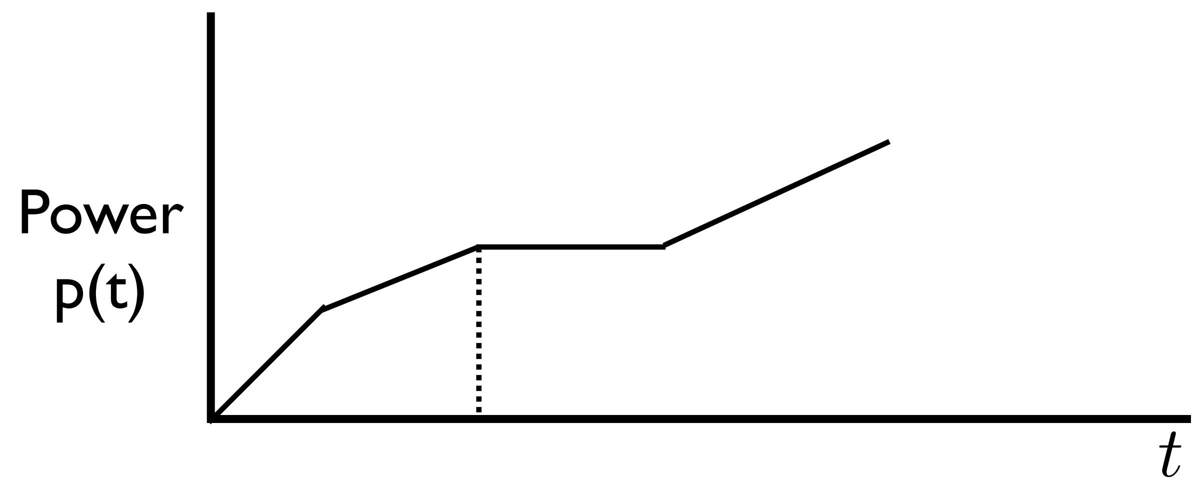
- No stop-start needed
  - Delay the start of transmission if need be



Important Properties

- No stop-start needed
  - Delay the start of transmission if need be

Tx


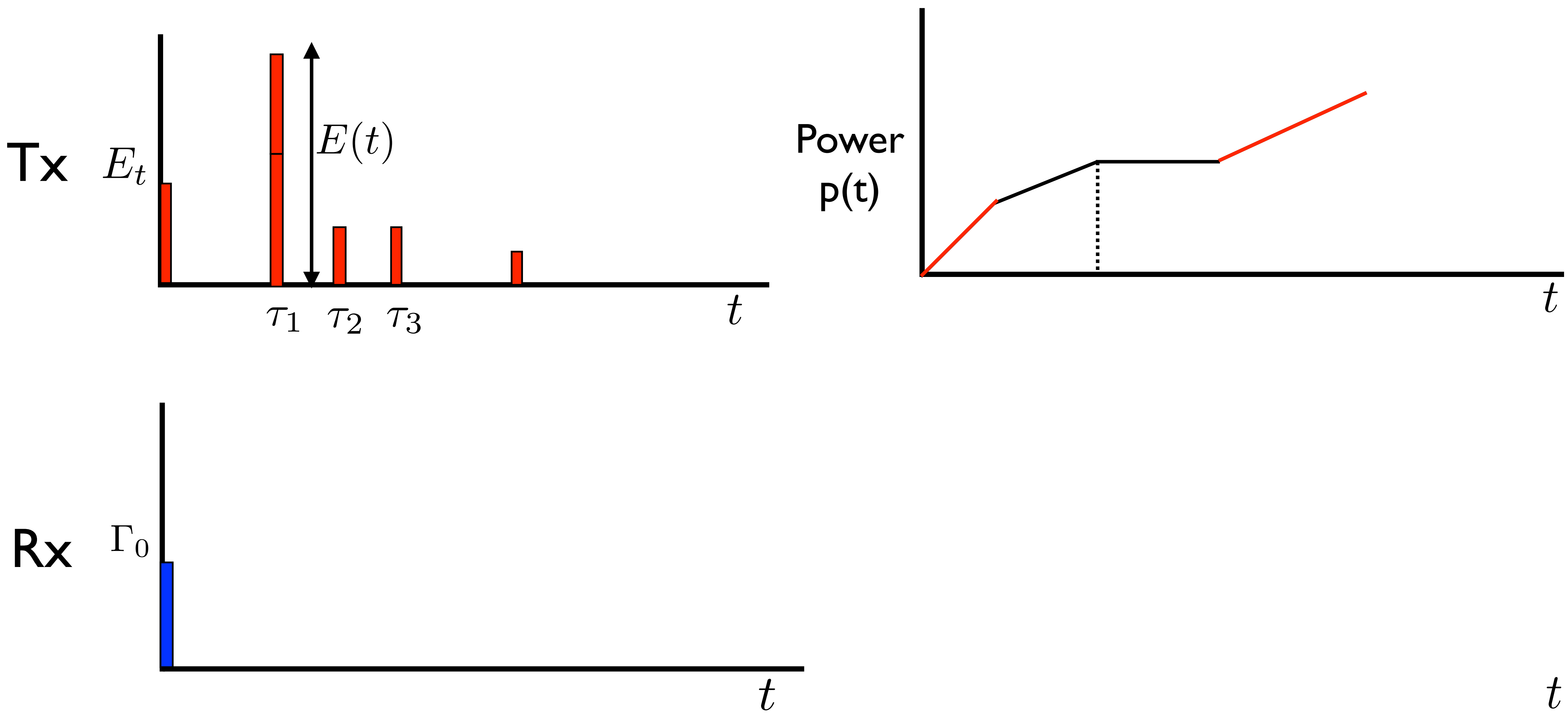


Rx


t

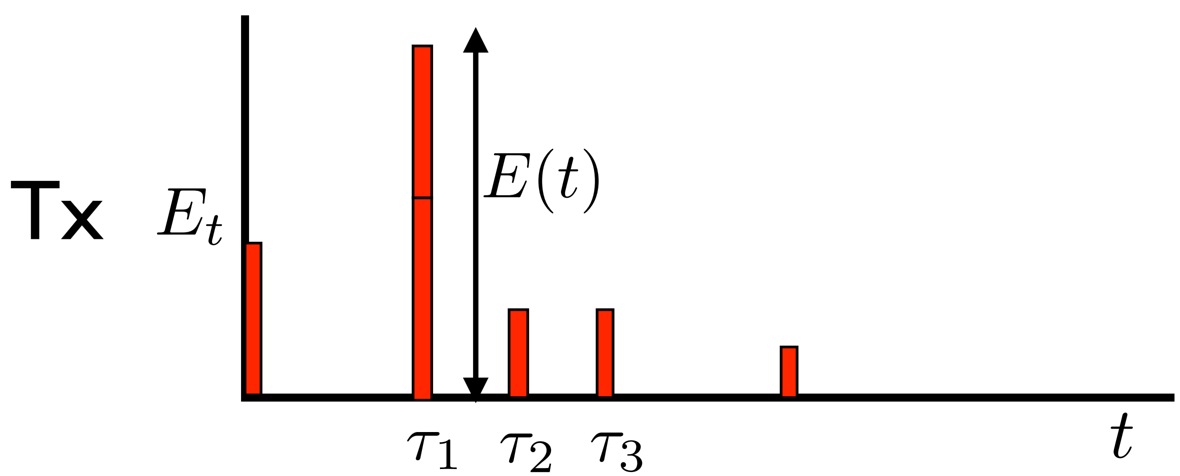
Important Properties

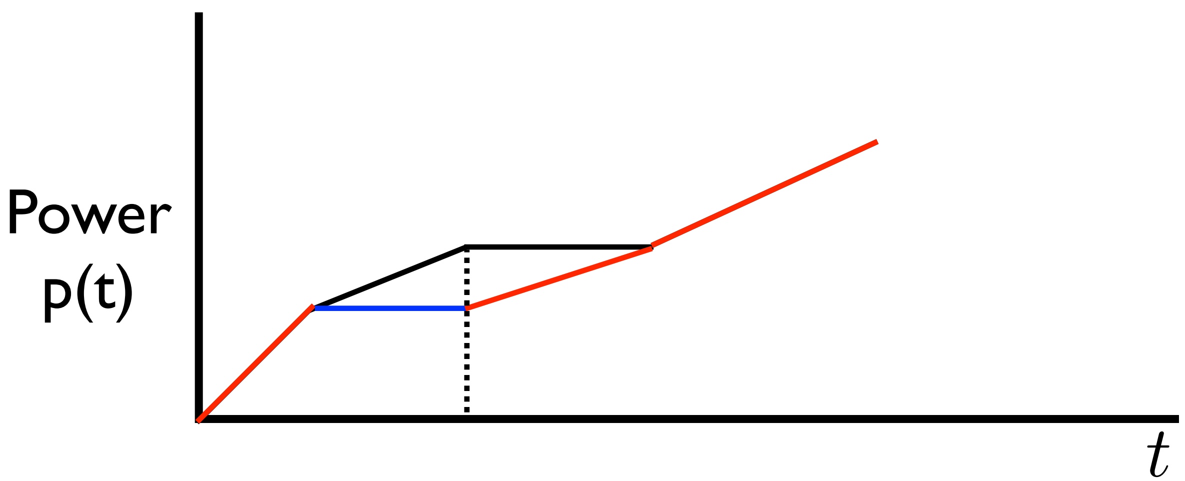
- No stop-start needed
  - Delay the start of transmission if need be



Important Properties

- No stop-start needed
  - Delay the start of transmission if need be

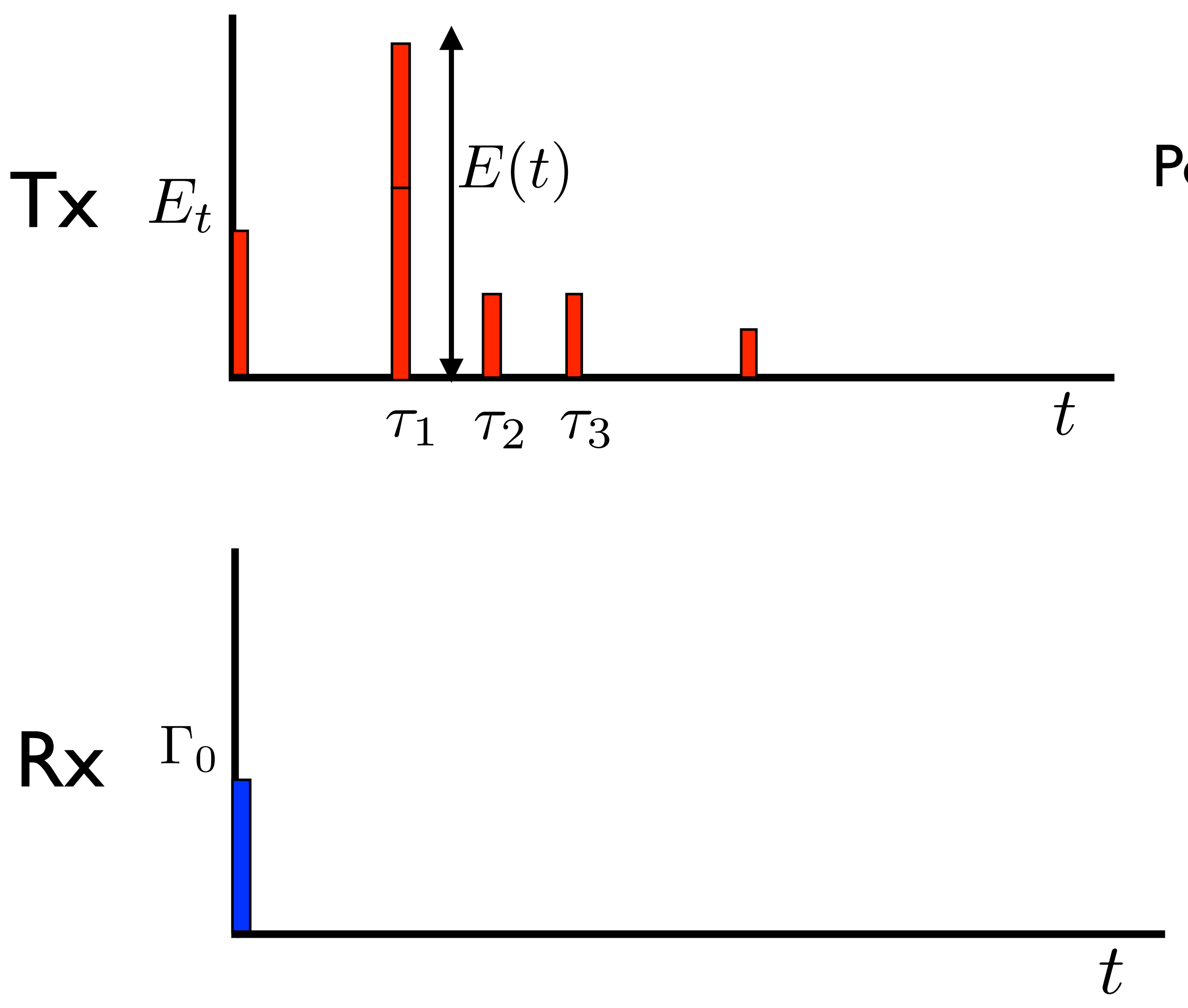


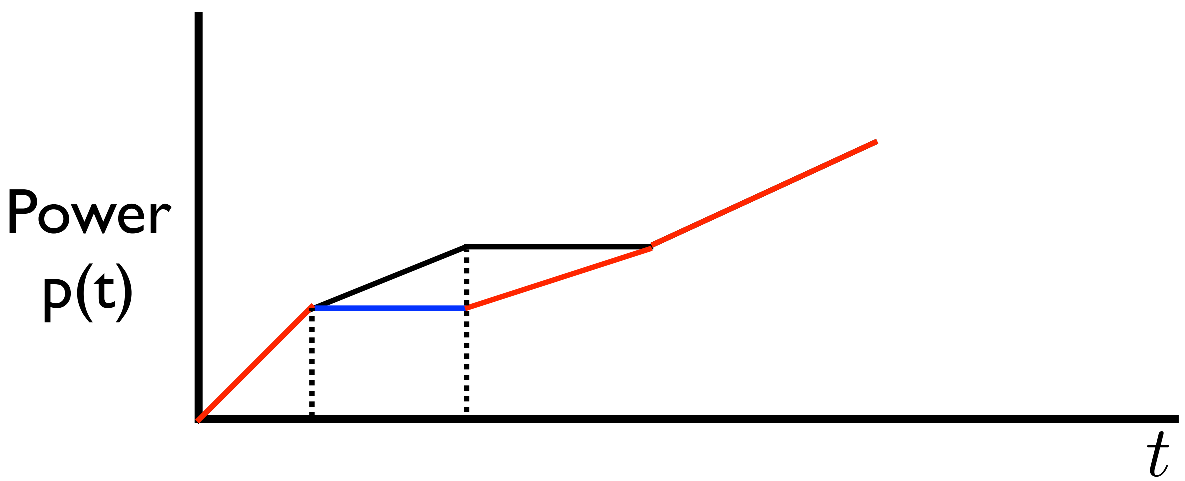


t

Important Properties

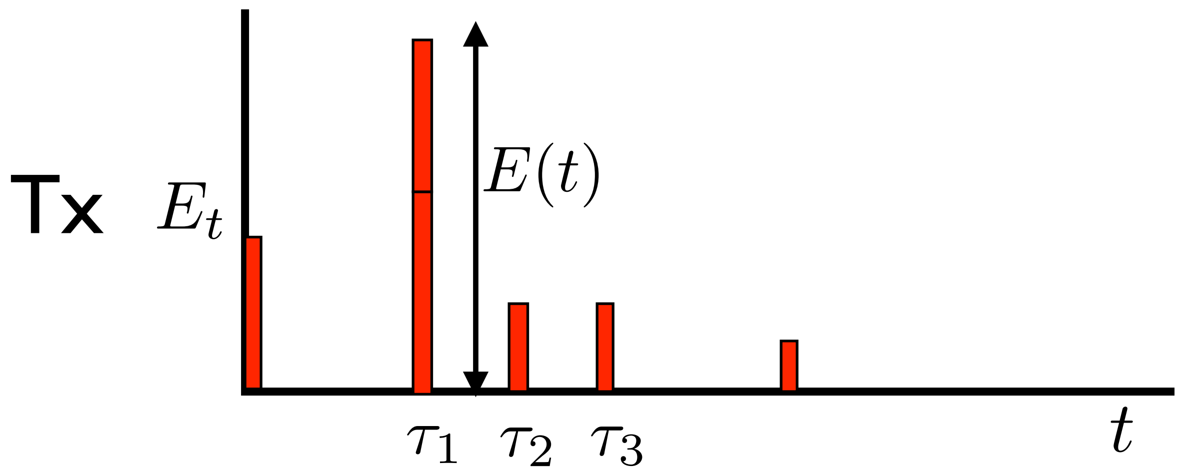
- No stop-start needed
  - Delay the start of transmission if need be

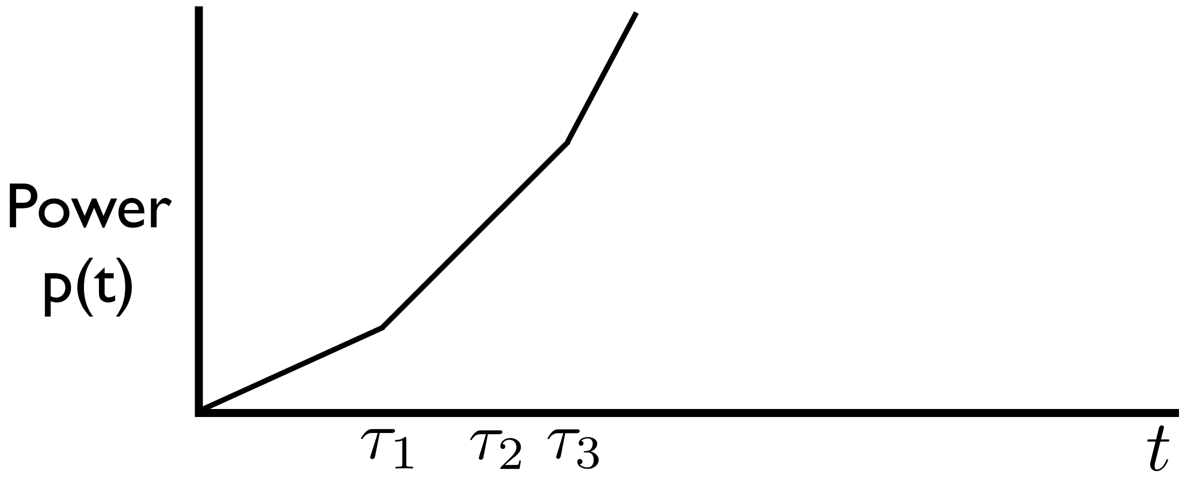




Important Properties

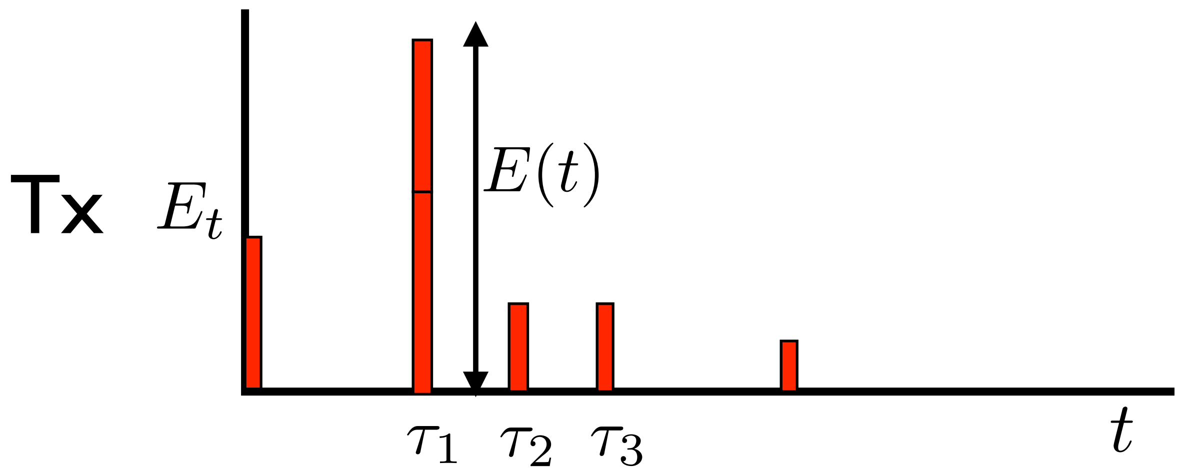
- powers are non-decreasing in time
- powers change only at Tx arrival epoch if at all

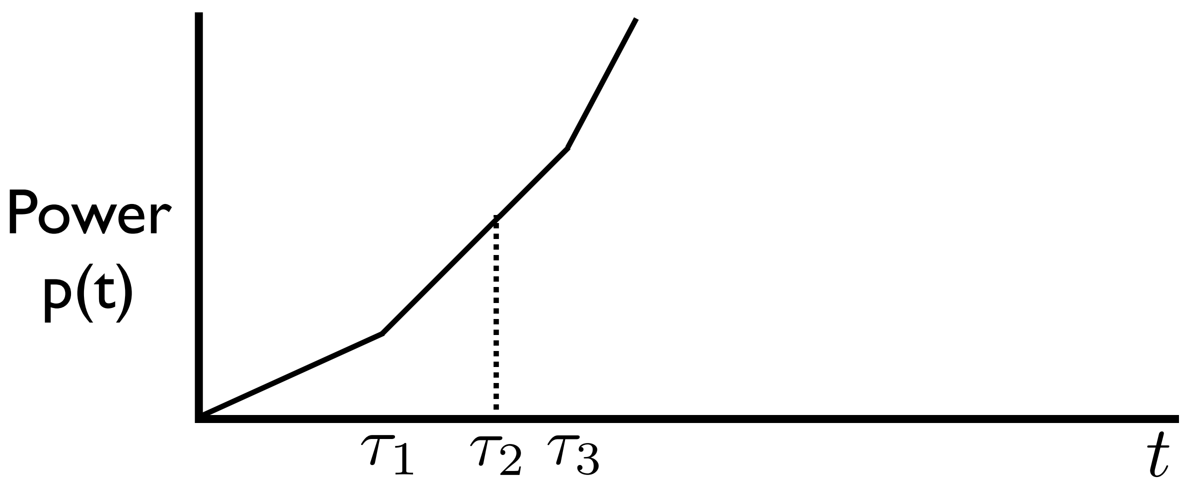




Important Properties

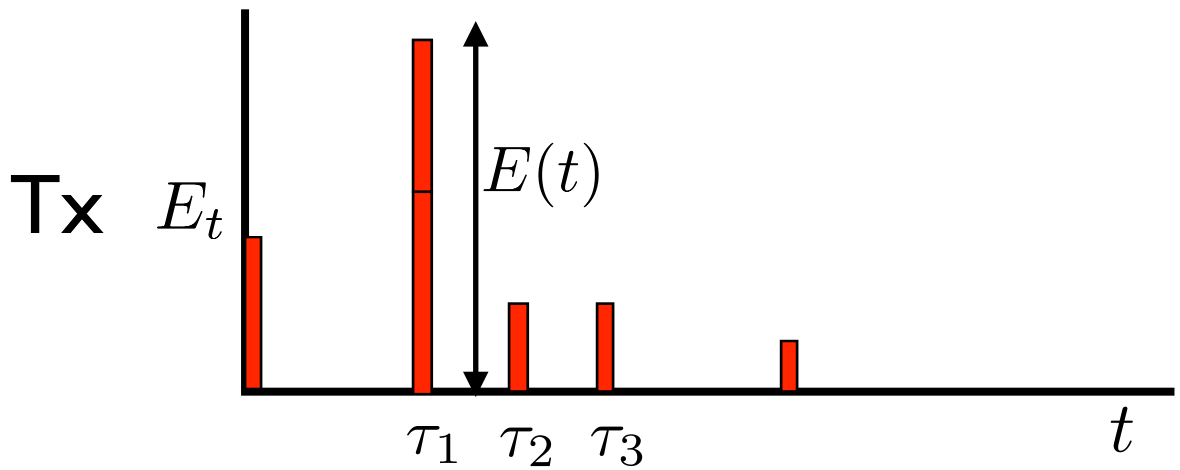
- powers are non-decreasing in time
- powers change only at Tx arrival epoch if at all

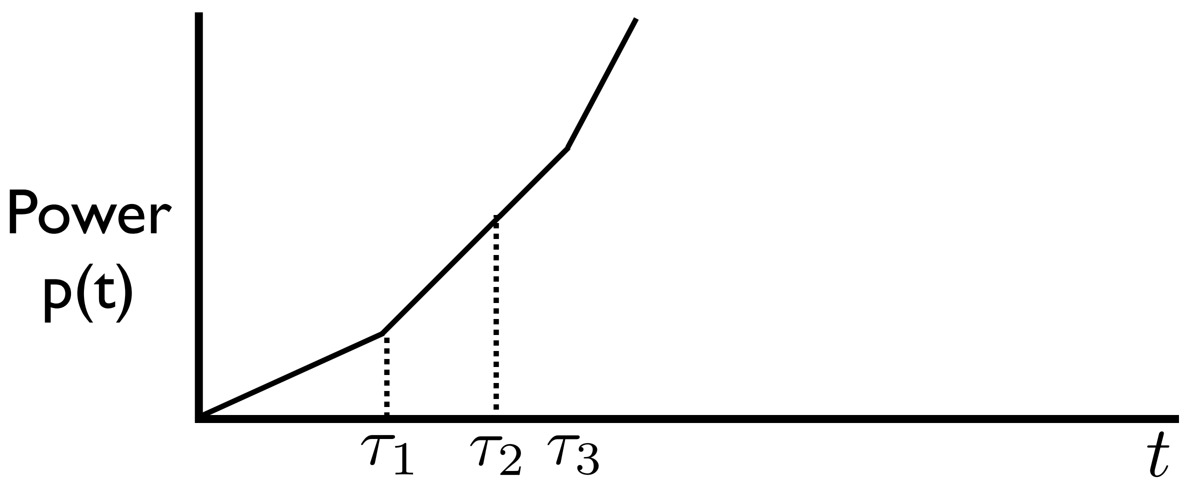




Important Properties

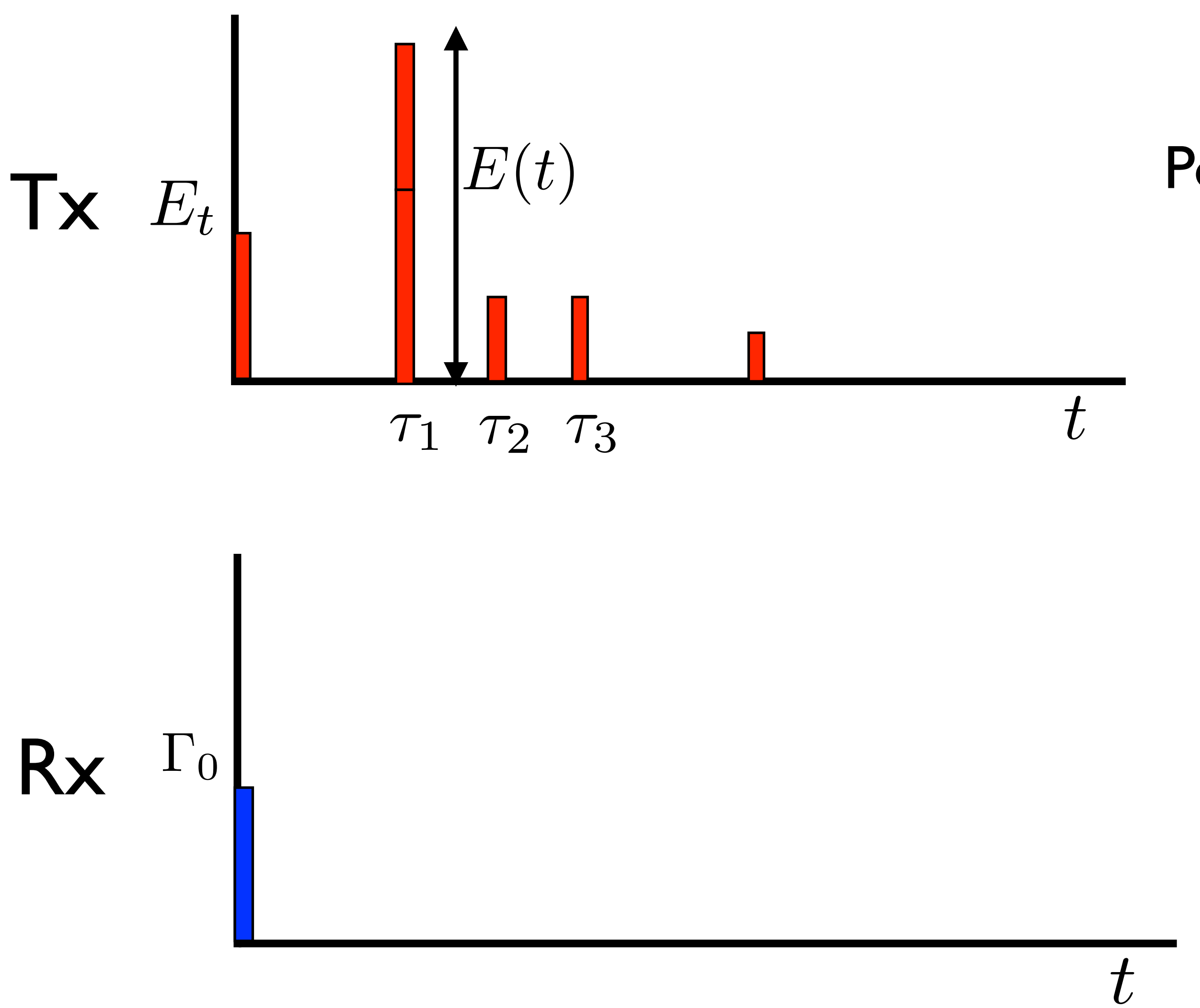
- powers are non-decreasing in time
- powers change only at Tx arrival epoch if at all

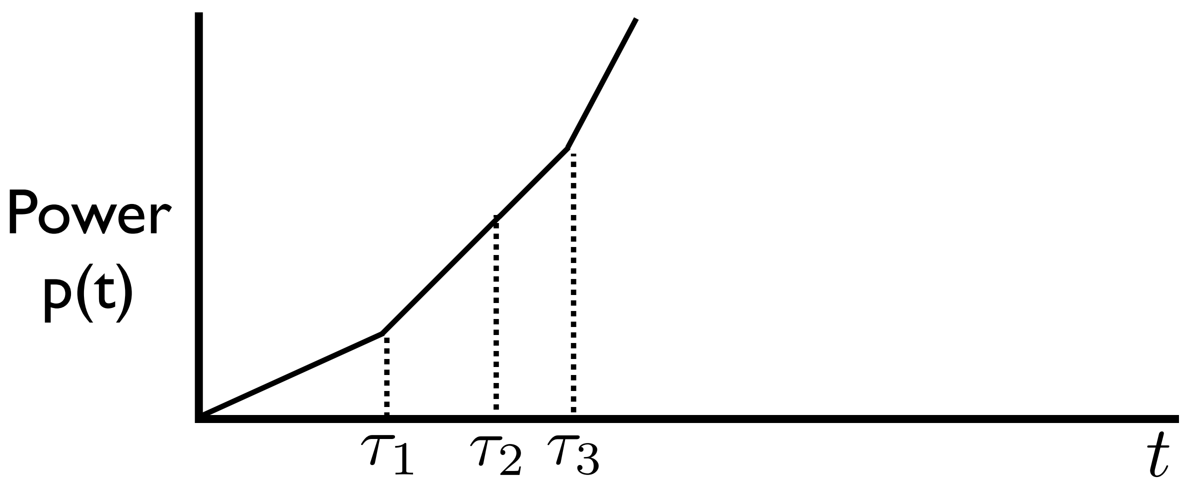




Important Properties

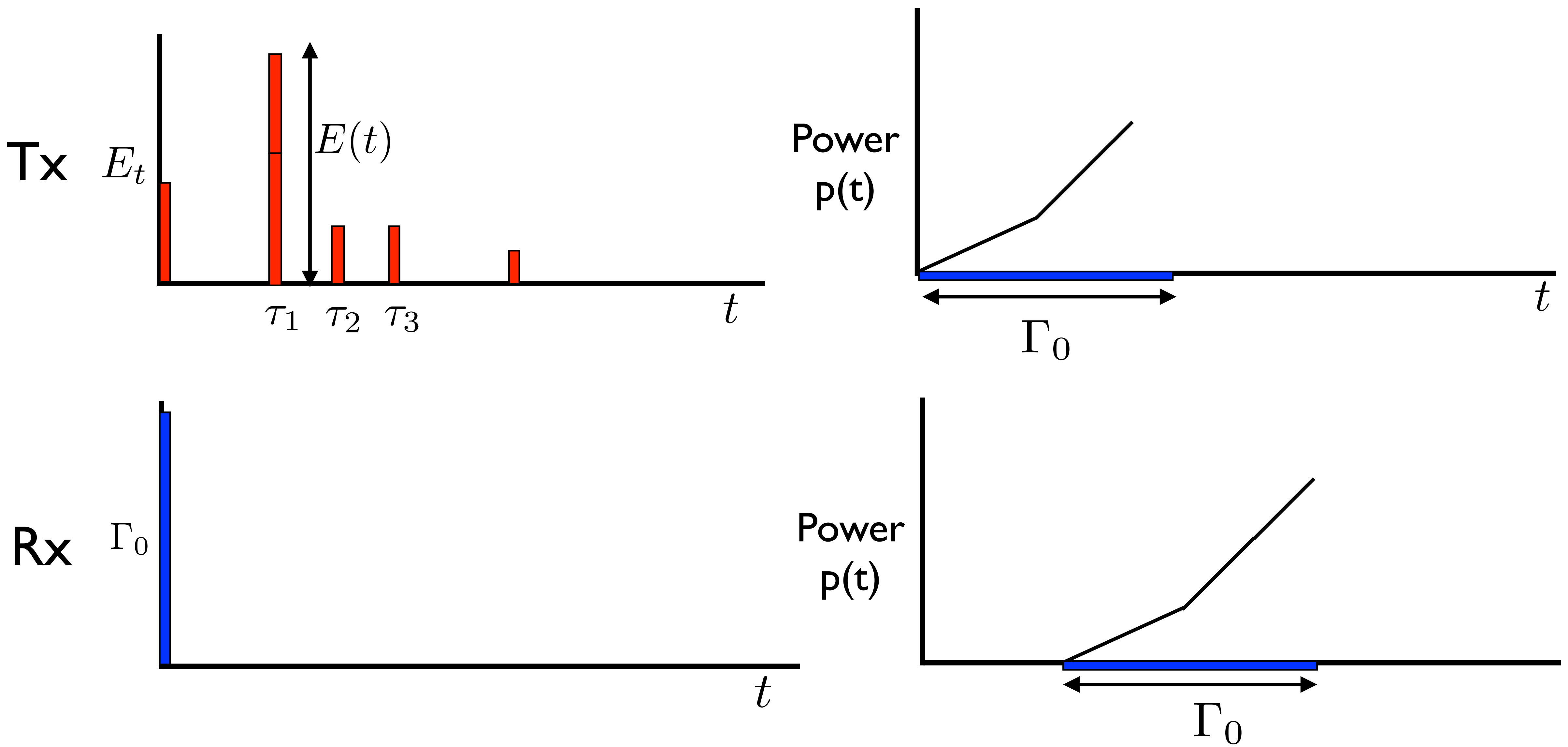
- powers are non-decreasing in time
- powers change only at Tx arrival epoch if at all





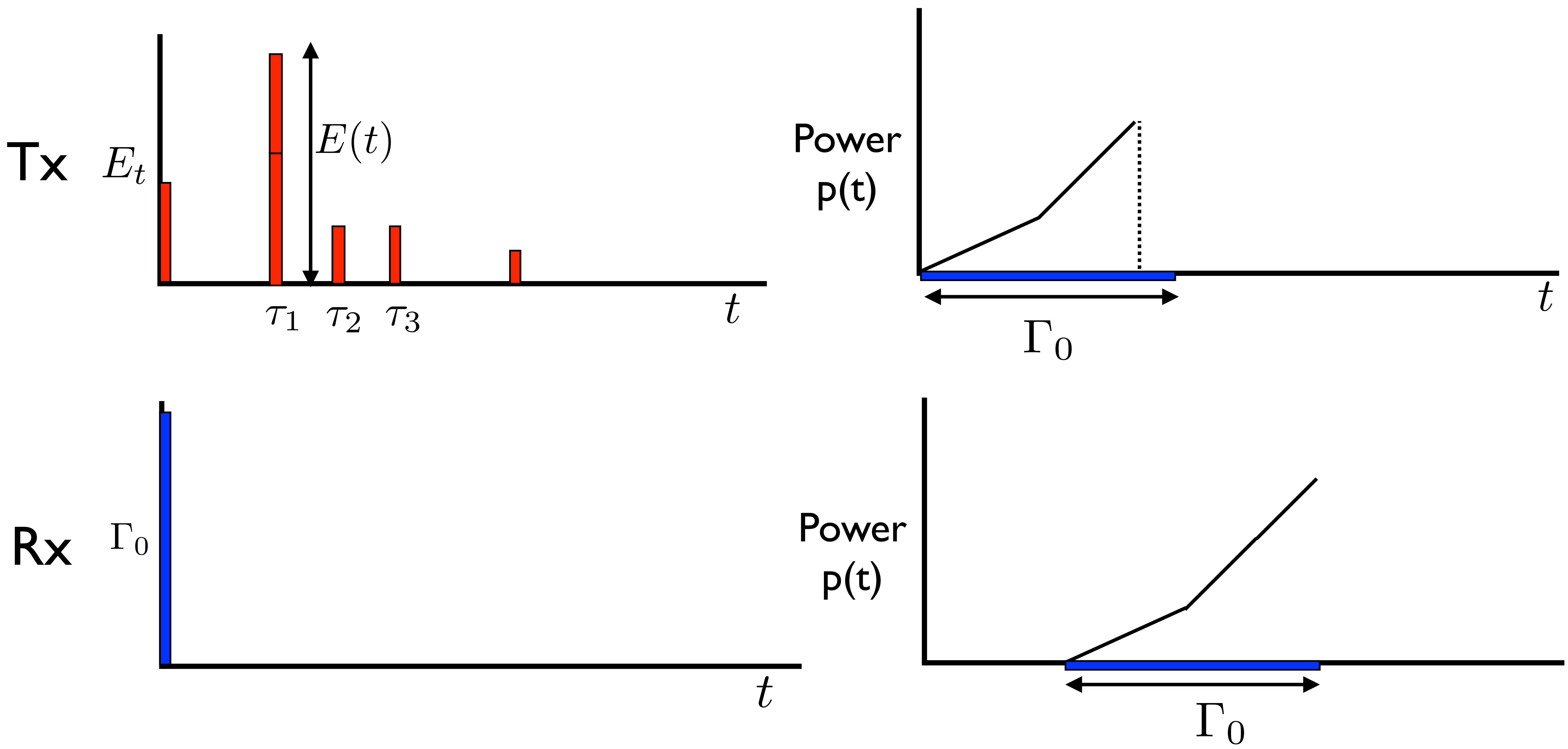
Final Important Property

- either transmission starts at origin transmission time is < Γ_0
- or transmission time is Γ_0



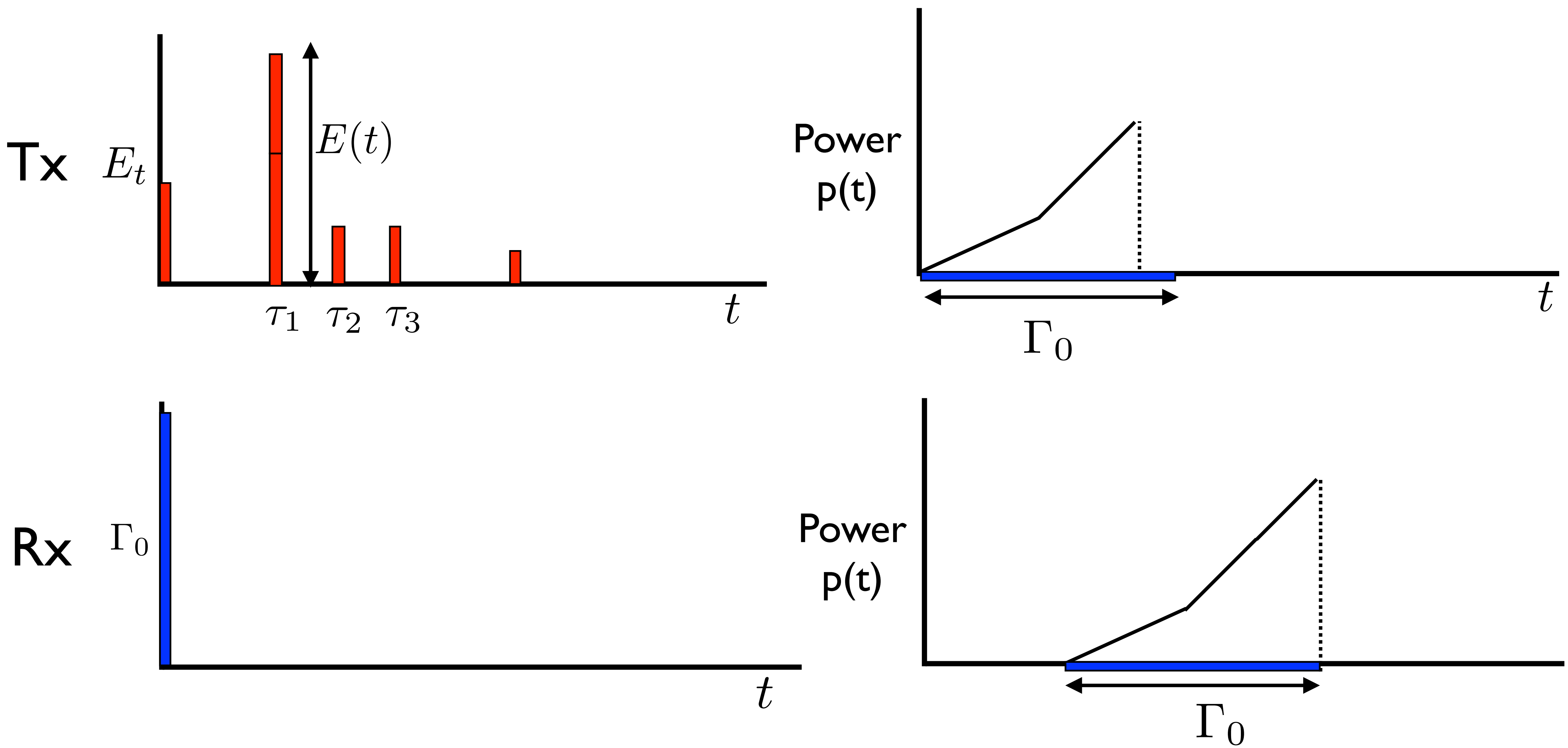
Final Important Property

- either transmission starts at origin transmission time is < Γ_0
- or transmission time is Γ_0



Final Important Property

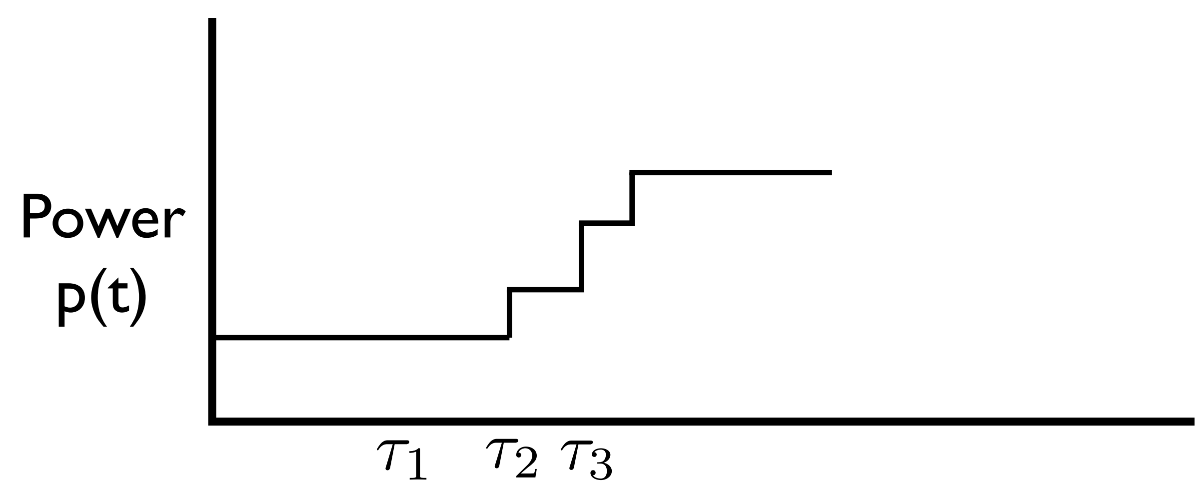
- either transmission starts at origin transmission time is < Γ_0
- or transmission time is Γ_0



Optimal Policy

- Three phase

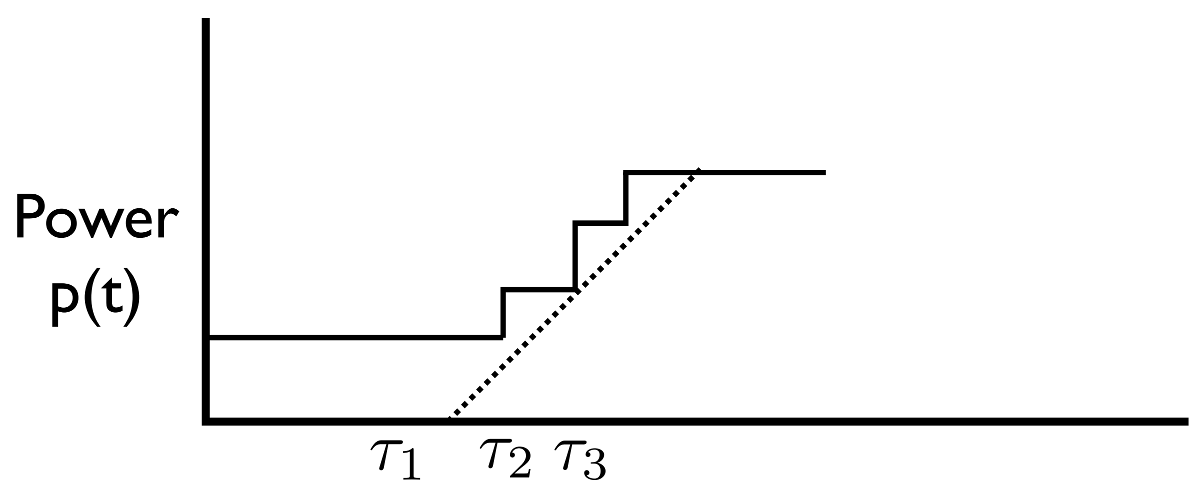
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

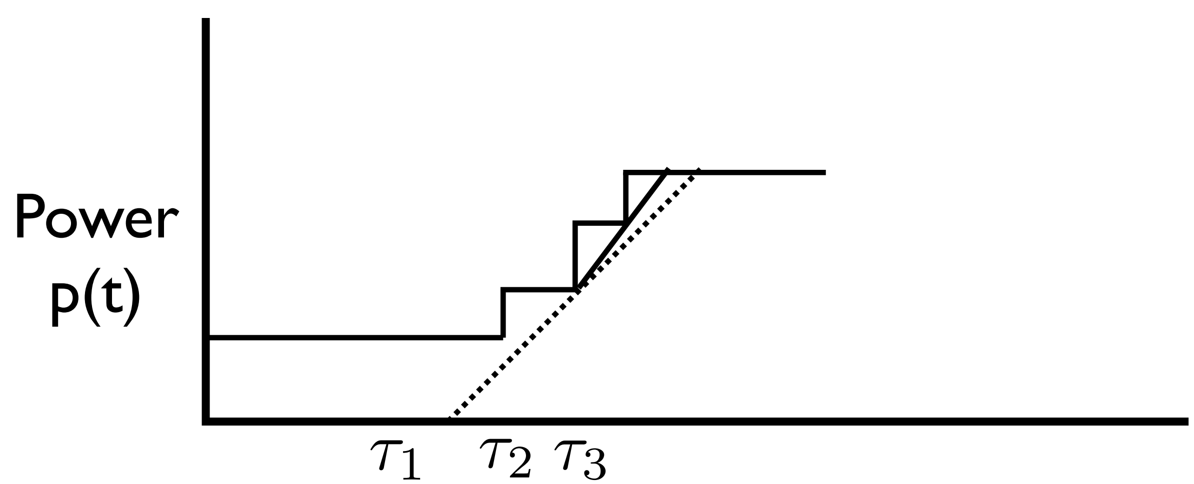
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

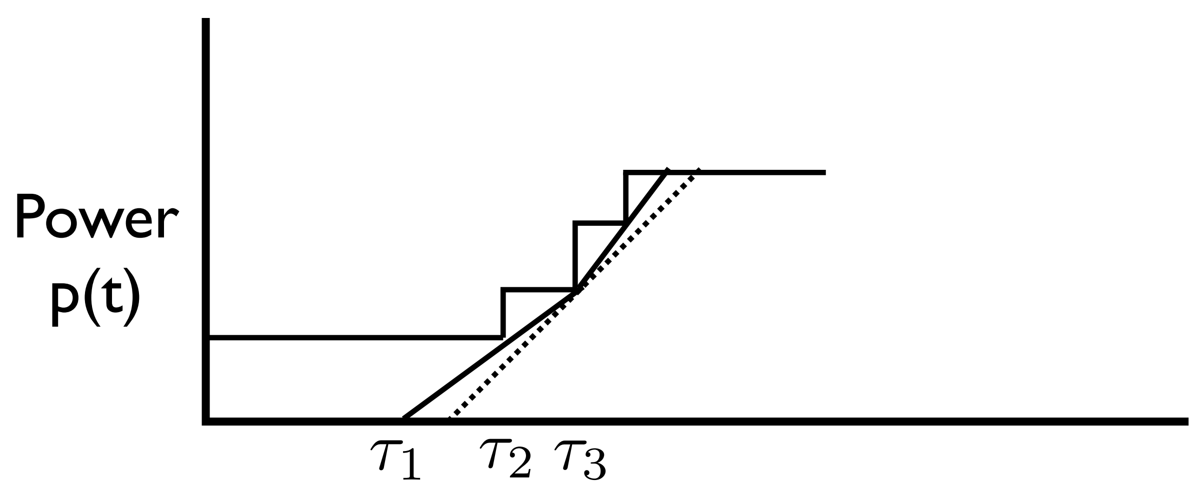
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

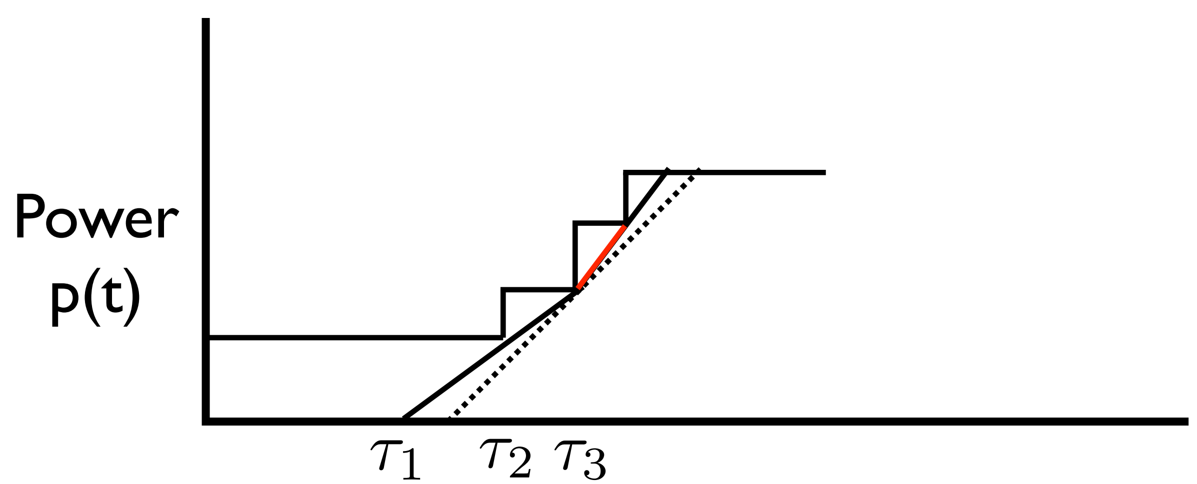
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

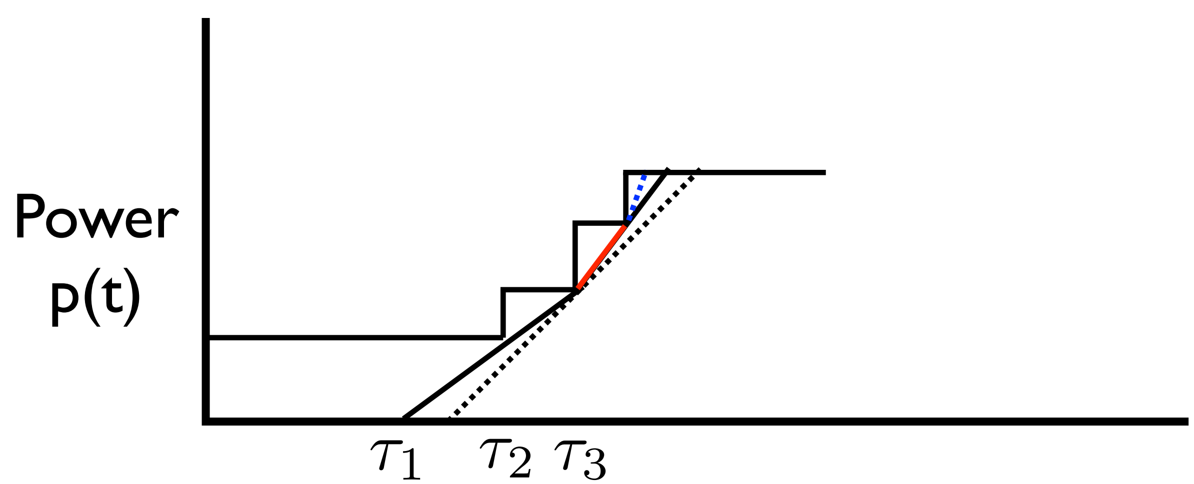
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

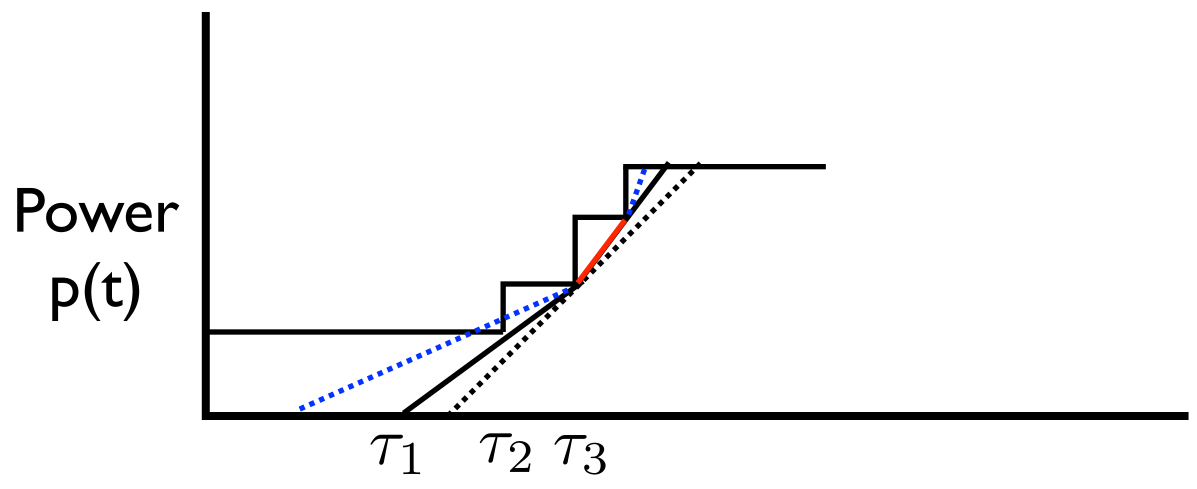
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

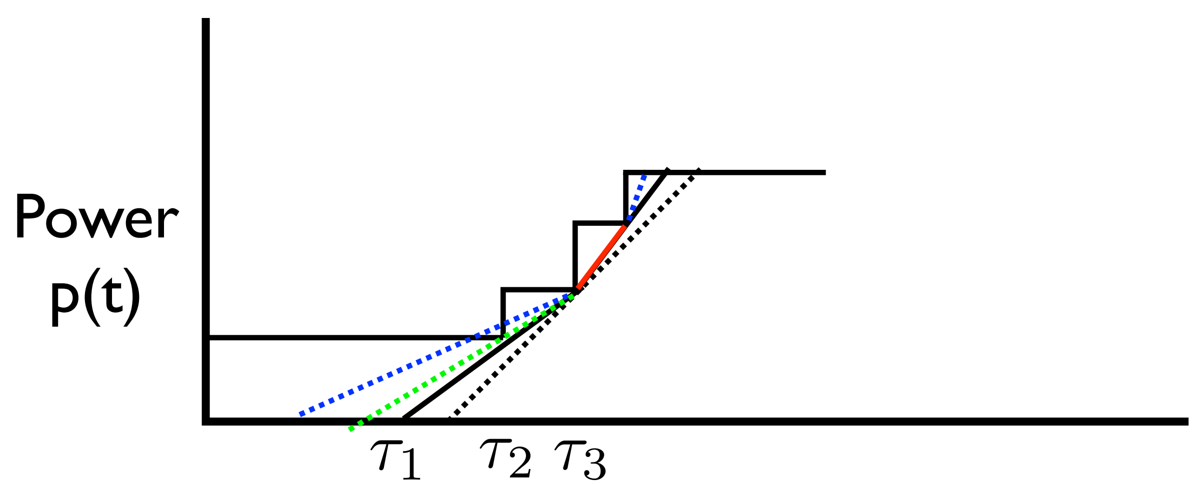
1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



Optimal Policy

- Three phase

1. find a feasible constant power policy that starts earliest
2. iteratively update first and last transmit power
3. make sure that transmission time = Γ_0 if not started from origin



online it is, Damn it!

Quantifying Performance of Online Algorithm

Quantifying Performance of Online Algorithm

Compare against the best offline strategy
(everything known in future)

Quantifying Performance of Online Algorithm

Compare against the best offline strategy
(everything known in future)

Let σ energy arrival sequence

Quantifying Performance of Online Algorithm

Compare against the best offline strategy
(everything known in future)

Let σ energy arrival sequence

$T^\star$ be time taken by an optimal offline algorithm

Quantifying Performance of Online Algorithm

Compare against the best offline strategy
(everything known in future)

Let σ energy arrival sequence

$T^\star$ be time taken by an optimal offline algorithm

T_O be time taken by an online algorithm O

Quantifying Performance of Online Algorithm

Compare against the best offline strategy (everything known in future)

Let σ energy arrival sequence

$T^\star$ be time taken by an optimal offline algorithm

T_O be time taken by an online algorithm O

$$\text{Then the competitive ratio of } O \text{ is } r_O = \max_\sigma \frac{T_O}{T^\star}$$

Quantifying Performance of Online Algorithm

Compare against the best offline strategy (everything known in future)

Let σ energy arrival sequence

$T^\star$ be time taken by an optimal offline algorithm

T_O be time taken by an online algorithm O

Then the competitive ratio of O is $r_O = \max_\sigma \frac{T_O}{T^\star}$

Objective is to find $O^\star$ such that $O^\star = \arg\min_O \max_\sigma \frac{T_O}{T^\star}$

Typical Strategy

Produce an O to upper bound $\max_{\sigma} \frac{T_O}{T^{\star}}$

Derive an online algorithm independent lower bound on

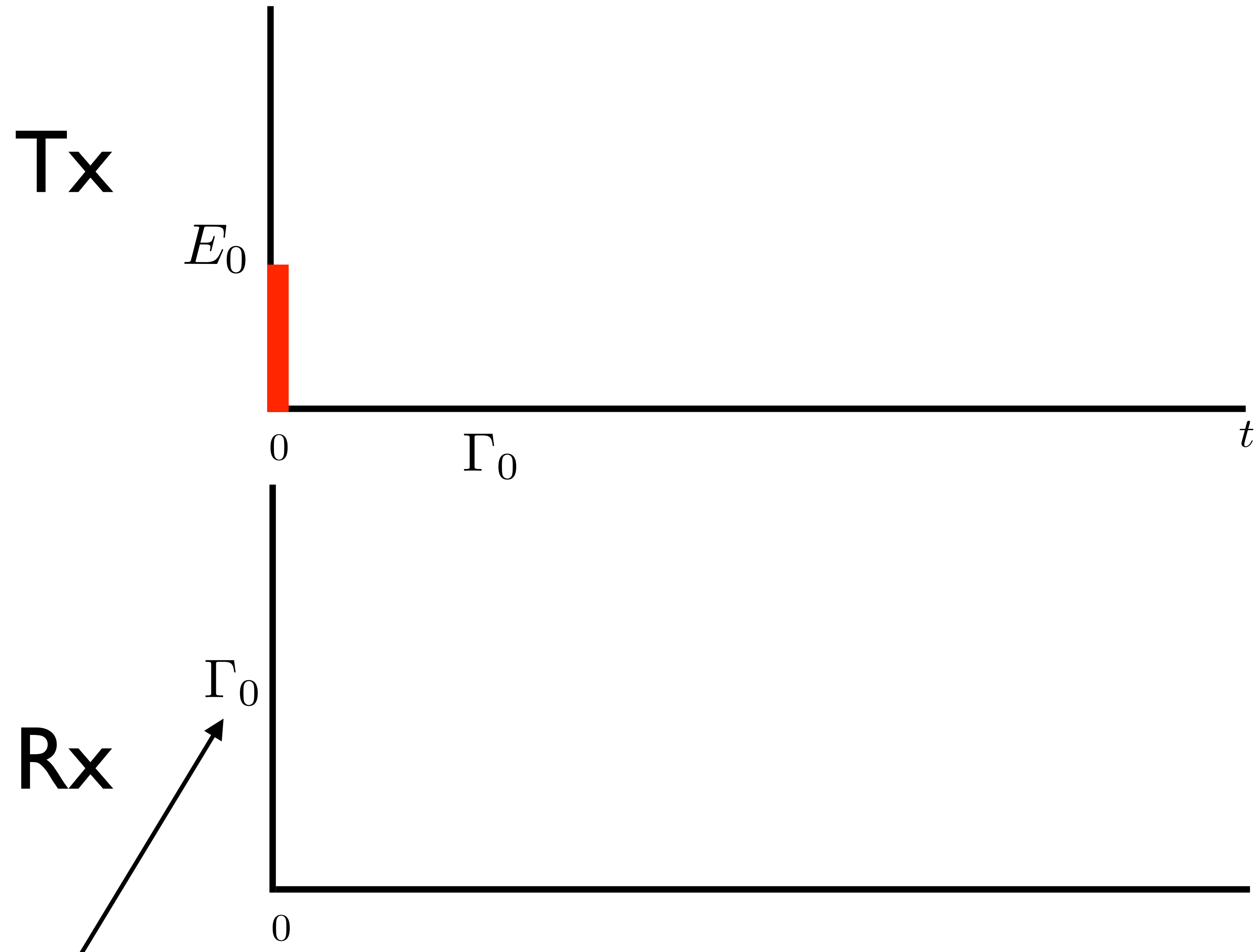
$$\min_{O} \max_{\sigma} \frac{T_O}{T^{\star}}$$

Hope that they match !

Result

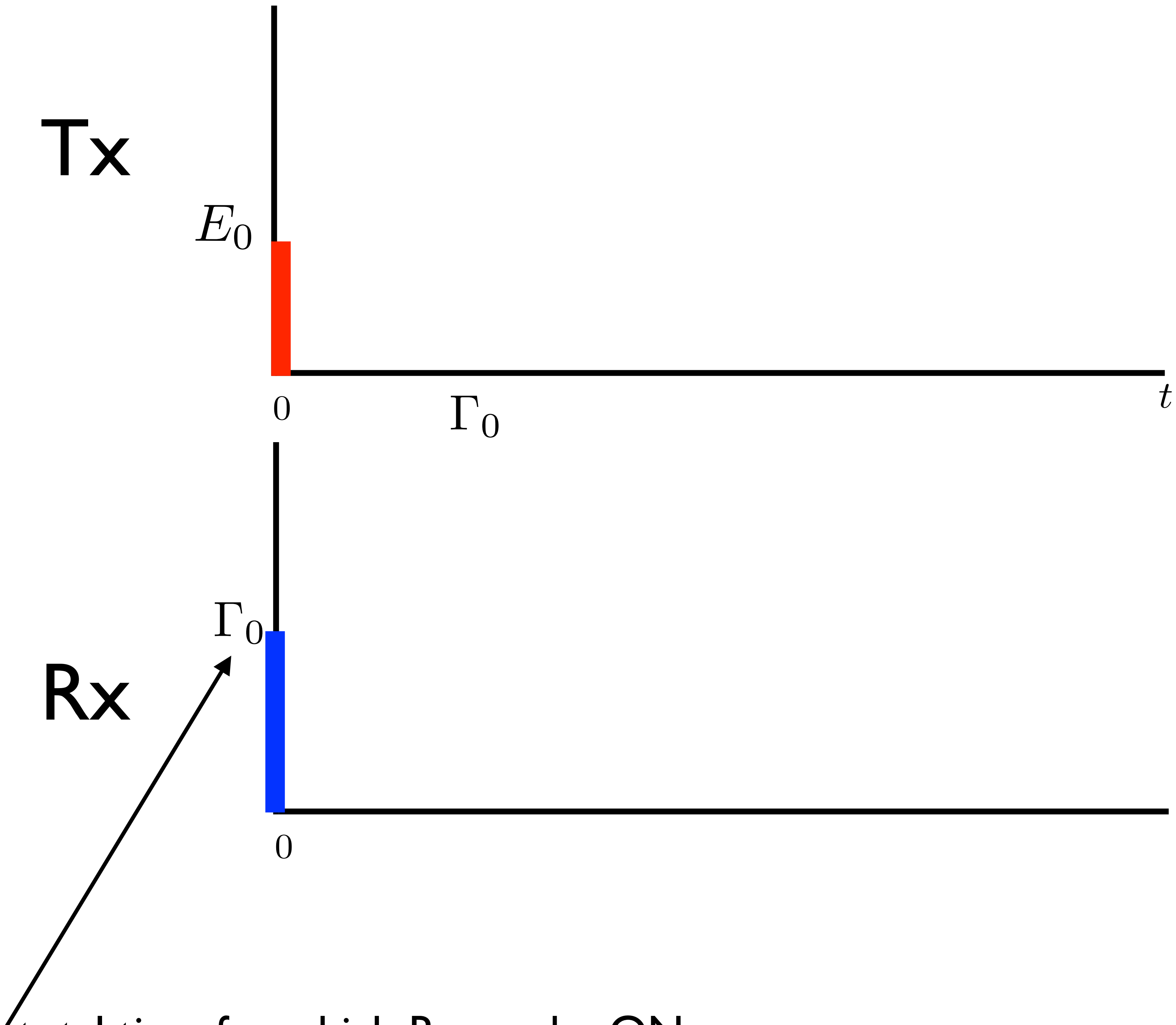
$$\min_{O} \max_{\sigma} \frac{T_O}{T^\star} = 2$$

Lazy (Best effort delivery) Online Algorithm



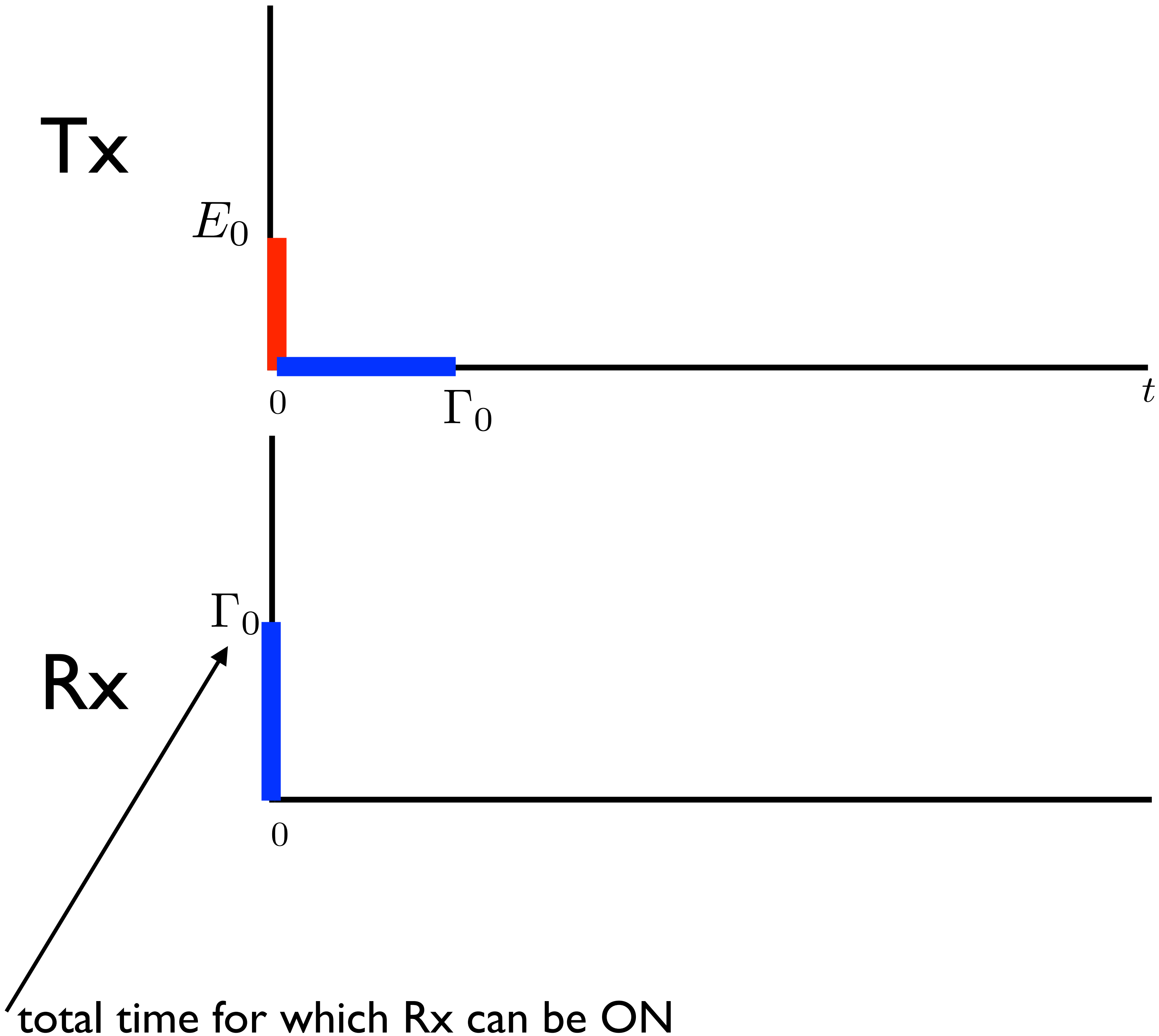
total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm

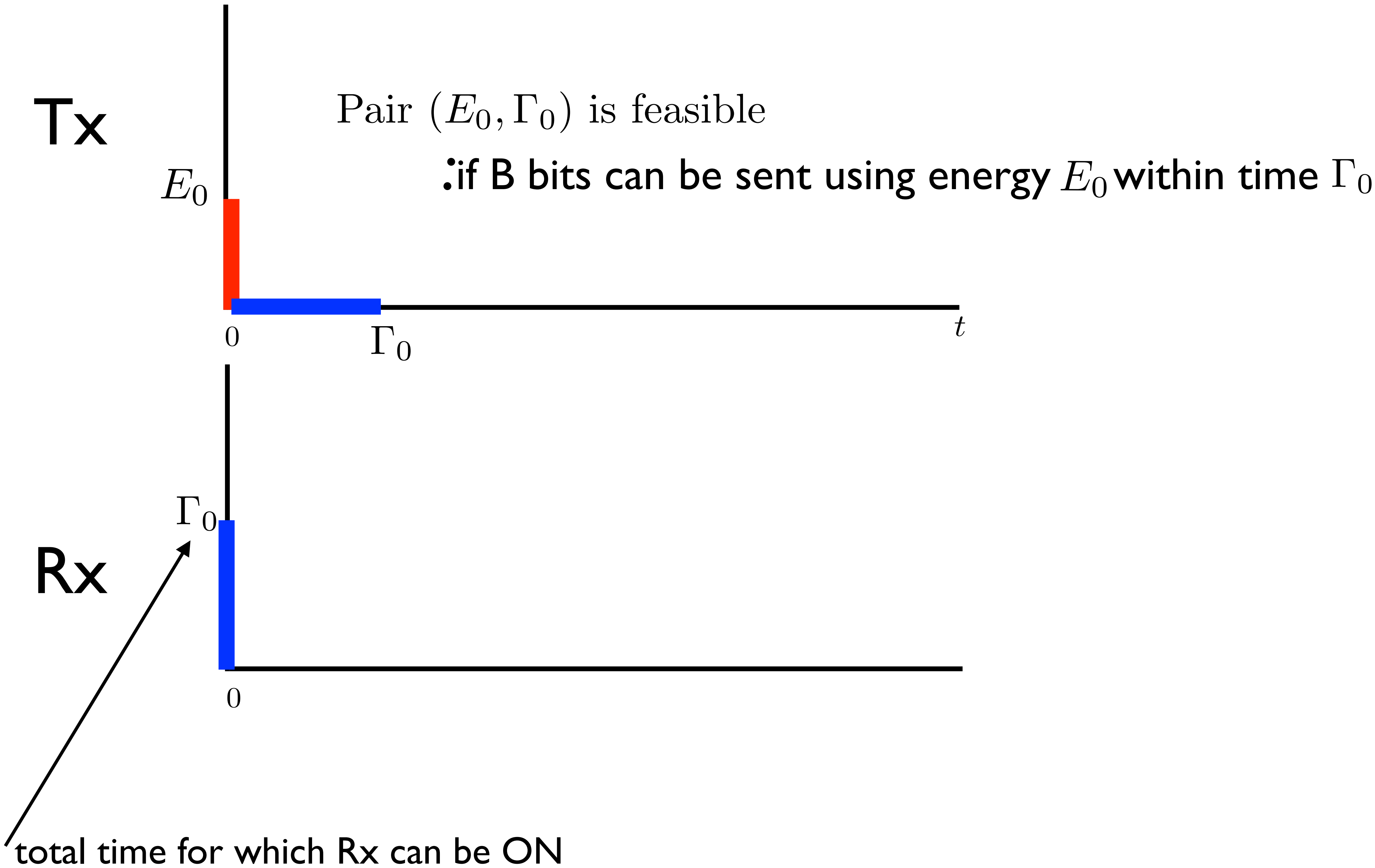


total time for which Rx can be ON

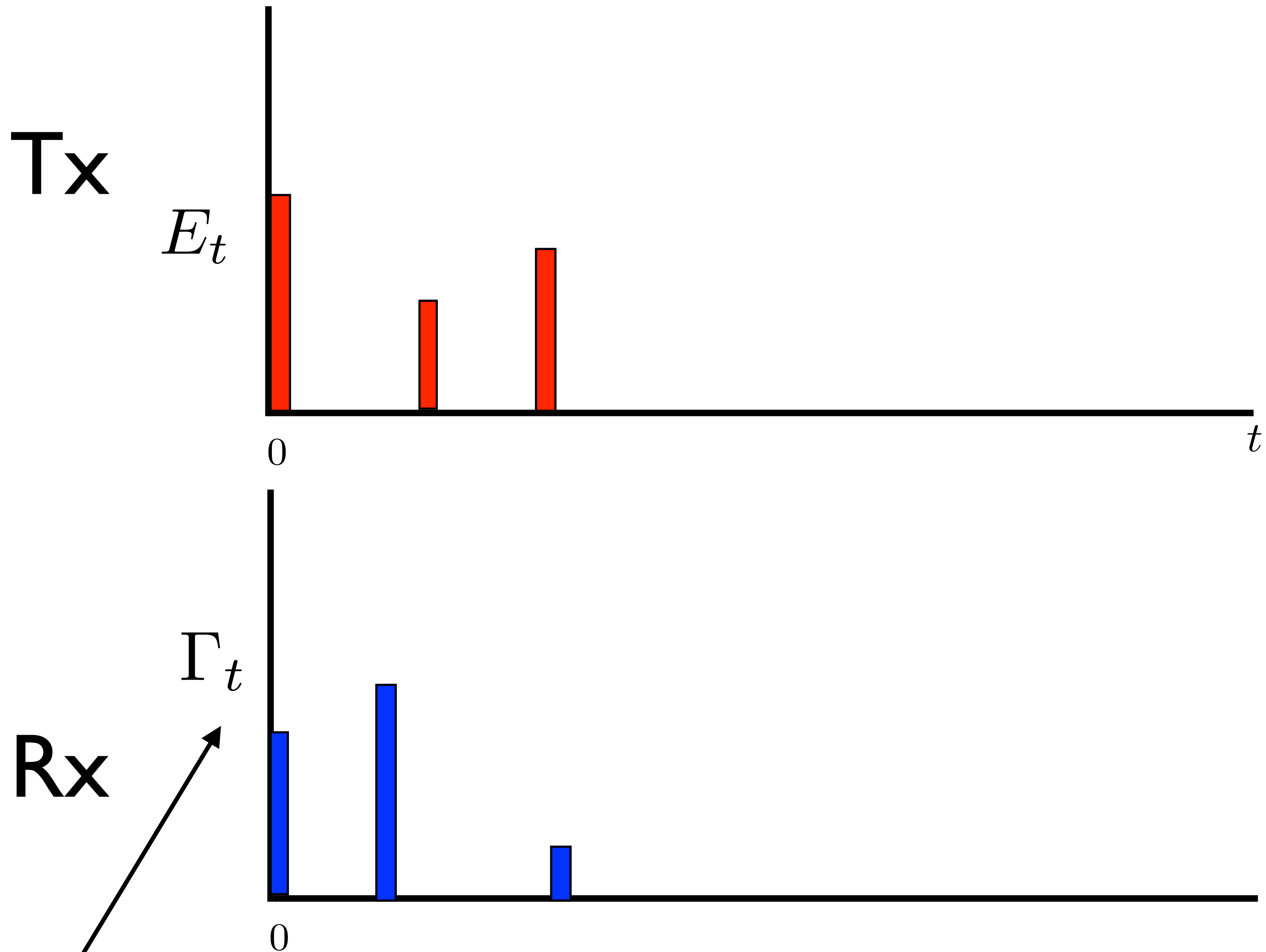
Lazy (Best effort delivery) Online Algorithm



Lazy (Best effort delivery) Online Algorithm



Lazy (Best effort delivery) Online Algorithm

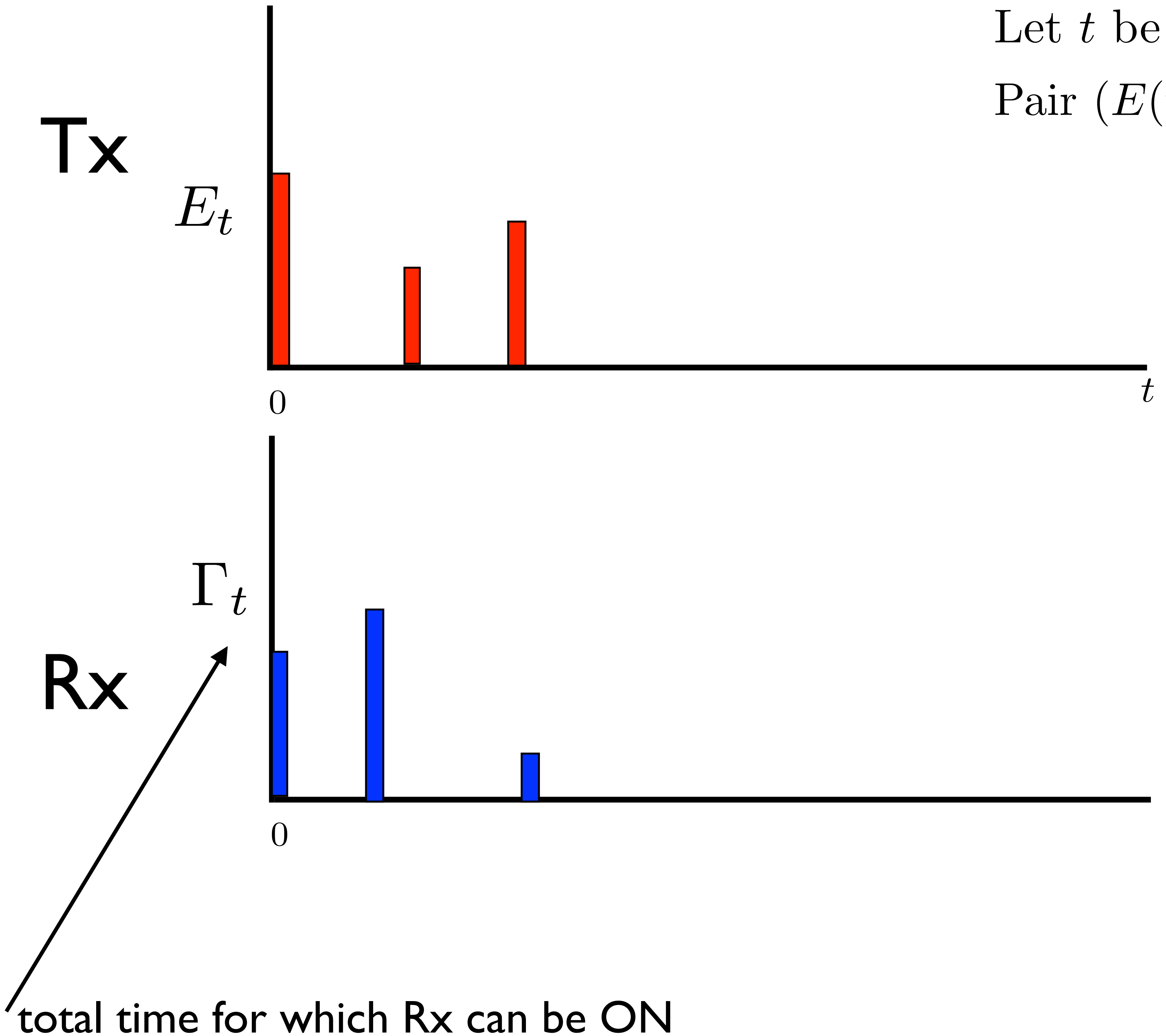


total time for which Rx can be ON

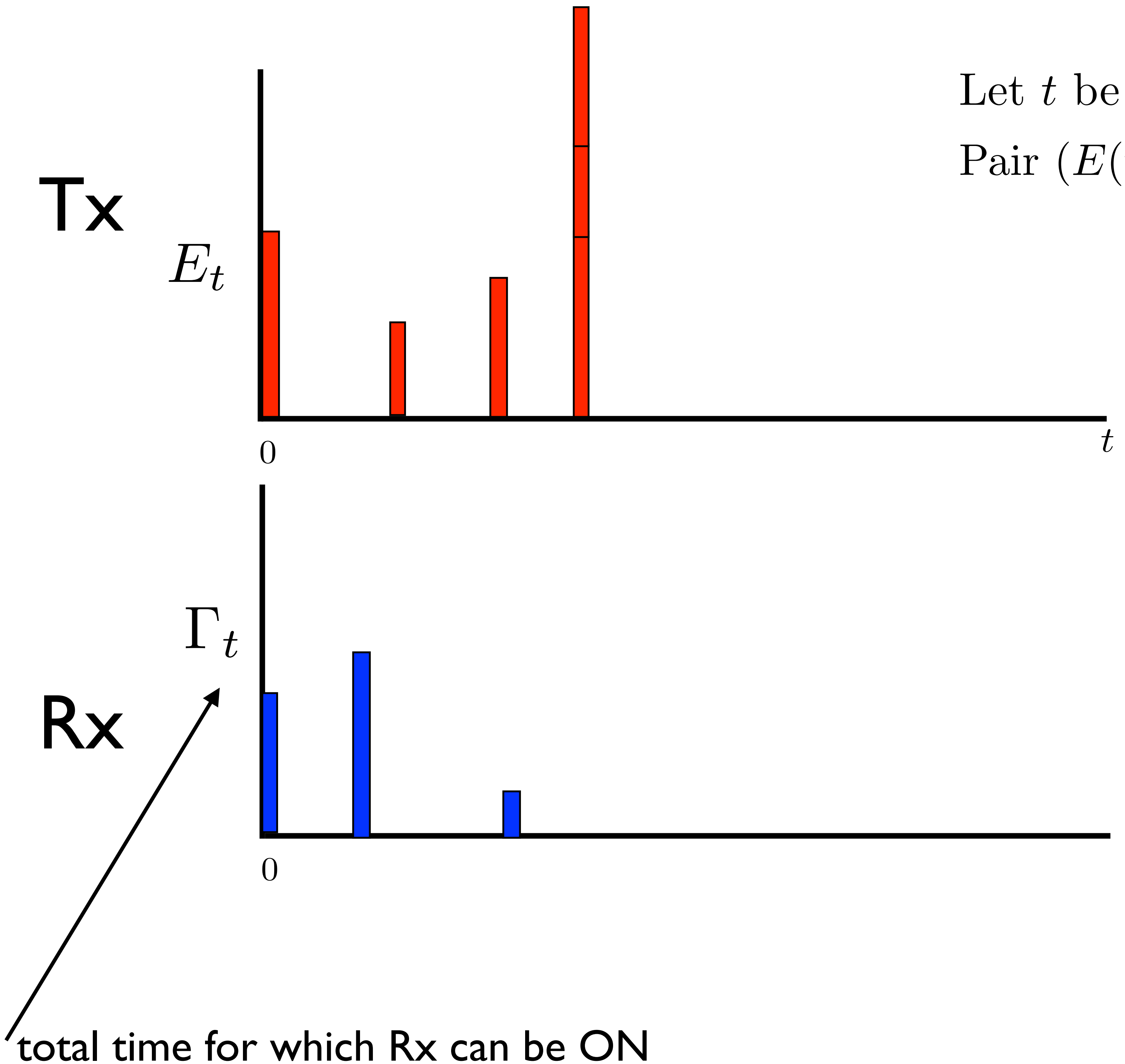
Lazy (Best effort delivery) Online Algorithm

Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible



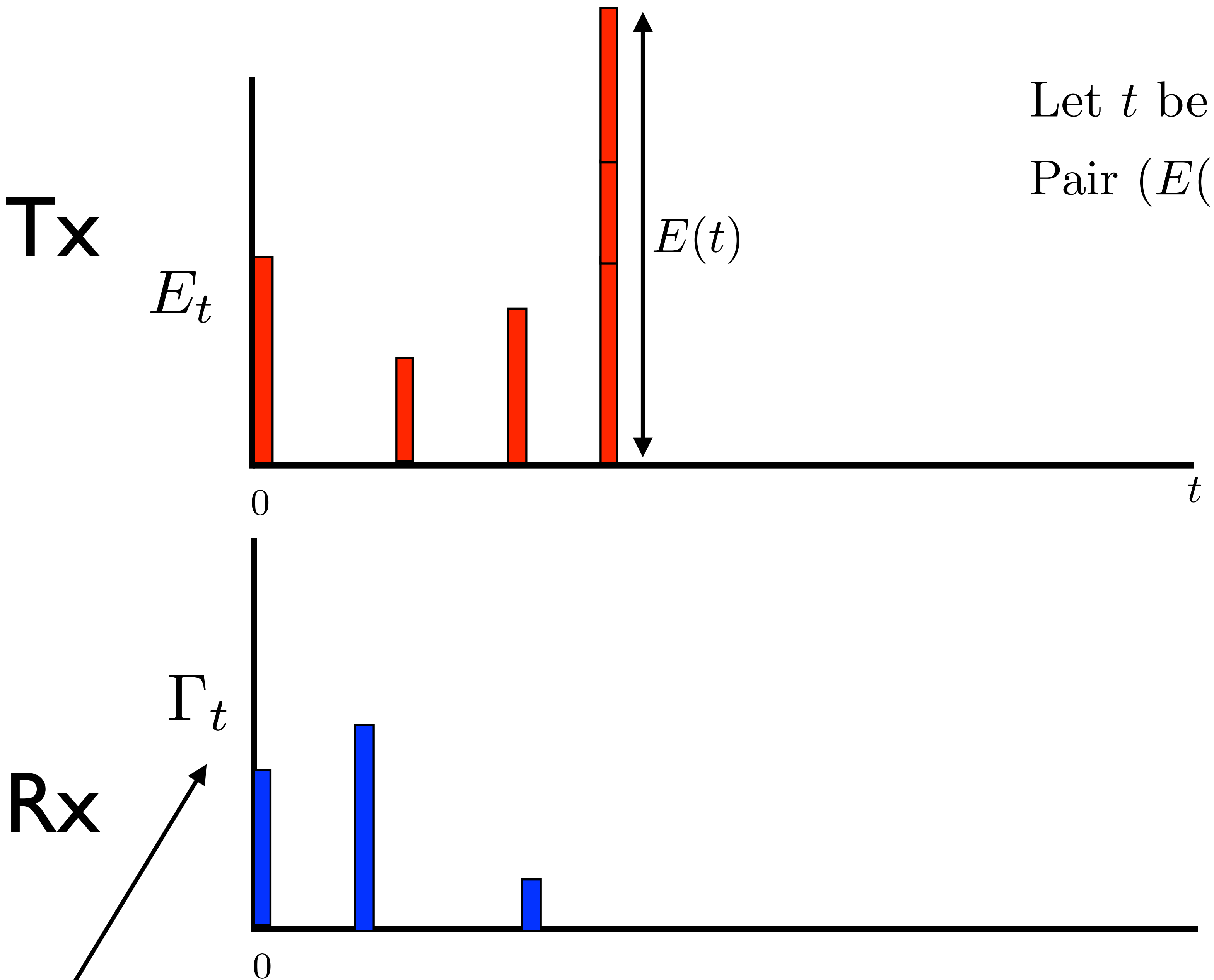
Lazy (Best effort delivery) Online Algorithm



Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

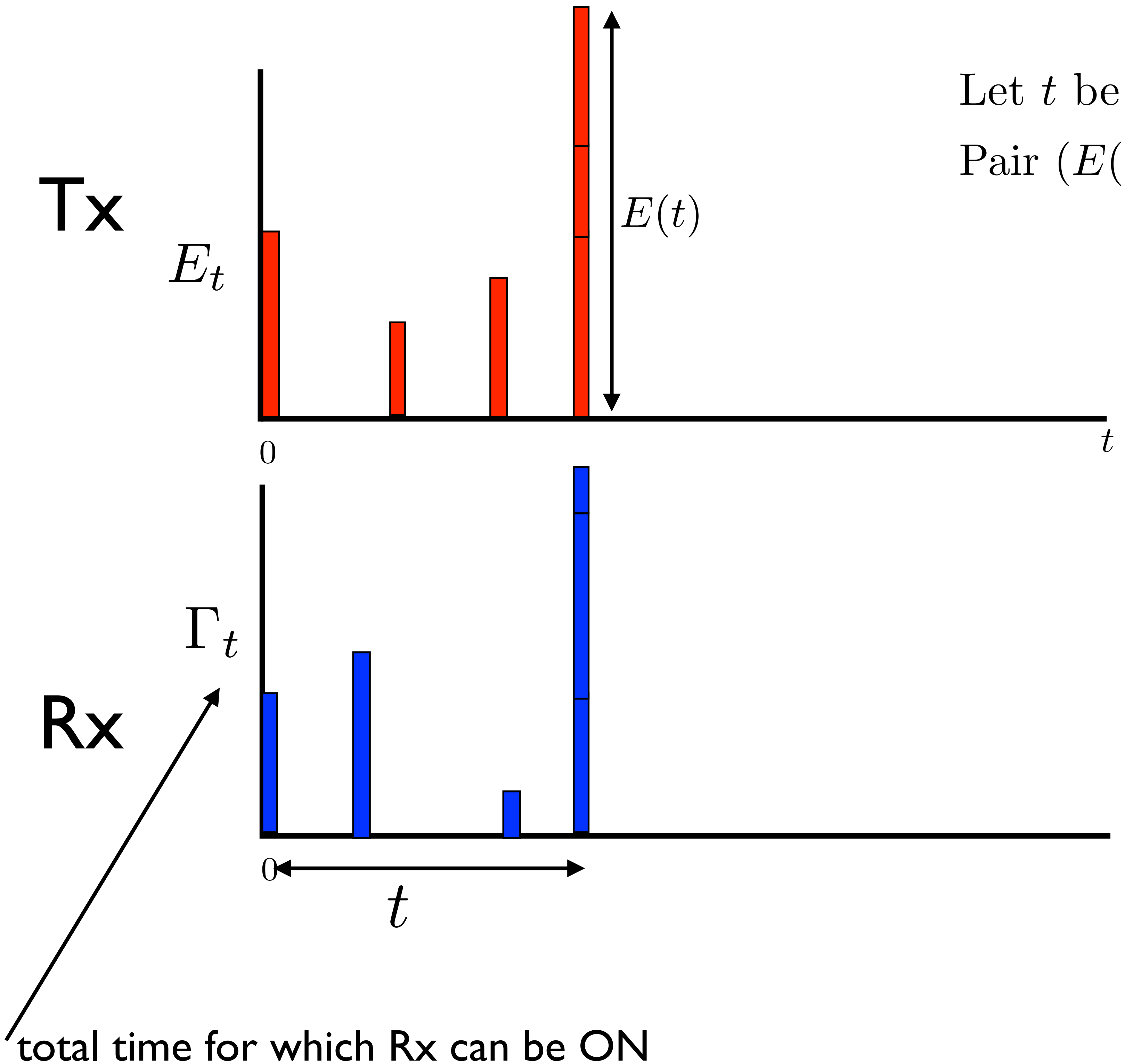
Lazy (Best effort delivery) Online Algorithm



Let t be the earliest time, where
Pair $(E(t), \Gamma(t))$ is feasible

total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm



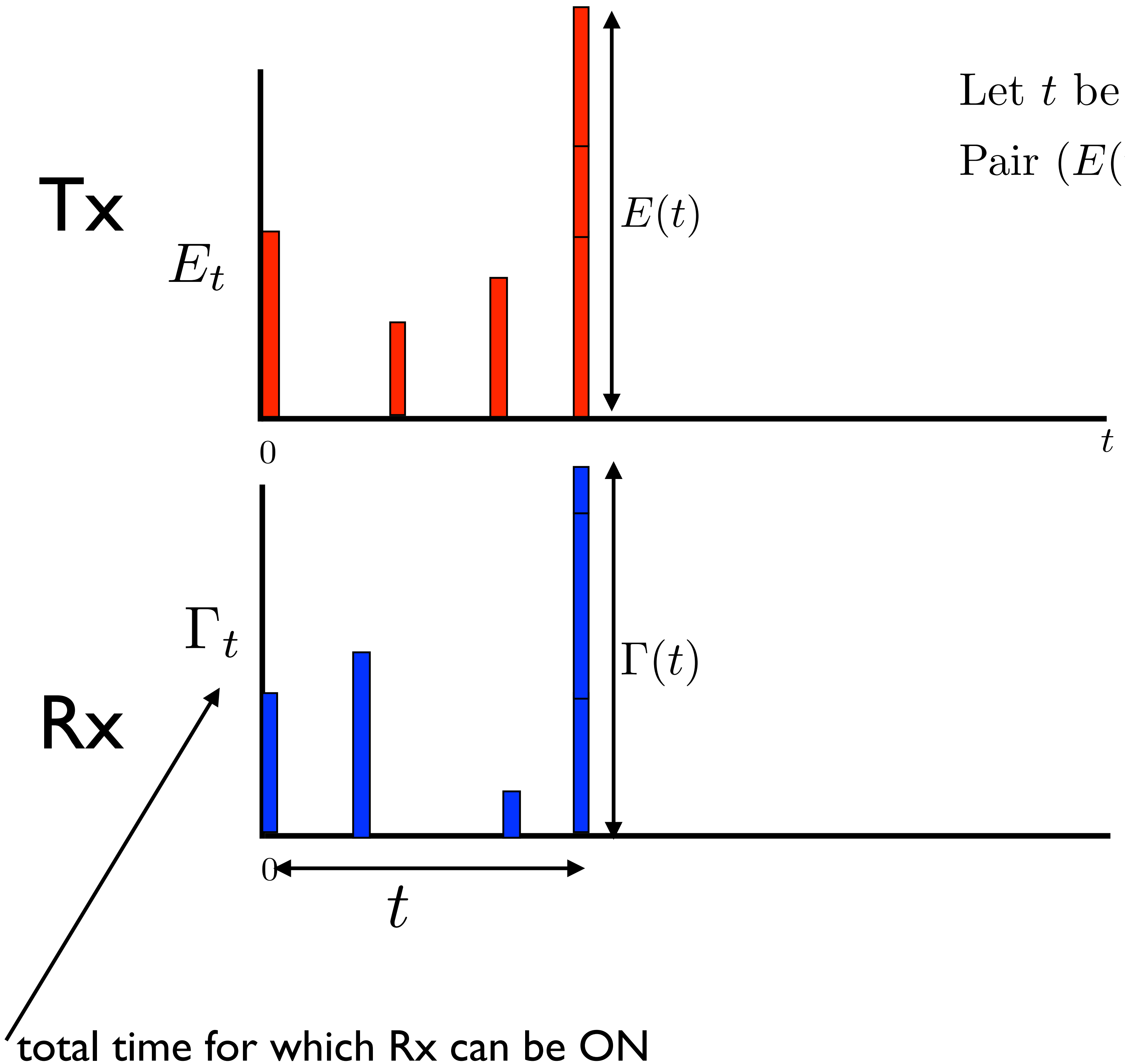
Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

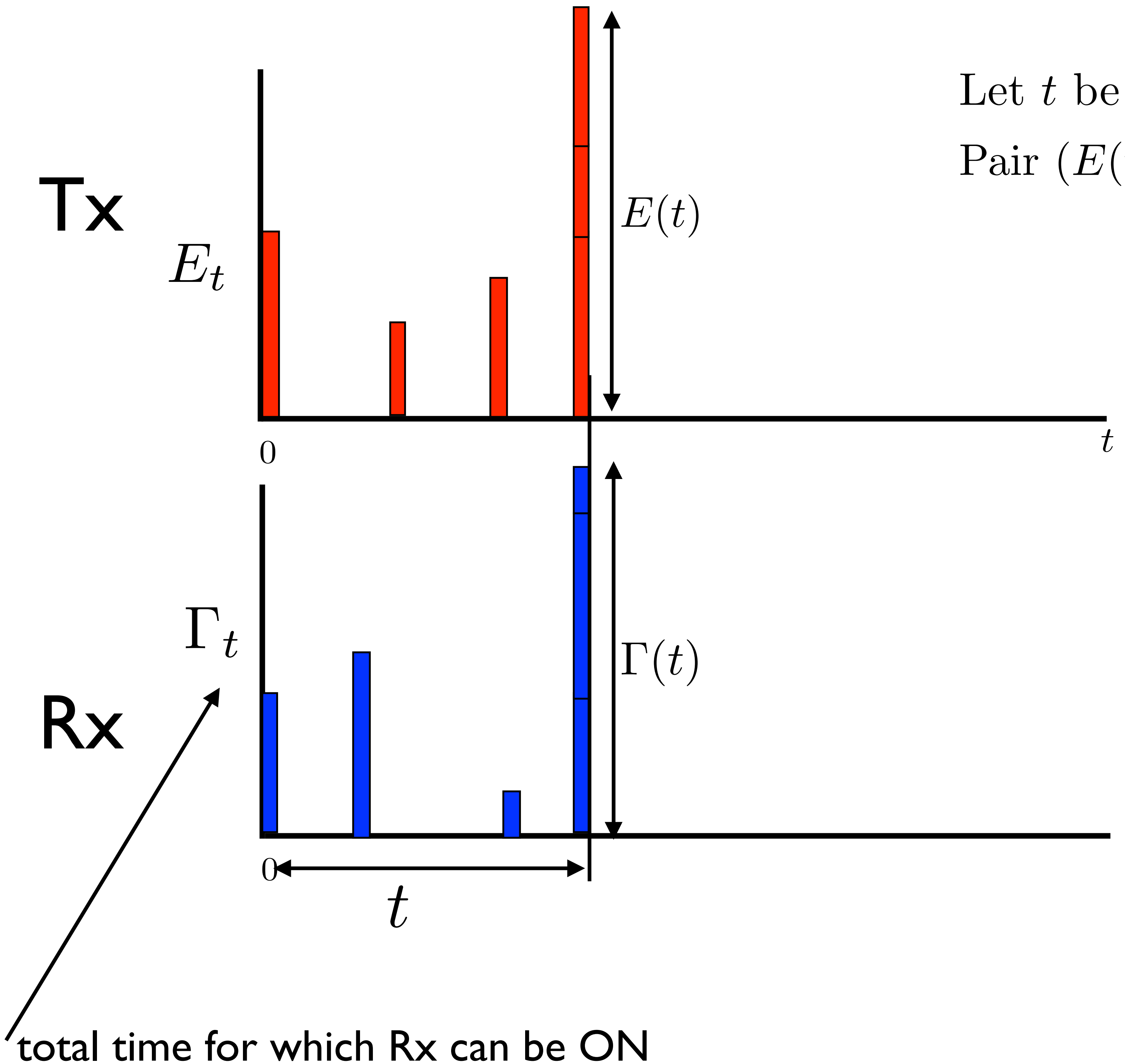
Lazy (Best effort delivery) Online Algorithm

Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible



Lazy (Best effort delivery) Online Algorithm

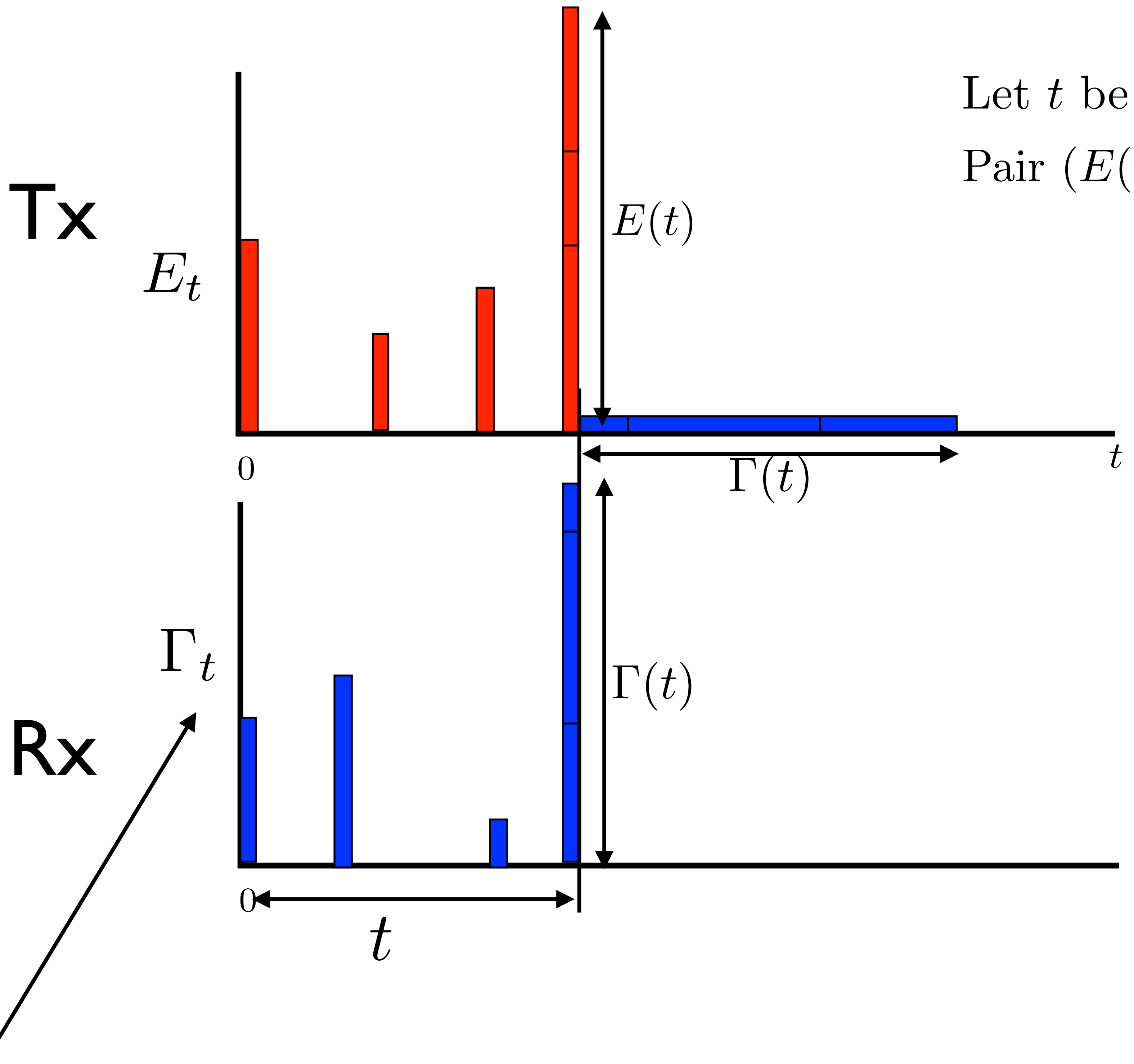


Let t be the earliest time, where Pair $(E(t), \Gamma(t))$ is feasible

Lazy (Best effort delivery) Online Algorithm

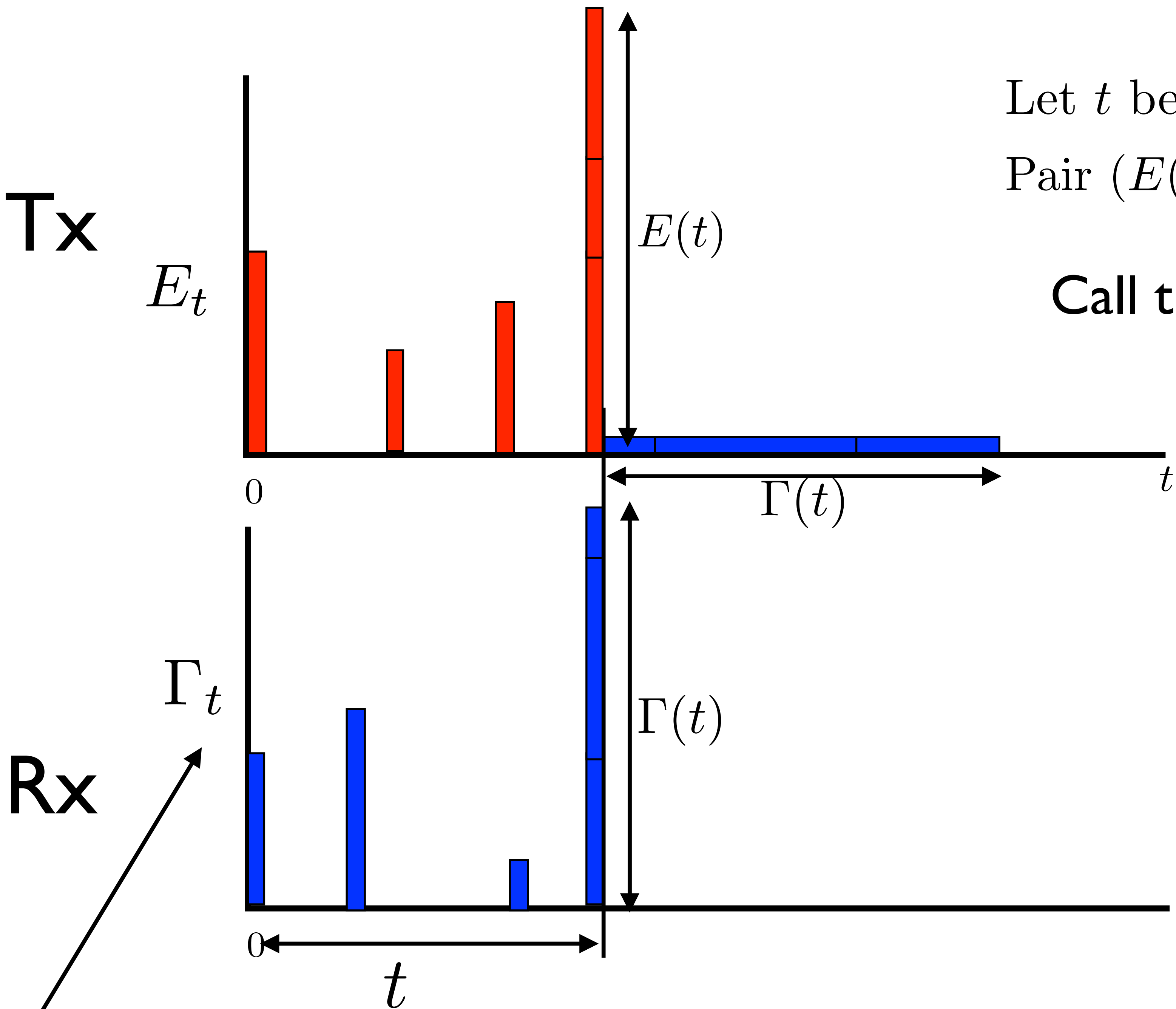
Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible



total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm

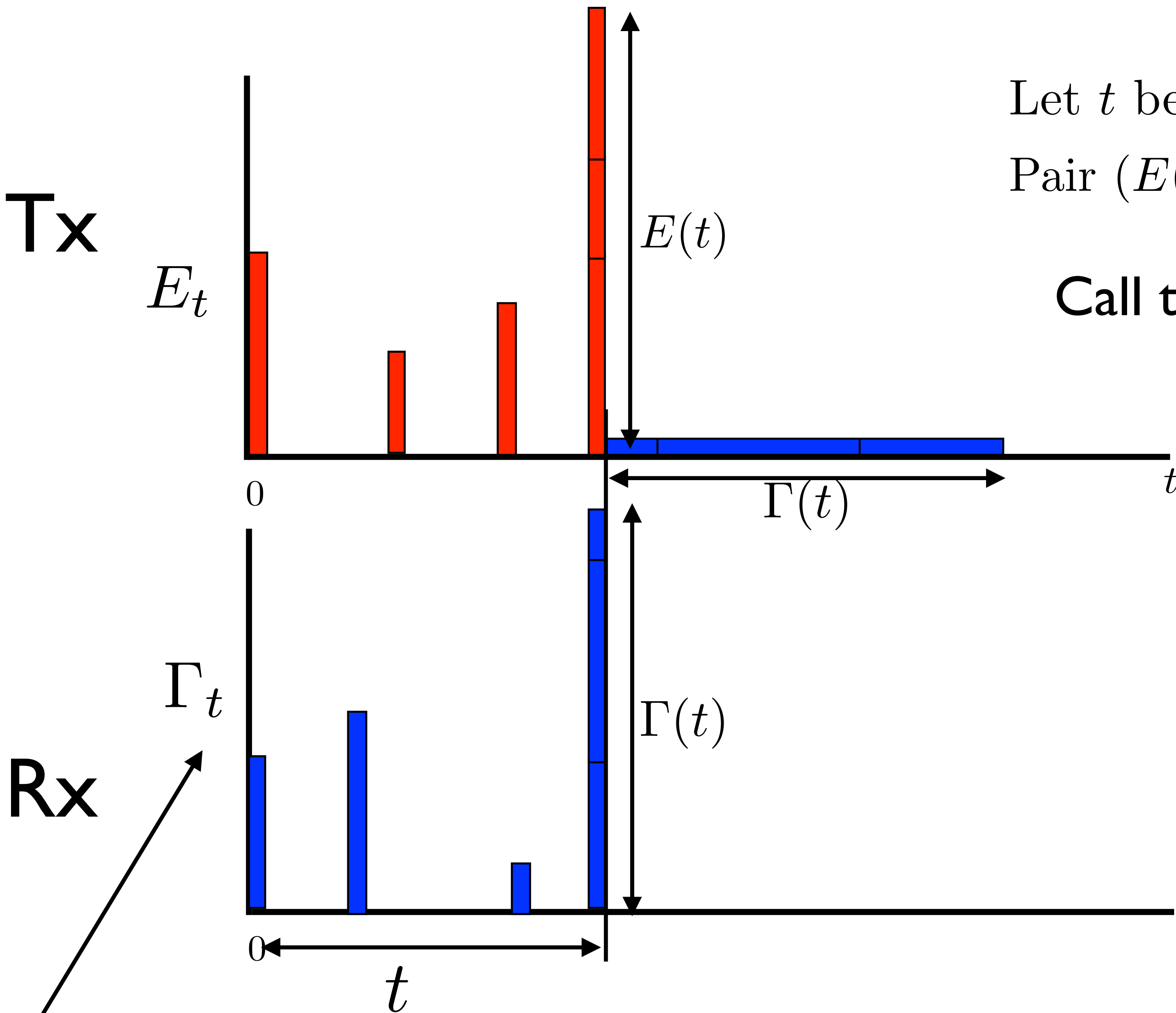


Let t be the earliest time, where
Pair $(E(t), \Gamma(t))$ is feasible

Call this t, T_{start}

total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm



Let t be the earliest time, where

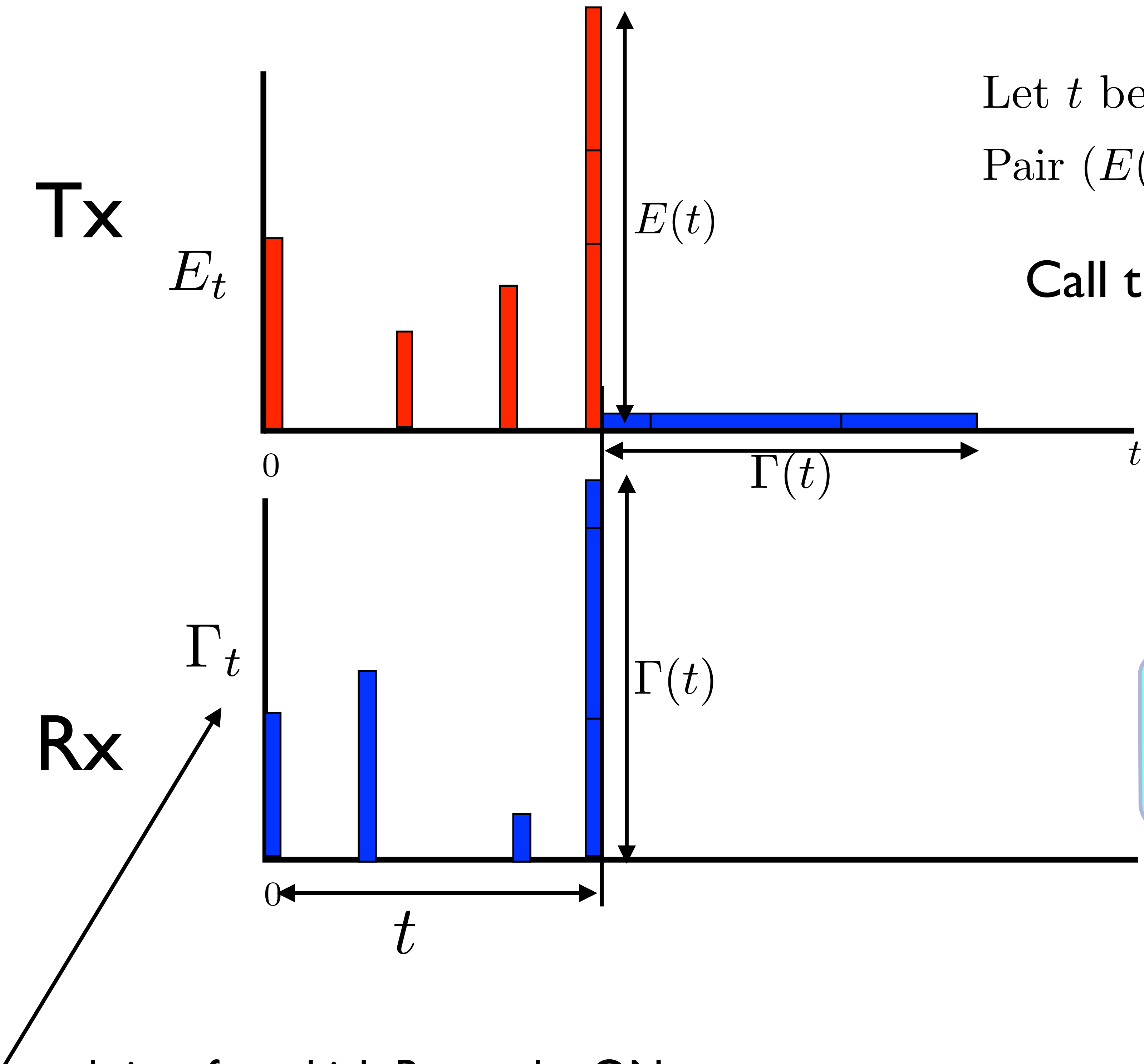
Pair $(E(t), \Gamma(t))$ is feasible

Call this t, T_{start}

transmit with power p_1

total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm



total time for which Rx can be ON

Let t be the earliest time, where

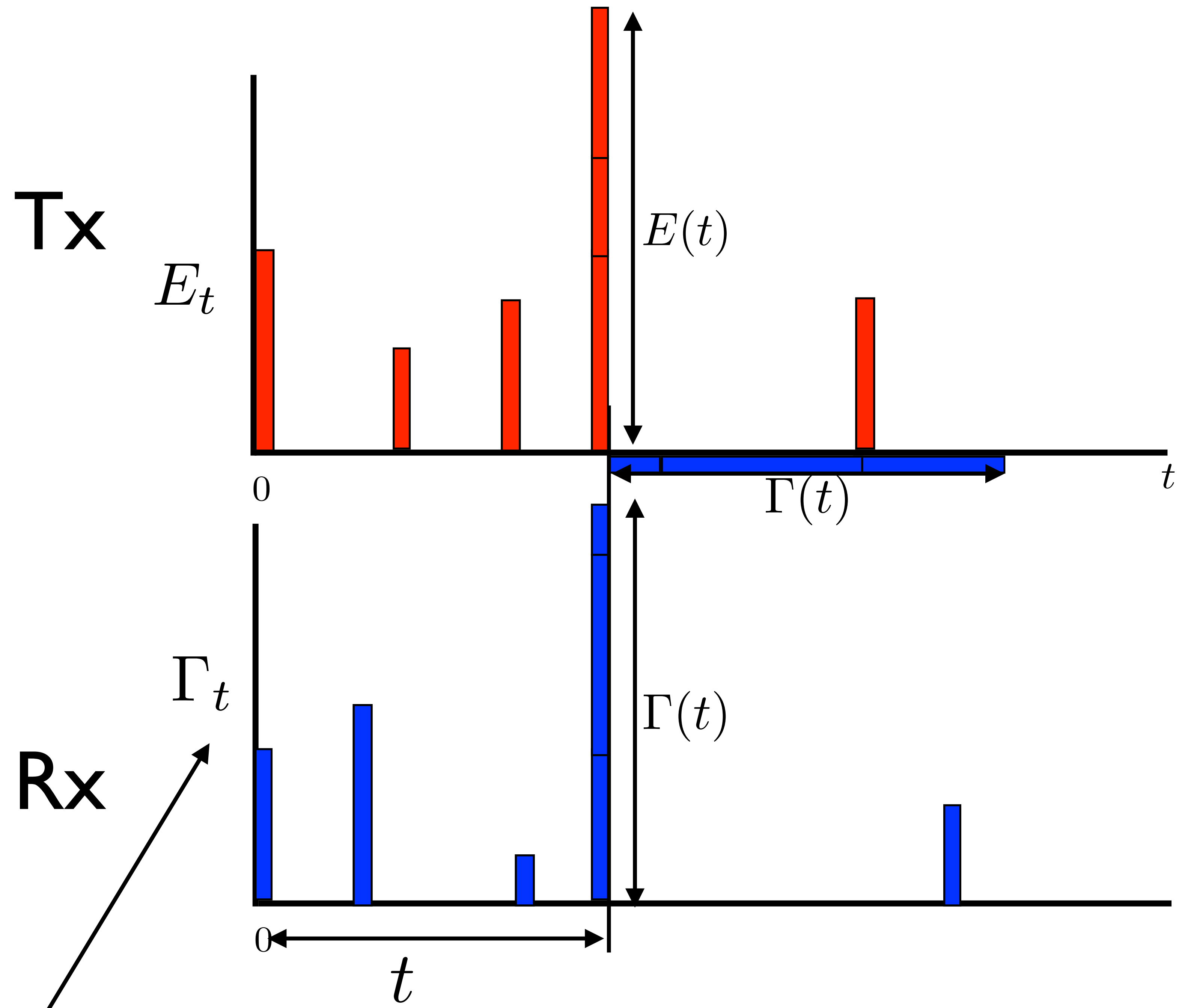
Pair $(E(t), \Gamma(t))$ is feasible

Call this t, T_{start}

transmit with power p_1

$$g(p_1)\frac{E(t)}{p_1} = B$$

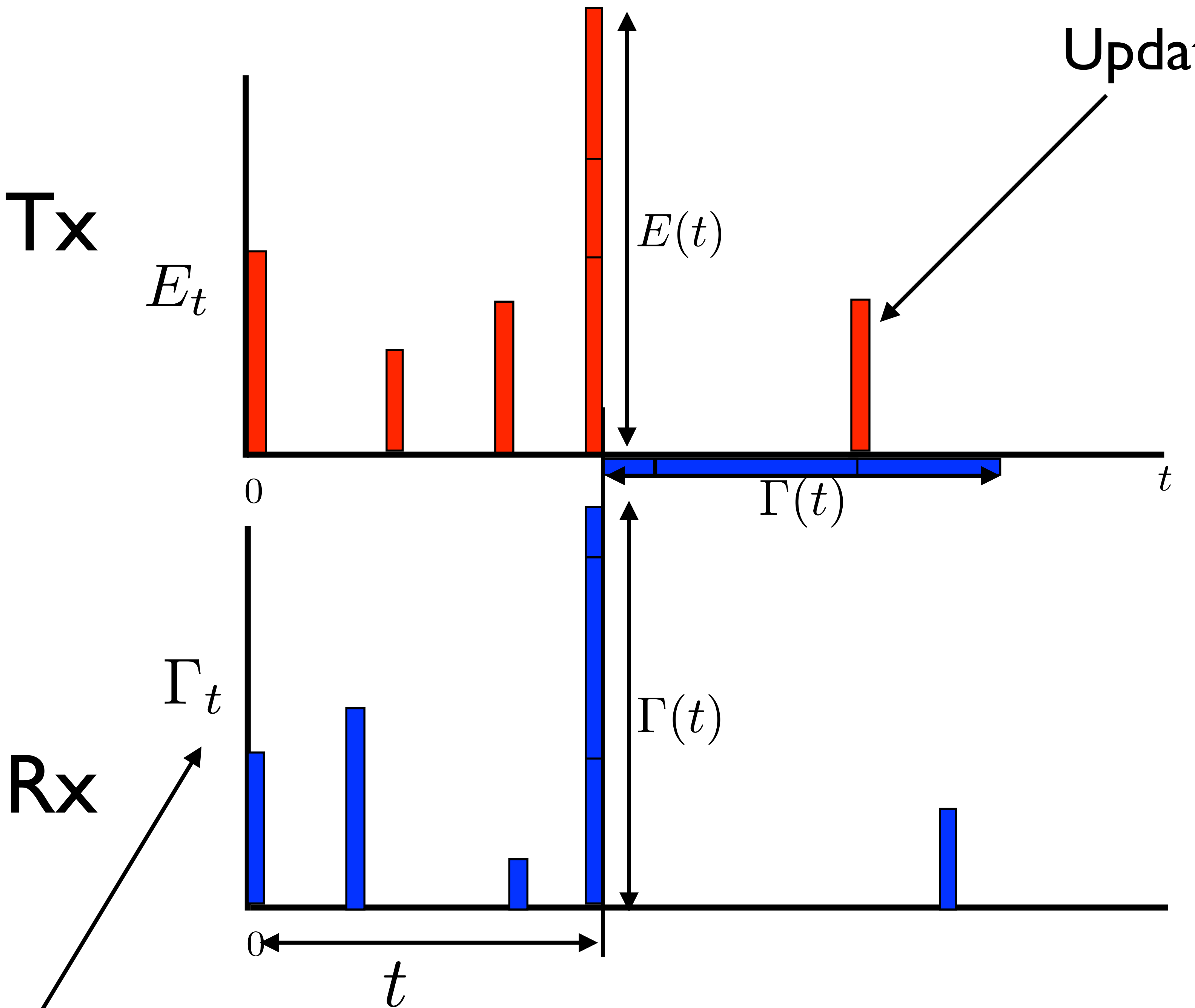
Lazy (Best effort delivery) Online Algorithm



total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm

Update transmit power at each energy arrival at Tx

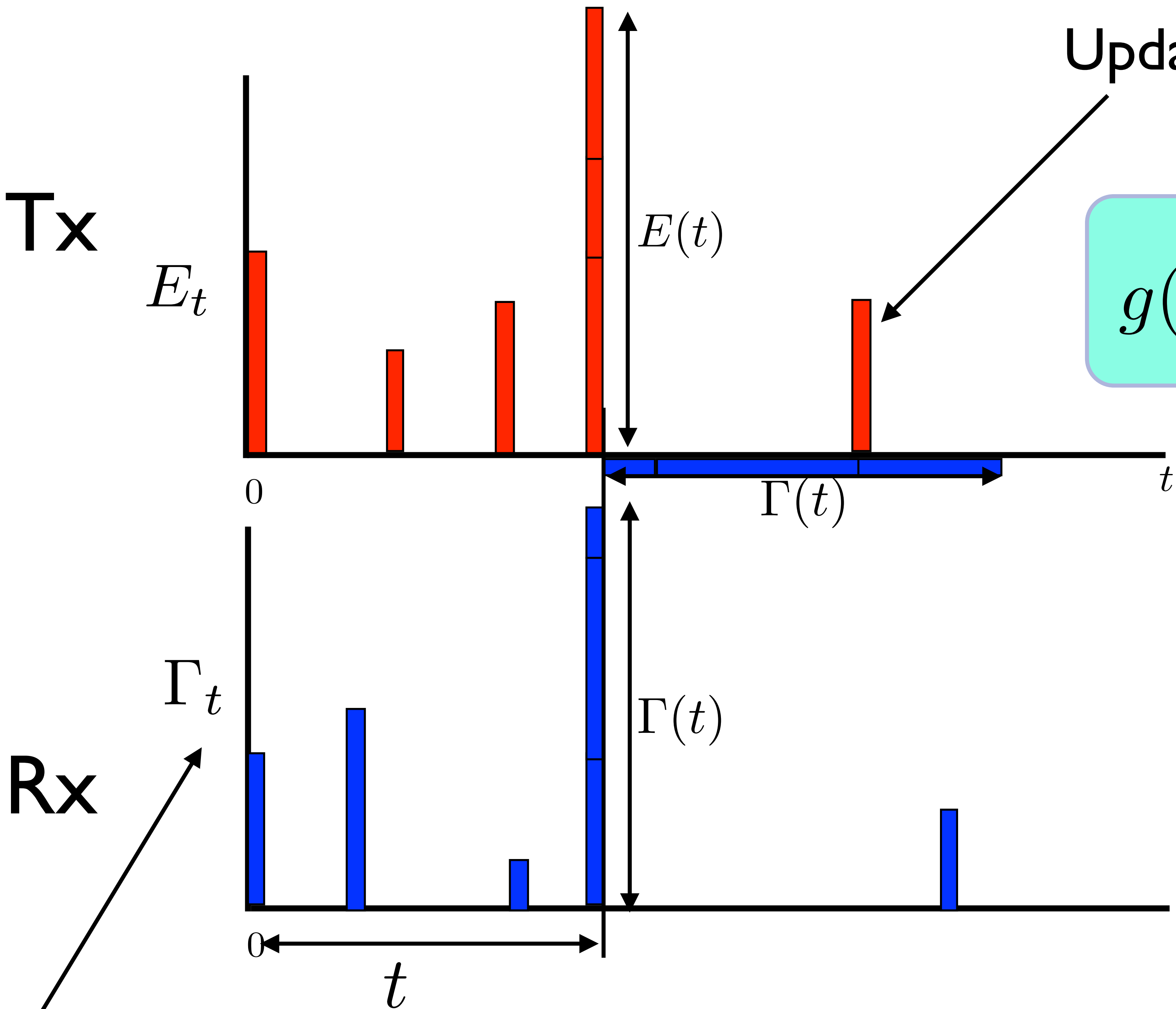


total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm

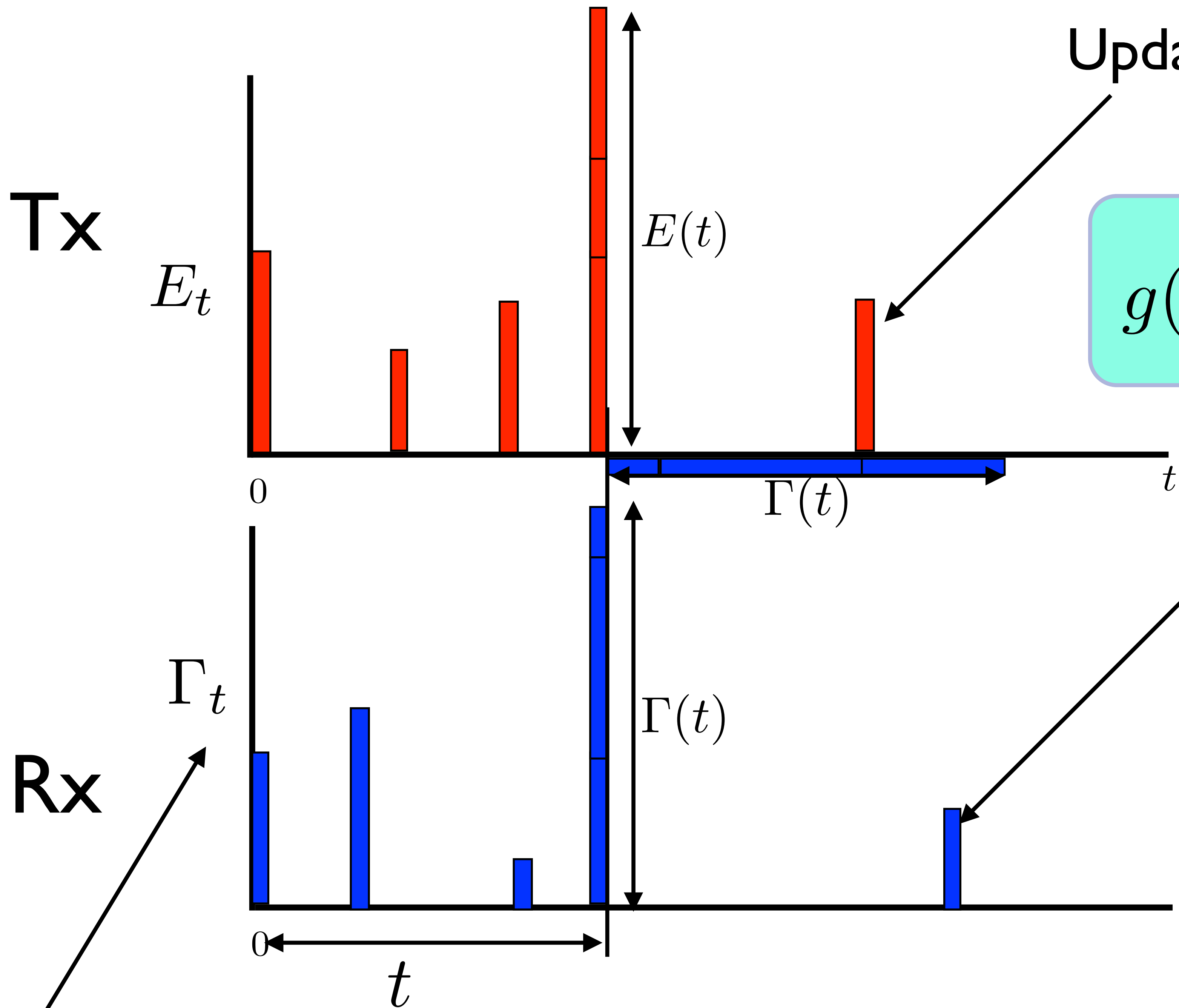
Update transmit power at each energy arrival at Tx

$$g(p_i)\frac{E_{rem}}{p_i} = B_{rem}$$



total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm



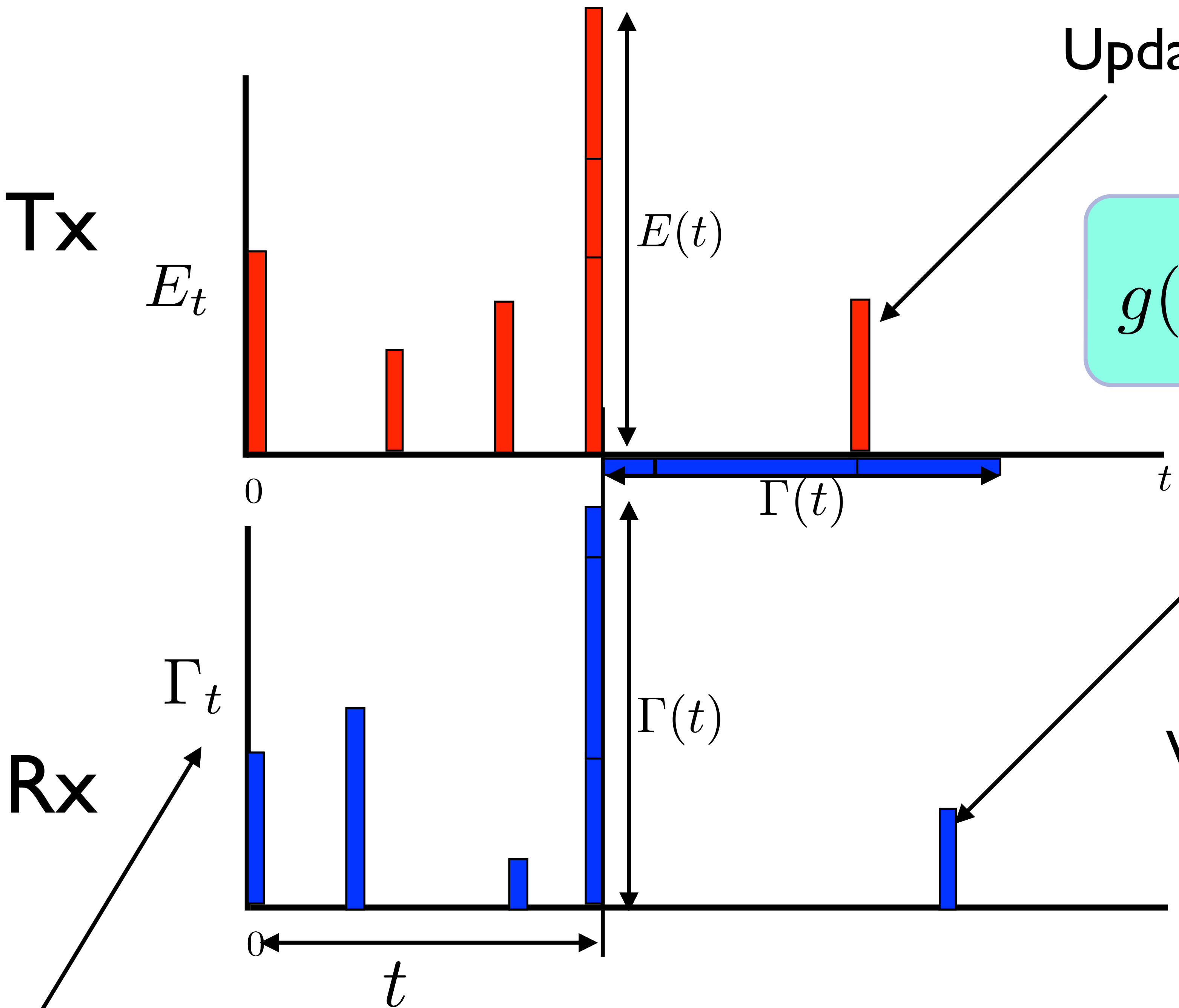
Update transmit power at each energy arrival at Tx

$$g(p_i)\frac{E_{rem}}{p_i} = B_{rem}$$

No change at any energy arrival at Rx

total time for which Rx can be ON

Lazy (Best effort delivery) Online Algorithm



Update transmit power at each energy arrival at Tx

$$g(p_i)\frac{E_{rem}}{p_i} = B_{rem}$$

No change at any energy arrival at Rx

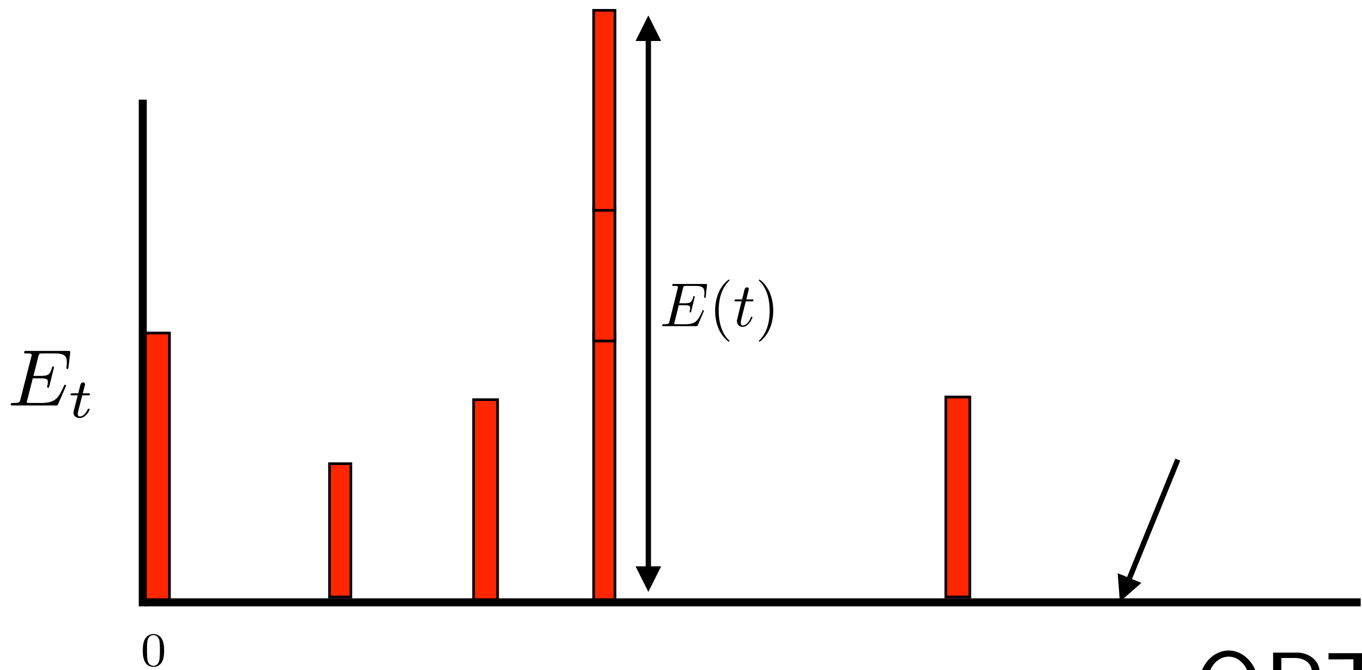
We have already checked for feasibility at T_{start}

total time for which Rx can be ON

Competitive Ratio of Lazy Online Algorithm

Theorem: The competitive ratio of Lazy Online Algorithm is < 2.

Proof - 2 step approach

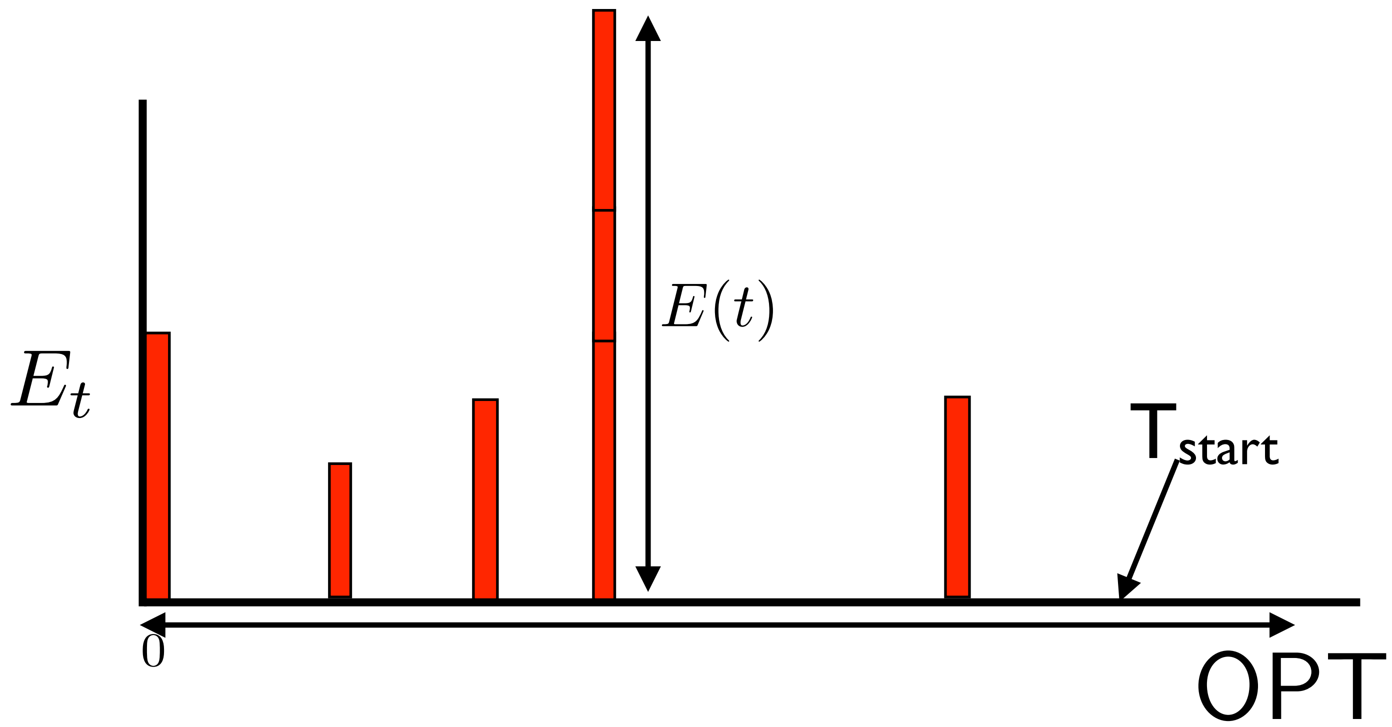


OPT finish time of optimal offline alg.

Claim 1: $T_{start} < OPT$

Thus online alg. starts before offline finishes

Proof - 2 step approach

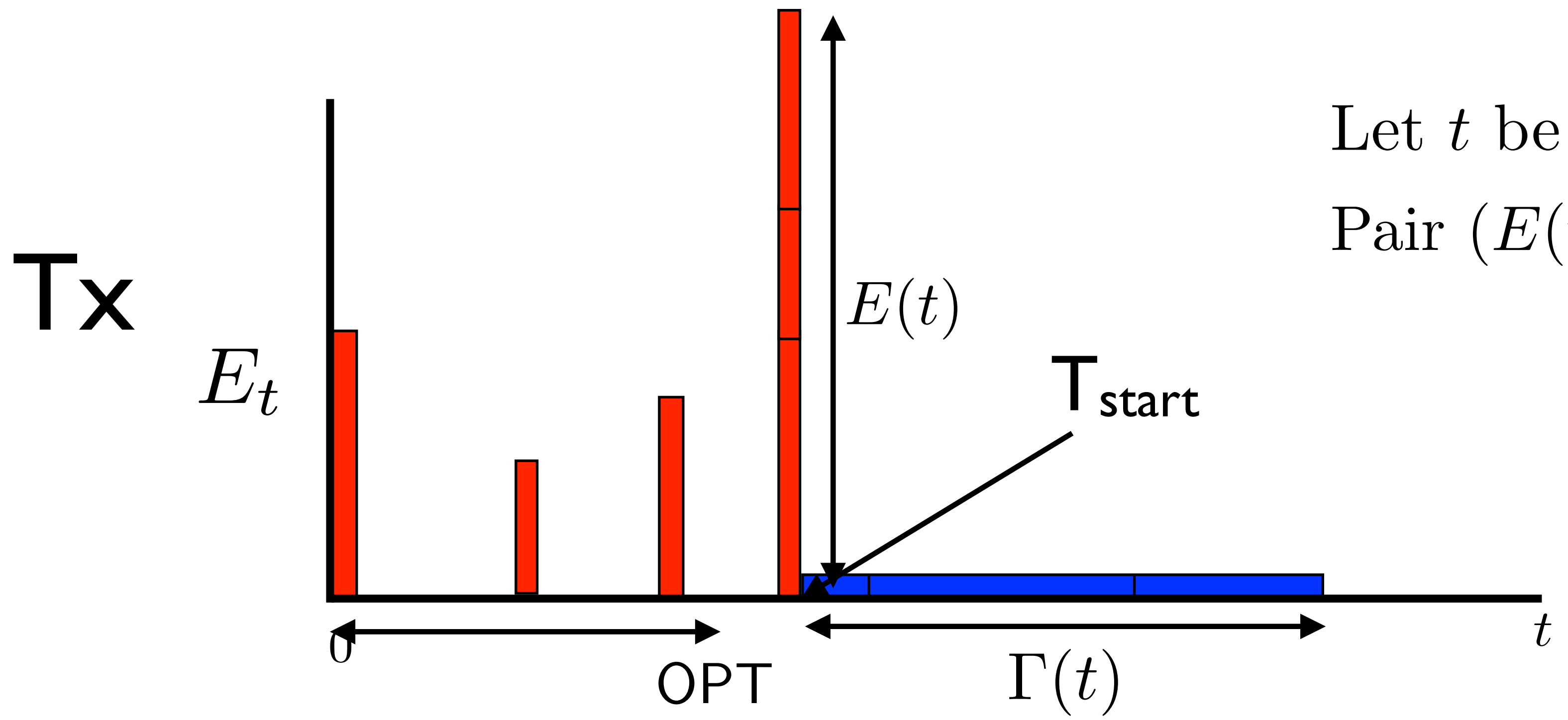


finish time of optimal offline alg.

Claim 1: $T_{start} < OPT$

Thus online alg. starts before offline finishes

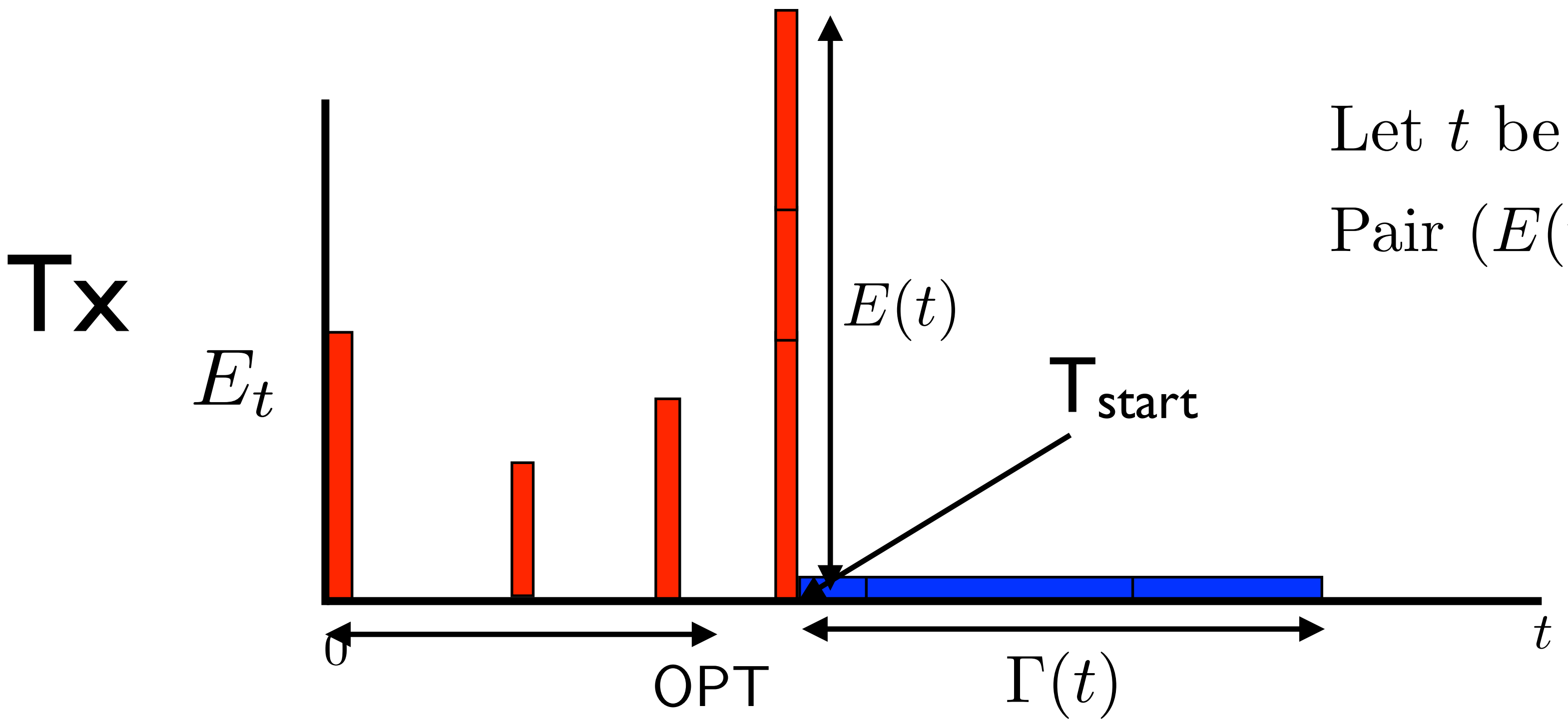
Idea: Claim 1



Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

Idea: Claim 1

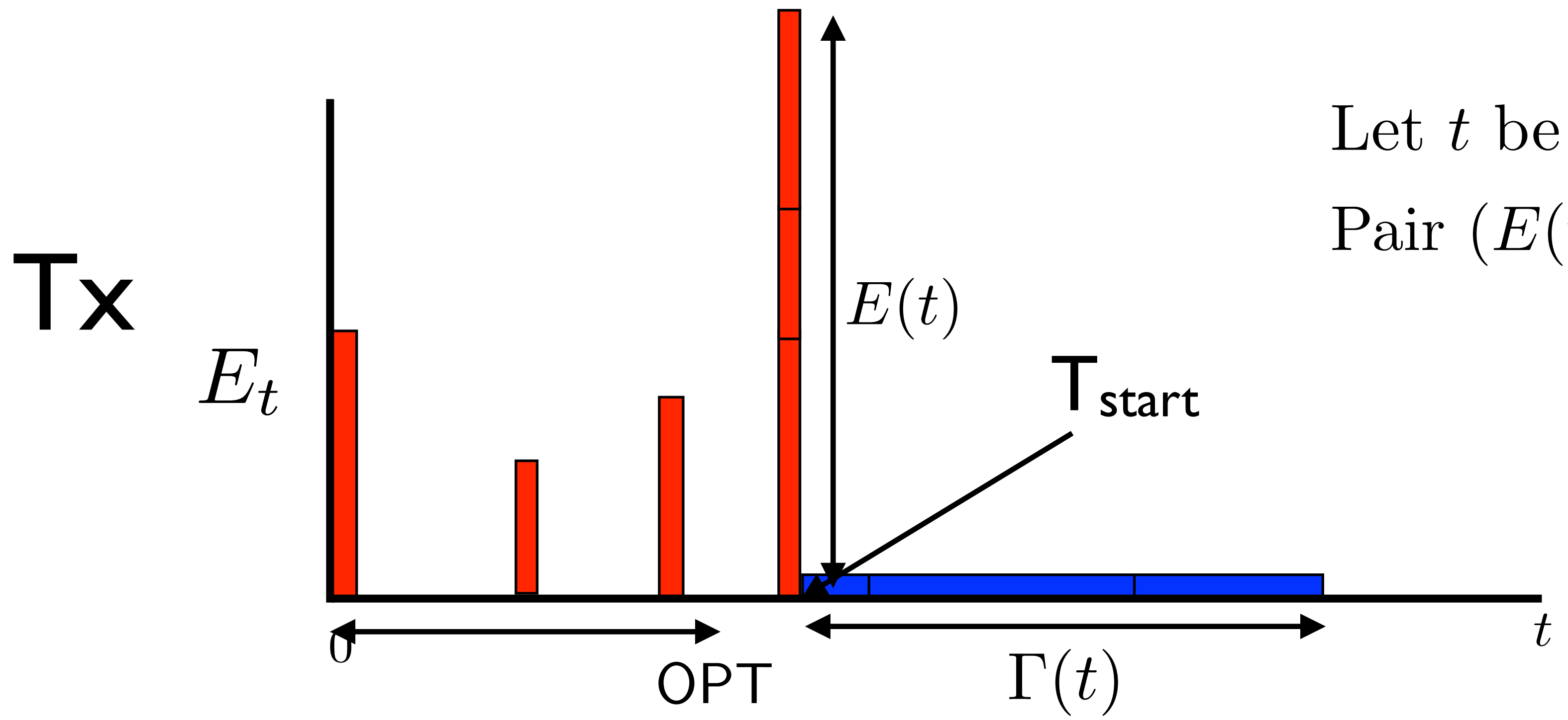


Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

total energy for OPT is at most $E(T^-_{start})$

Idea: Claim 1



Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

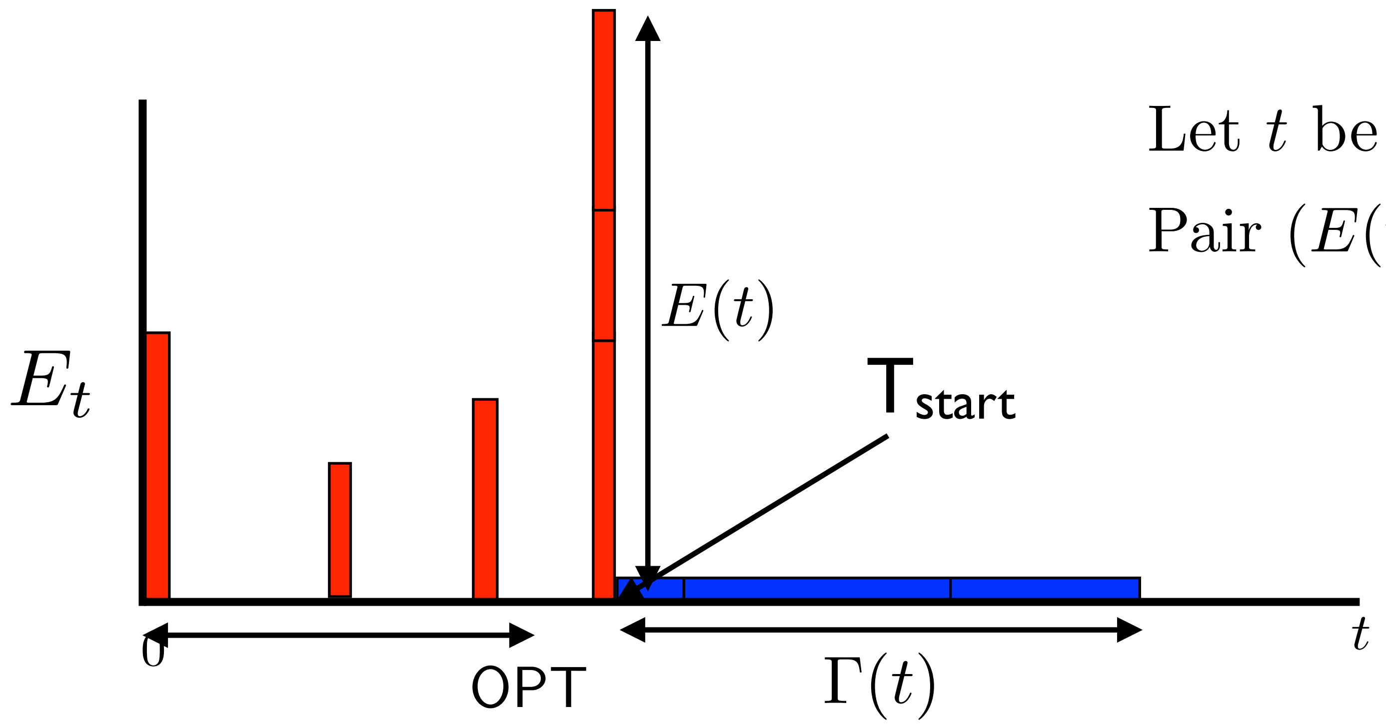
total energy for OPT is at most $E(T^{-}_{start})$

Thus using energy at most $E(T^{-}_{start})$

and receiver time at most $\Gamma(T^{-}_{start})$

Idea: Claim 1

Tx



Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

total energy for OPT is at most $E(T^-_{start})$

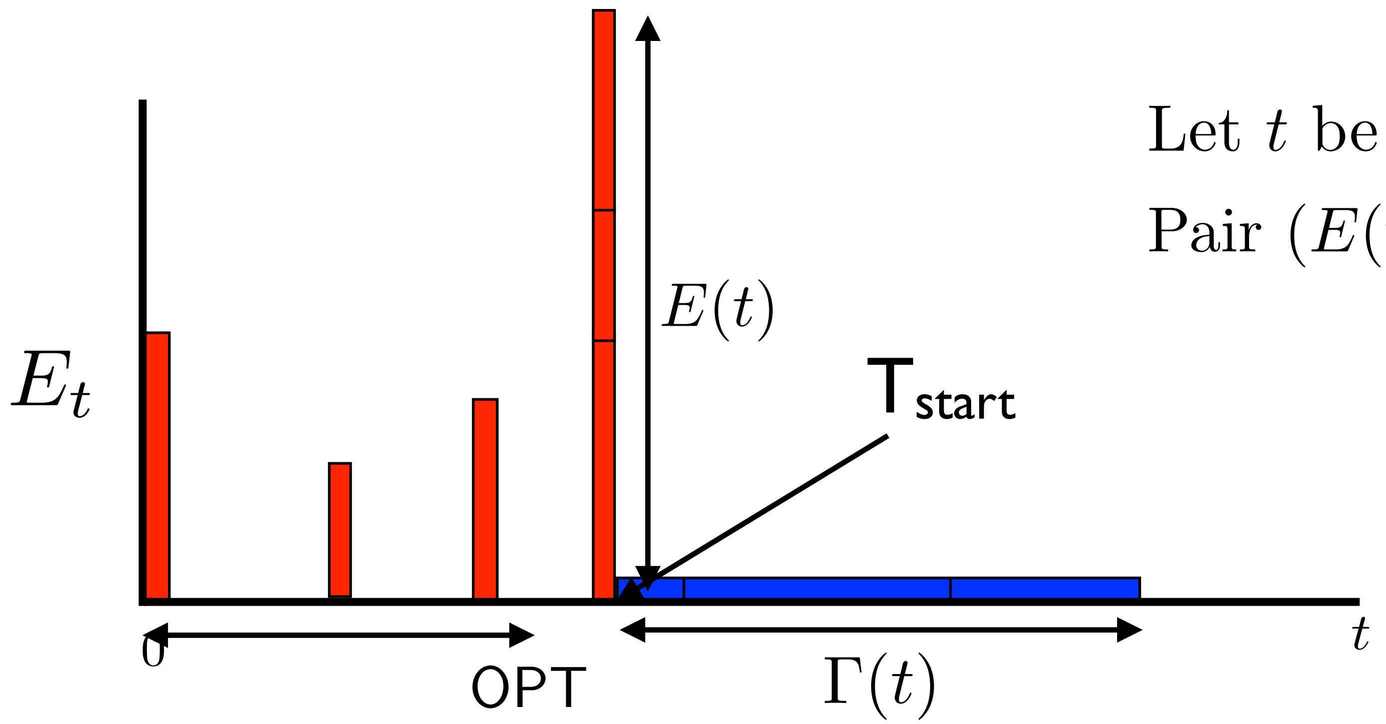
Thus using energy at most $E(T^-_{start})$

and receiver time at most $\Gamma(T^-_{start})$

OPT can transmit B bits

Idea: Claim 1

Tx



Let t be the earliest time, where

Pair $(E(t), \Gamma(t))$ is feasible

total energy for OPT is at most $E(T^-_{start})$

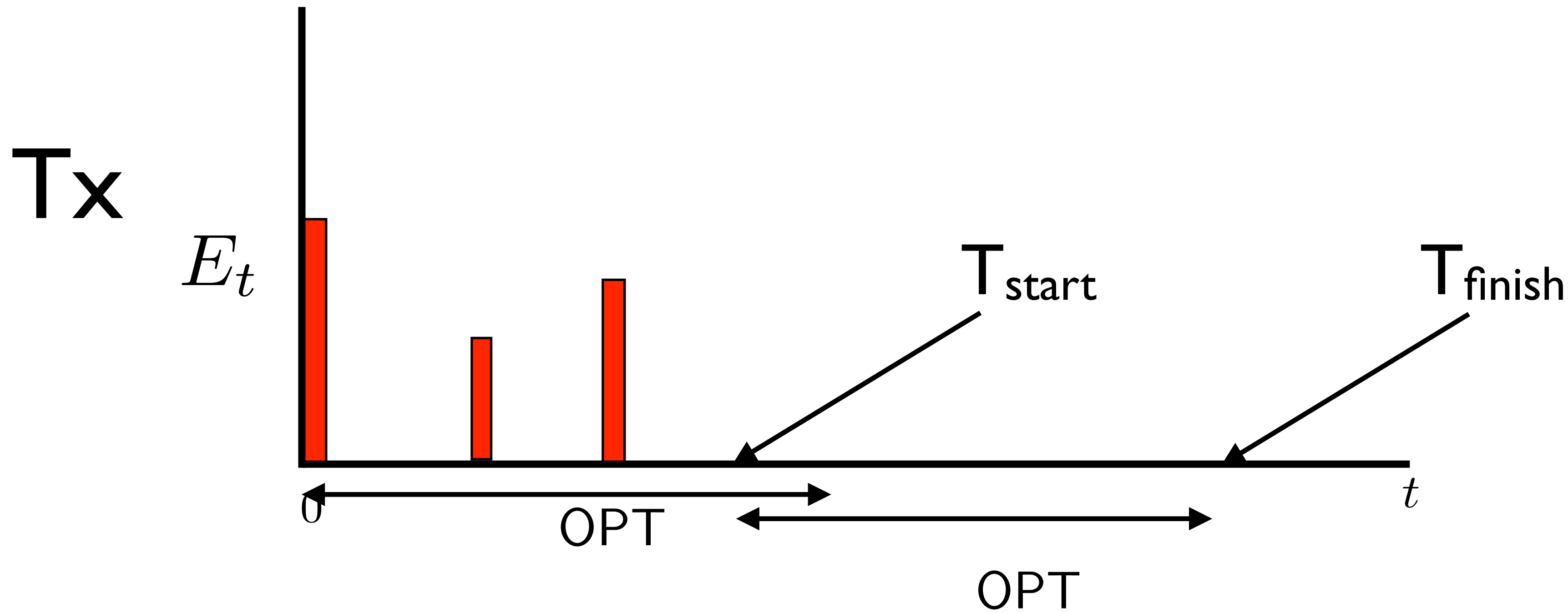
Thus using energy at most $E(T^-_{start})$

and receiver time at most $\Gamma(T^-_{start})$

OPT can transmit B bits

Contradiction to the definition of T_{start}

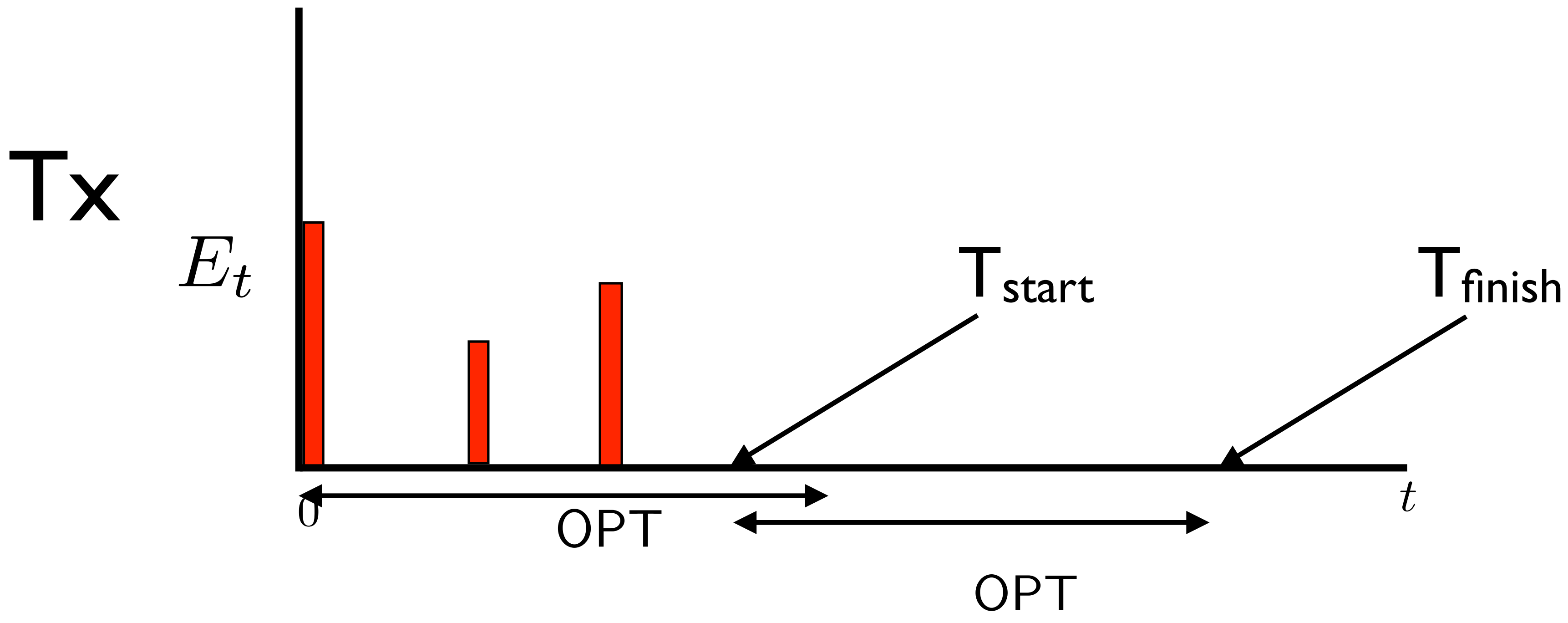
Proof - claim 2



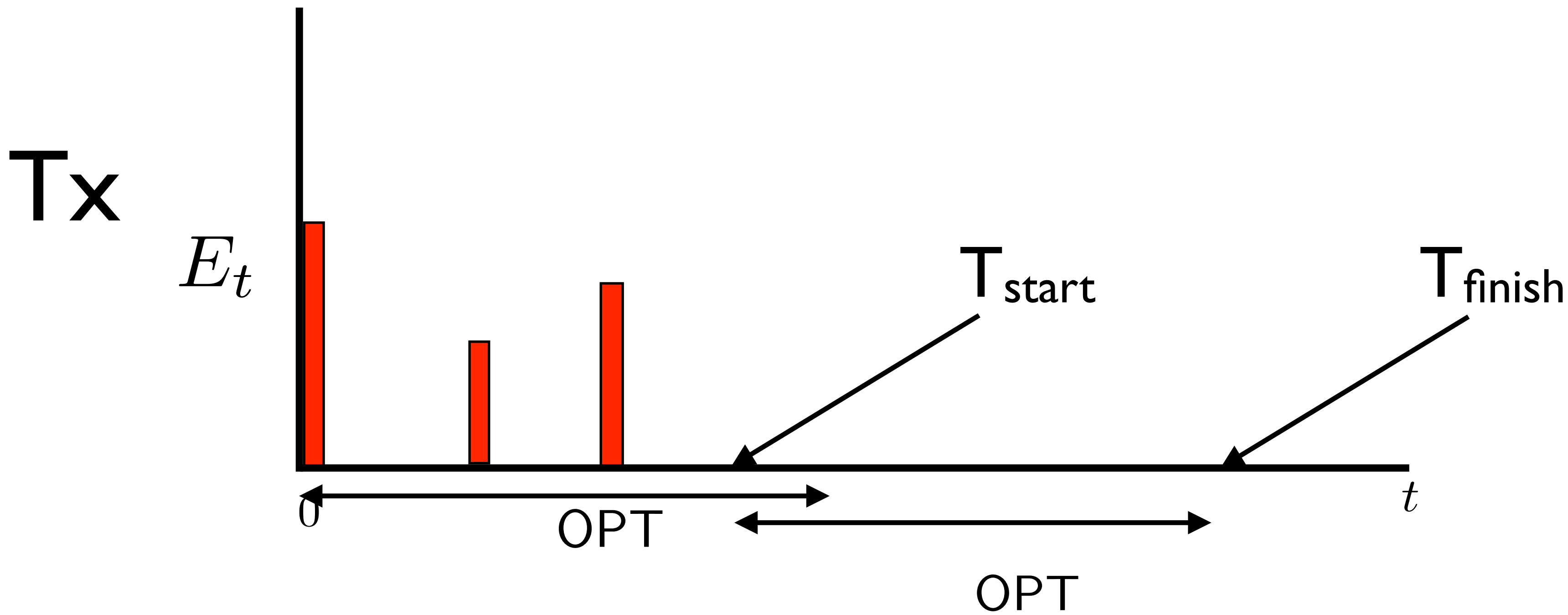
Claim 2: Once the alg. starts it takes at most OPT time

Thus total time is < 2 OPT

Idea - Claim 2



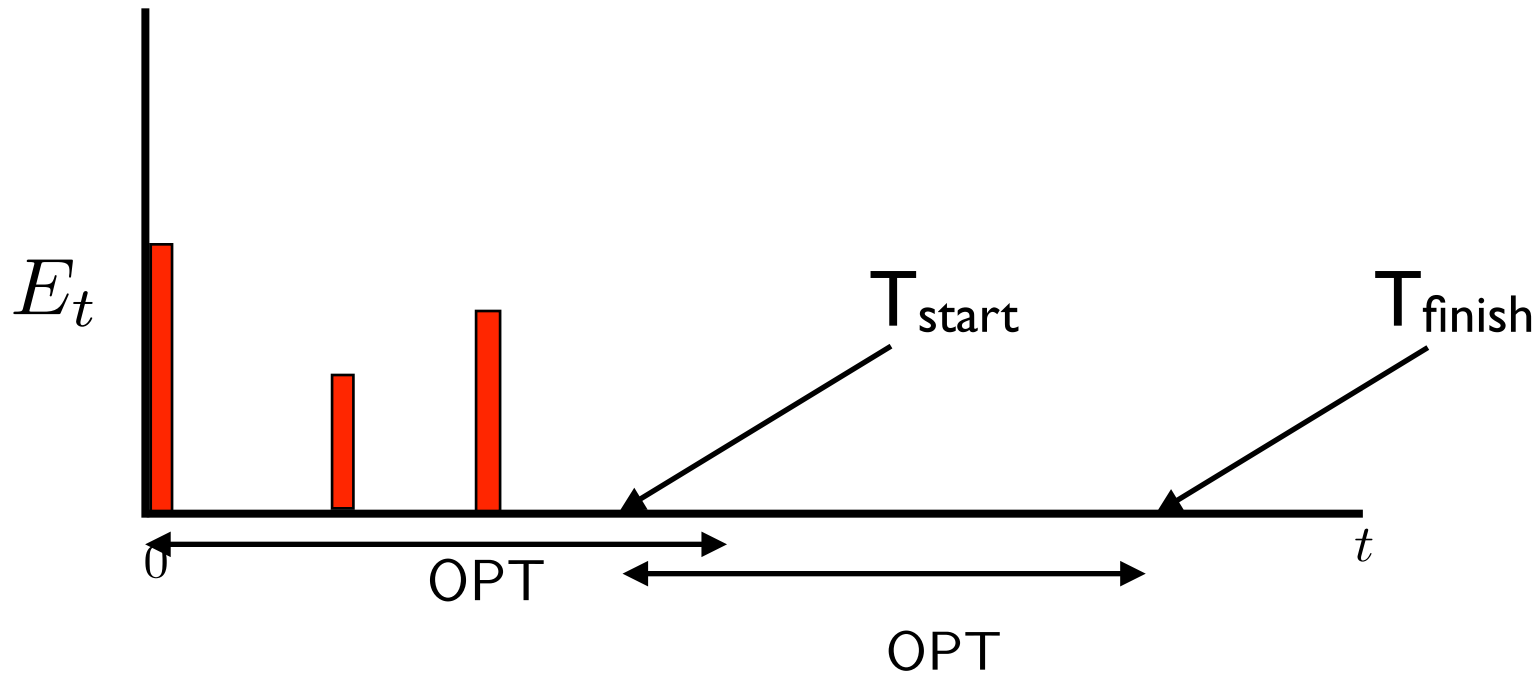
Idea - Claim 2



Since OFF finishes by OPT $\quad$ Clearly, $B \leq \mathsf{OPT}\, g\left(\frac{E(\mathsf{OPT}^-)}{\mathsf{OPT}}\right)$

Idea - Claim 2

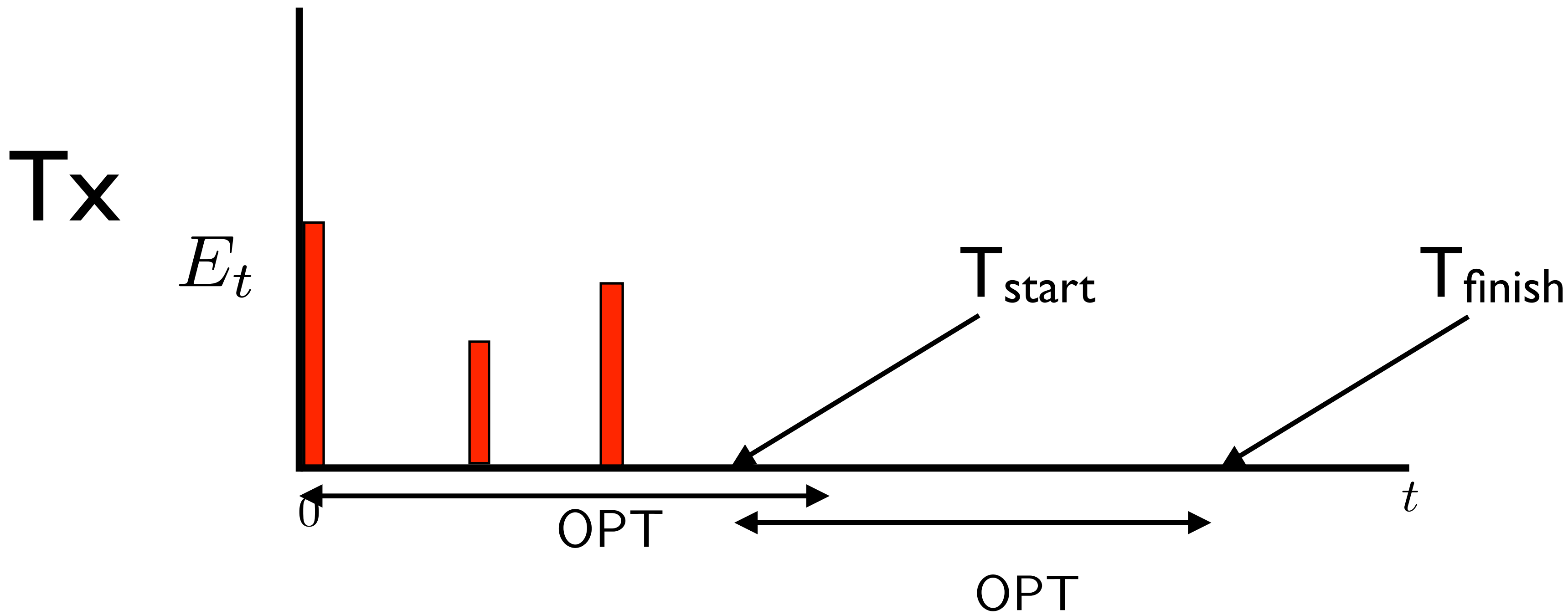
Tx



Since OFF finishes by OPT $\quad$ $\text{Clearly, } B \leq \mathsf{OPT} g\left(\frac{E(\mathsf{OPT}^-)}{\mathsf{OPT}}\right)$

Moreover from Alg. at any time t

Idea - Claim 2



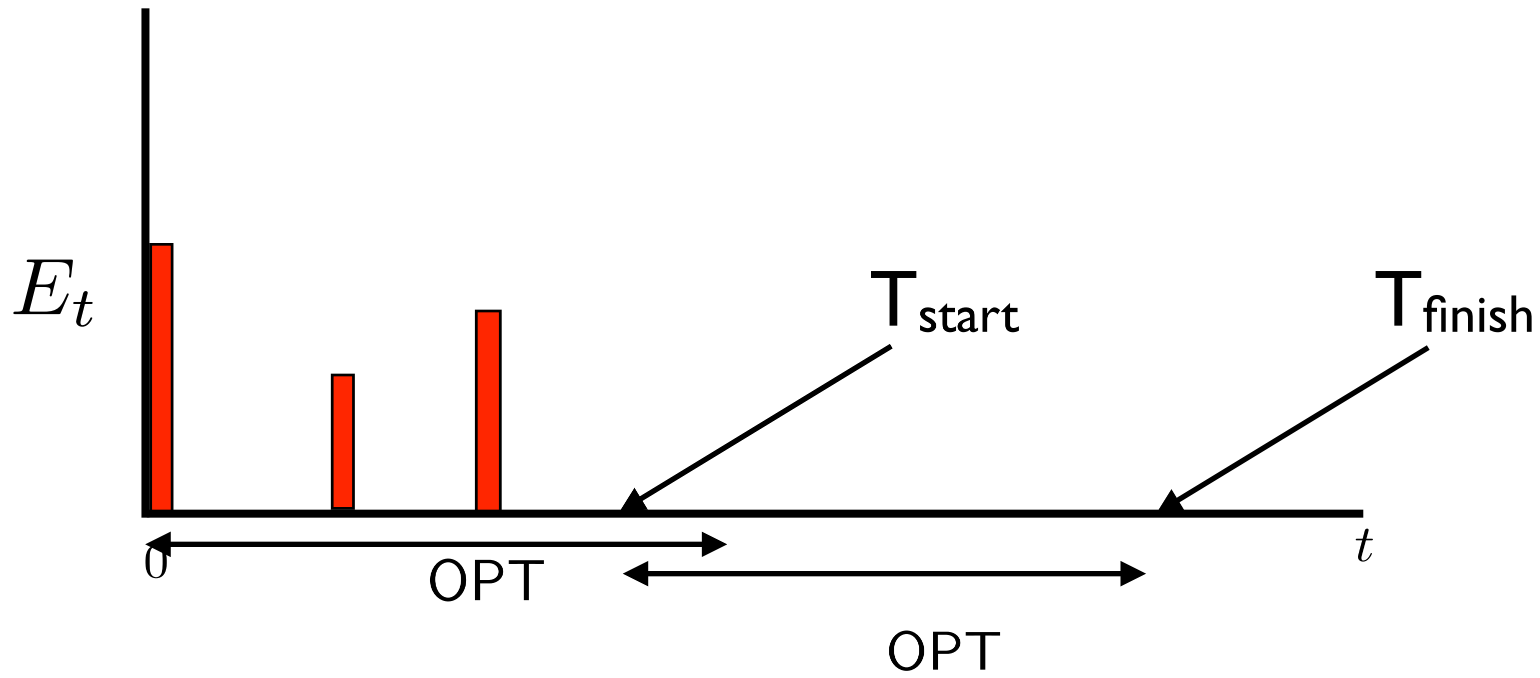
Since OFF finishes by OPT $\quad$ Clearly, $B \leq \mathsf{OPT} g\left(\frac{E(\mathsf{OPT}^-)}{\mathsf{OPT}}\right)$

Moreover from Alg. at any time t

$$\frac{E(t)}{p} g(p) \leq B \text{ with equality only at } t = T_{start}$$

Idea - Claim 2

Tx



Since OFF finishes by OPT

$$\text{Clearly, } B \leq \mathsf{OPT} g\left(\frac{E(\mathsf{OPT}^-)}{\mathsf{OPT}}\right)$$

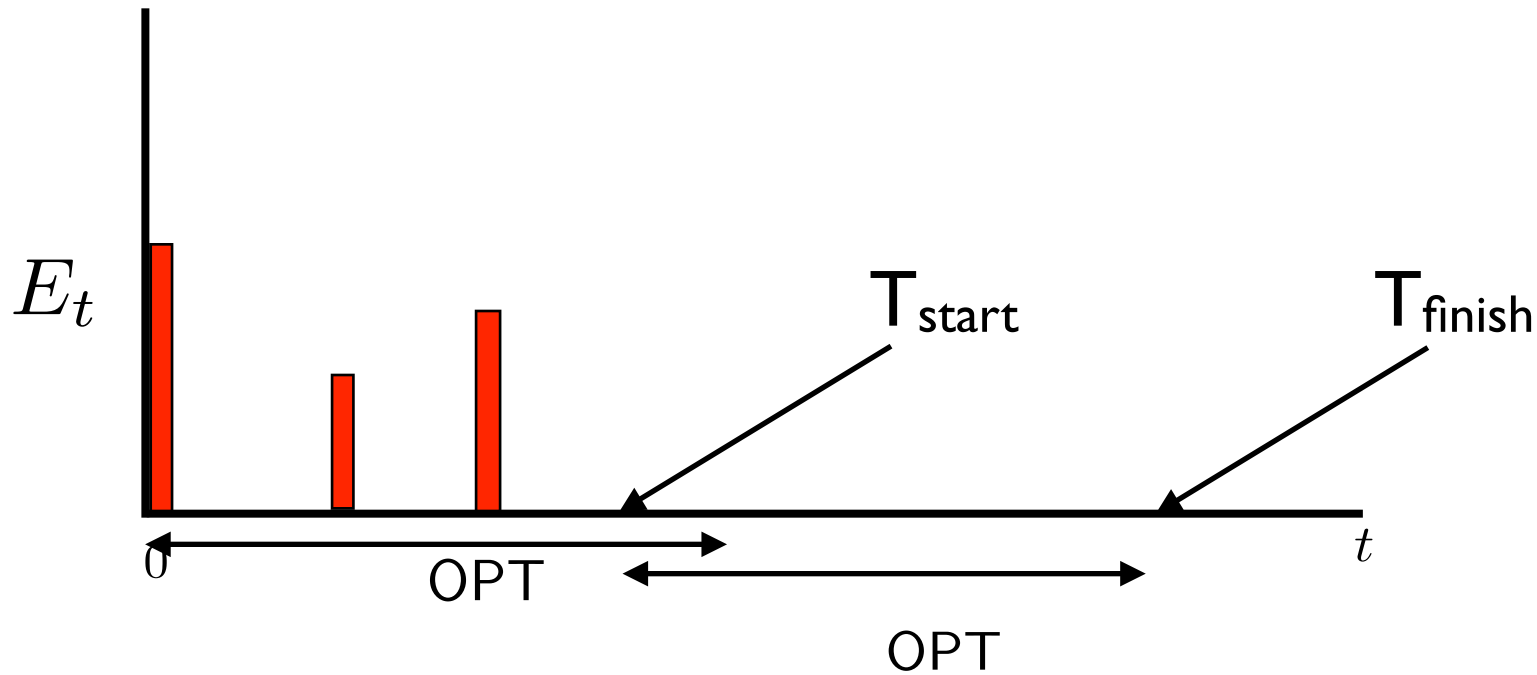
Moreover from Alg. at any time t

$$\frac{E(t)}{p} g(p) \leq B \text{ with equality only at } t = T_{start}$$

$$\text{At } t = OPT^-,\ \frac{E(OPT^-)}{p} g(p) \leq B \leq \mathsf{OPT} g\left(\frac{E(\mathsf{OPT}^-)}{\mathsf{OPT}}\right)$$

Idea - Claim 2

Tx



Since OFF finishes by OPT

Clearly, $B \leq \text{OPT} g\left(\frac{E(\text{OPT}^-)}{\text{OPT}}\right)$

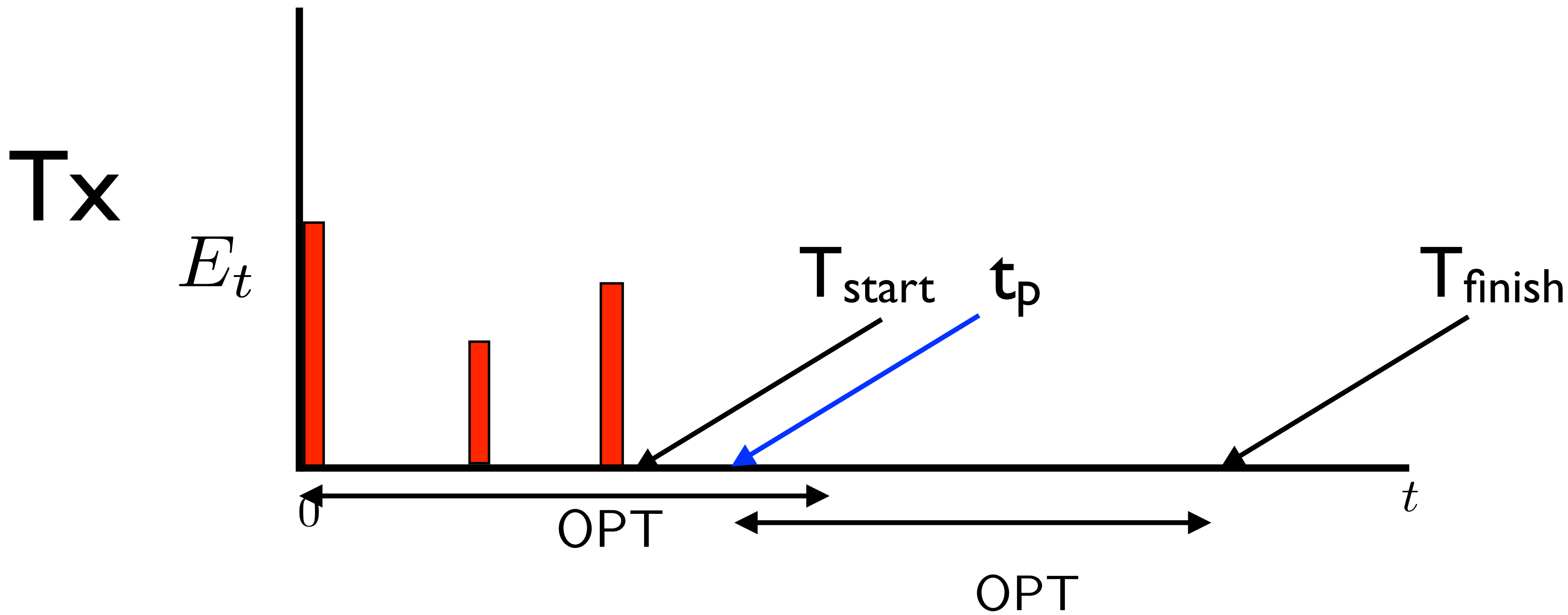
Moreover from Alg. at any time t

$$\frac{E(t)}{p} g(p) \leq B \text{ with equality only at } t = T_{start}$$

At $t = OPT^-$, $\frac{E(OPT^-)}{p} g(p) \leq B \leq \text{OPT} g\left(\frac{E(\text{OPT}^-)}{\text{OPT}}\right)$

$$\frac{g(p)}{p} \text{ is monotonic, } \frac{E(\text{OPT}^-)}{p} < \text{OPT}$$

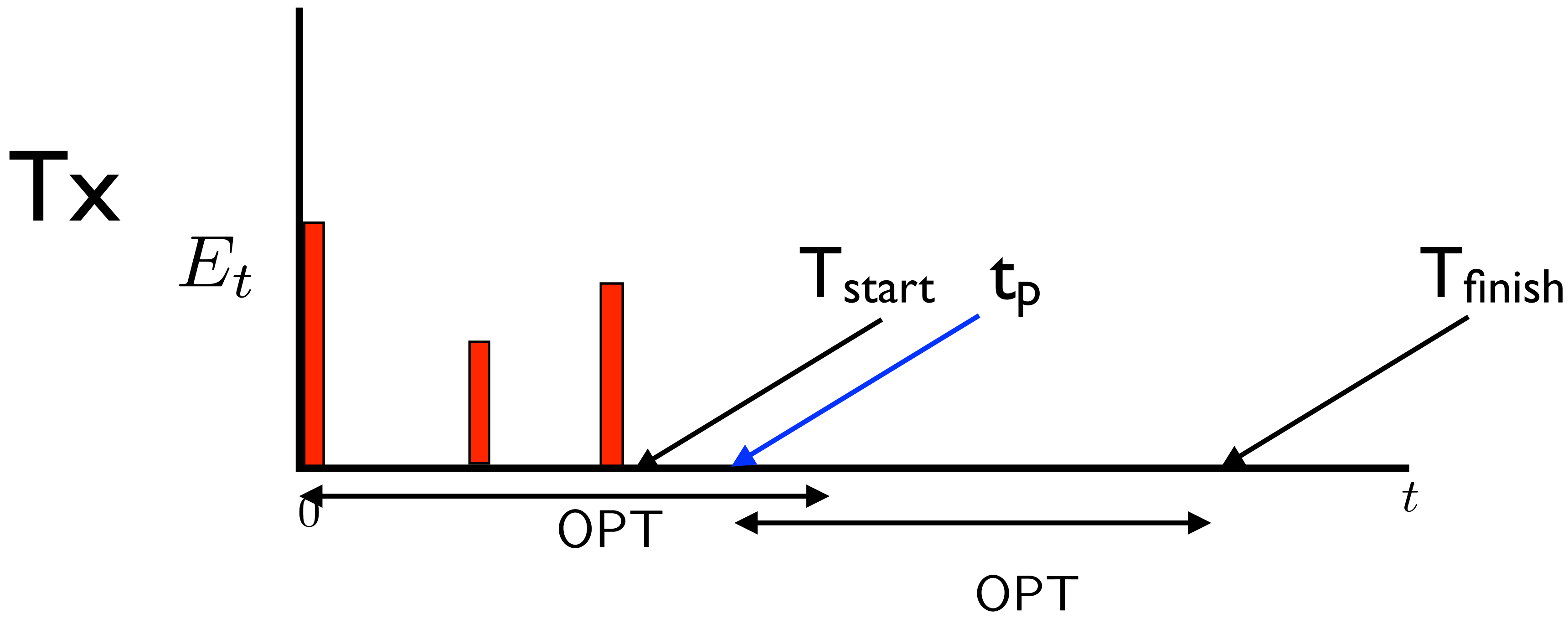
Idea - Claim 2



$$T_{finish} - t_p \le \frac{B_{rem}(t_p)}{g(p)} = \frac{E_{rem}(t_p)}{p} \le \frac{E(t_p)}{p} \le \frac{E(\mathsf{OPT}^-)}{p}$$

Earlier assertion

Idea - Claim 2

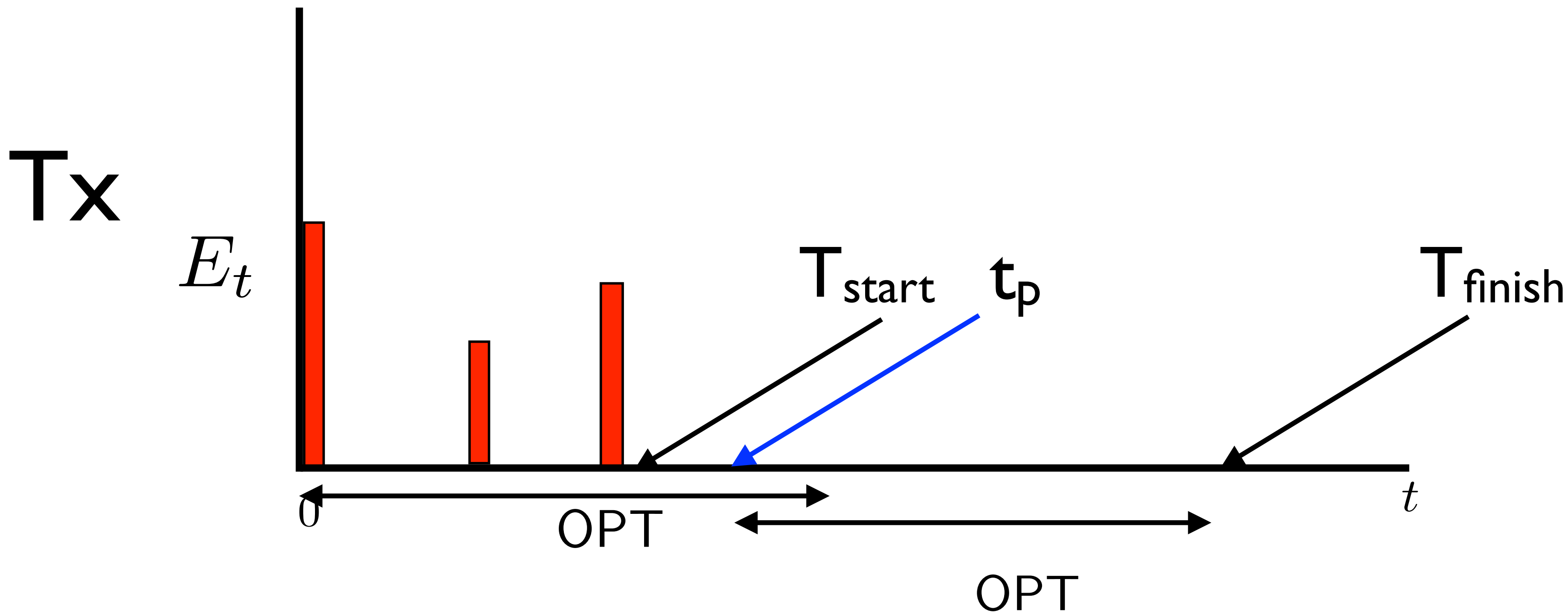


$$T_{finish} - t_p \le \frac{B_{rem}(t_p)}{g(p)} = \frac{E_{rem}(t_p)}{p} \le \frac{E(t_p)}{p} \le \frac{E(\mathsf{OPT}^-)}{p}$$

Earlier assertion

$$\frac{g(p)}{p} \text{ is monotonic, } \frac{E(\mathsf{OPT}^-)}{p} < \mathsf{OPT}$$

Idea - Claim 2



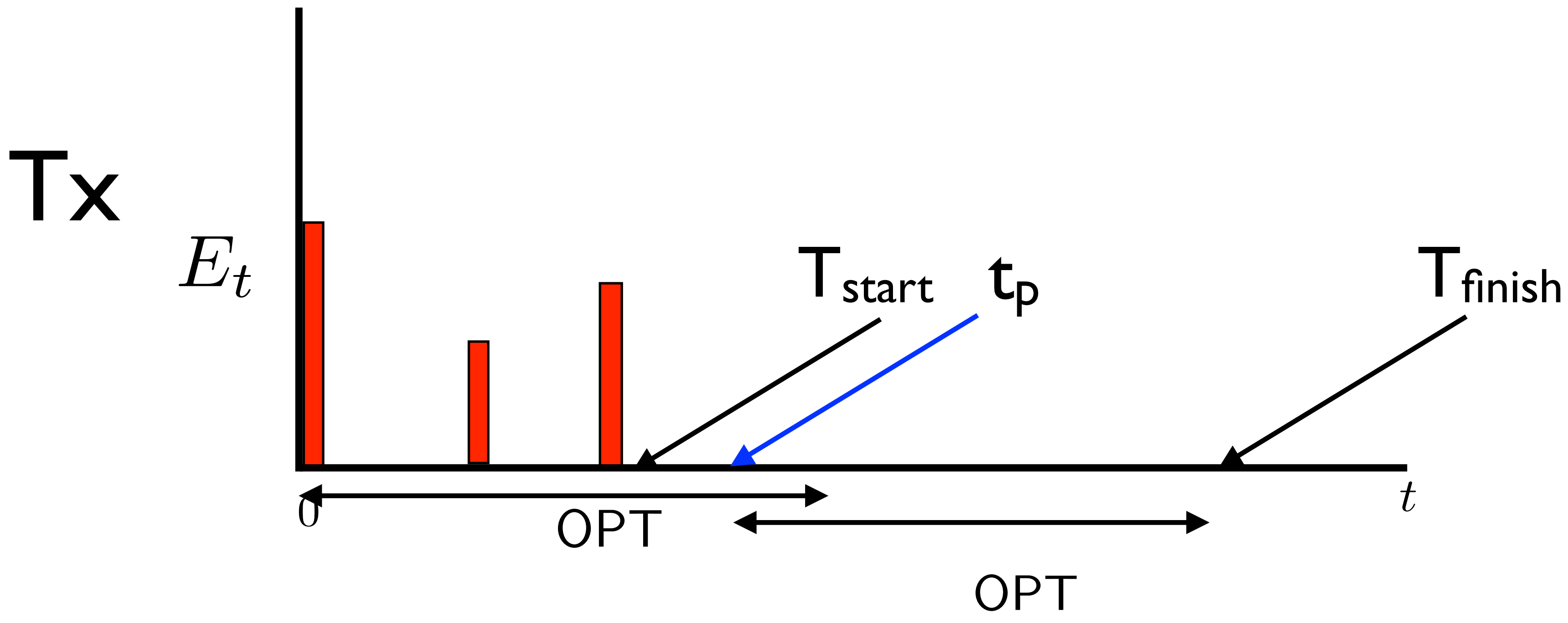
$$T_{finish} - t_p \le \frac{B_{rem}(t_p)}{g(p)} = \frac{E_{rem}(t_p)}{p} \le \frac{E(t_p)}{p} \le \frac{E(\mathsf{OPT}^-)}{p}$$

Earlier assertion

$$\frac{g(p)}{p} \text{ is monotonic, } \frac{E(\mathsf{OPT}^-)}{p} < \mathsf{OPT}$$

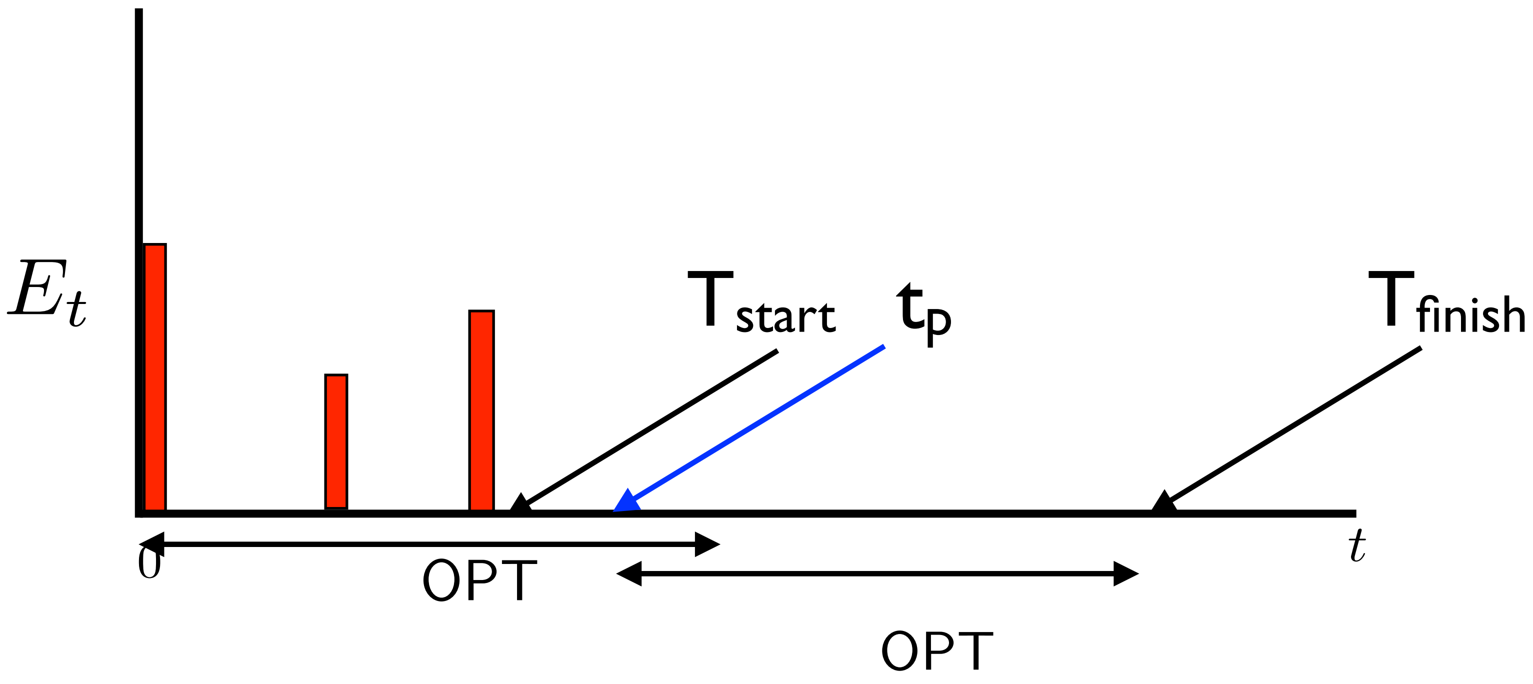
$$T_{finish} - t_p \le \mathsf{OPT}$$

Idea - Claim 2



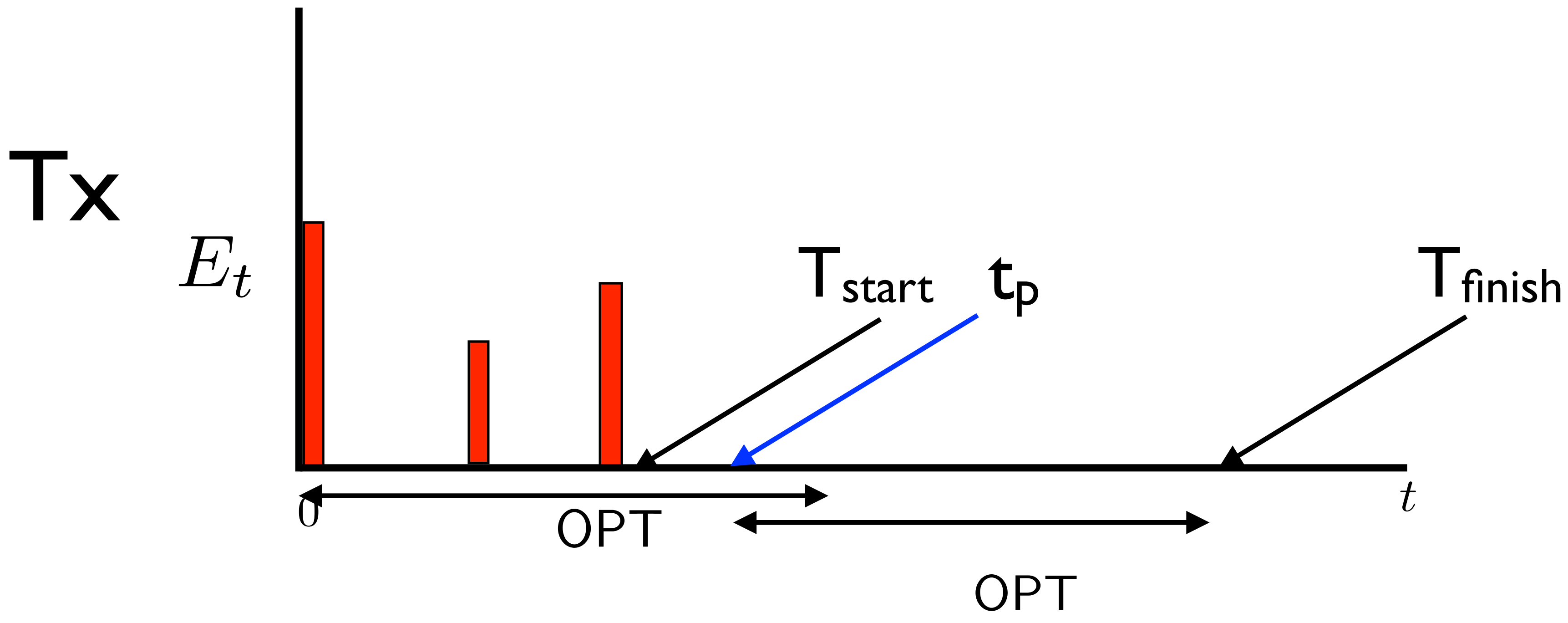
Idea - Claim 2

Tx



Let power at OPT$^-$ be p

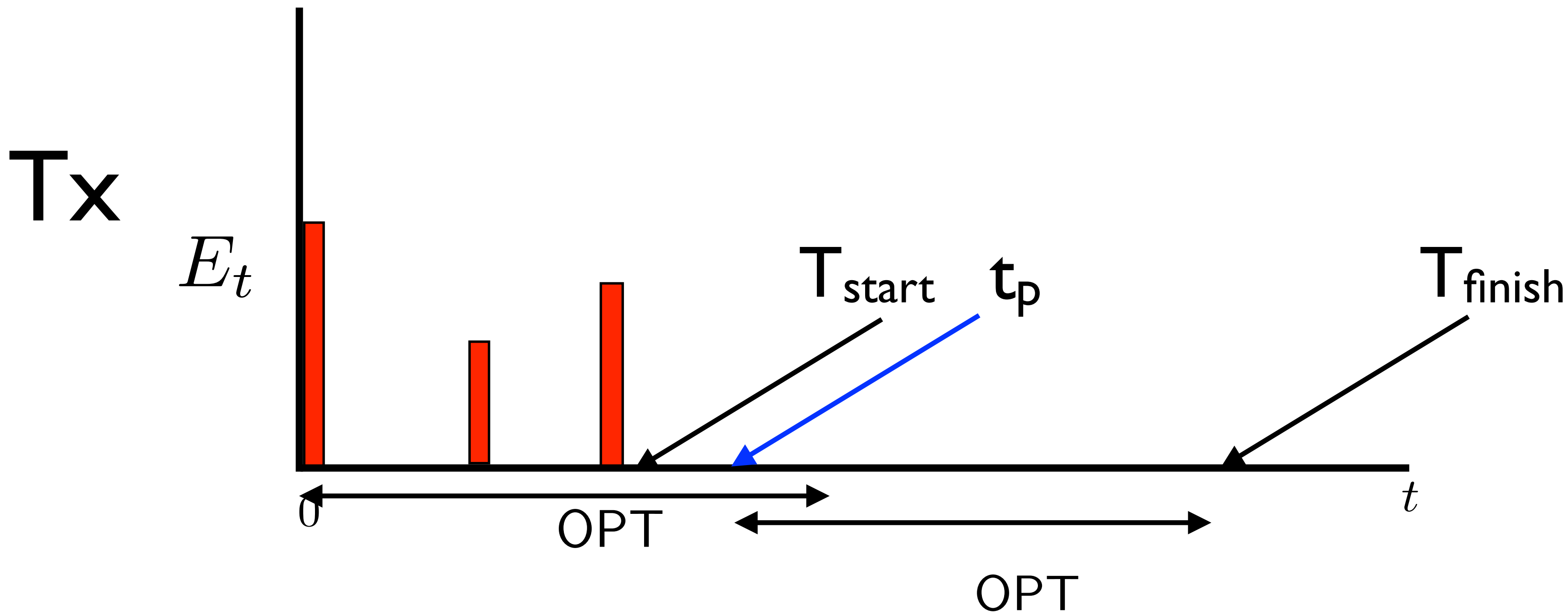
Idea - Claim 2



Let power at OPT^- be p

Since Alg. start before OPT p > 0

Idea - Claim 2



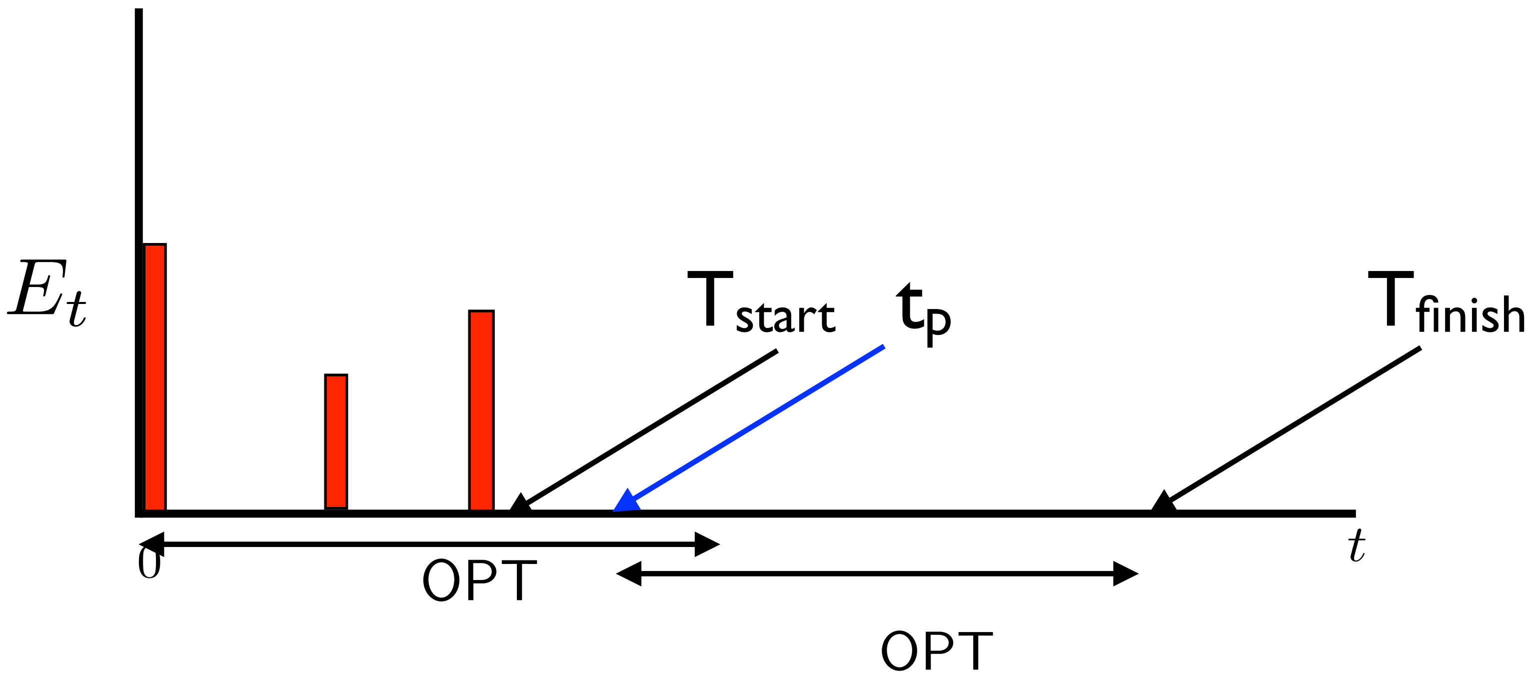
Let power at OPT^- be p

Since Alg. start before OPT p > 0

Let time t_p < OPT where transmission with power p starts

Idea - Claim 2

Tx



Let power at OPT⁻ be p

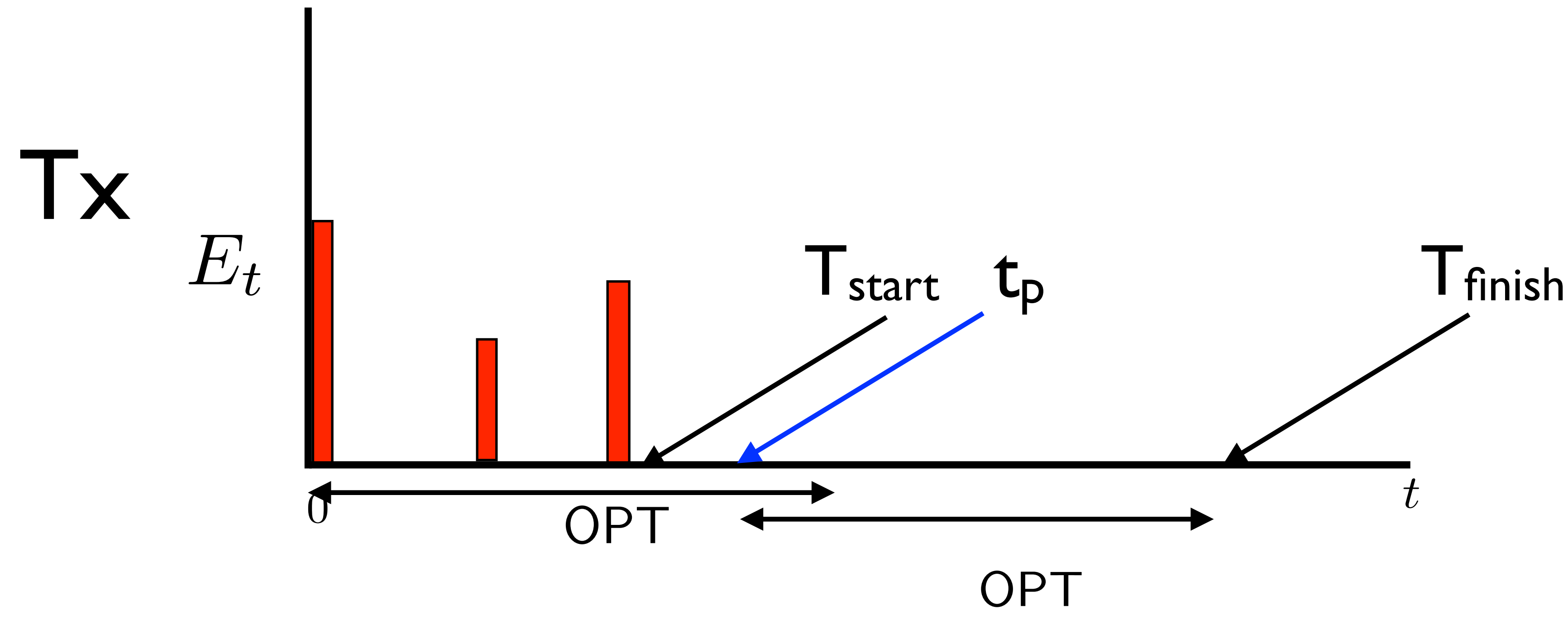
Since Alg. start before OPT p > 0

Let time t_p < OPT where transmission with power p starts

$$T_{finish} - t_p \leq \frac{B_{rem}(t_p)}{g(p)}$$

can be shown

Idea - Claim 2



Let power at OPT^- be p

Since Alg. start before OPT $p > 0$

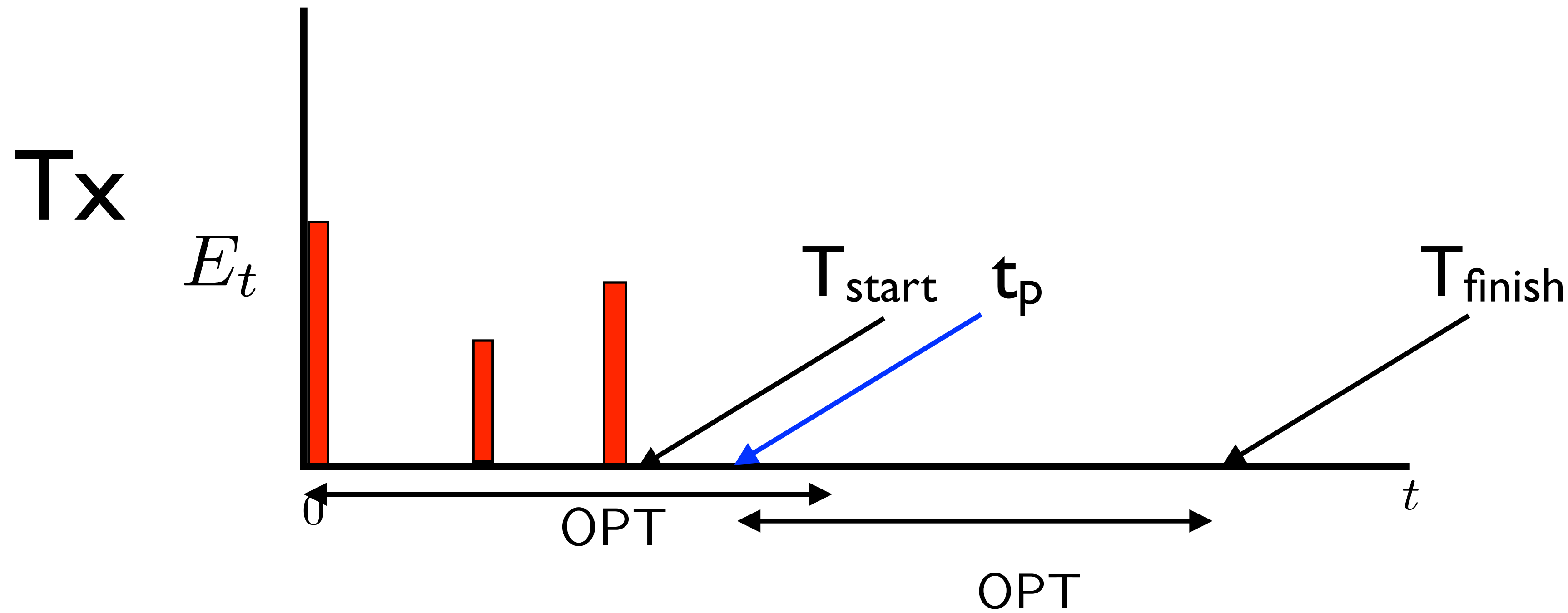
Let time $t_p < OPT$ where transmission with power p starts

$$T_{finish} - t_p \leq \frac{B_{rem}(t_p)}{g(p)} = \frac{E_{rem}(t_p)}{p}$$

can be shown

Alg. definition

Idea - Claim 2



Let power at OPT^{-} be p

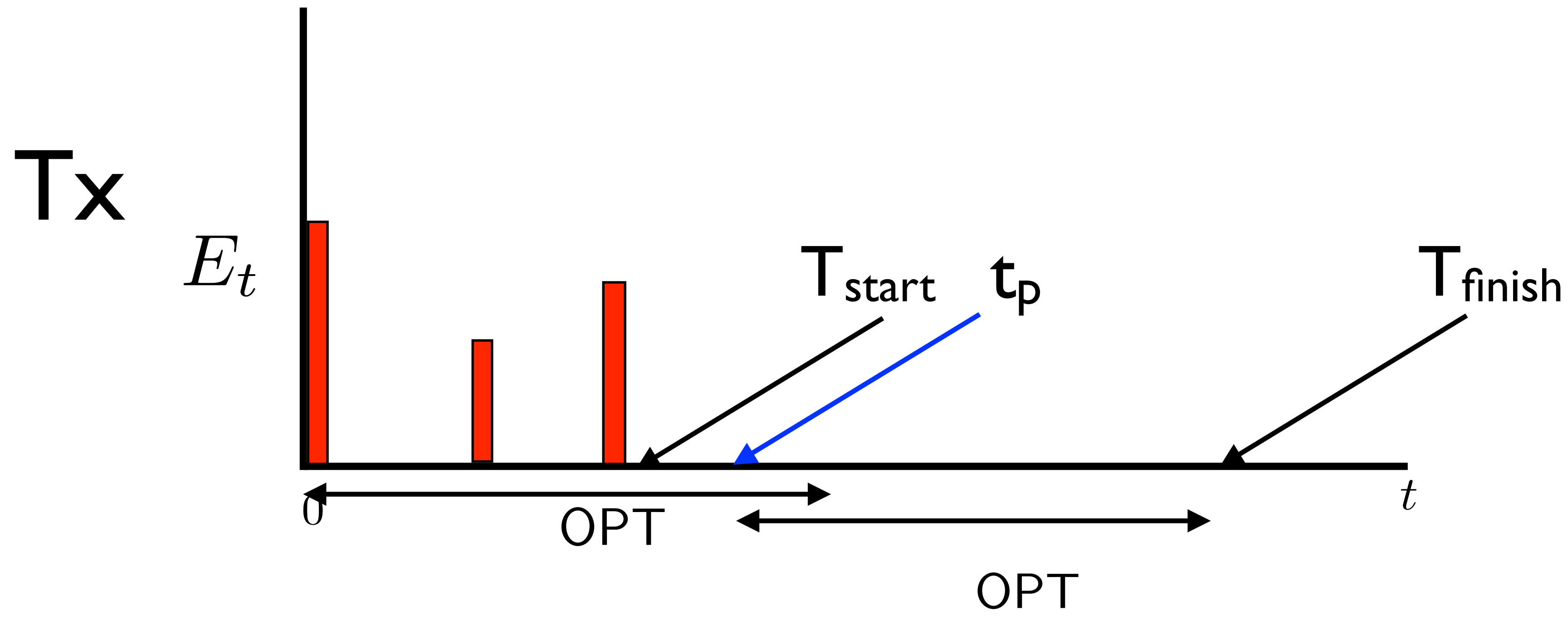
Since Alg. start before OPT p > 0

Let time $t_p < OPT$ where transmission with power p starts

$$T_{finish} - t_p \leq \frac{B_{rem}(t_p)}{g(p)} = \frac{E_{rem}(t_p)}{p} \leq \frac{E(t_p)}{p}$$

can be shown Alg. definition energy constraint

Idea - Claim 2



Let power at OPT^- be p

Since Alg. start before OPT p > 0

Let time t_p < OPT where transmission with power p starts

$$T_{finish} - t_p \leq \frac{B_{rem}(t_p)}{g(p)} = \frac{E_{rem}(t_p)}{p} \leq \frac{E(t_p)}{p} \leq \frac{E(\mathrm{OPT}^-)}{p}$$

can be shown Alg. definition energy constraint t_p < OPT

Lower Bound on the Competitive Ratio of any Online Algorithm

Lower Bound on the Competitive Ratio of any Online Algorithm

Derive an online algorithm independent lower bound on

$$\min_{O} \max_{\sigma} \frac{T_O}{T^\star}$$

Lower Bound on the Competitive Ratio of any Online Algorithm

Derive an online algorithm independent lower bound on

$$\min_{O} \max_{\sigma} \frac{T_O}{T^\star}$$

Sufficient to consider some specific input seq. $\{\sigma_i\}$

Lower Bound on the Competitive Ratio of any Online Algorithm

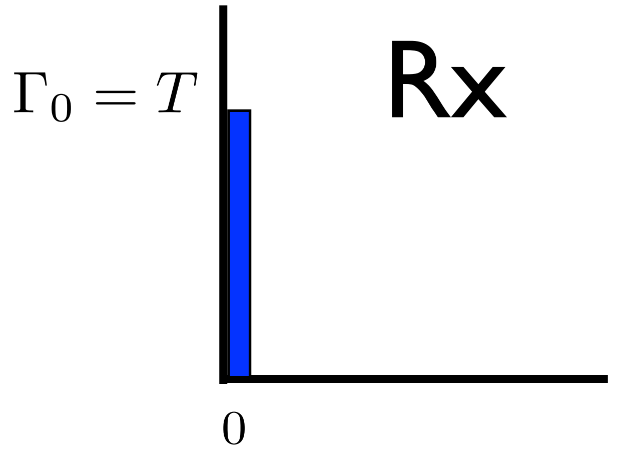
Derive an online algorithm independent lower bound on

$$\min_{O} \max_{\sigma} \frac{T_O}{T^\star}$$

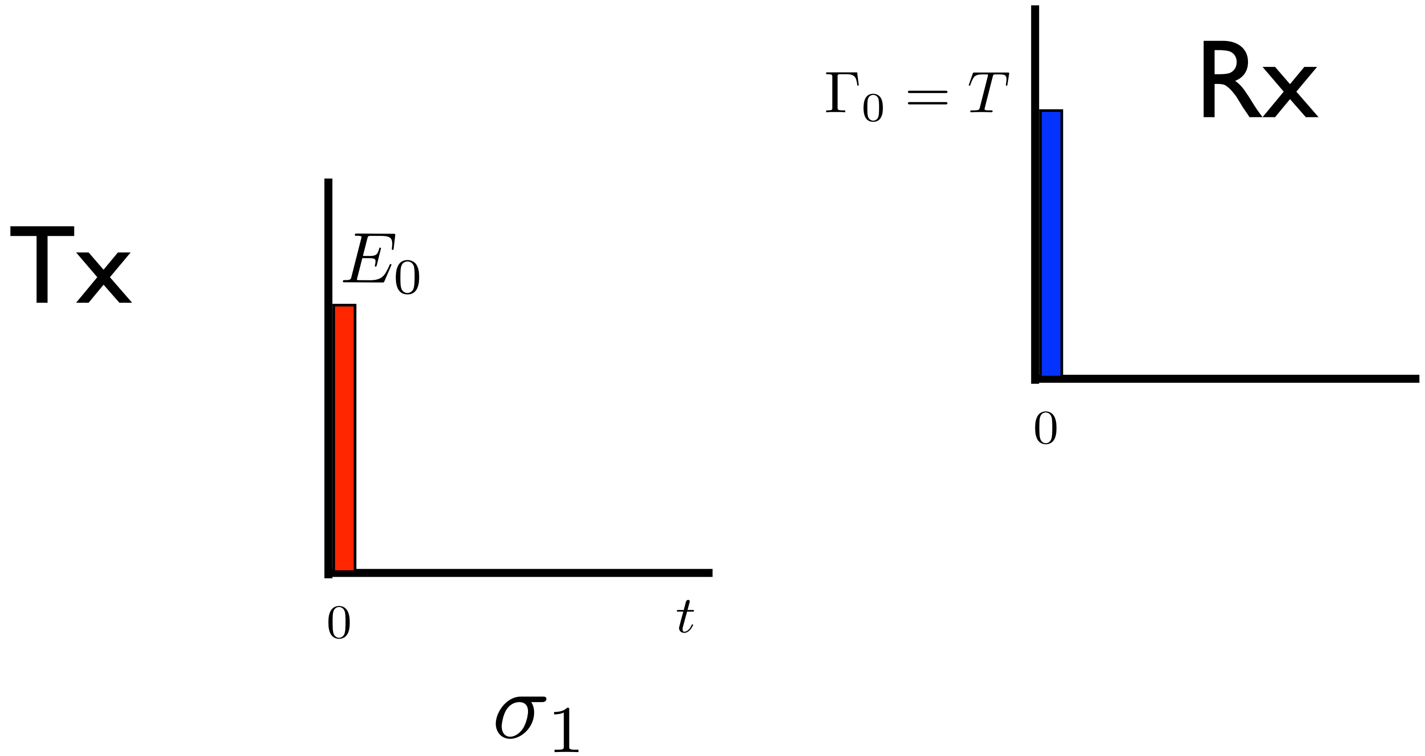
Sufficient to consider some specific input seq. $\{\sigma_i\}$

Theorem: The competitive ratio of any Online Algorithm is > 2-a for any a >0.

Idea

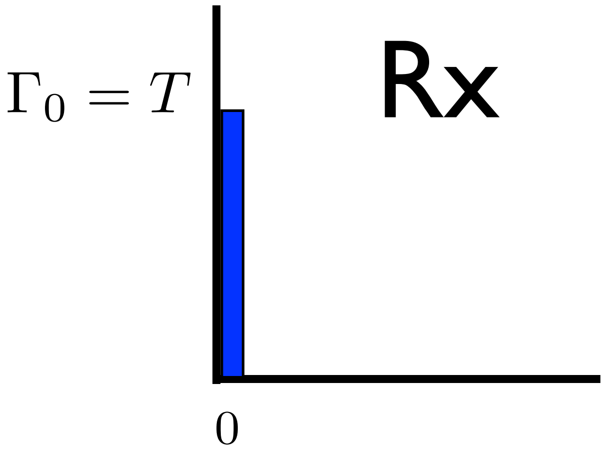


Idea

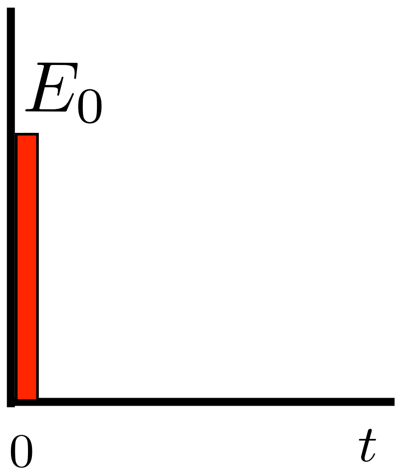


Idea

Rx



Tx



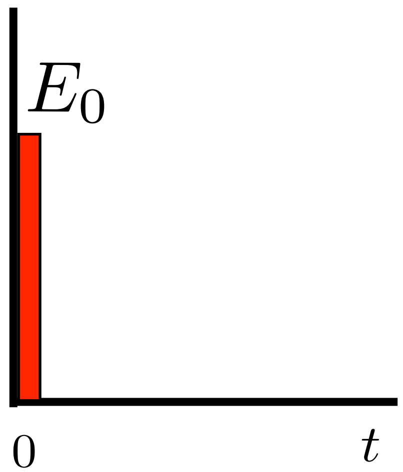
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$

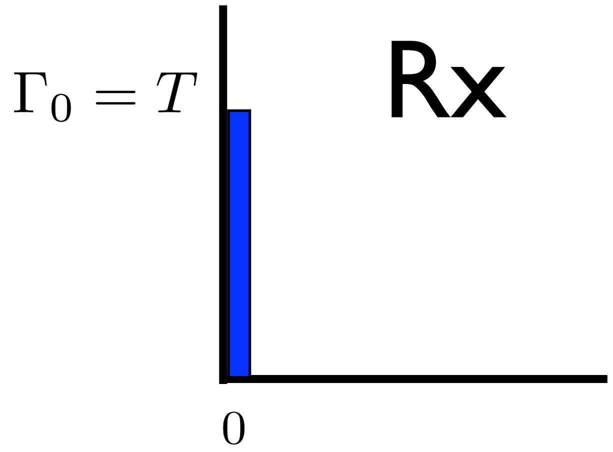
Idea

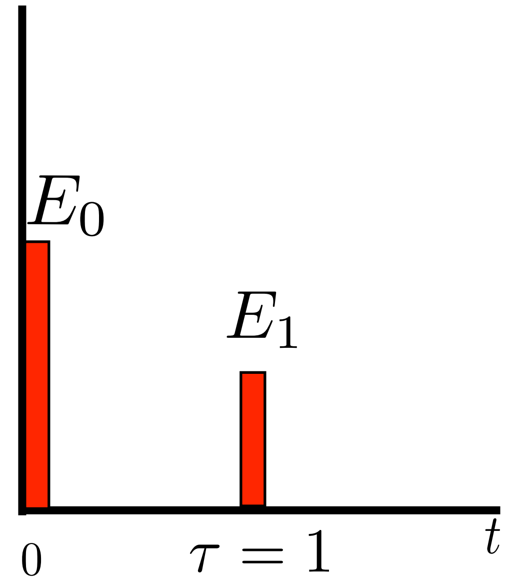
Rx

Tx



σ_1



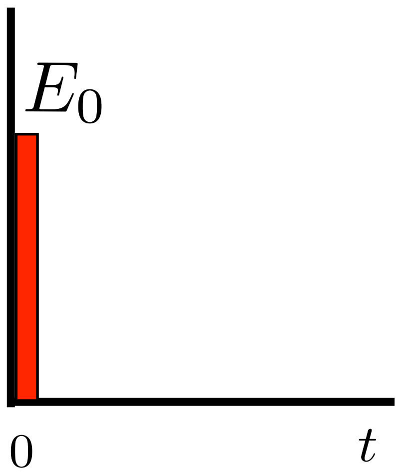


σ_2

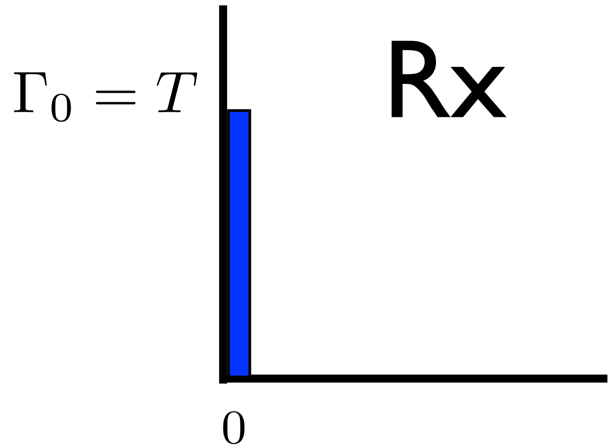
$$B = Tg\left(\frac{E_0}{T}\right)$$

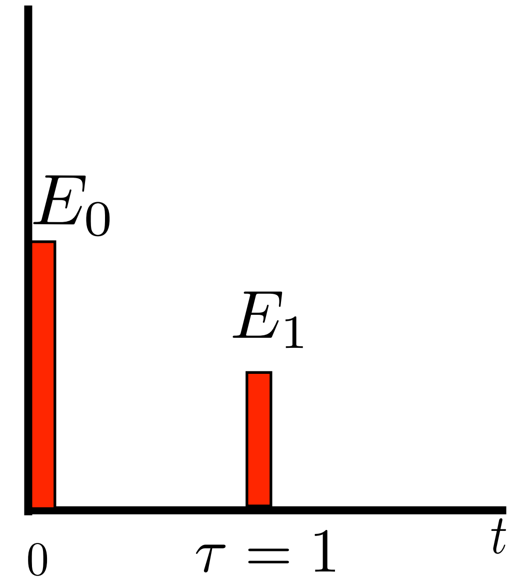
Idea

Tx



Rx





σ_1

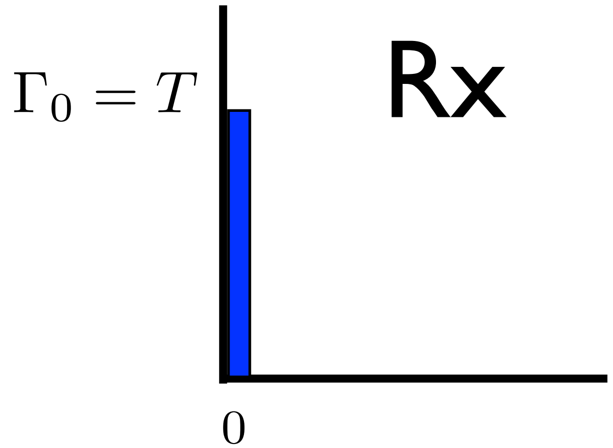
$$B = Tg\left(\frac{E_0}{T}\right)$$

σ_2

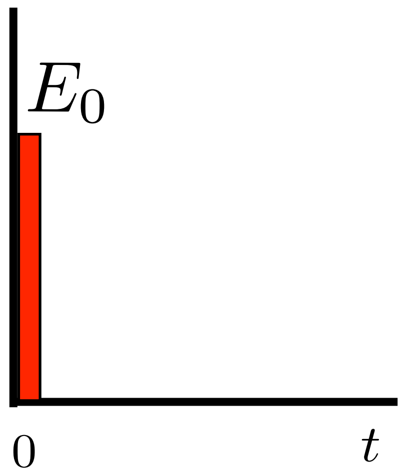
$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$

Idea

Rx

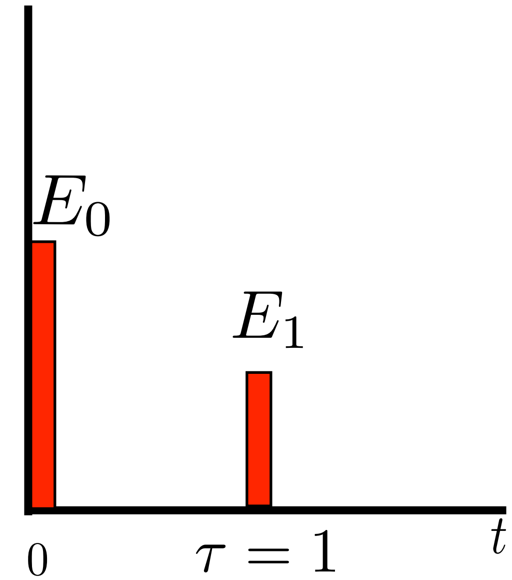


Tx



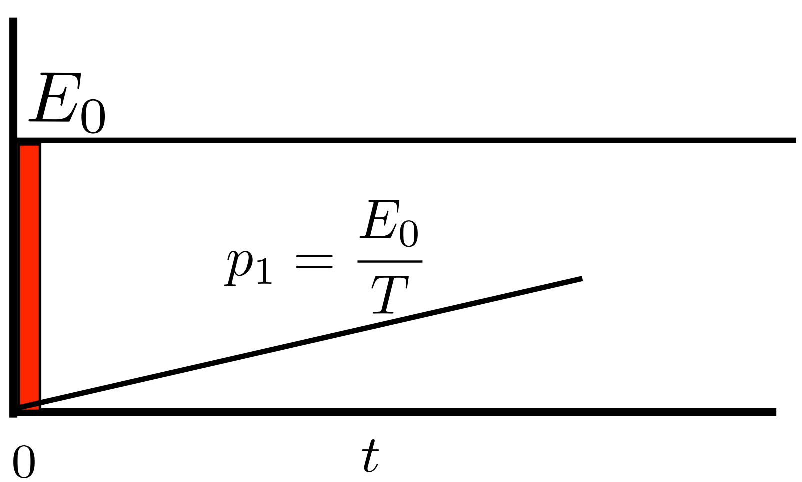
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$



σ_2

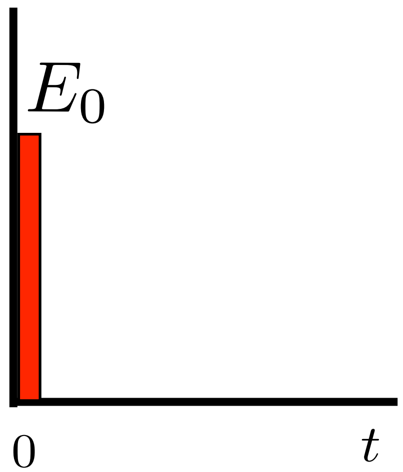
$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$

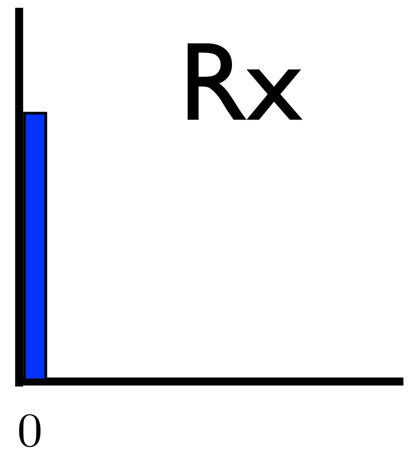


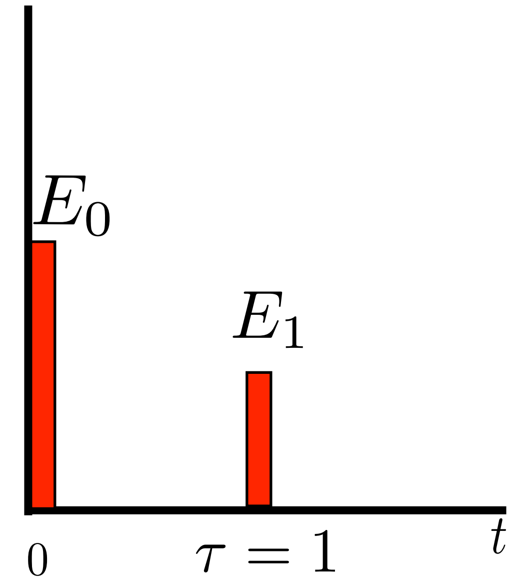
online = offline

Idea

Tx





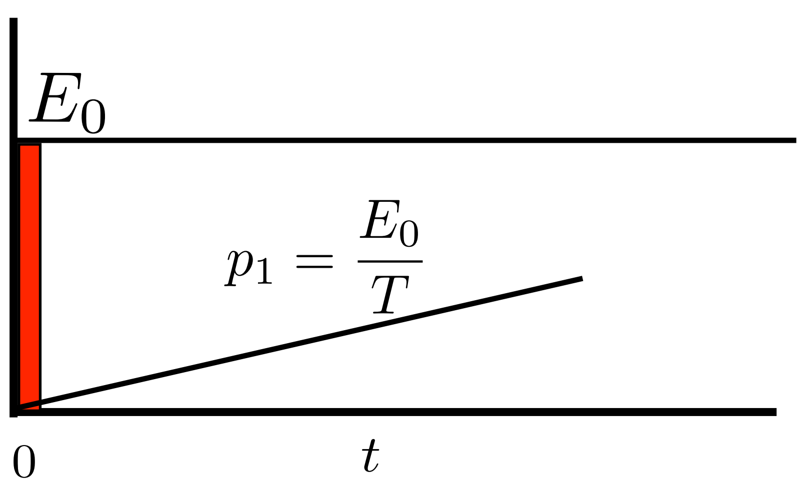


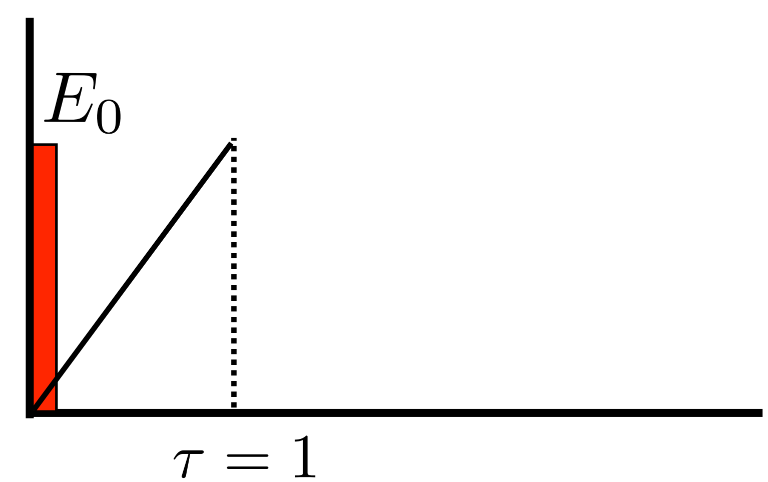
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$

σ_2

$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$

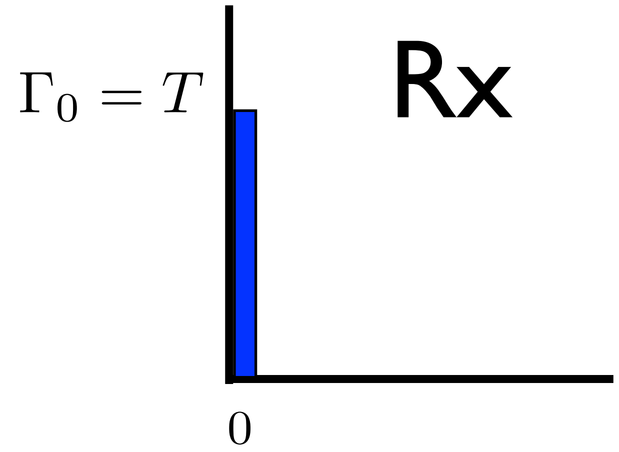




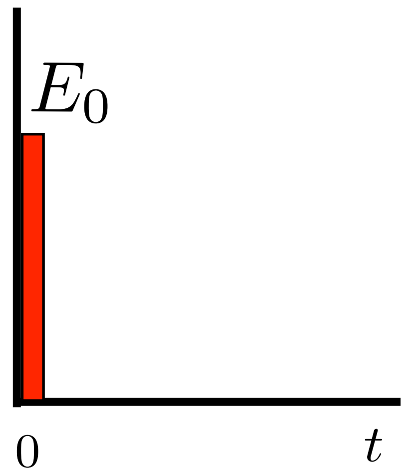
online = offline

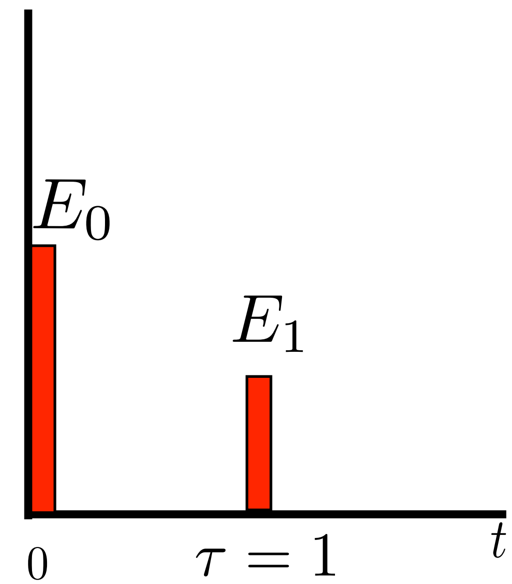
Idea

Rx



Tx



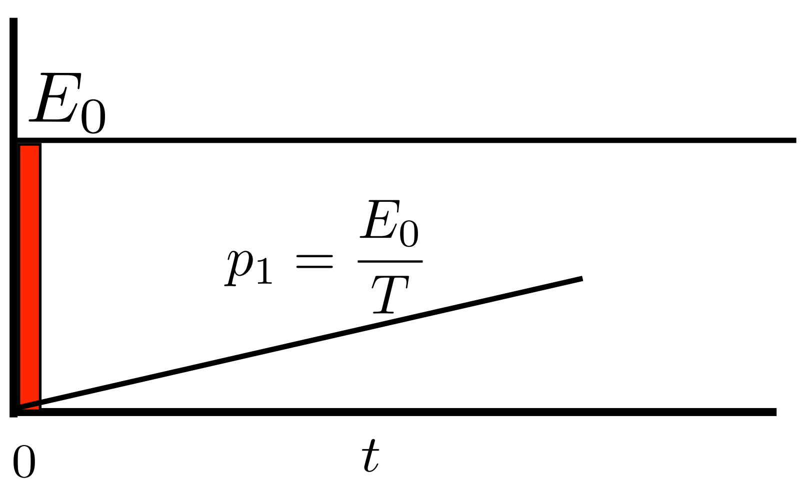


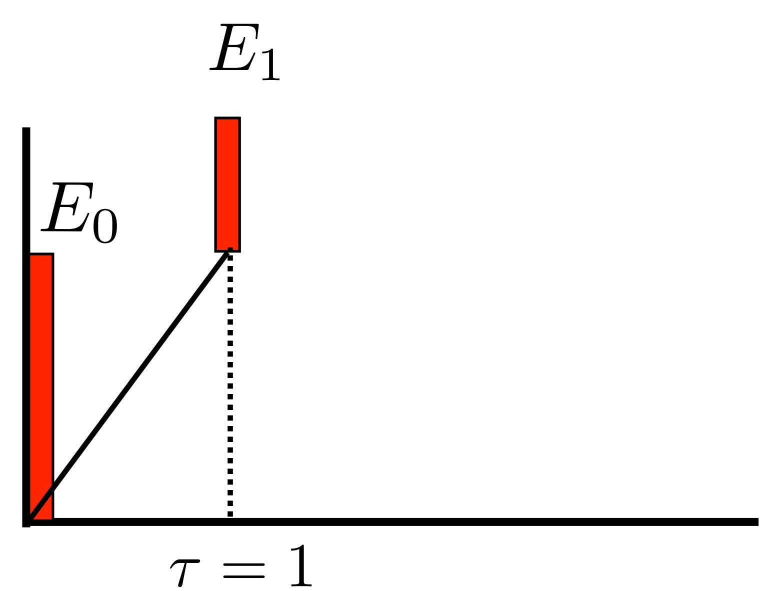
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$

σ_2

$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$



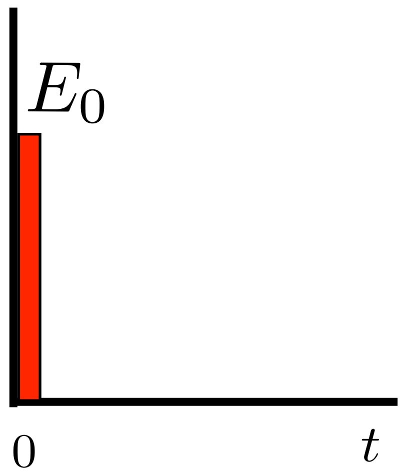


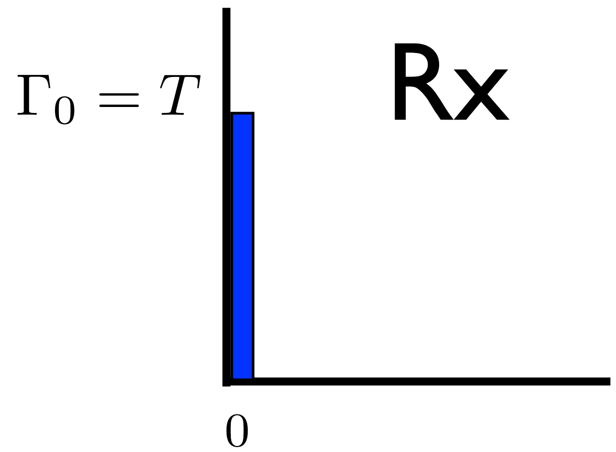
online = offline

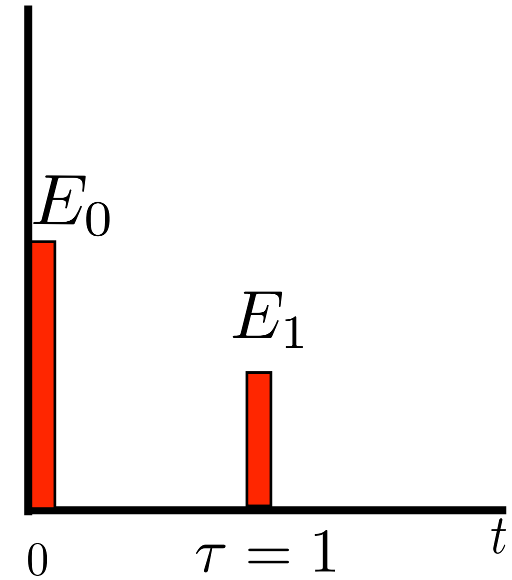
Idea

Tx

Rx





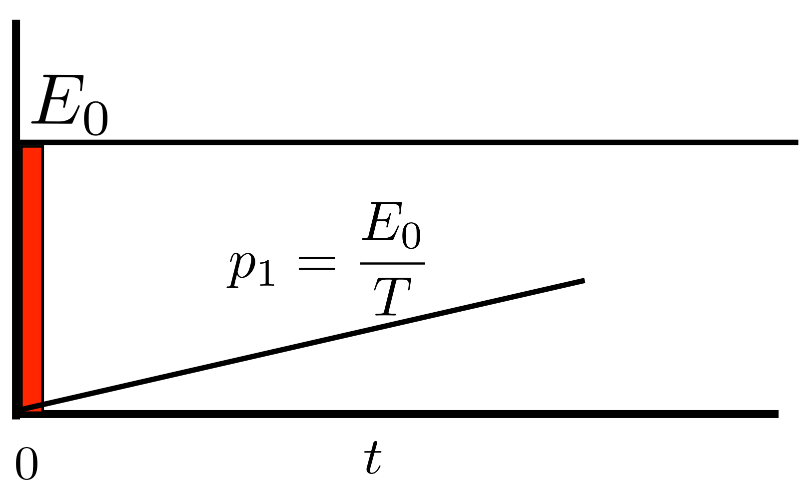


σ_1

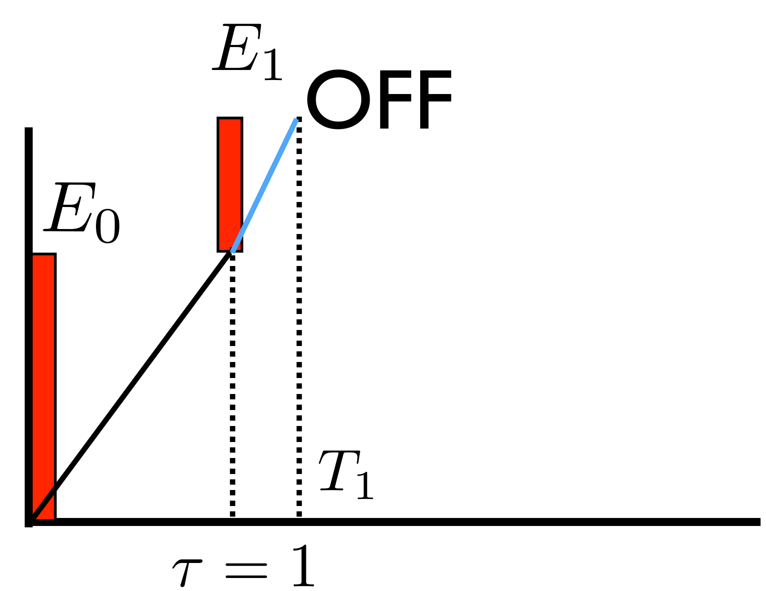
$$B = Tg\left(\frac{E_0}{T}\right)$$

σ_2

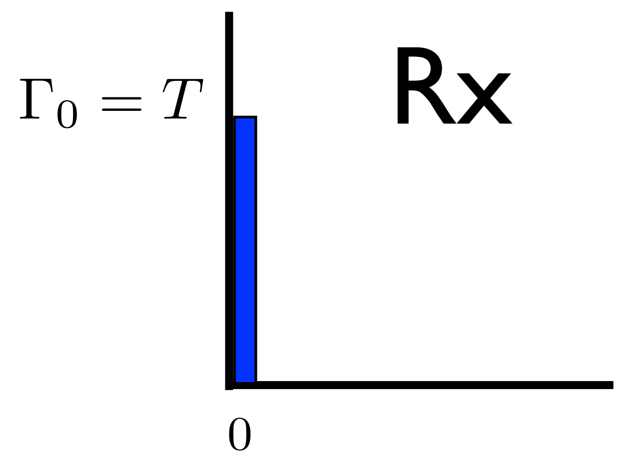
$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$



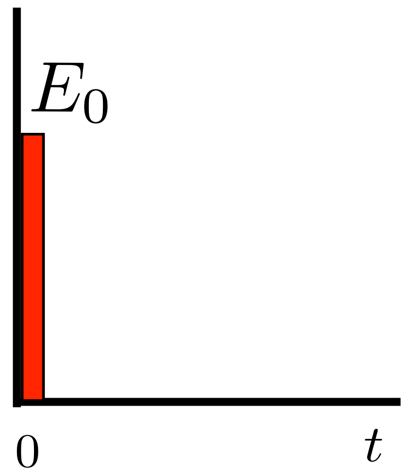
online = offline



Idea

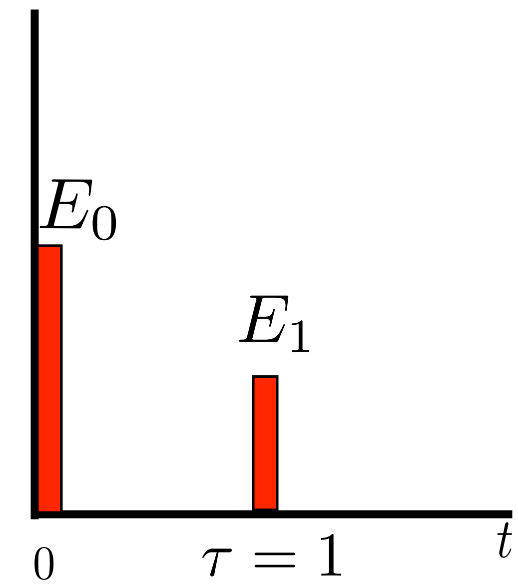


Tx



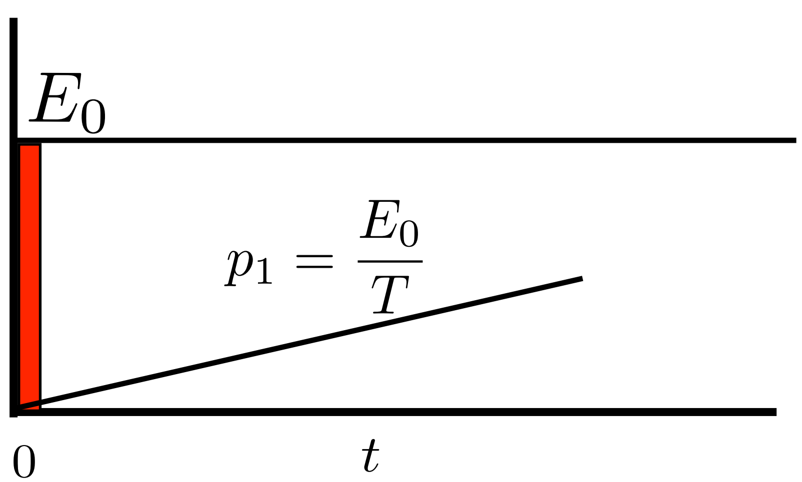
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$

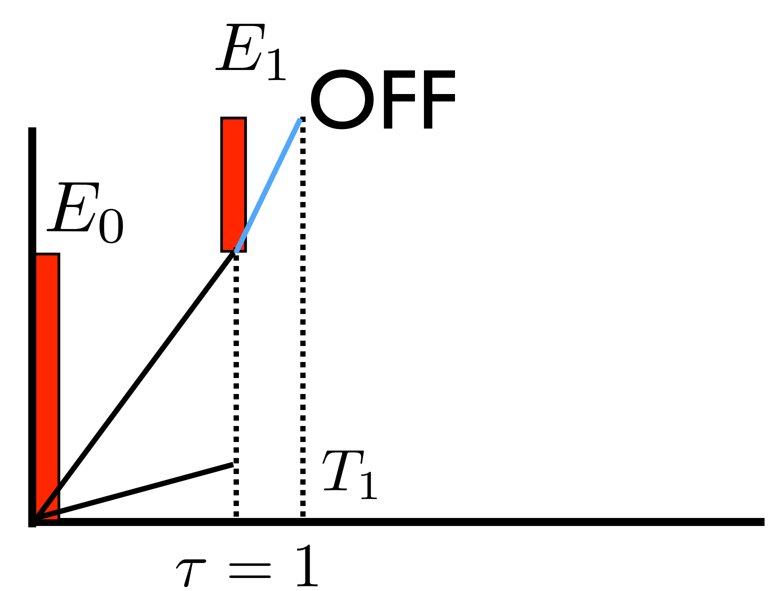


σ_2

$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$

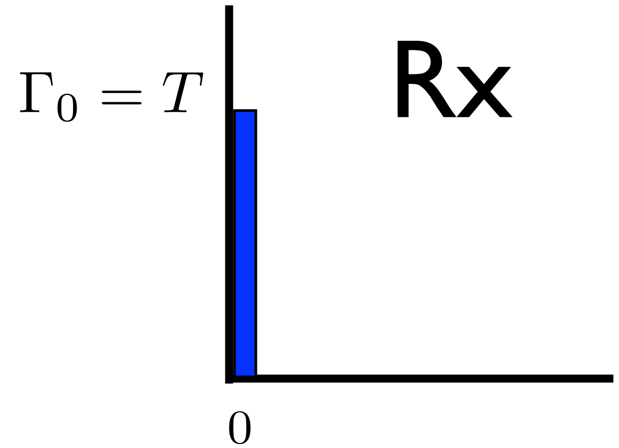


online = offline

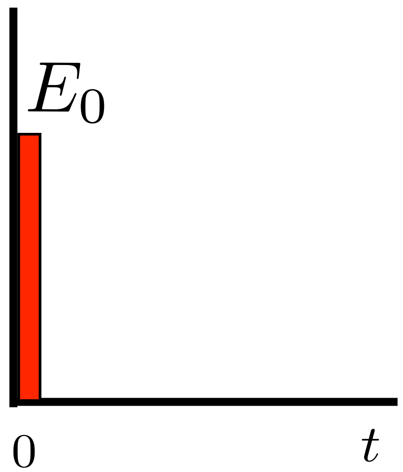


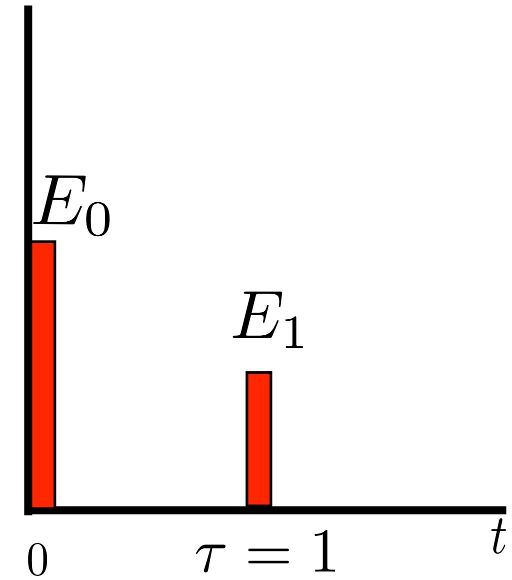
Idea

Rx



Tx



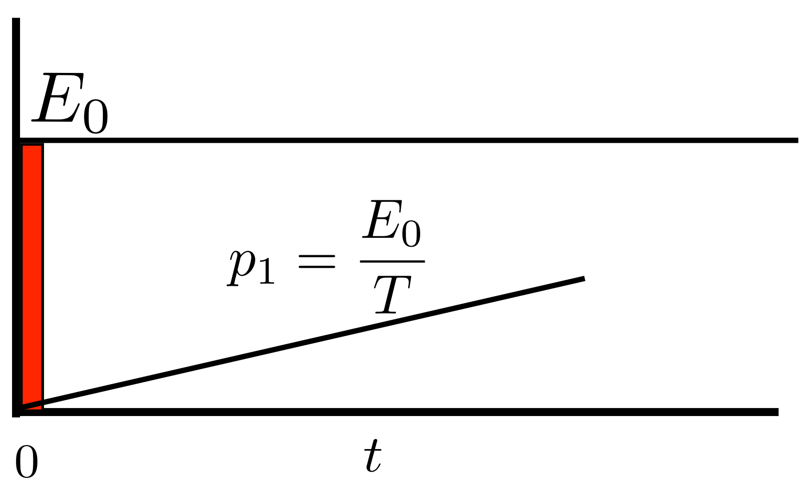


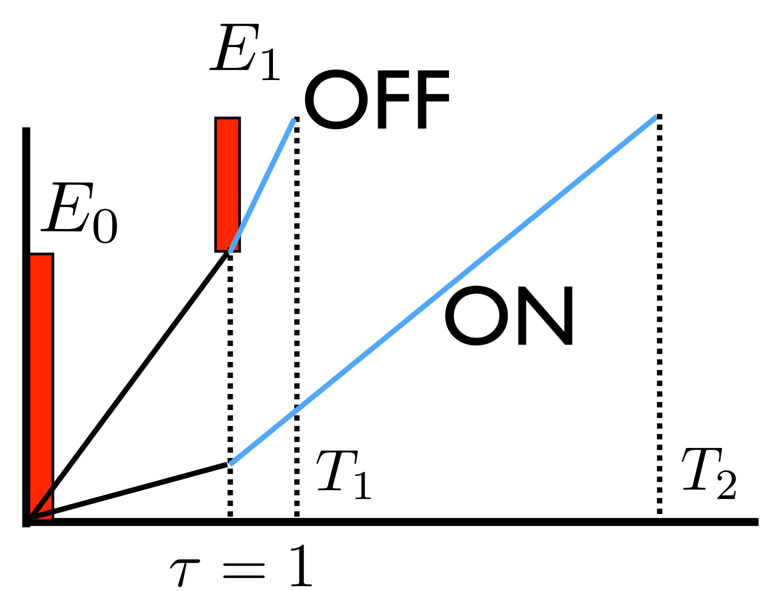
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$

σ_2

$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$

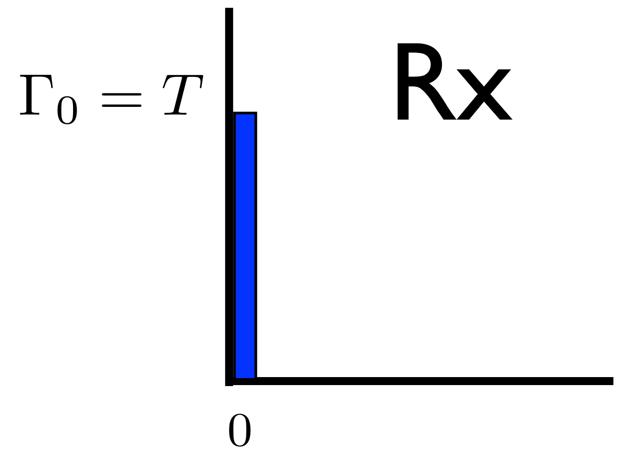




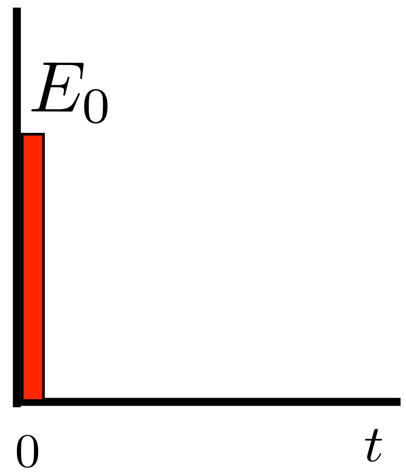
online = offline

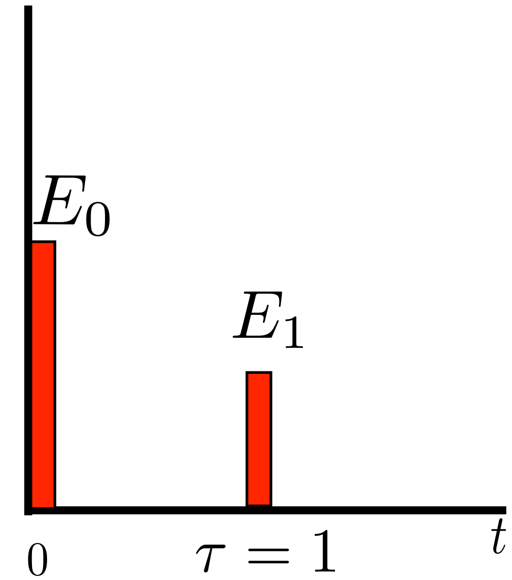
Idea

Rx



Tx



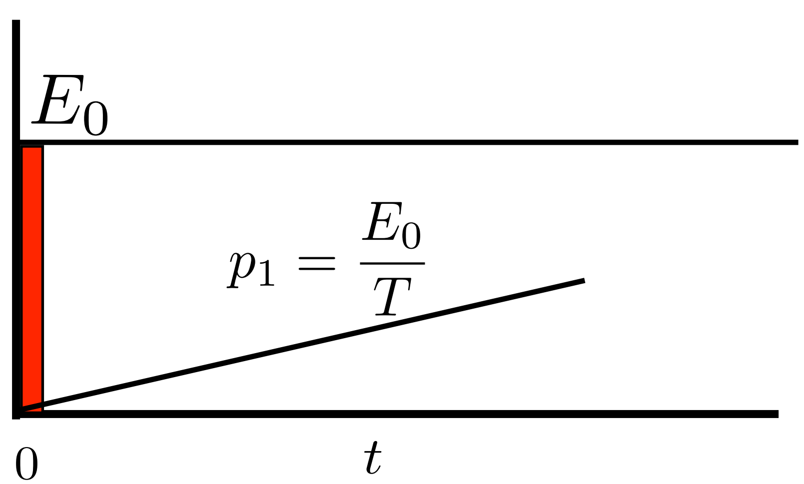


σ_1

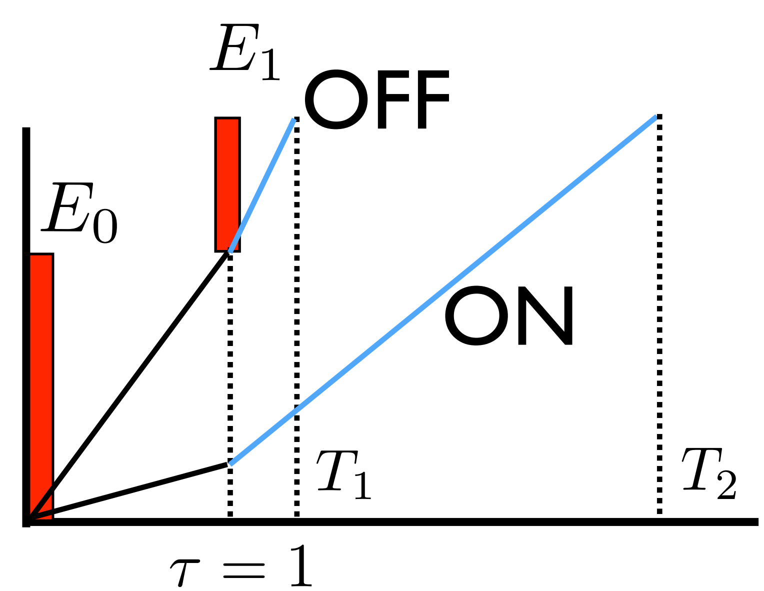
$$B = Tg\left(\frac{E_0}{T}\right)$$

σ_2

$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$



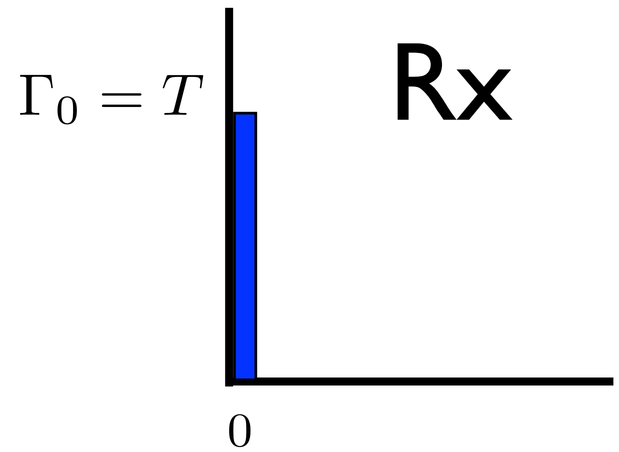
online = offline



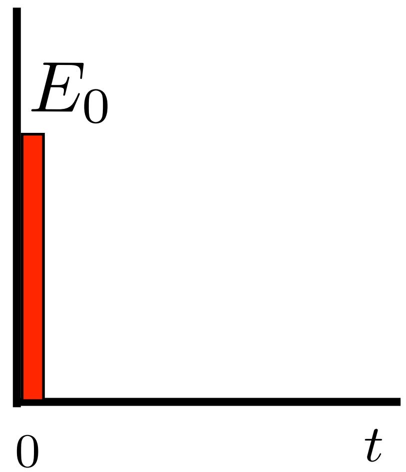
online (T_2), offline(T_1)

Idea

Rx

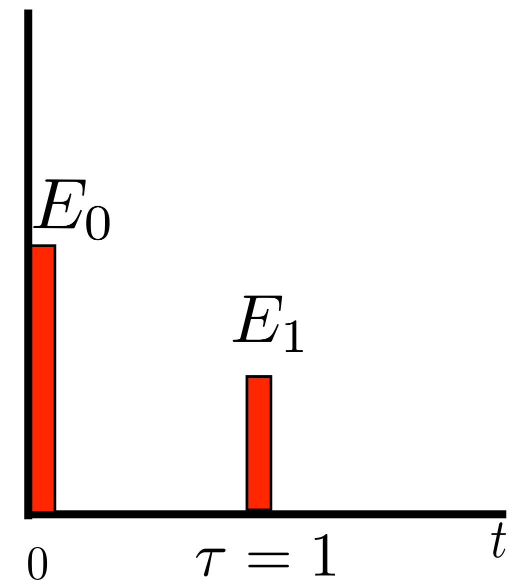


Tx



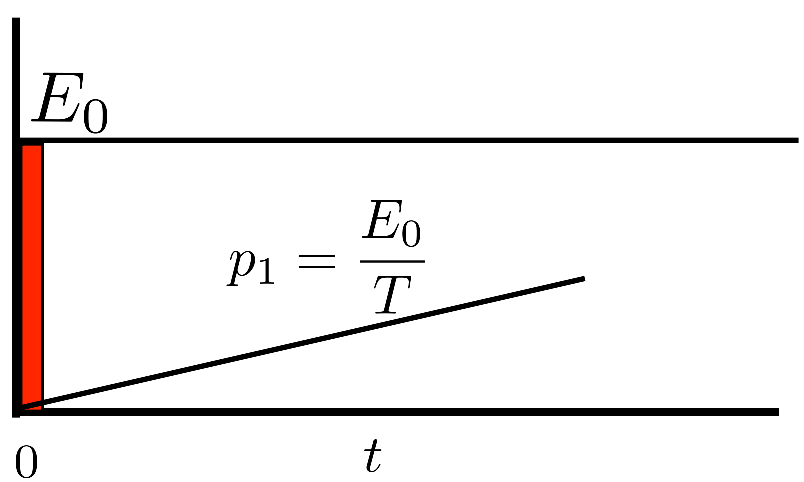
σ_1

$$B = Tg\left(\frac{E_0}{T}\right)$$

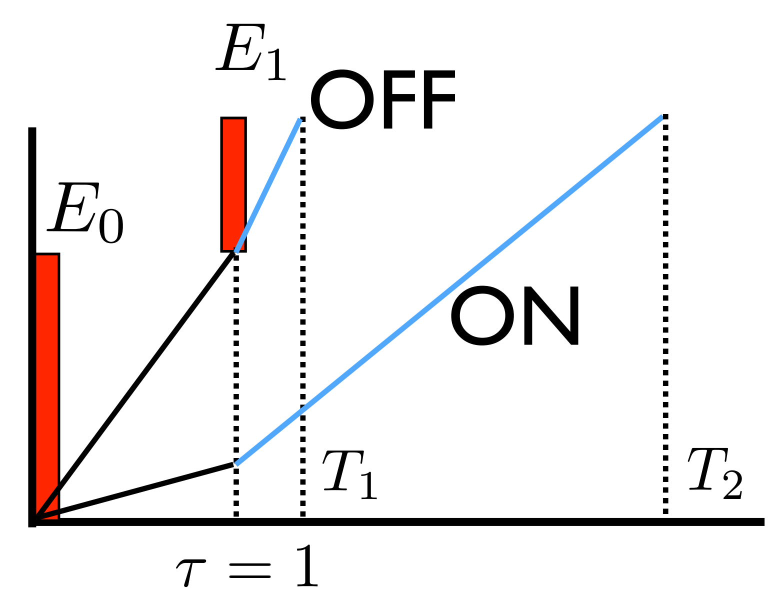


σ_2

$$B = \tau g\left(\frac{E_0 + E_1}{\tau}\right)$$



$$r \geq \frac{T_2}{T_1}$$

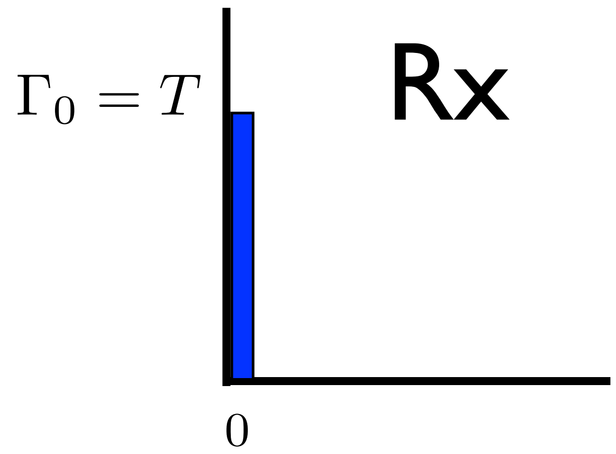


online = offline

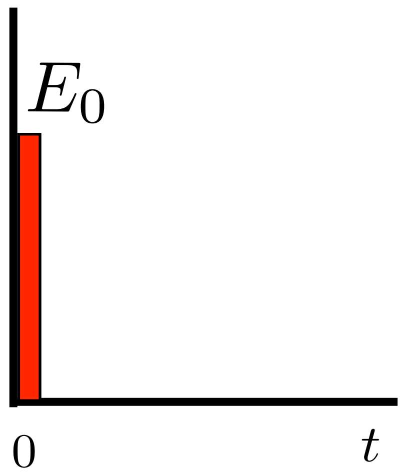
online (T_2), offline(T_1)

Proposed Online is best for these seq.

Rx

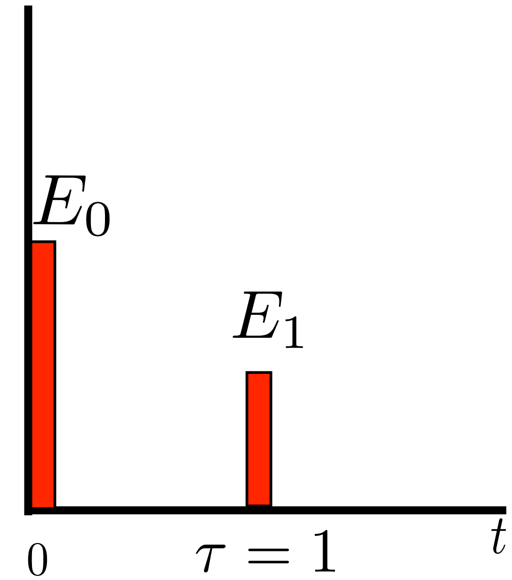


Tx



σ_1

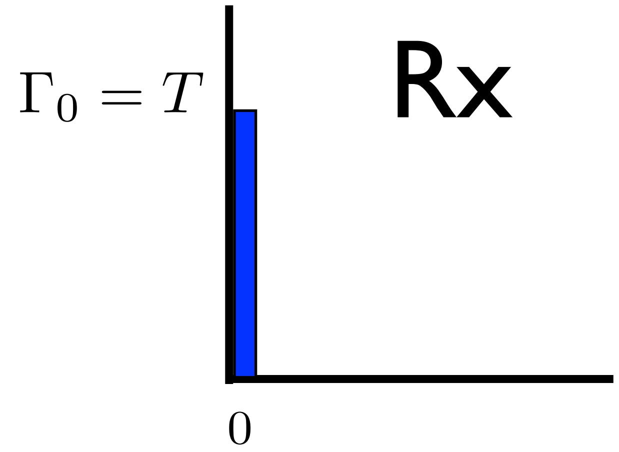
$$B = Tg\left(\frac{E_0}{T}\right)$$



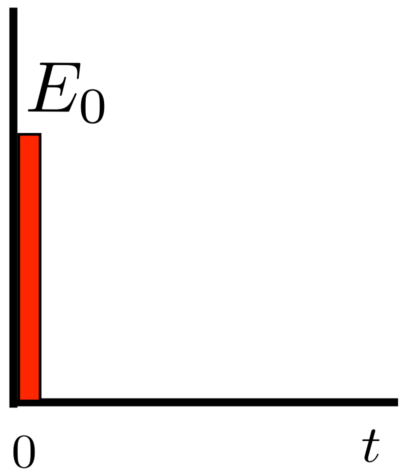
σ_2

Proposed Online is best for these seq.

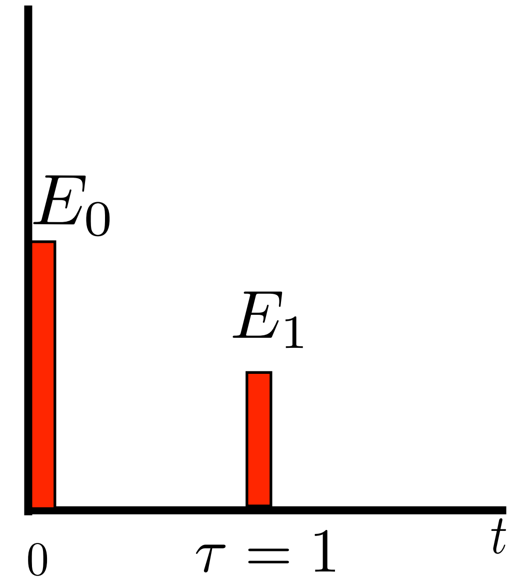
Rx



Tx



σ_1



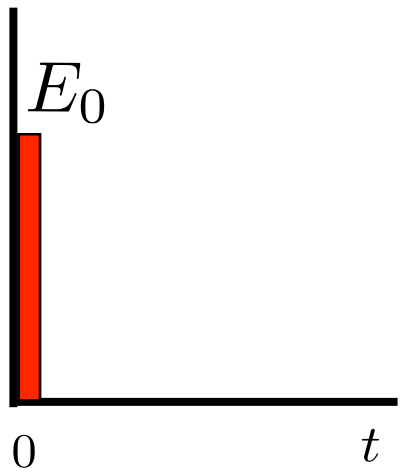
σ_2

$$B = Tg\left(\frac{E_0}{T}\right)$$

Look at any other online algorithm

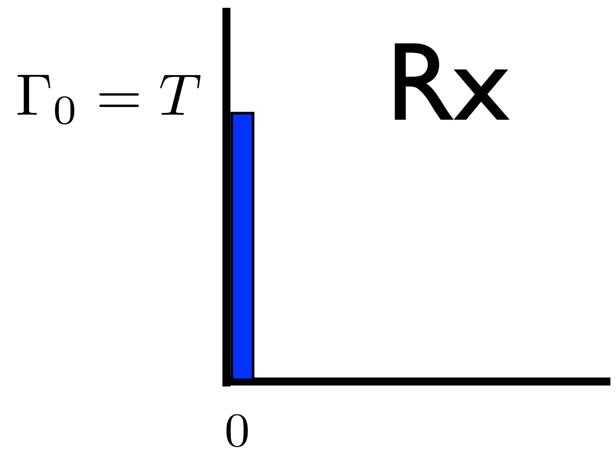
Proposed Online is best for these seq.

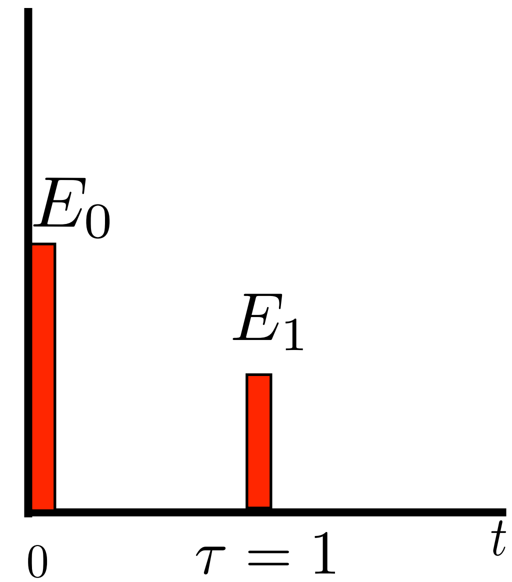
Tx



σ_1

Rx





σ_2

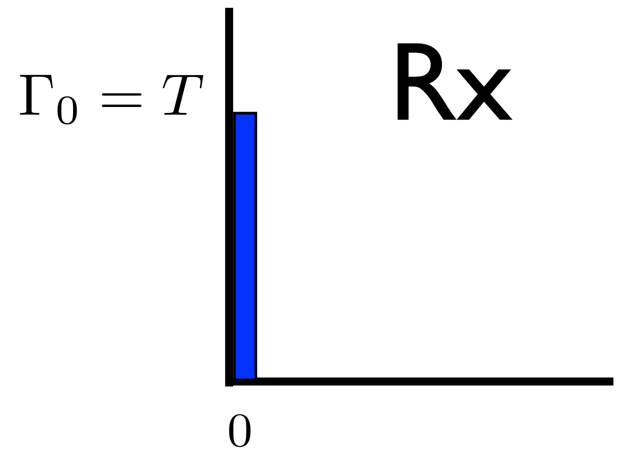
$$B = Tg\left(\frac{E_0}{T}\right)$$

Look at any other online algorithm

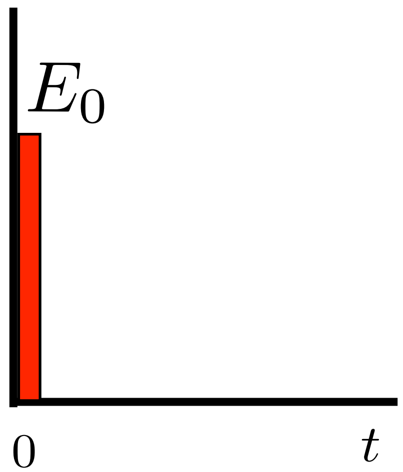
Until $\tau = 1$ let it use α fraction of E_0

Proposed Online is best for these seq.

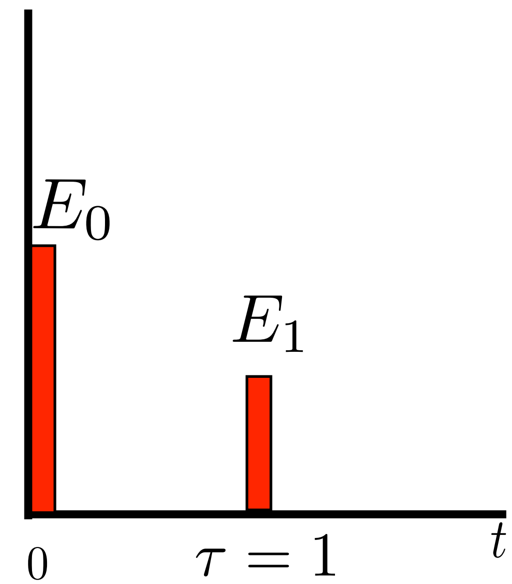
Rx



Tx



σ_1

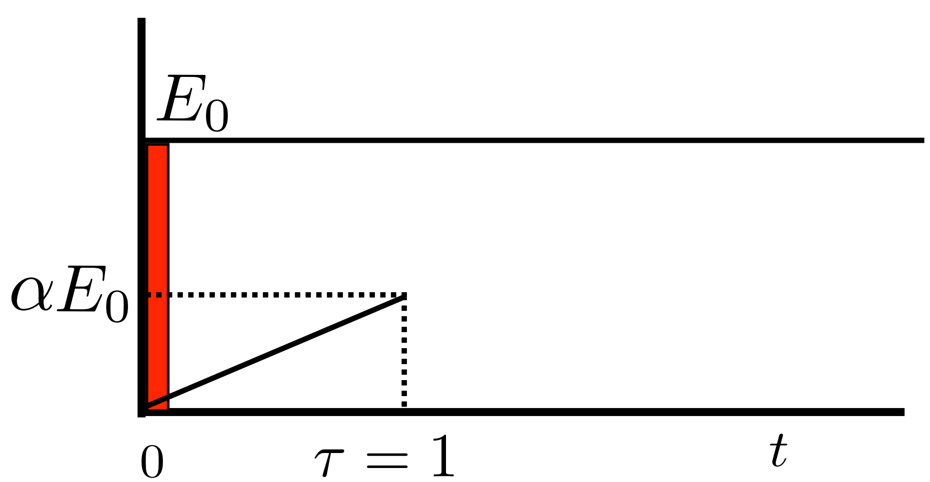


σ_2

$$\boxed{B = Tg\left(\frac{E_0}{T}\right)}$$

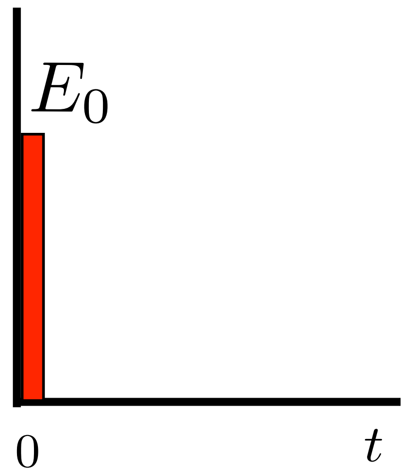
Look at any other online algorithm

Until $\tau = 1$ let it use α fraction of E_0

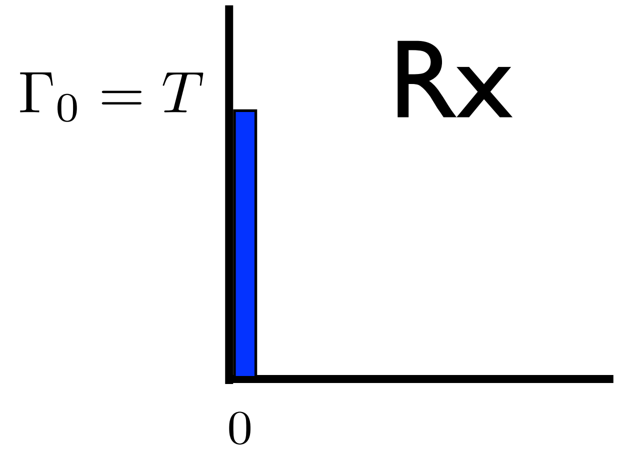


Proposed Online is best for these seq.

Tx

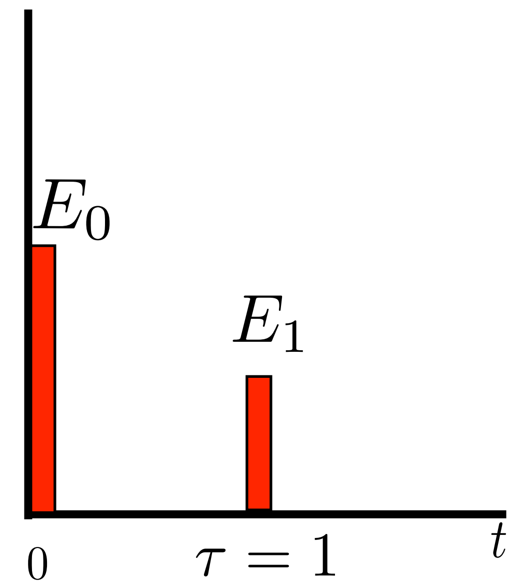


σ_1



Rx

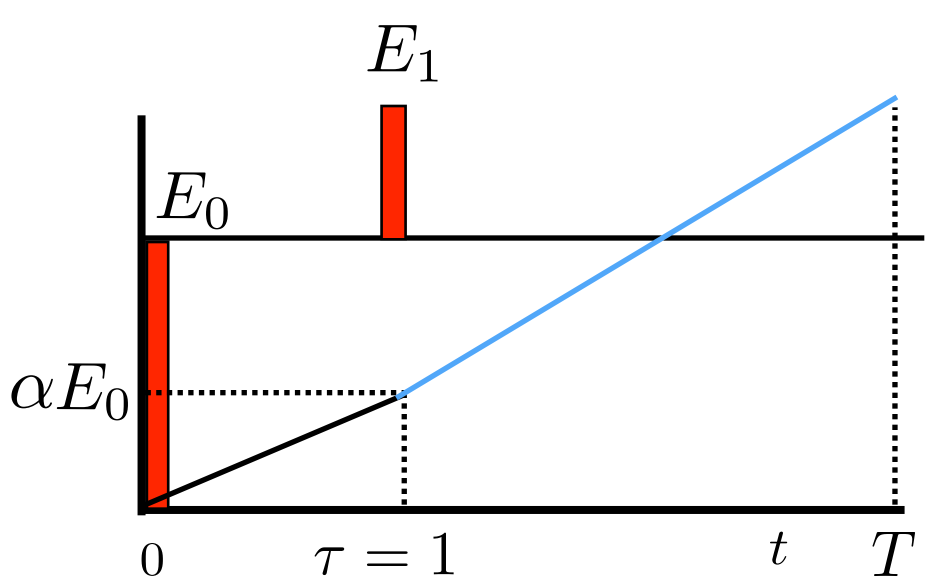
$$B = Tg\left(\frac{E_0}{T}\right)$$



σ_2

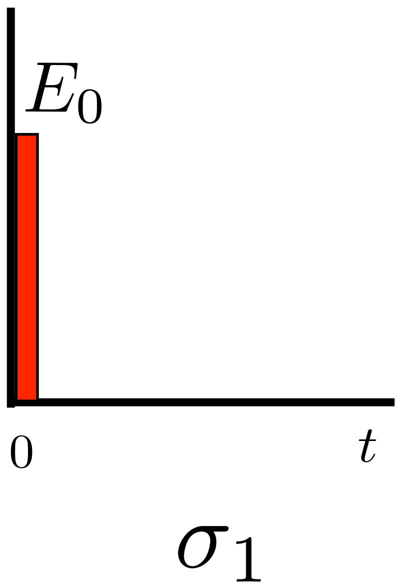
Look at any other online algorithm

Until $\tau = 1$ let it use α fraction of E_0

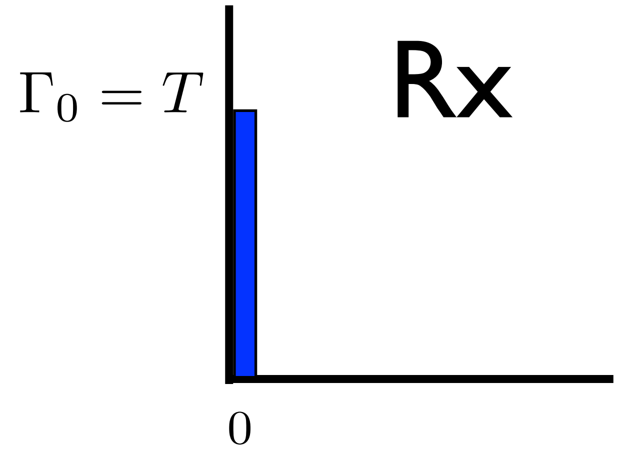


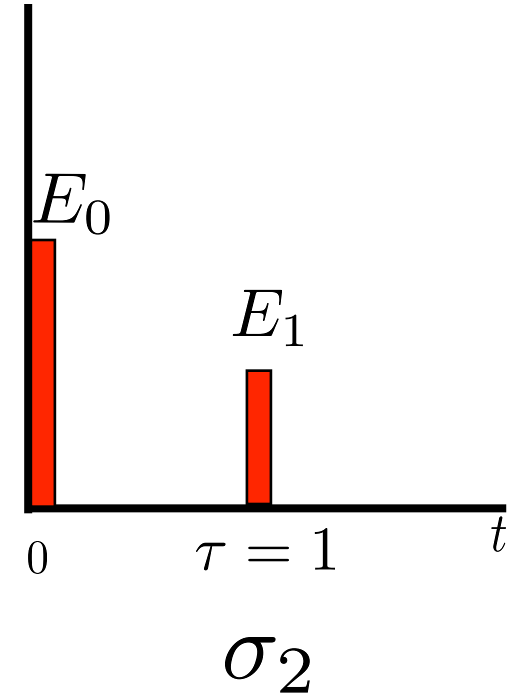
Proposed Online is best for these seq.

Tx



Rx

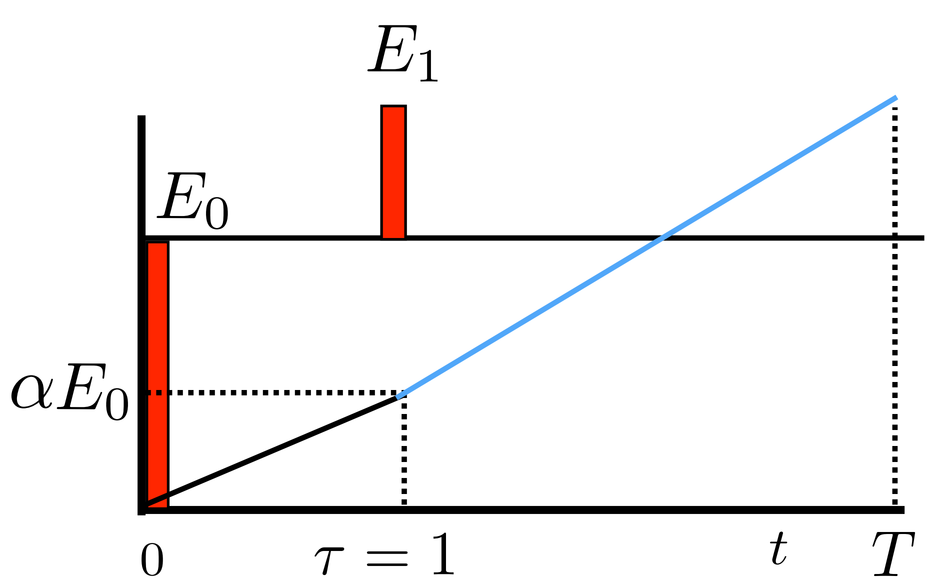




$$B = Tg\left(\frac{E_0}{T}\right)$$

Look at any other online algorithm

Until $\tau = 1$ let it use α fraction of E_0

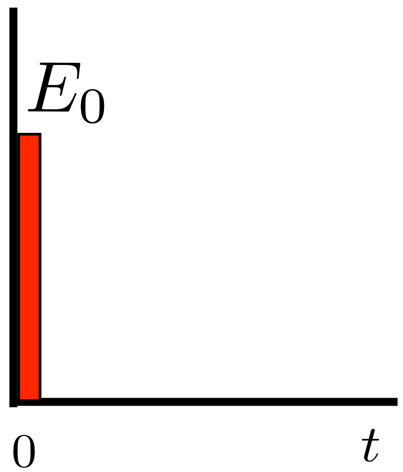


Max bits sent within time T (Rx constraint)

$$B_\alpha = \tau g\left(\frac{\alpha E_0}{\tau}\right) + (T-\tau) g\left(\frac{(1-\alpha)E_0 + E_1}{T-\tau}\right)$$

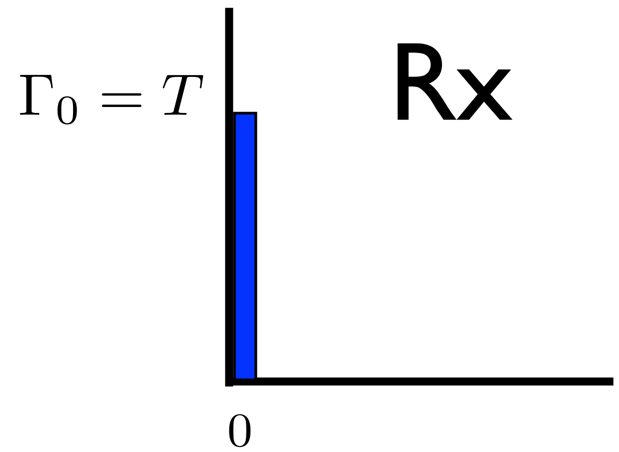
Proposed Online is best for these seq.

Tx

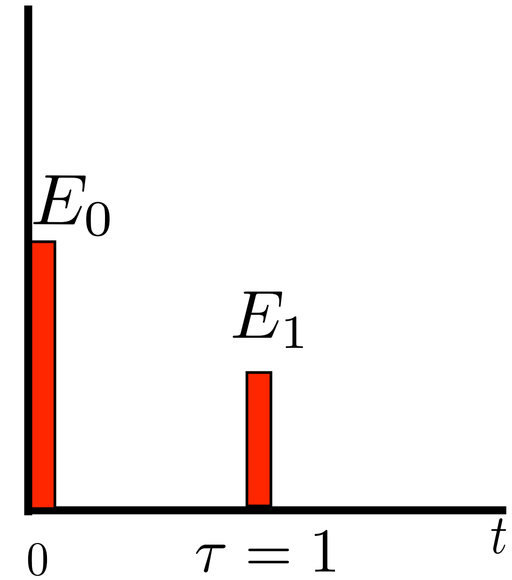


σ_1

Rx



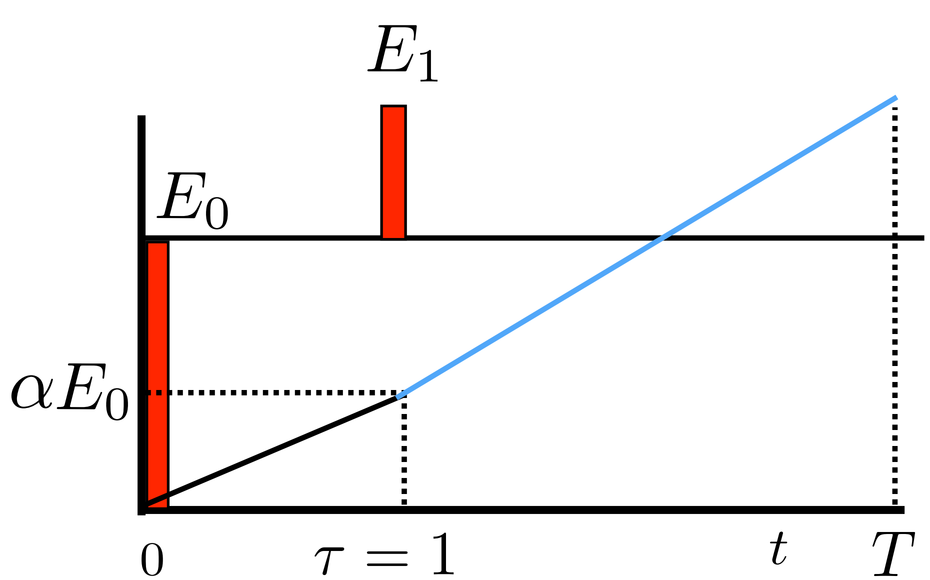
$$B = Tg\left(\frac{E_0}{T}\right)$$



σ_2

Look at any other online algorithm

Until $\tau = 1$ let it use α fraction of E_0



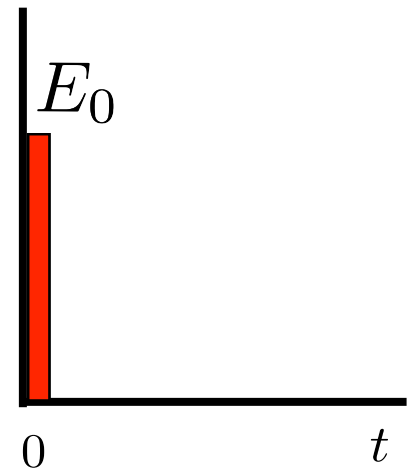
Max bits sent within time T (Rx constraint)

$$B_\alpha = \tau g\left(\frac{\alpha E_0}{\tau}\right) + (T-\tau) g\left(\frac{(1-\alpha)E_0 + E_1}{T-\tau}\right)$$

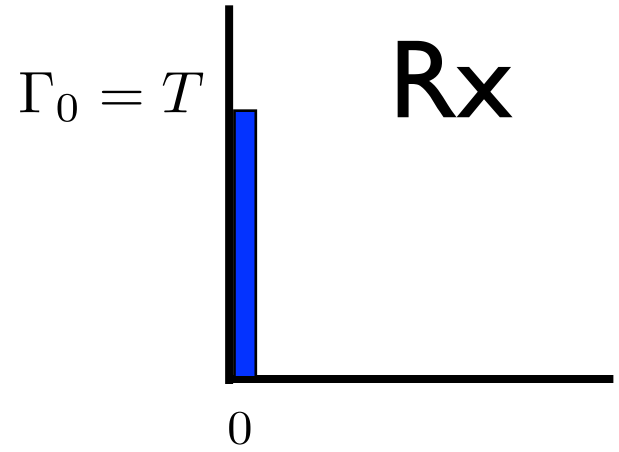
concavity $\quad B_\alpha \leq B$

Proposed Online is best for these seq.

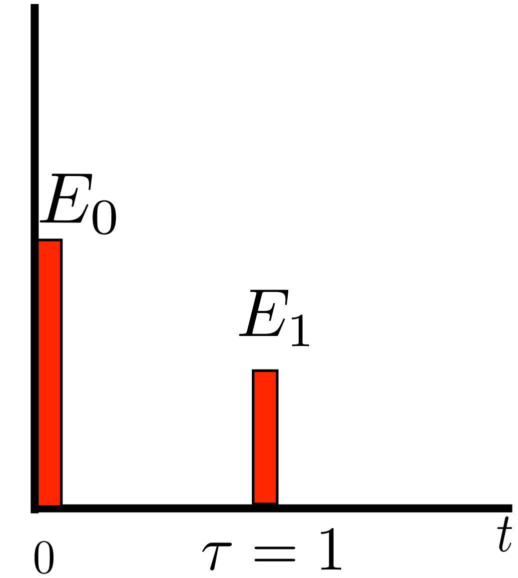
Tx



σ_1



Rx

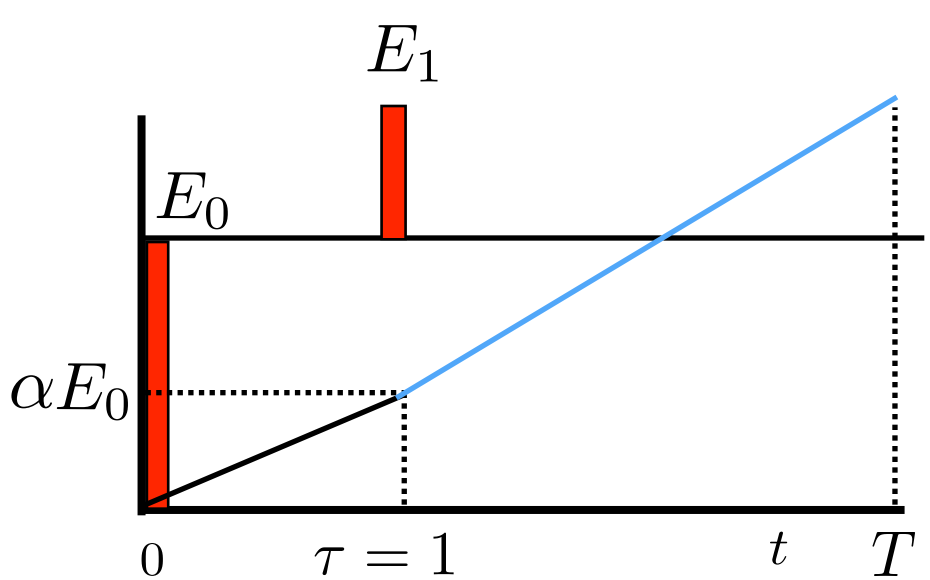


σ_2

$$\boxed{B = Tg\left(\frac{E_0}{T}\right)}$$

Look at any other online algorithm

Until $\tau = 1$ let it use α fraction of E_0



Max bits sent within time T (Rx constraint)

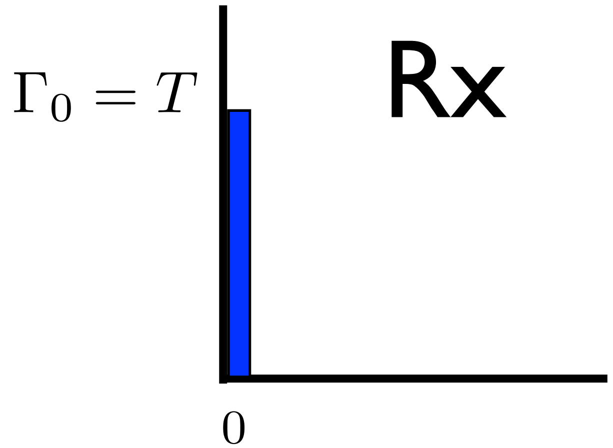
$$B_\alpha = \tau g\left(\frac{\alpha E_0}{\tau}\right) + (T-\tau)g\left(\frac{(1-\alpha)E_0 + E_1}{T-\tau}\right)$$

concavity $\quad B_\alpha \leq B$

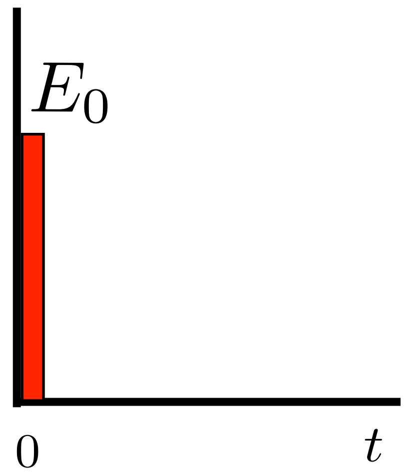
Equality if $\alpha = 1 - 1/T$ Prop. Online

Idea

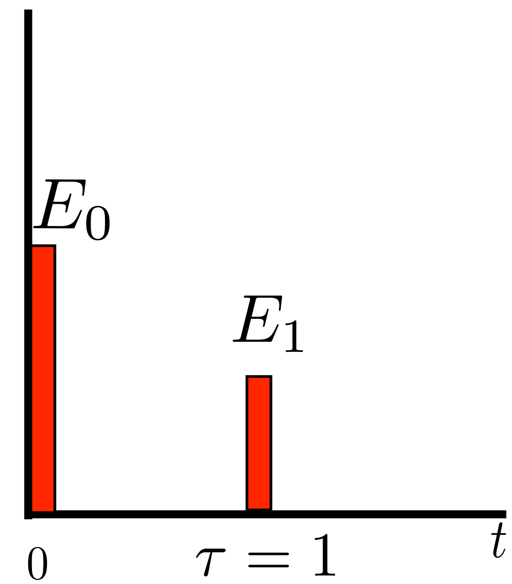
Rx



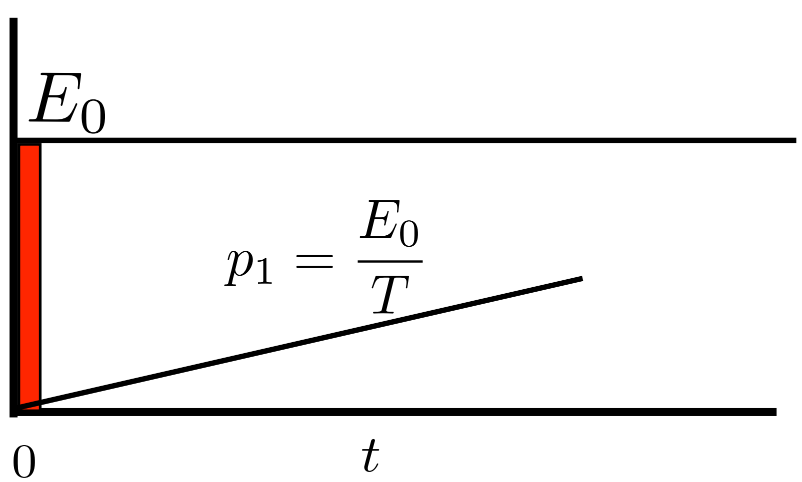
Tx



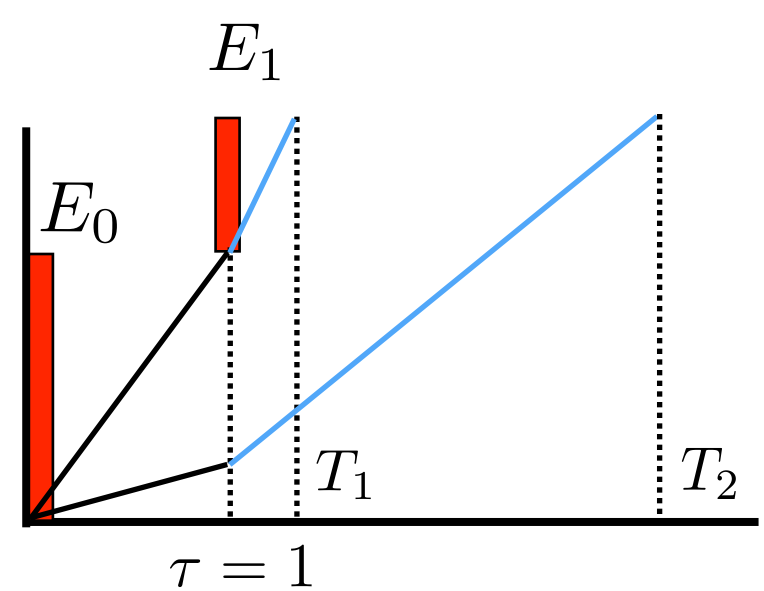
σ_1



σ_2



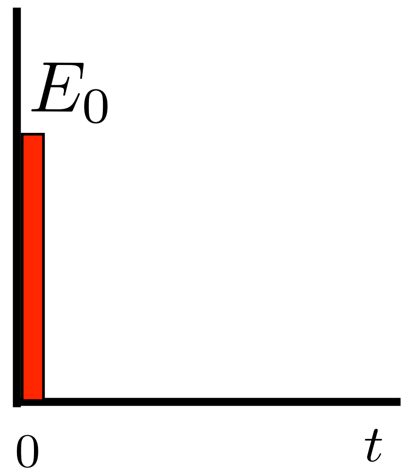
online = offline



online (T_2), offline(T_1)

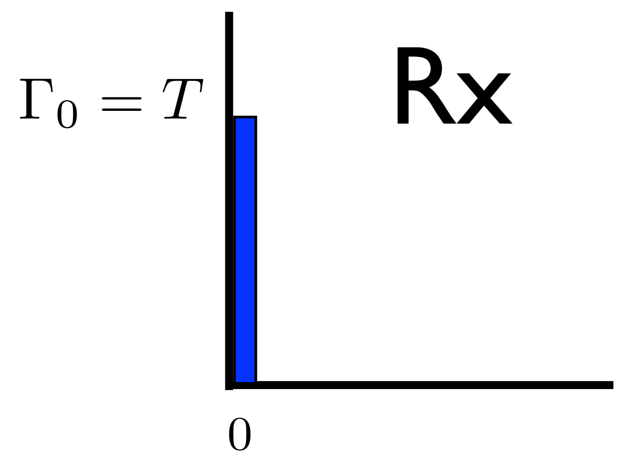
Idea

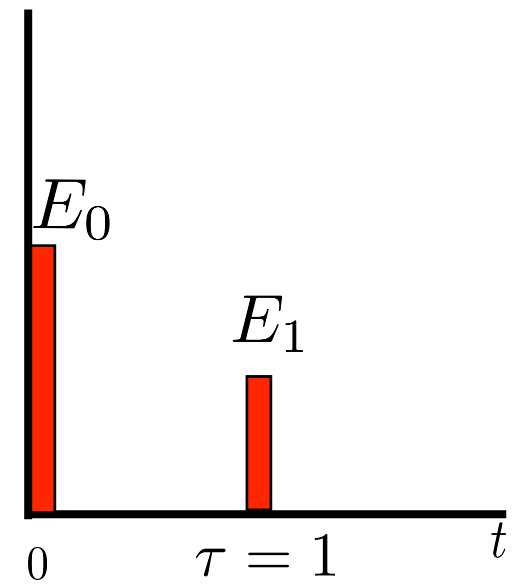
Tx



σ_1

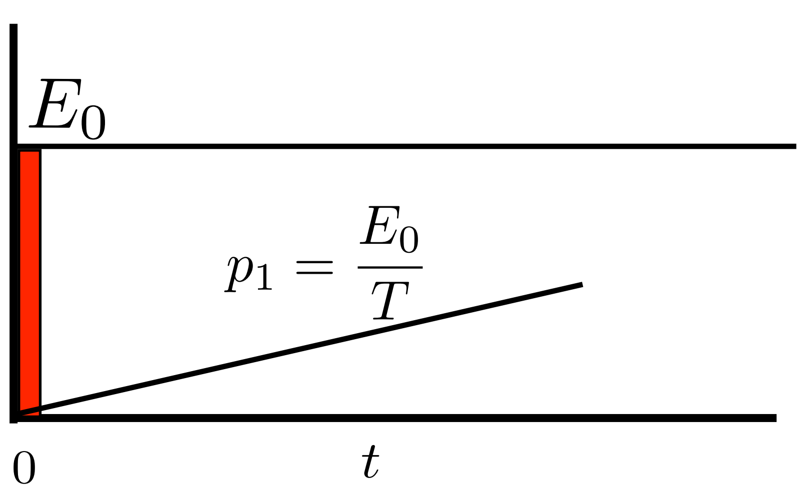
Rx



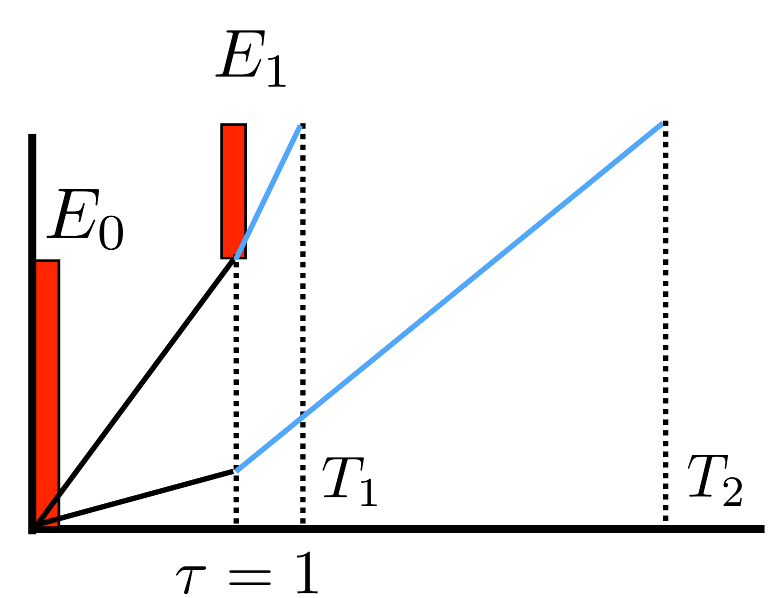


σ_2

For these seq. Prop. Online is OPT



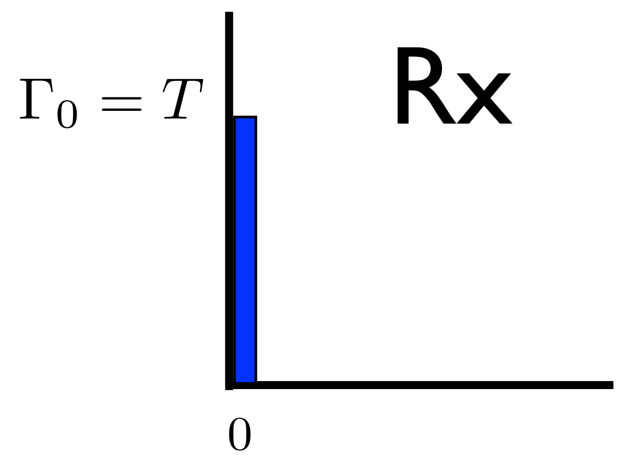
online = offline



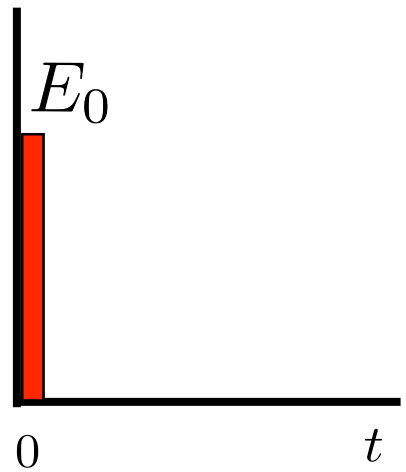
online (T_2), offline(T_1)

Idea

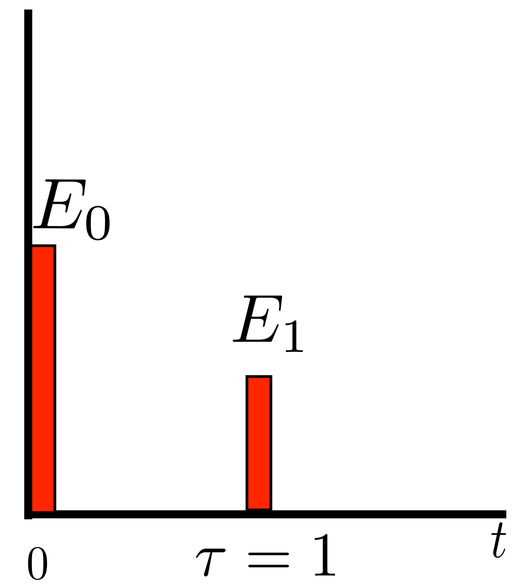
Rx



Tx



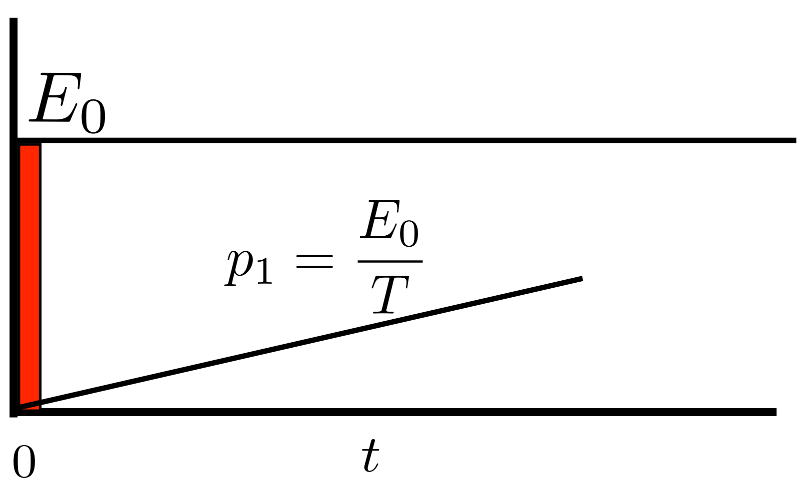
σ_1



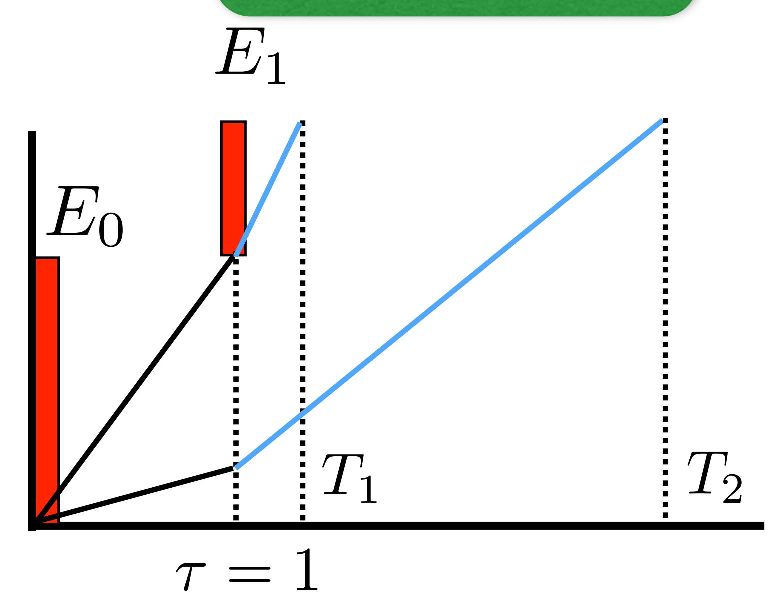
σ_2

For these seq. Prop. Online is OPT

r $\geq \frac{T_2}{T_1}$



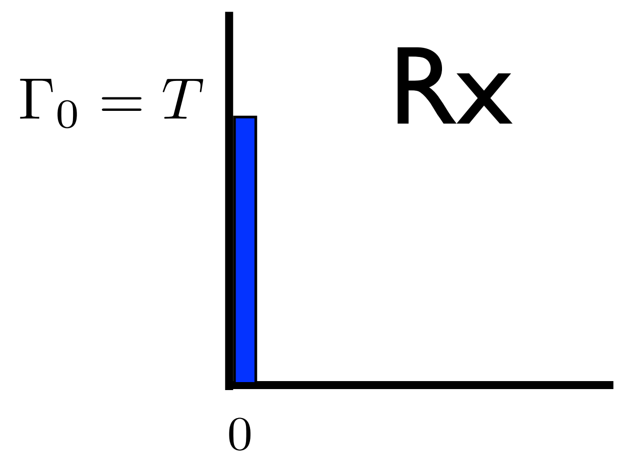
online = offline



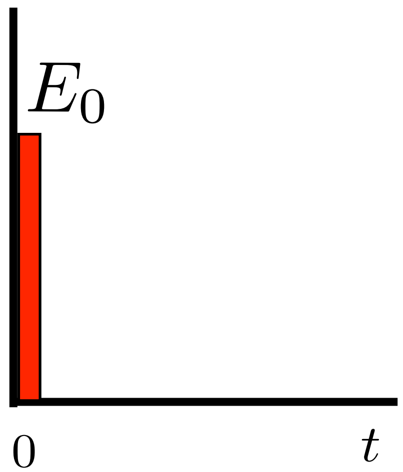
online (T_2), offline(T_1)

Idea

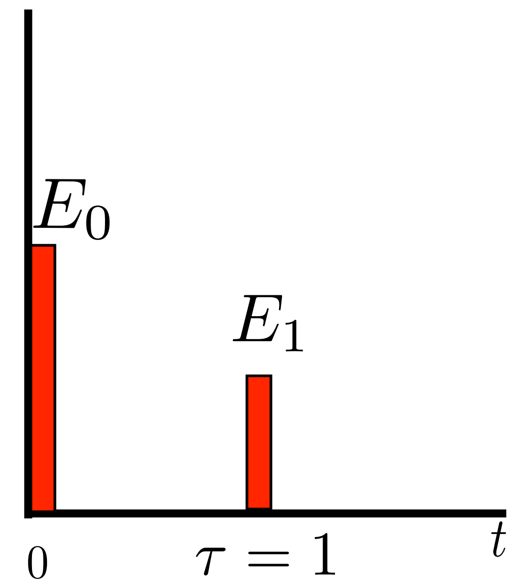
Rx



Tx



σ_1

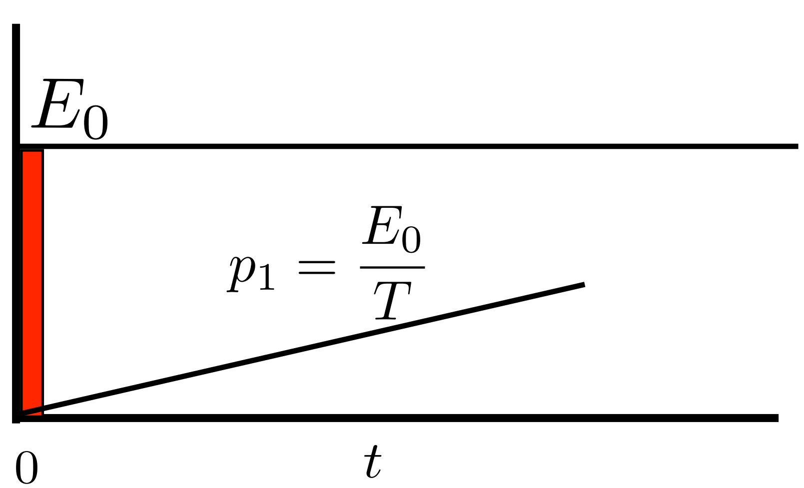


σ_2

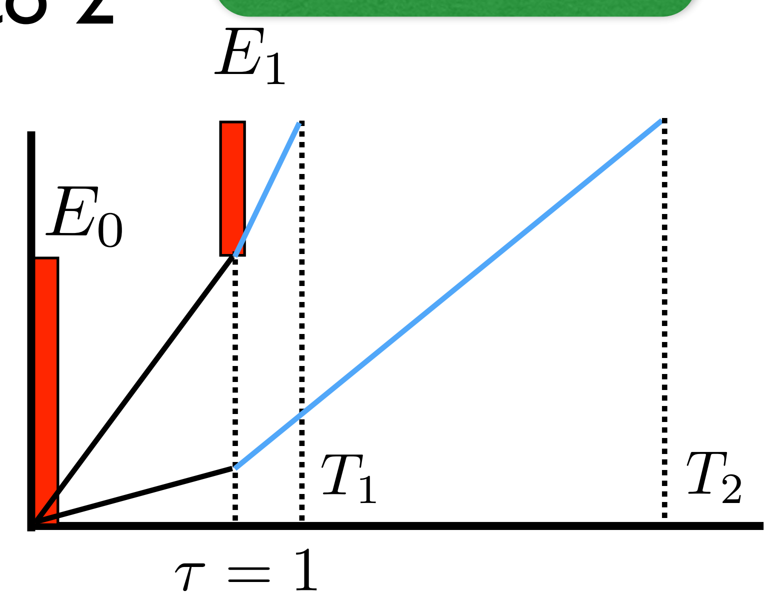
For these seq. Prop. Online is OPT

Choose parameters to make this ratio as close to 2

$$r \geq \frac{T_2}{T_1}$$



online = offline



online (T_2), offline(T_1)

Finite Battery Online Algorithm

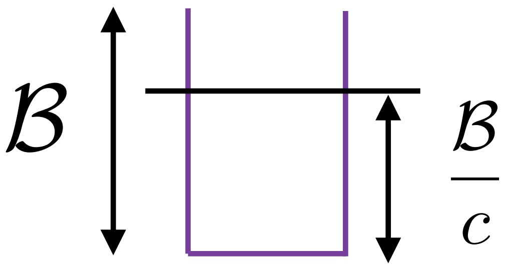
Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.

Finite Battery Online Algorithm

Alg: Accumulate&Dump

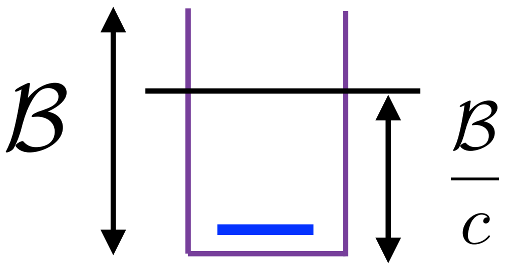
Wait for acc. energy to cross a fixed fraction of battery cap.

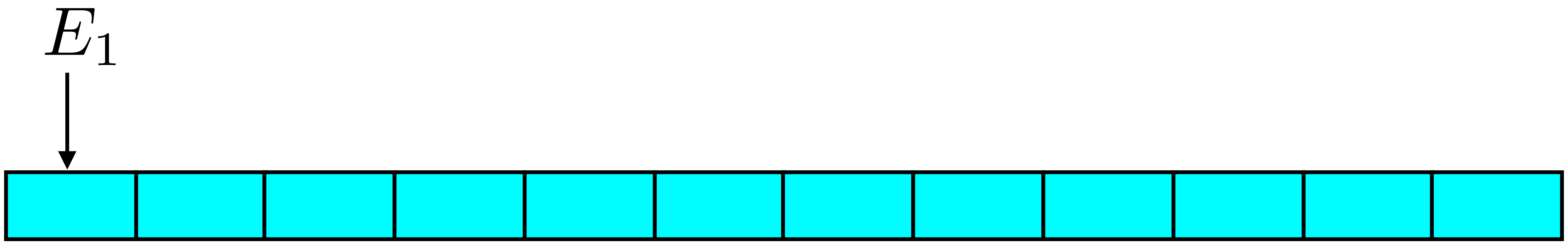


Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.

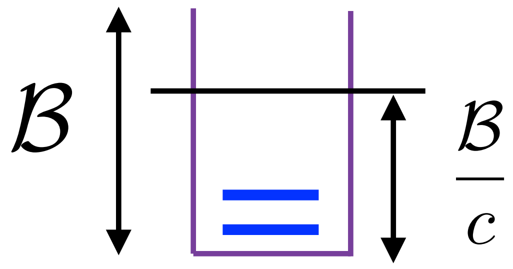


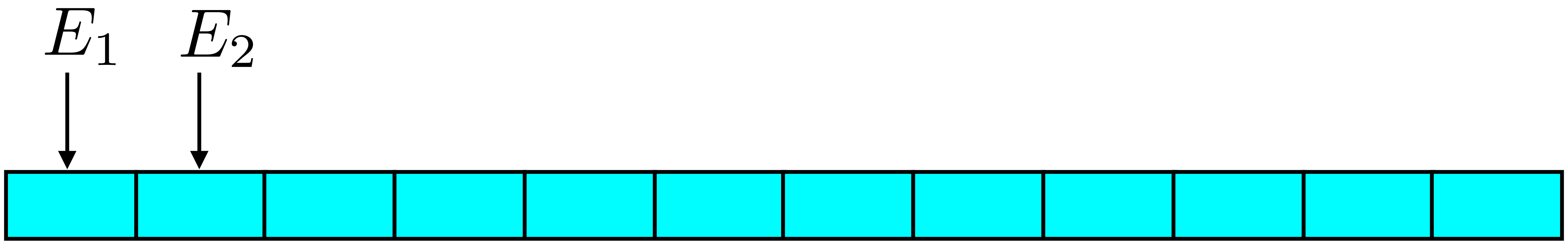


Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.

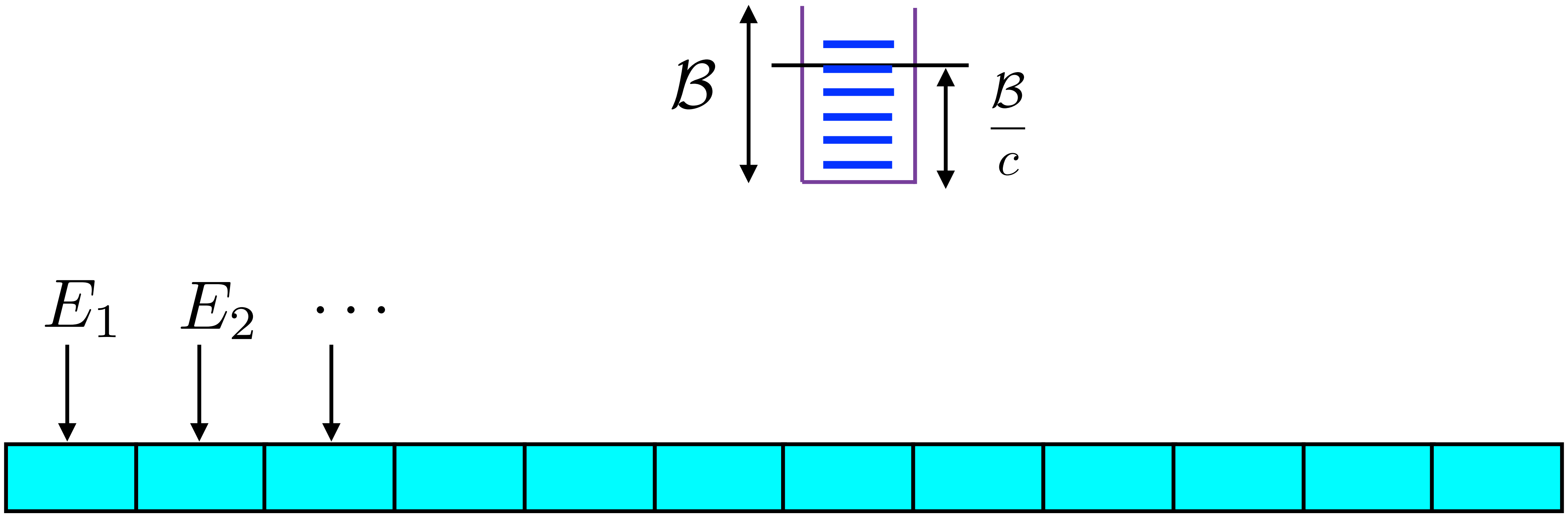




Finite Battery Online Algorithm

Alg: Accumulate&Dump

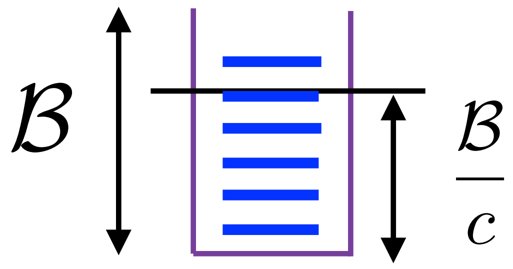
Wait for acc. energy to cross a fixed fraction of battery cap.

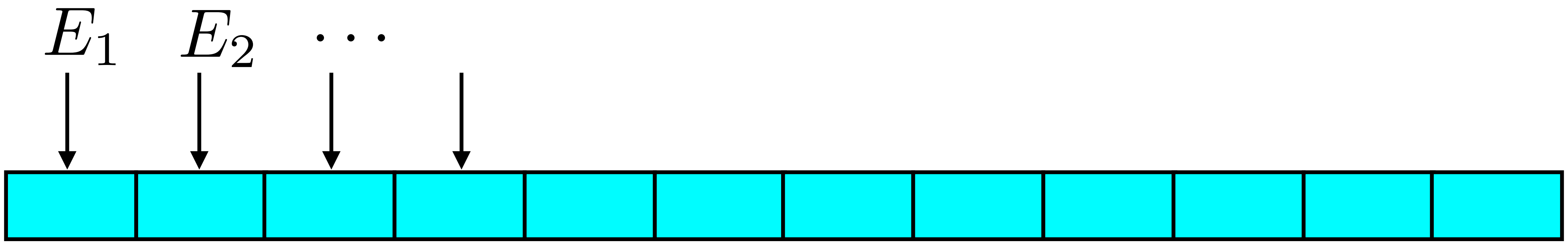


Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.

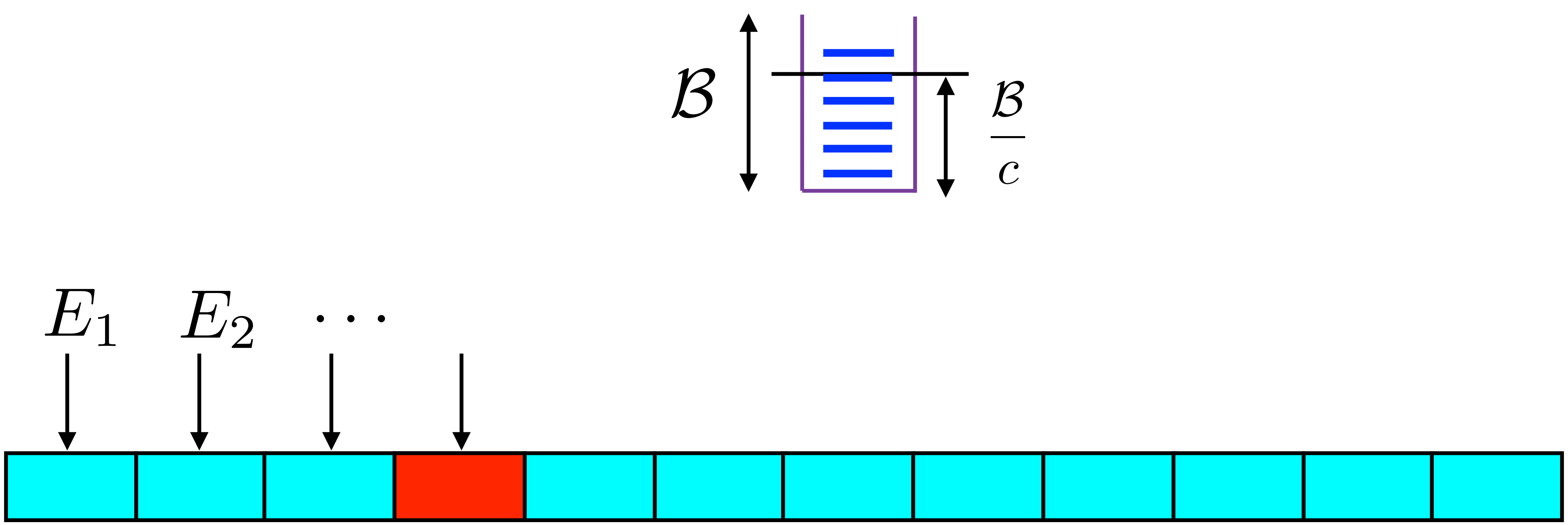




Finite Battery Online Algorithm

Alg: Accumulate&Dump

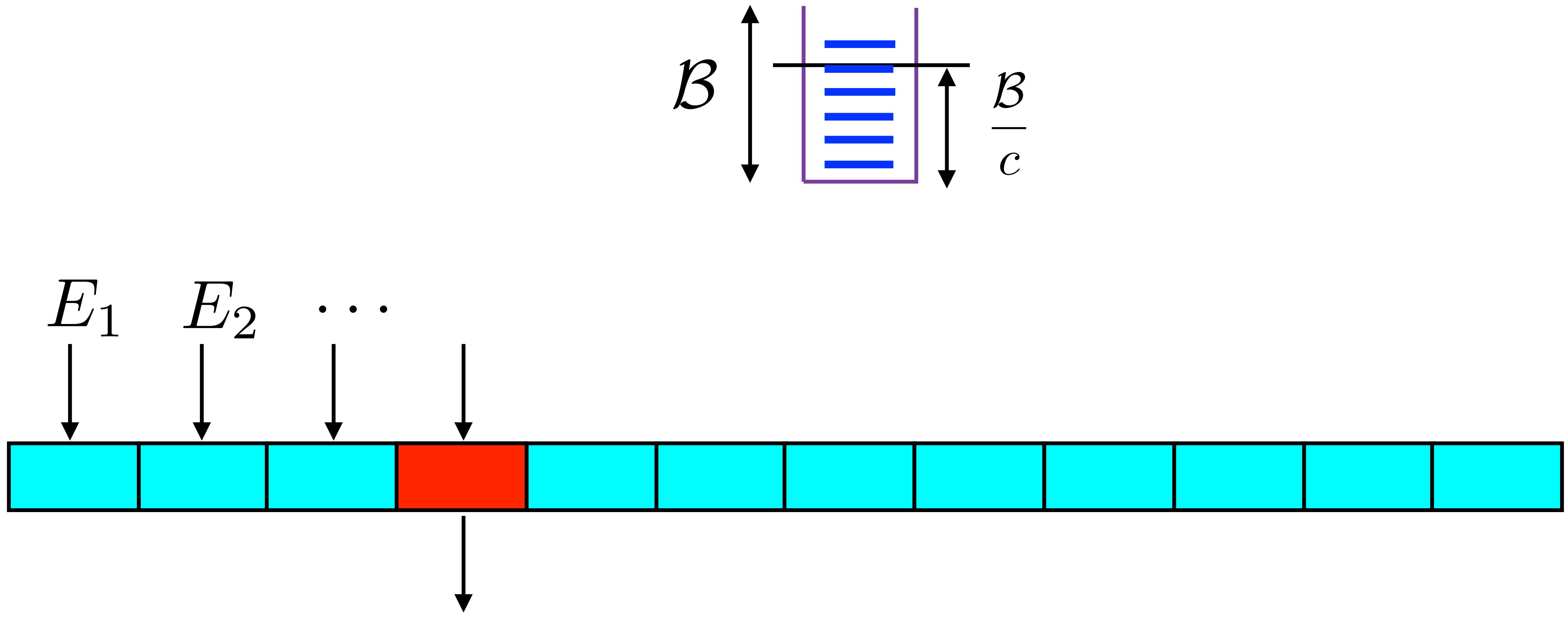
Wait for acc. energy to cross a fixed fraction of battery cap.



Finite Battery Online Algorithm

Alg: Accumulate&Dump

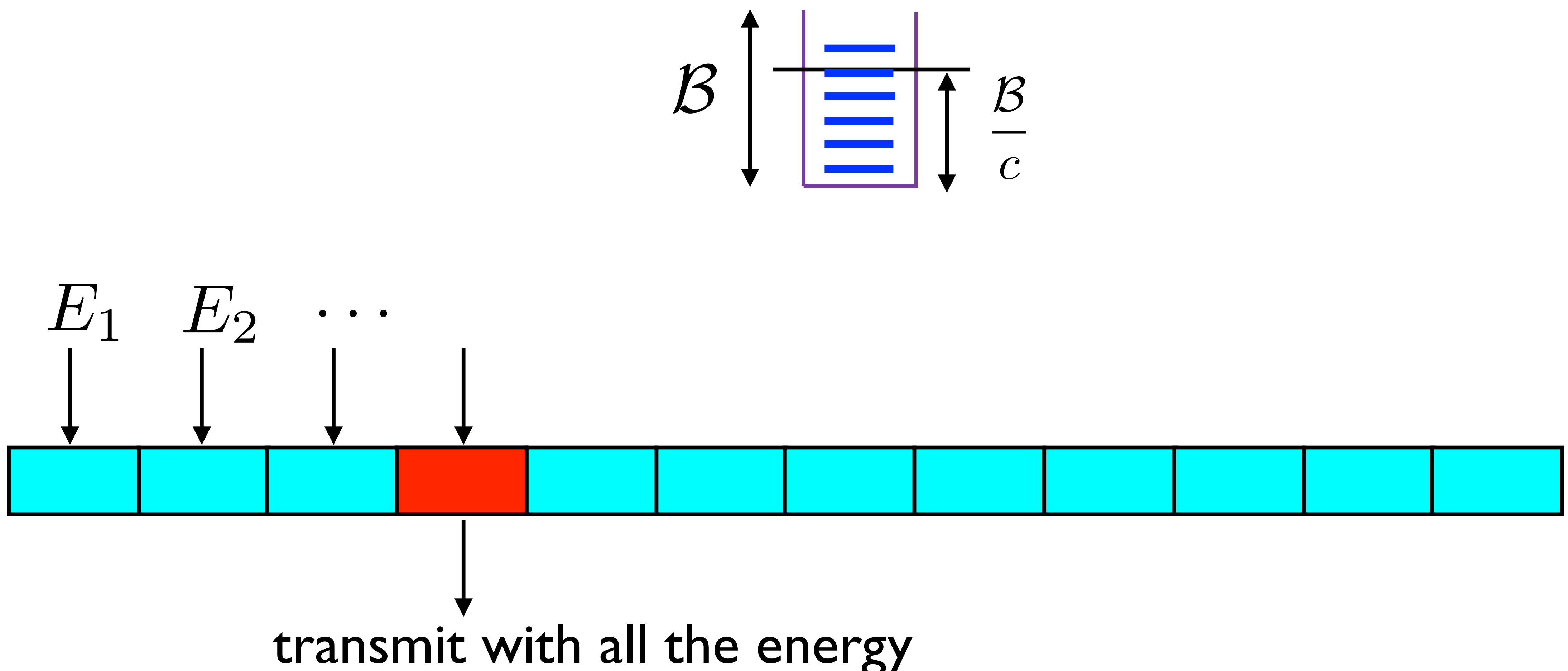
Wait for acc. energy to cross a fixed fraction of battery cap.



Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.



Finite Battery Online Algorithm

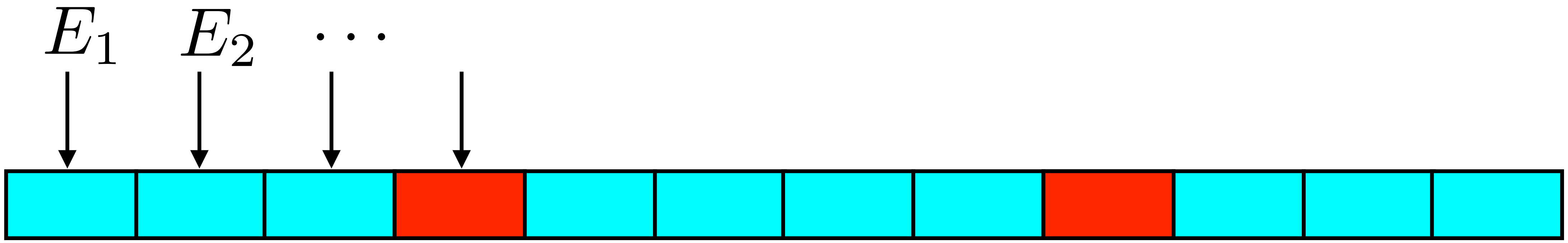
Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.

Finite Battery Online Algorithm

Alg: Accumulate&Dump

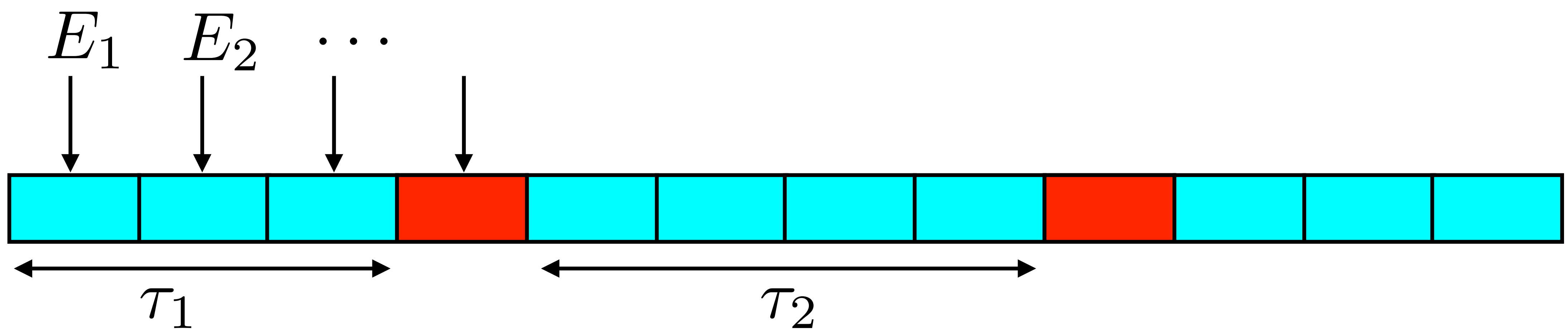
Wait for acc. energy to cross a fixed fraction of battery cap.



Finite Battery Online Algorithm

Alg: Accumulate&Dump

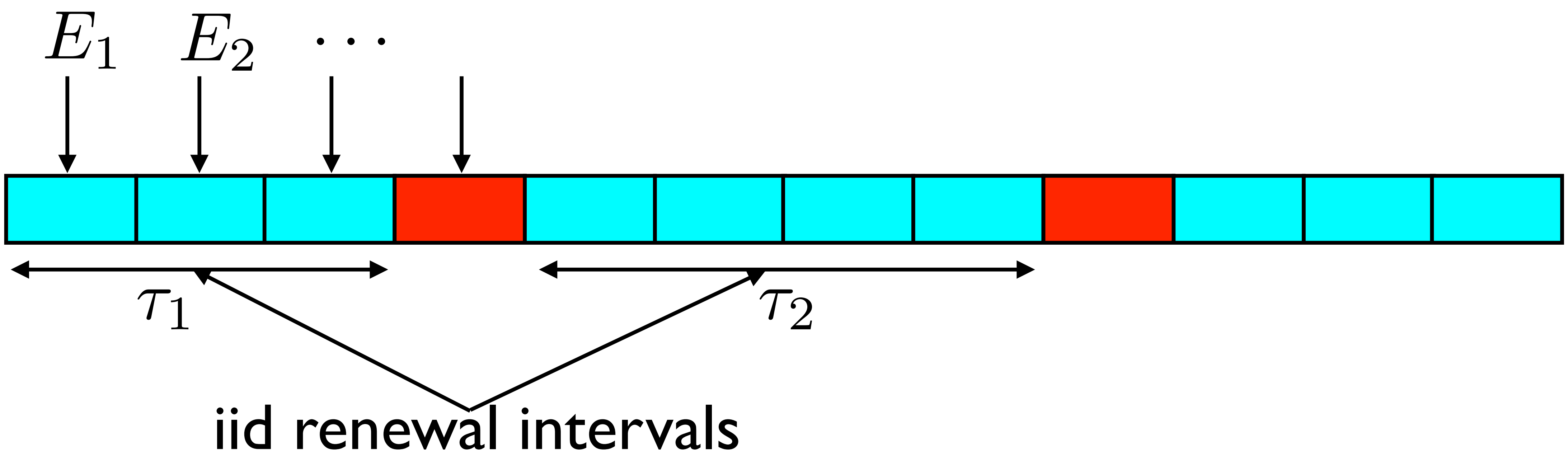
Wait for acc. energy to cross a fixed fraction of battery cap.



Finite Battery Online Algorithm

Alg: Accumulate&Dump

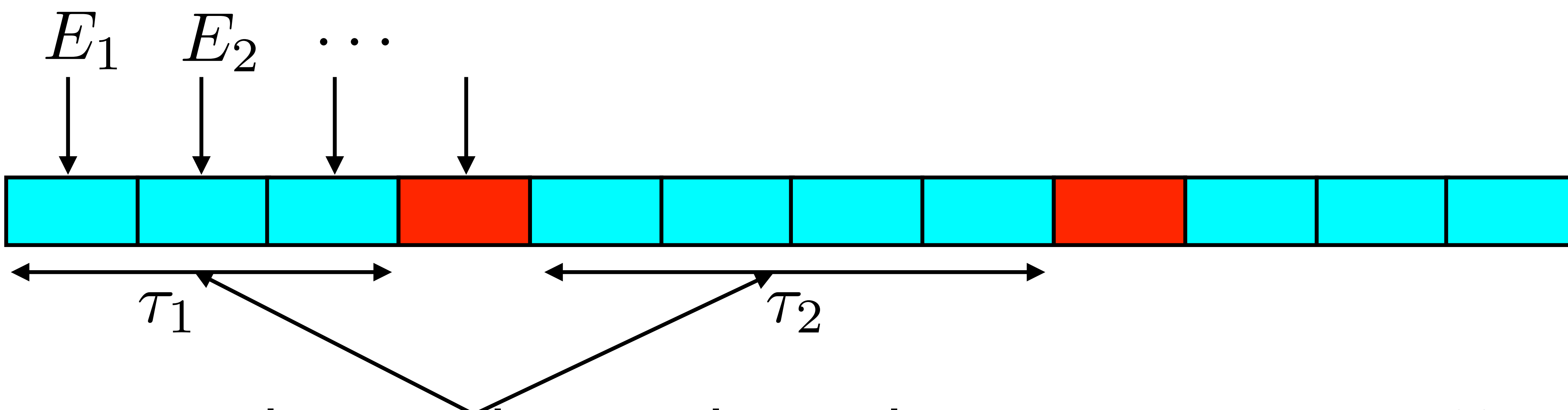
Wait for acc. energy to cross a fixed fraction of battery cap.



Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.

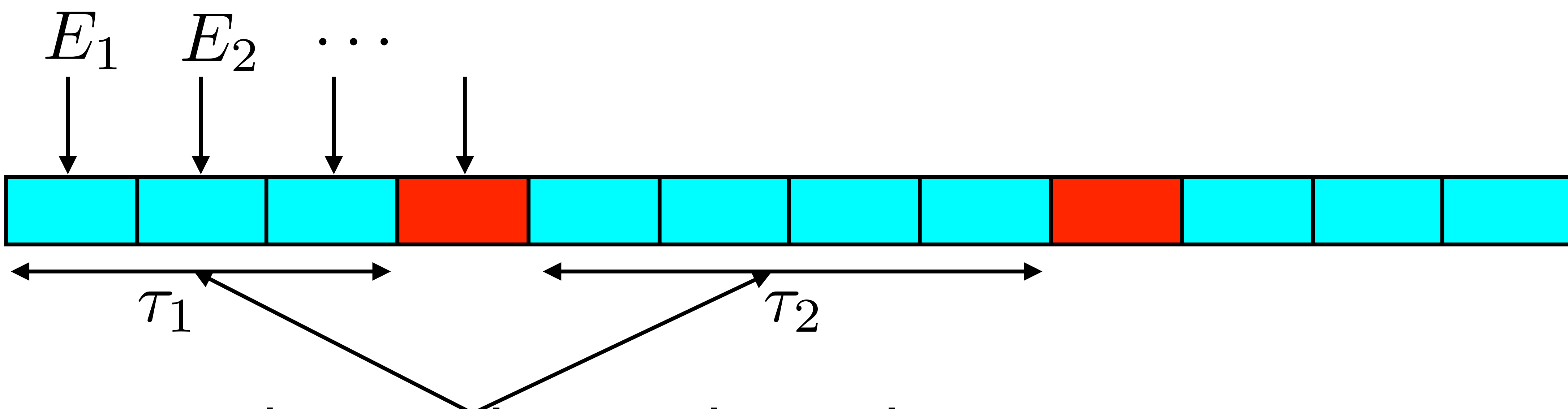


iid renewal intervals also stopping times $E(t) \geq \frac{\mathcal{B}}{c}$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.



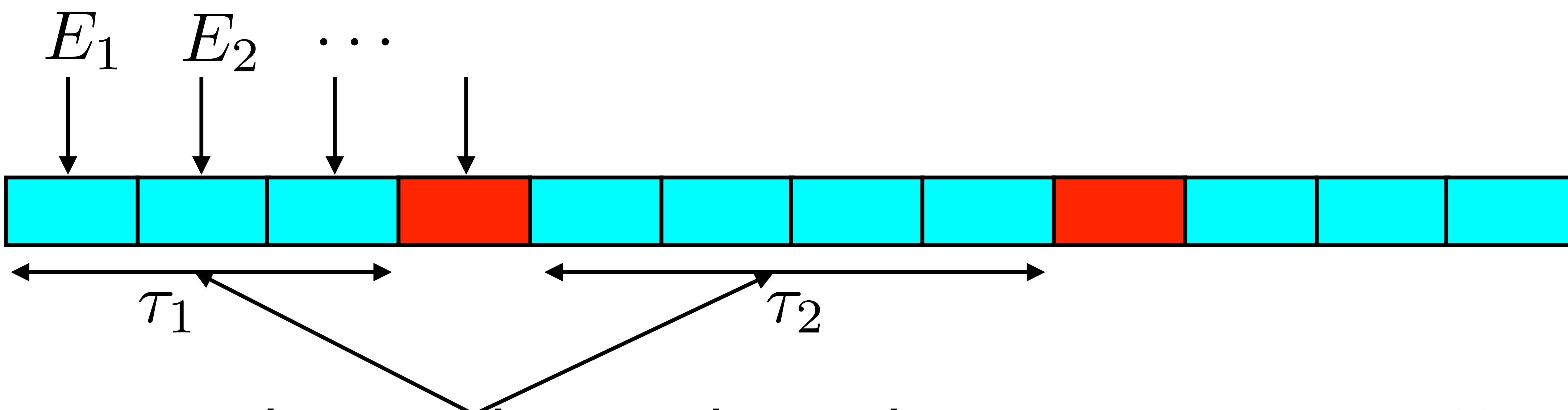
iid renewal intervals also stopping times $E(t) \geq \frac{\mathcal{B}}{c}$

Wald's inequality tells us $\mathbf{E}\{\tau\} = \frac{\mathbf{E}\{\sum_{i=0}^{\tau} E_i\}}{\mathbf{E}\{E\}}$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.



iid renewal intervals also stopping times $E(t) \geq \frac{\mathcal{B}}{c}$

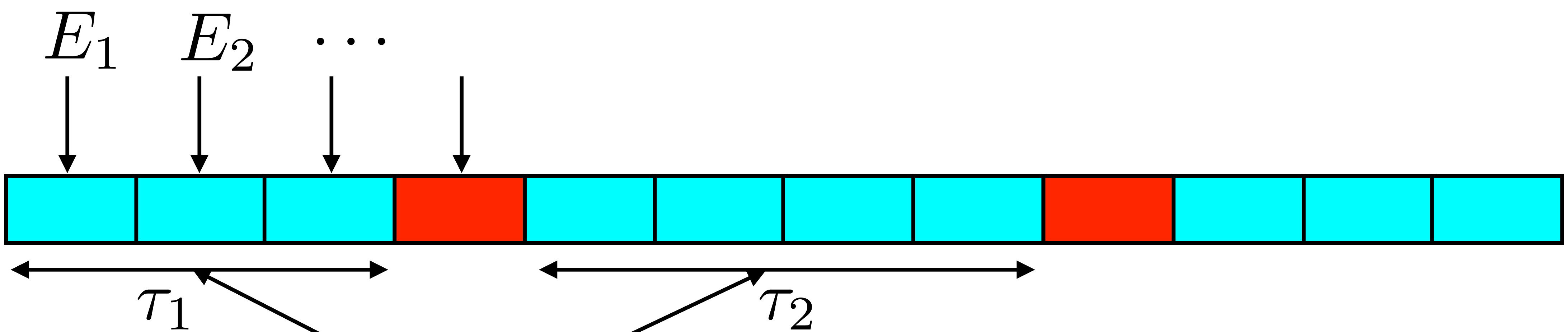
Wald's inequality tells us $$\mathbf{E}\{\tau\} = \frac{\mathbf{E}\{\sum_{i=0}^{\tau} E_i\}}{\mathbf{E}\{E\}}$$

For nice distributions, *Exp*, *Unif* $$\mathbf{E}\{\sum_{i=0}^{\tau} E_i\} \leq \frac{\mathcal{B}}{c} + \mathbf{E}\{E\}$$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.



iid renewal intervals   also stopping times $E(t) \geq \frac{\mathcal{B}}{c}$

Wald's inequality tells us $\mathbf{E}\{\tau\} = \frac{\mathbf{E}\{\sum_{i=0}^{\tau} E_i\}}{\mathbf{E}\{E\}}$

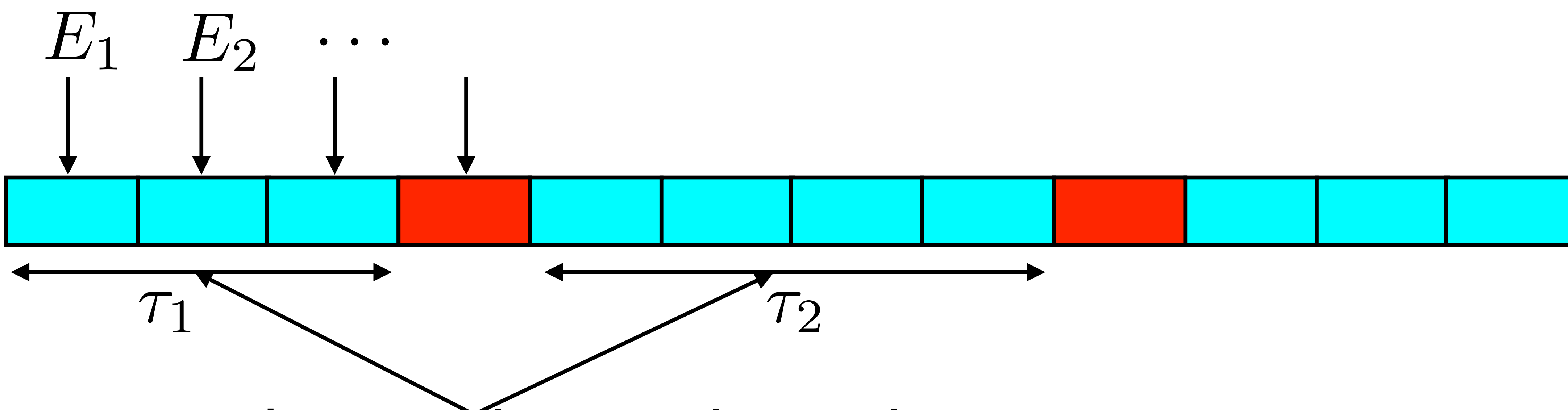
For nice distributions, *Exp, Unif* $\mathbf{E}\{\sum_{i=0}^{\tau} E_i\} \leq \frac{\mathcal{B}}{c} + \mathbf{E}\{E\}$

$$\boxed{\mathbf{E}\{\tau\} \leq \frac{\frac{\mathcal{B}}{c}}{\mathbf{E}\{E\}} + 1}$$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

Wait for acc. energy to cross a fixed fraction of battery cap.



iid renewal intervals also stopping times $E(t) \geq \frac{\mathcal{B}}{c}$

Wald's inequality tells us $$\mathbf{E}\{\tau\} = \frac{\mathbf{E}\{\sum_{i=0}^{\tau} E_i\}}{\mathbf{E}\{E\}}$$

For nice distributions, *Exp*, *Unif* $$\mathbf{E}\{\sum_{i=0}^{\tau} E_i\} \leq \frac{\mathcal{B}}{c} + \mathbf{E}\{E\}$$

$$\boxed{\mathbf{E}\{\tau\} \leq \frac{\frac{\mathcal{B}}{c}}{\mathbf{E}\{E\}} + 1}$$

$$\boxed{\mathbf{E}\{S_{on}\} \leq \mathbf{E}\{\tau\} \frac{B}{g(\mathcal{B}/c)}}$$

Finite Battery Online Algorithm

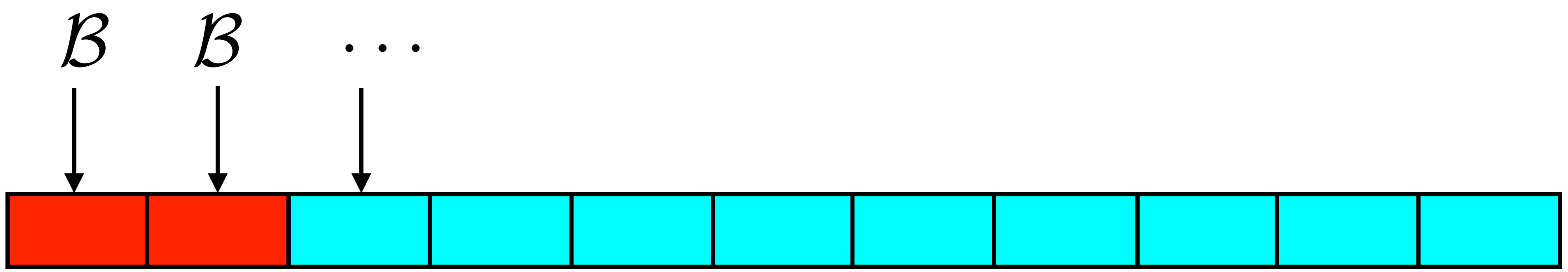
Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot

Finite Battery Online Algorithm

Alg: Accumulate&Dump

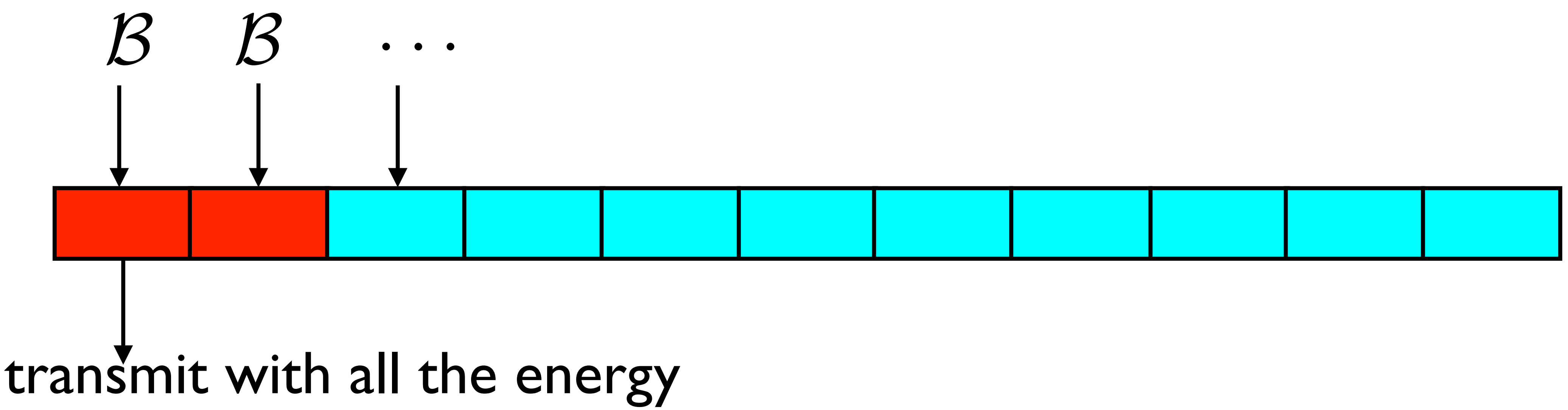
To compare look at an Alg. that gets energy = full batt. cap every slot



Finite Battery Online Algorithm

Alg: Accumulate&Dump

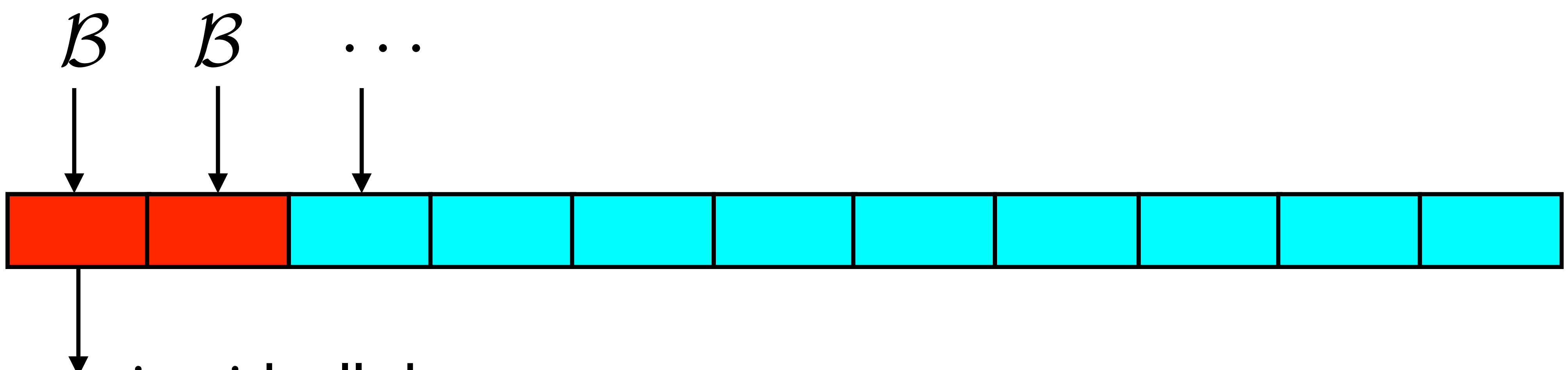
To compare look at an Alg. that gets energy = full batt. cap every slot



Finite Battery Online Algorithm

Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot



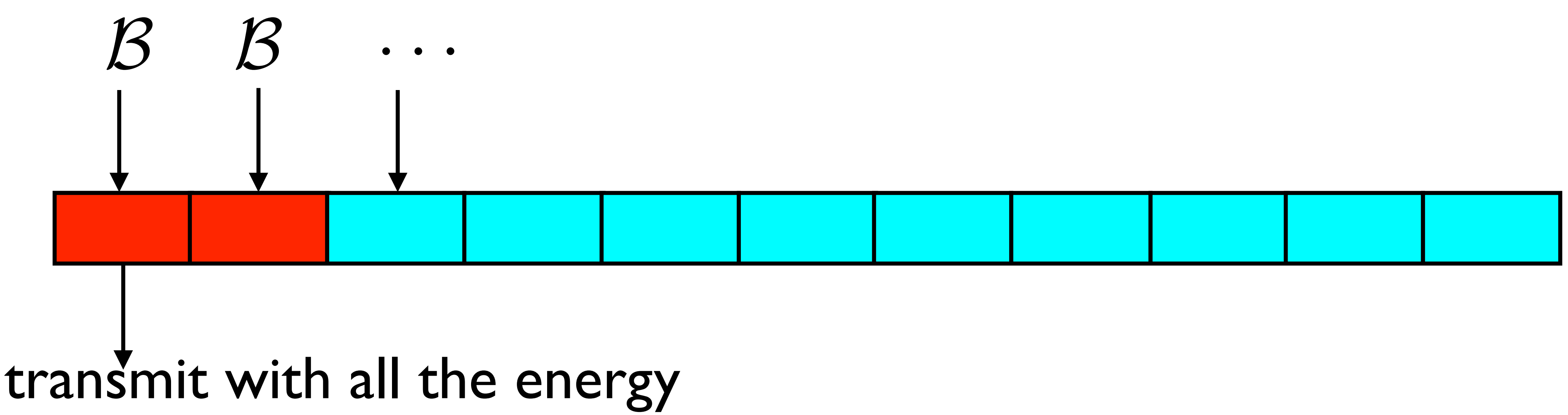
transmit with all the energy

Thus to transmit B bits at least $S_{ub} \geq \frac{B}{g(\mathcal{B})}$ slots are needed

Finite Battery Online Algorithm

Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot



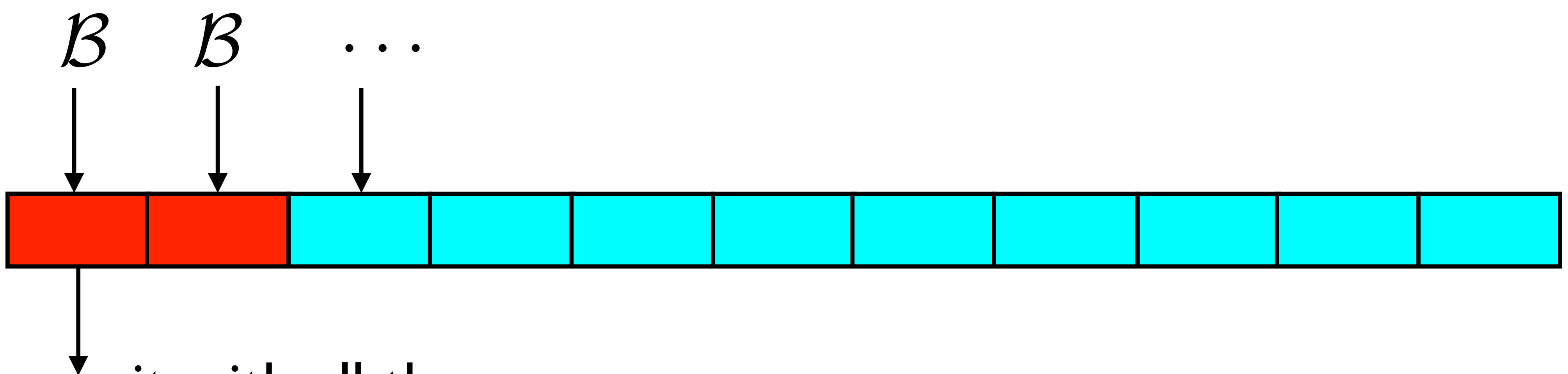
Thus to transmit B bits at least $S_{ub} \geq \dfrac{B}{g(\mathcal{B})}$ slots are needed

While for the online algorithm $\mathbf{E}\{S_{on}\} \leq \mathbf{E}\{\tau\}\dfrac{B}{g(\mathcal{B}/c)}$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot



transmit with all the energy

Thus to transmit B bits at least $S_{ub} \geq \dfrac{B}{g(\mathcal{B})}$ slots are needed

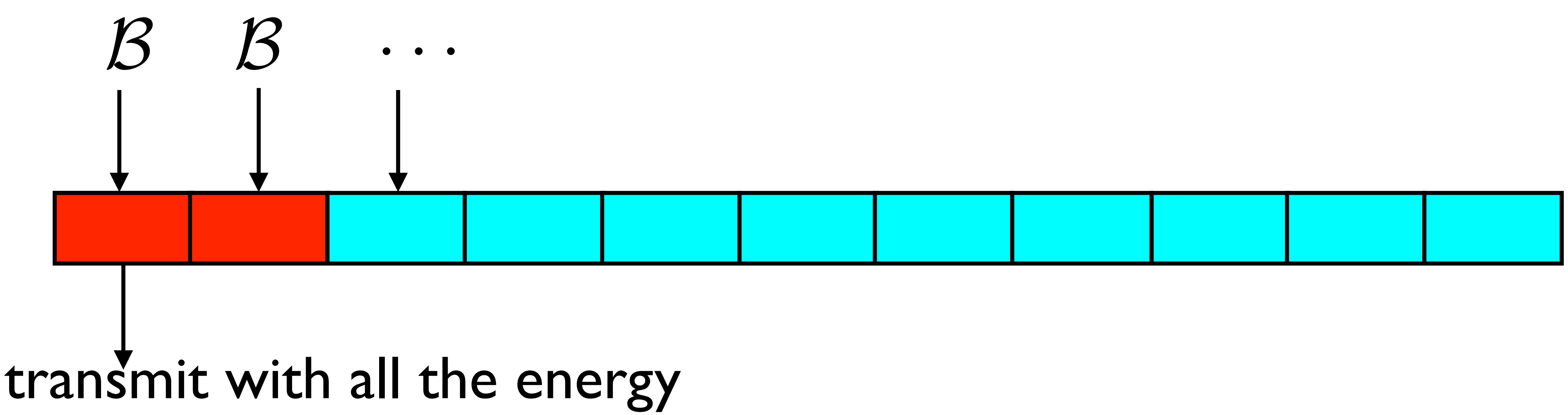
While for the online algorithm $\mathbf{E}\{S_{on}\} \leq \mathbf{E}\{\tau\}\dfrac{B}{g(\mathcal{B}/c)}$

Choosing

Finite Battery Online Algorithm

Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot



Thus to transmit B bits at least $S_{ub} \geq \dfrac{B}{g(\mathcal{B})}$ slots are needed

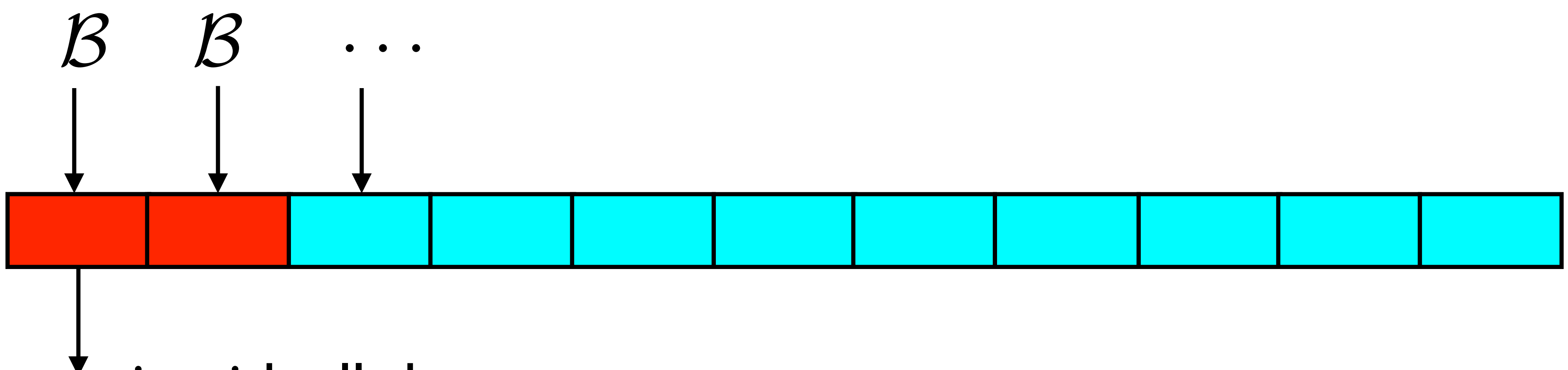
While for the online algorithm $\mathbf{E}\{S_{on}\} \leq \mathbf{E}\{\tau\}\dfrac{B}{g(\mathcal{B}/c)}$

Choosing $c = \dfrac{\mathcal{B}}{\mathbf{E}\{E\}},$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot



transmit with all the energy

Thus to transmit B bits at least $S_{ub} \geq \dfrac{B}{g(\mathcal{B})}$ slots are needed

While for the online algorithm $\mathbf{E}\{S_{on}\} \leq \mathbf{E}\{\tau\}\dfrac{B}{g(\mathcal{B}/c)}$

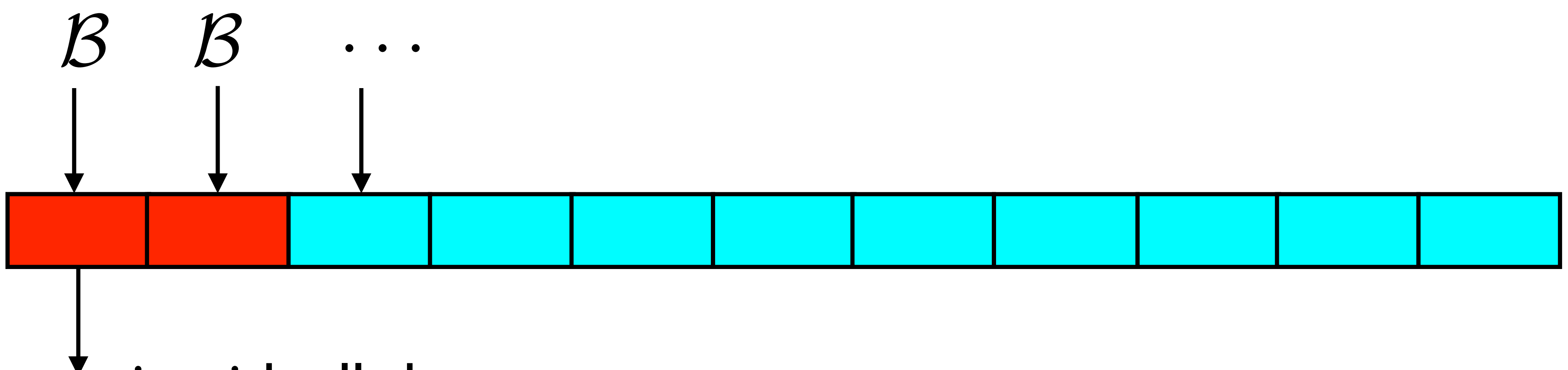
Choosing $c = \dfrac{\mathcal{B}}{\mathbf{E}\{E\}}$,

$$\mathbf{E}\{\mathrm{r}\} \leq \frac{\mathbf{E}\{S_{on}\}}{S_{ub}}$$

Finite Battery Online Algorithm

Alg: Accumulate&Dump

To compare look at an Alg. that gets energy = full batt. cap every slot



transmit with all the energy

Thus to transmit B bits at least $S_{ub} \geq \dfrac{B}{g(\mathcal{B})}$ slots are needed

While for the online algorithm $\mathbf{E}\{S_{on}\} \leq \mathbf{E}\{\tau\}\dfrac{B}{g(\mathcal{B}/c)}$

Choosing $c = \dfrac{\mathcal{B}}{\mathbf{E}\{E\}}$,

$\mathbf{E}\{\mathrm{r}\} \leq \dfrac{\mathbf{E}\{S_{on}\}}{S_{ub}}$ is a constant for $g(x) = \log(1+x)$

Conclusions

- Optimal Offline Algorithm - modular
- Online Algorithm *cr*=2
- Finite Battery can be handled as well

Rahul Vaze teaches at the School of Technology and Computer Science, Tata Institute of Fundamental Research, Mumbai. He obtained his PhD from the University of Texas at Austin. His research interests are in the areas of multiple antenna communication, ad hoc networks, and combinatorial resource allocation. He is the recipient of the Indian National Science Academy's young scientist award for the year 2013.

This book discusses the theoretical limits of information transfer in random wireless networks or ad hoc networks, where nodes are distributed uniformly random in space and there is no centralized control. Examples of ad hoc networks include sensor networks, military networks, and vehicular networks that have widespread applications. Decentralized nature of these networks makes them easily configurable, scalable, and inherently robust.

The author provides a detailed analysis of the two relevant notions of capacity for random wireless networks – transmission capacity and throughput capacity. The book starts with the transmission capacity framework that is first presented for the single-hop model and later extended to the multi-hop model with retransmissions. By reusing some of the tools developed for analysis of transmission capacity, few key long-standing questions about the performance analysis of cellular networks are also addressed for the benefit of students. To complete the throughput capacity characterization, the author finally discusses the concept of hierarchical cooperation that allows the throughput capacity to scale linearly with the number of nodes.

Rahul Vaze teaches at the School of Technology and Computer Science, Tata Institute of Fundamental Research, Mumbai

Cover image source: Kotkoa / Shutterstock

CAMBRIDGE
UNIVERSITY PRESS
www.cambridge.org

ISBN 978-1-107-10232-3

9 781107 102323

Vaze **Random Wireless Networks** CAMBRIDGE

Random Wireless Networks

An Information Theoretic Perspective

Rahul Vaze

The optimal role of multiple antennas, ARQ protocols, and scheduling protocols in random wireless networks is identified using the transmission capacity paradigm. This book provides a holistic view of all relevant tools and concepts used to analyse random wireless networks. A conscious attempt is made to bring out the connections between transmission and throughput capacity, between percolation theory and throughput capacity, and stochastic geometry and cellular networks. For effective understanding, an extensive effort is made to explain the physical interpretation of all results.