On-line Scheduling with level of Services

Ee-Chien Chang
School of Computing
National University of Singapore
changec@comp.nus.edu.sg

Chee Yap
Courant Institute of Mathematical Science
New York University

On-line Scheduling with Level of Services

Motivated by an application in visualization across network, we study an abstract on-line scheduling problem.

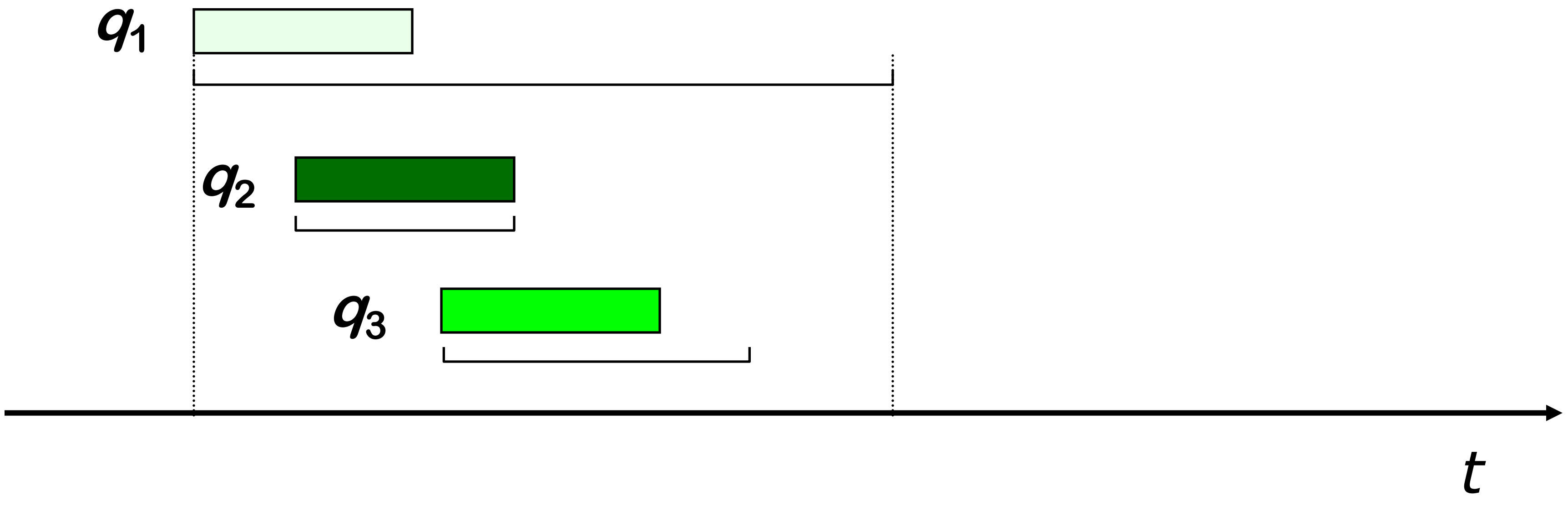
Our schedulers can gain partial merit from a partially served request. Thus the problem embodies a notion of “level of services”.

We give 2 schedulers FirstFit and EndFit which based on 2 simple heuristics. Both are 2-competitive.
We generalize them to a class of Greedy schedulers.
Any greedy scheduler is 3-competitive.

On-line Scheduling with Level of Services

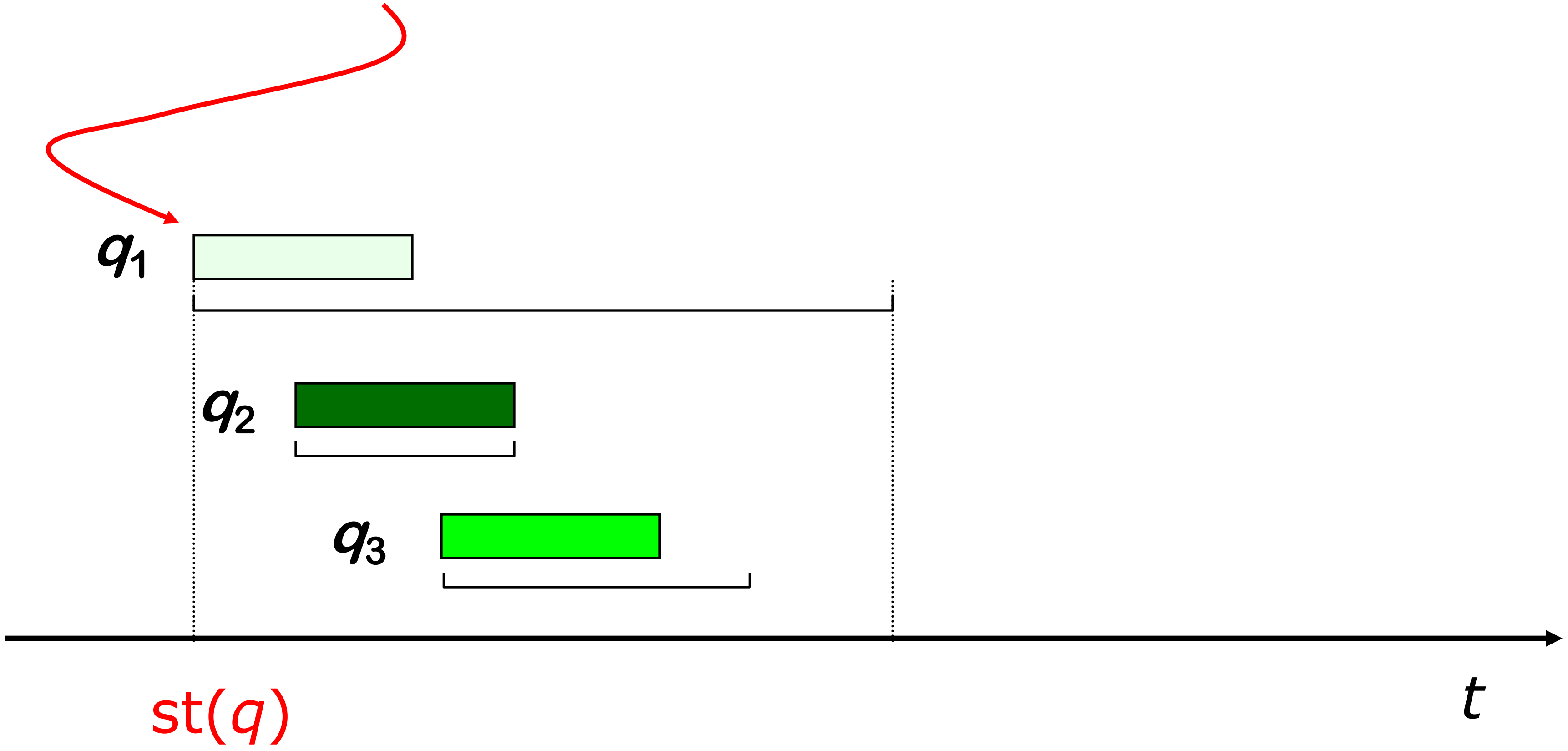
An instance I is a sequence of n requests.

Each request is parameterized by
q = (start-time, deadline, volume, weight).



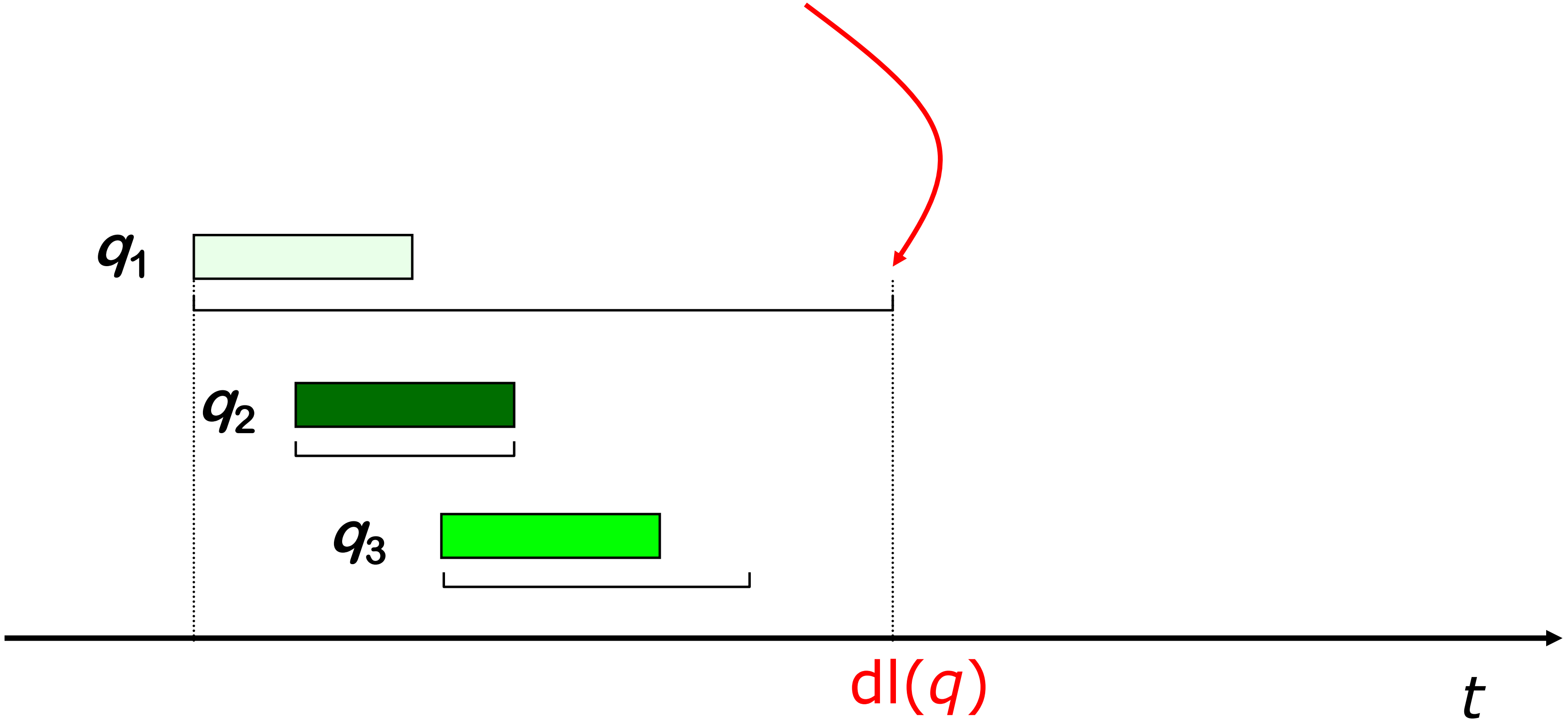
Each request is parameterized by

q = (start-time, deadline, volume, weight).



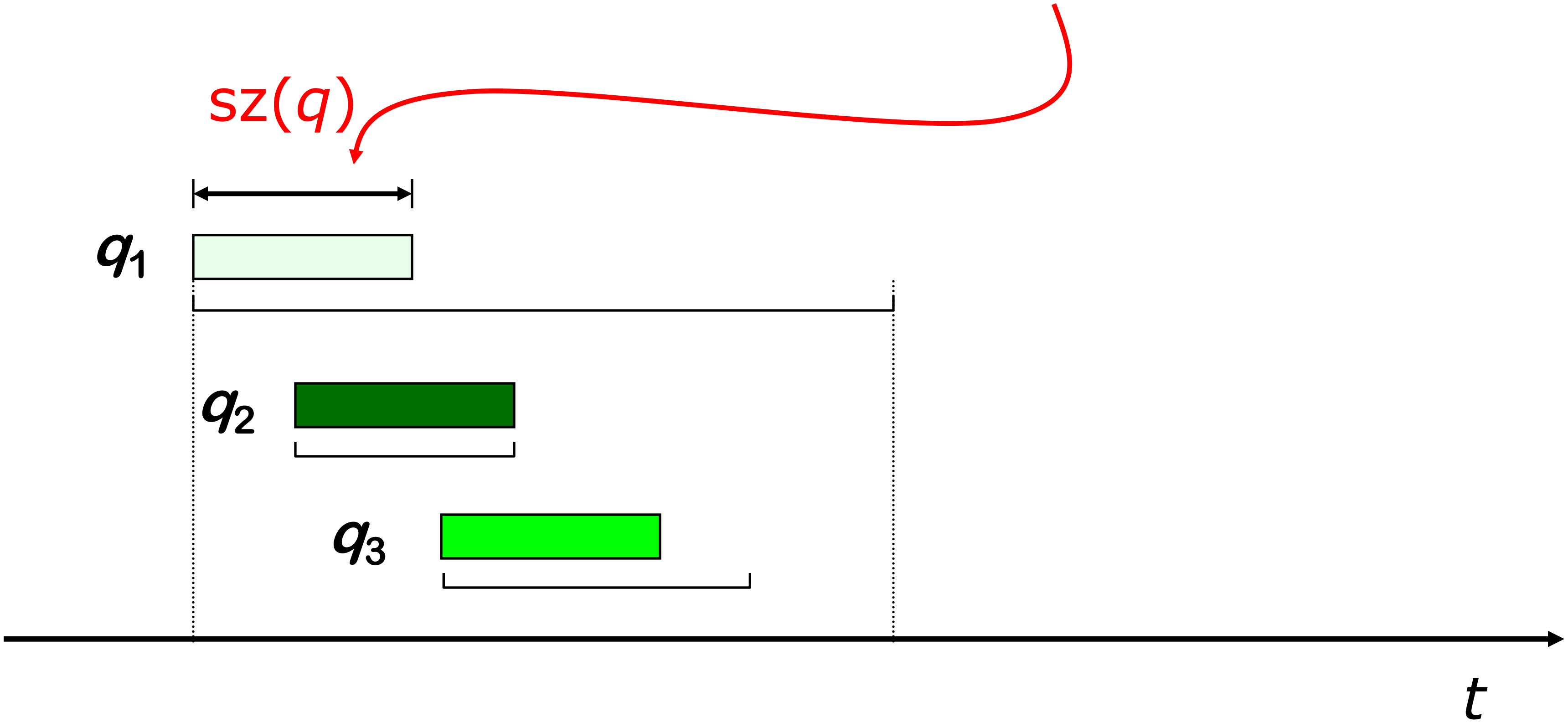
Each request is parameterized by

q = (start-time, deadline, volume, weight).



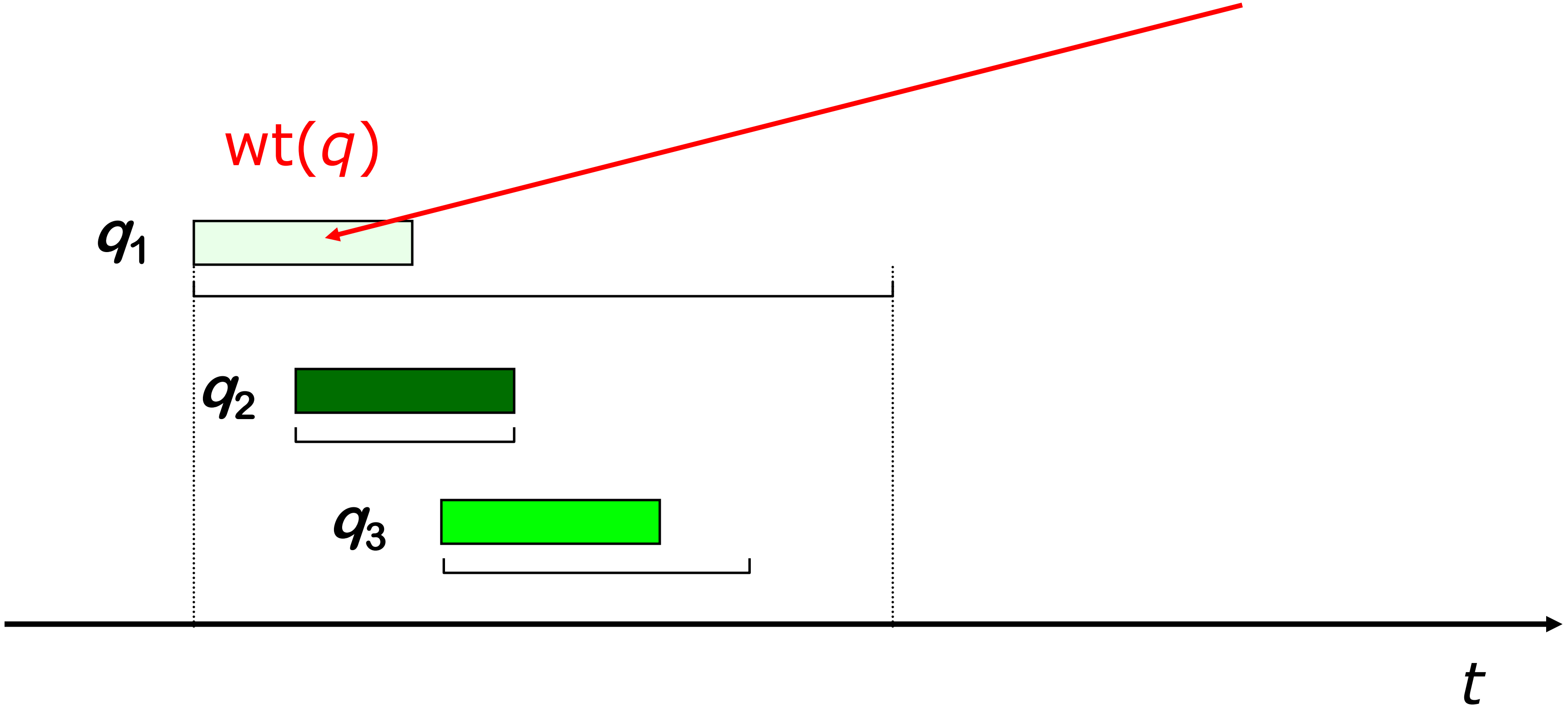
Each request is parameterized by

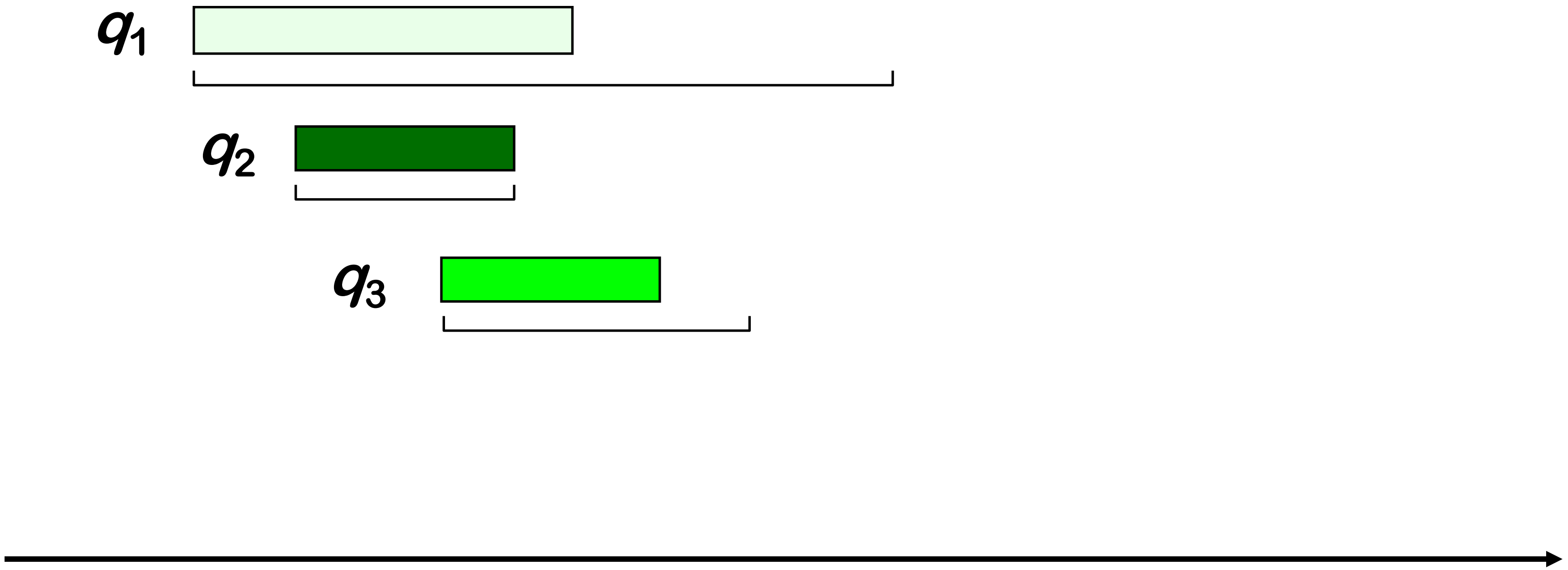
q = (start-time, deadline, volume, weight).



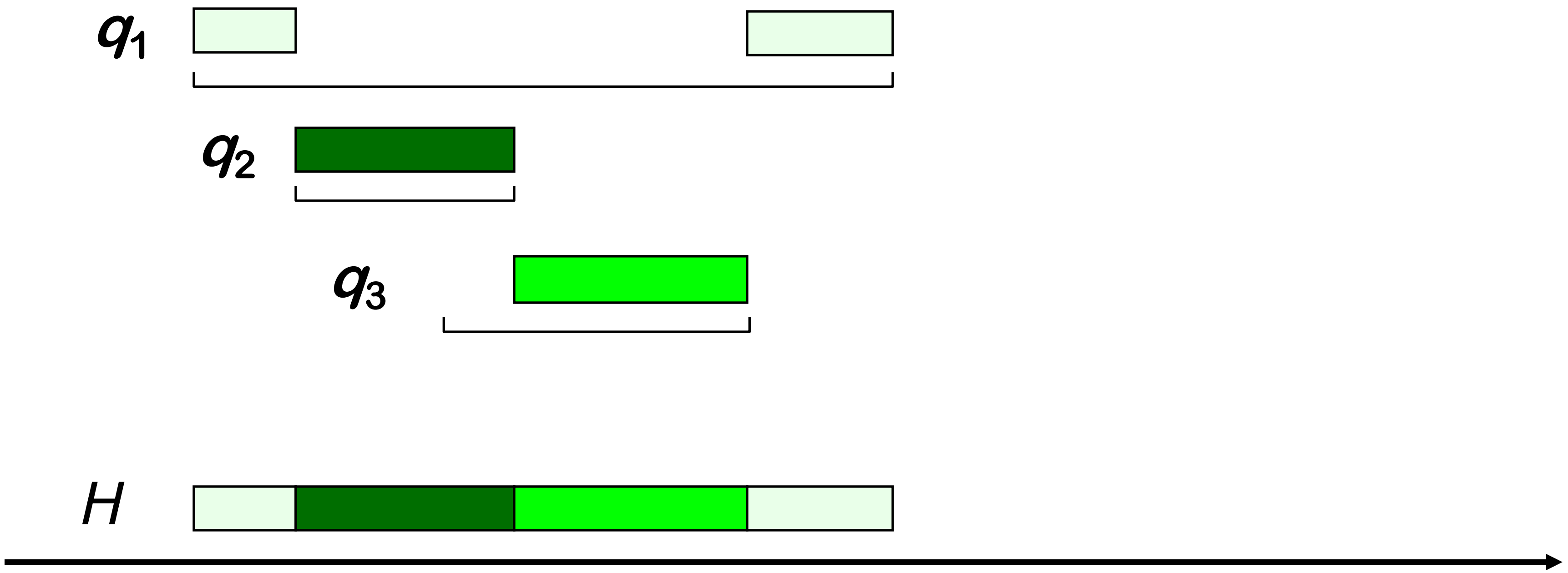
Each request is parameterized by

q = (start-time, deadline, volume, weight).




q_1
q_2
q_3

A valid schedule *H*.



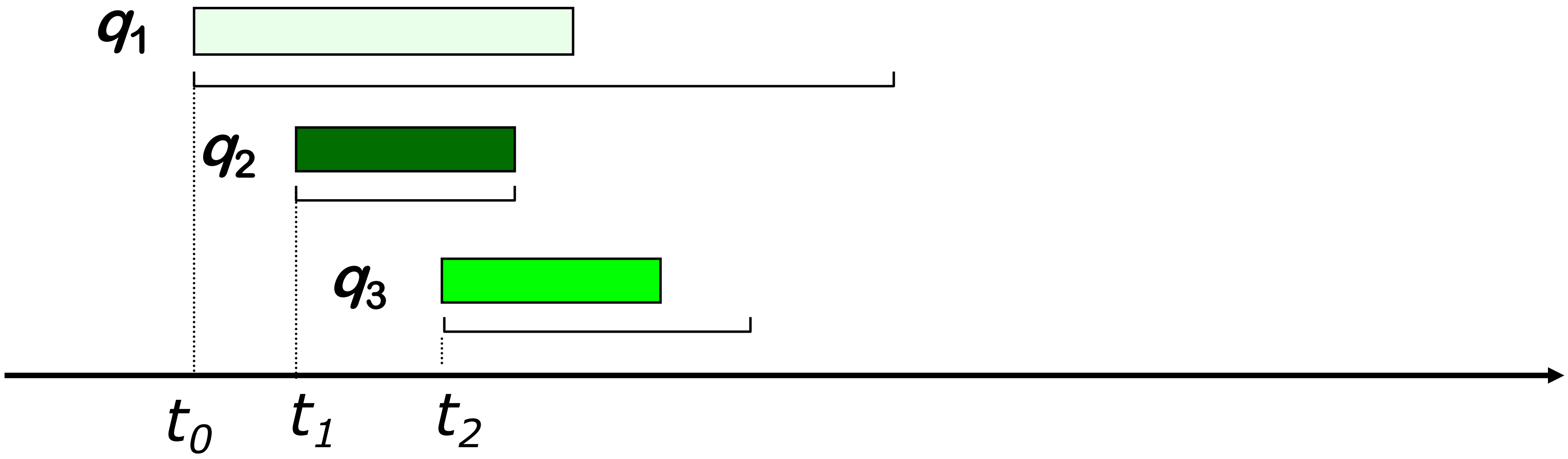
merit (*H*) =

$$\sum_{q} \text{(weight of } q\text{)} \;\; * \;\; \text{(total size of } q \text{ served in } H\text{)}$$

1. Unlike most scheduling problems, a partially served request contributes to the merit.

2. Each request can be broken into finite number of pieces.

3. We consider online scheduling, i.e., at time *t,* the server only sees requests whose start-time is earlier than *t.*



4. A scheduler S is c-competitive if for any I,

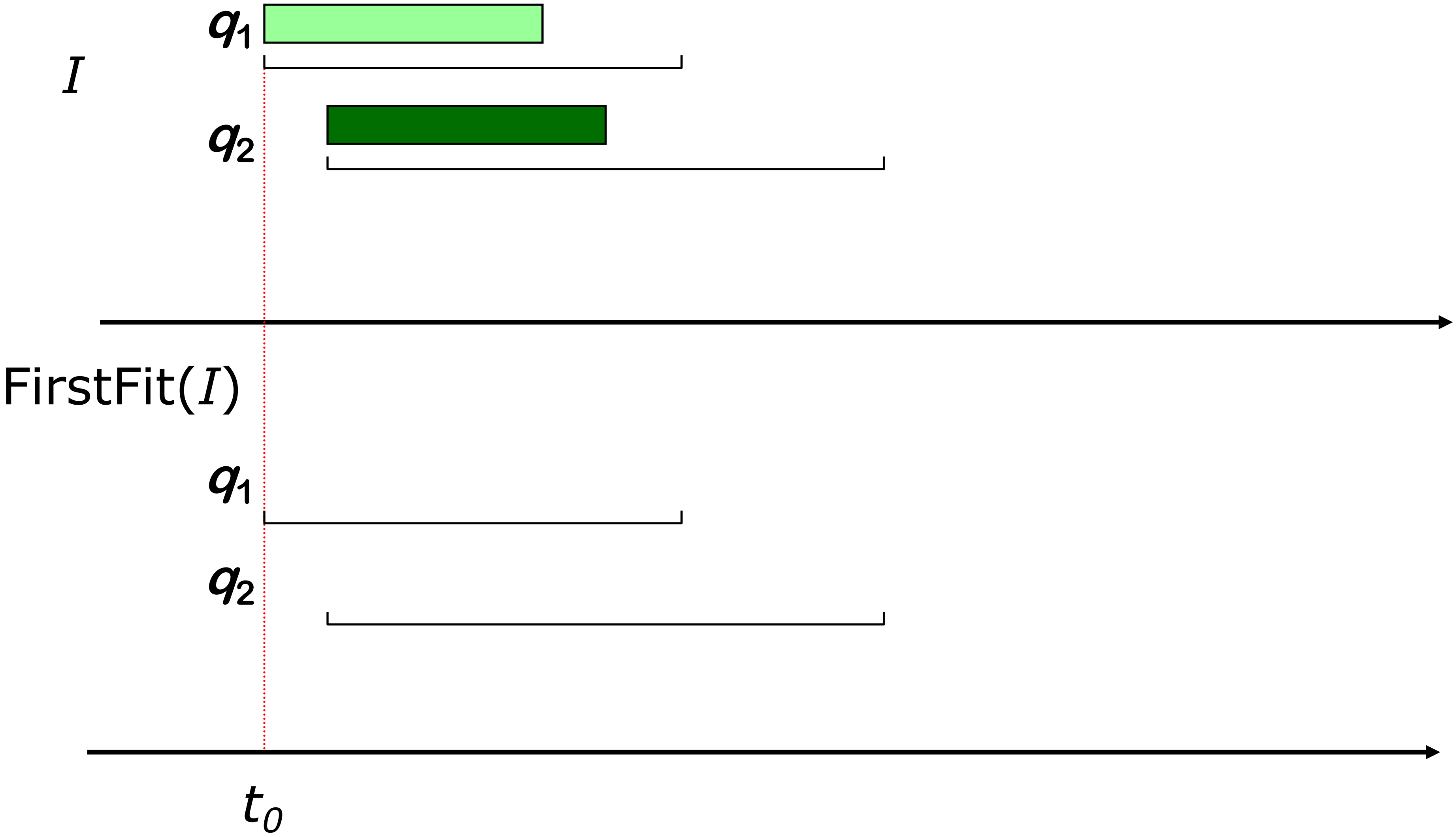
c merit ($S(I)$) $\geqslant$ merit (offline_optimal (I))

Two schedulers

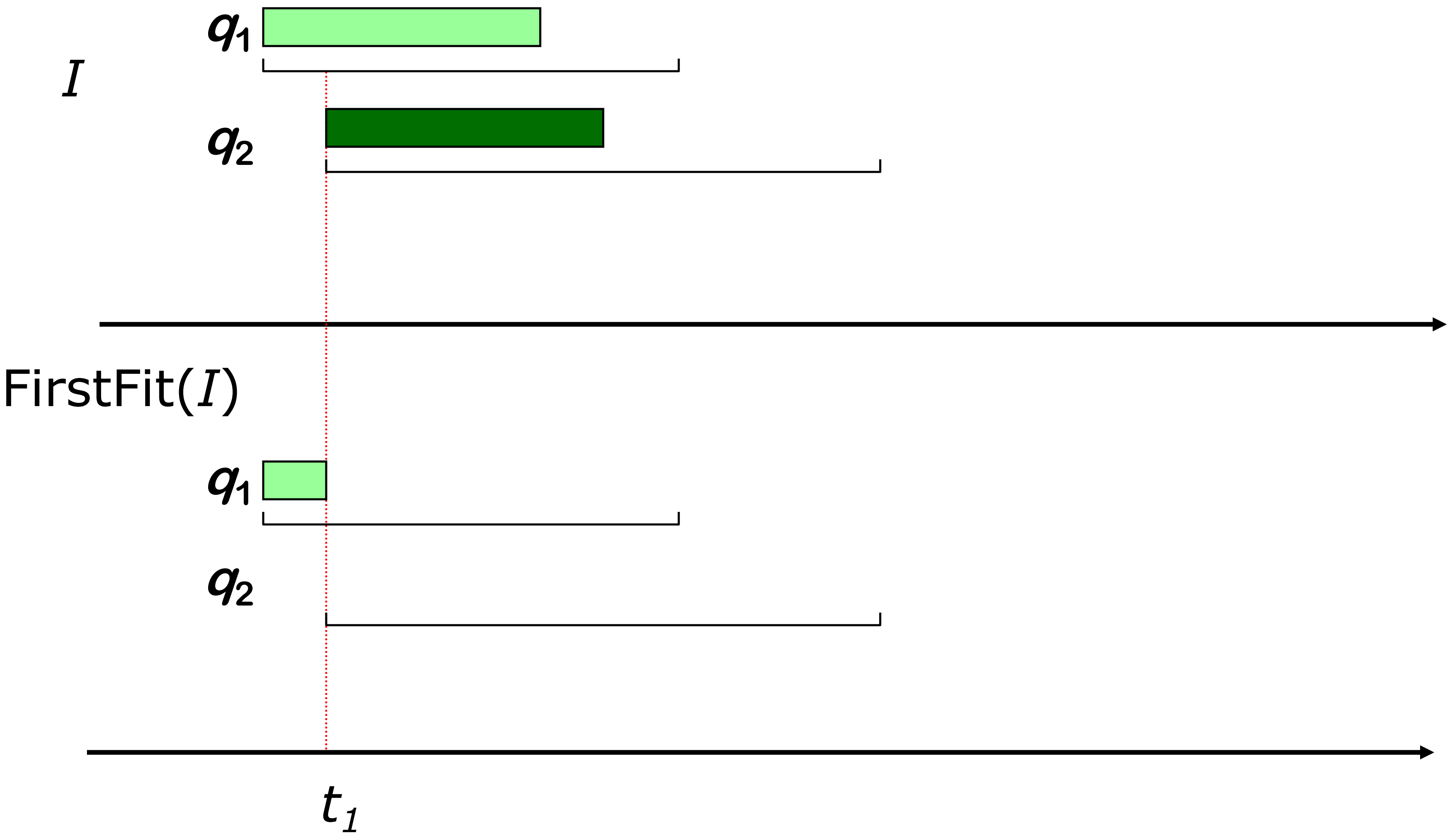
1. FirstFit: always serves the current heaviest residual request.

2. EndFit: always serves according to the off-line optimal schedule of the residual requests.

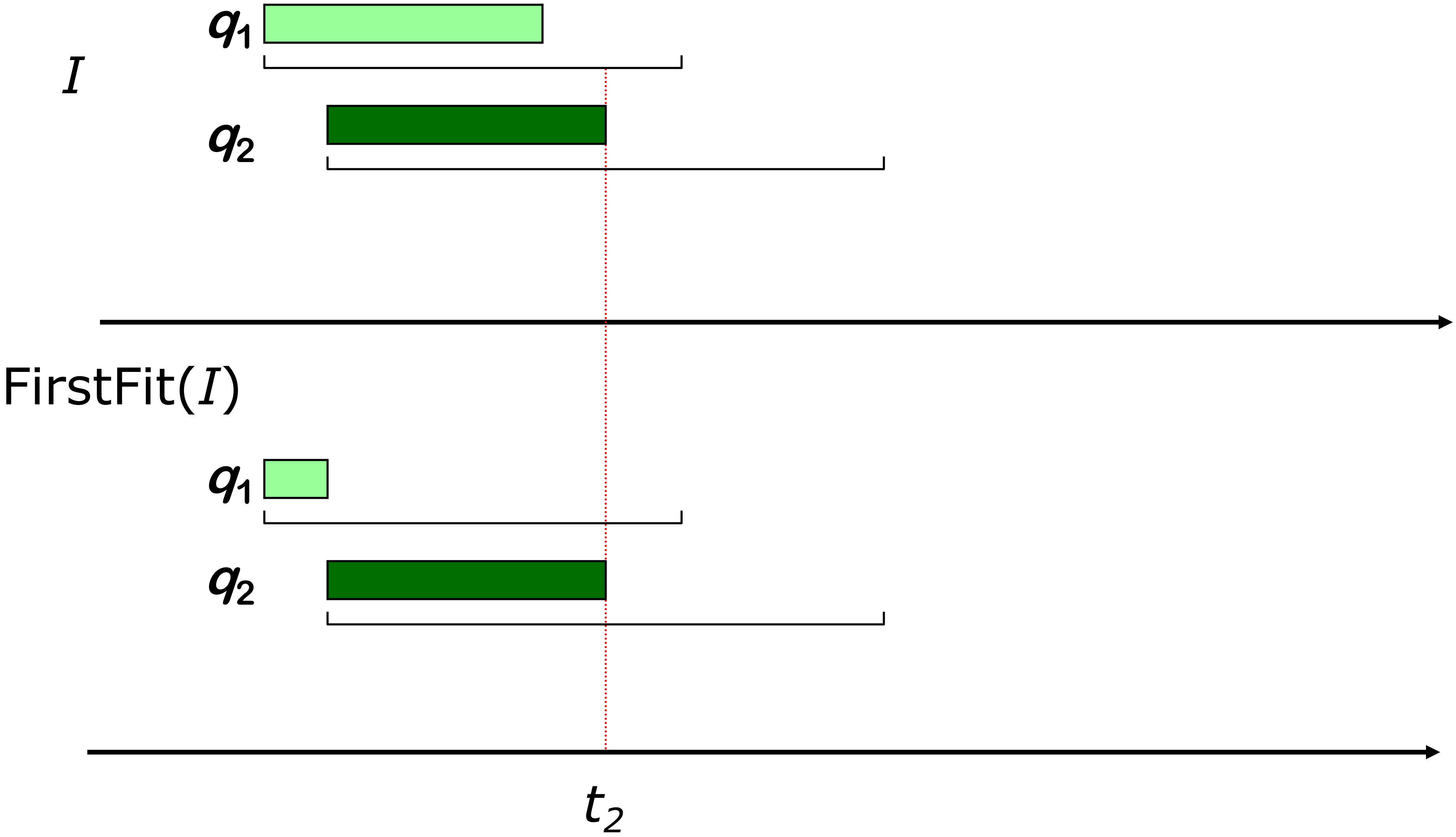
1. FirstFit: always serves the current heaviest residual request.



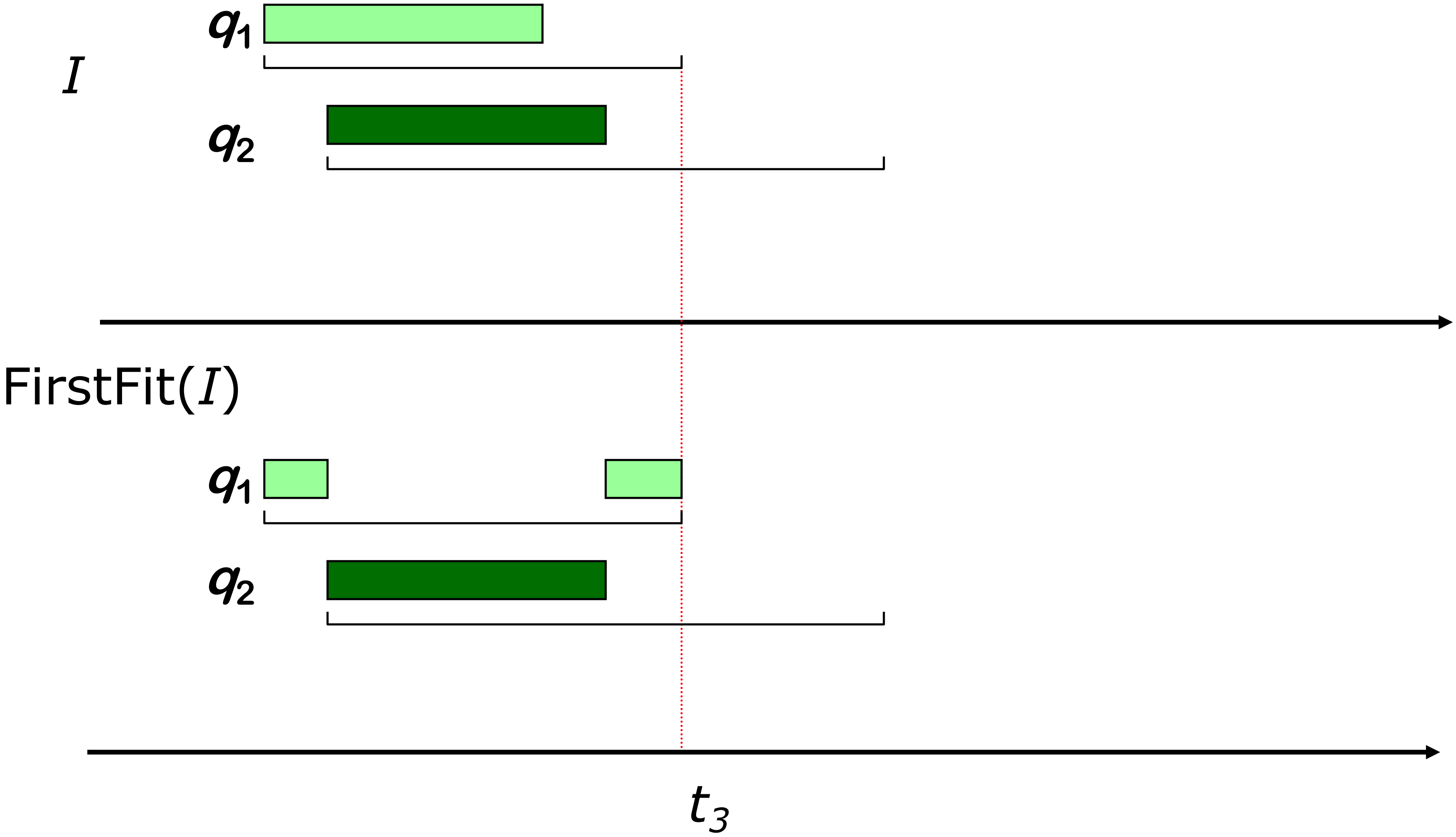
1. FirstFit: always serves the current heaviest residual request.



1. FirstFit: always serves the current heaviest residual request.



1. FirstFit: always serves the current heaviest residual request.



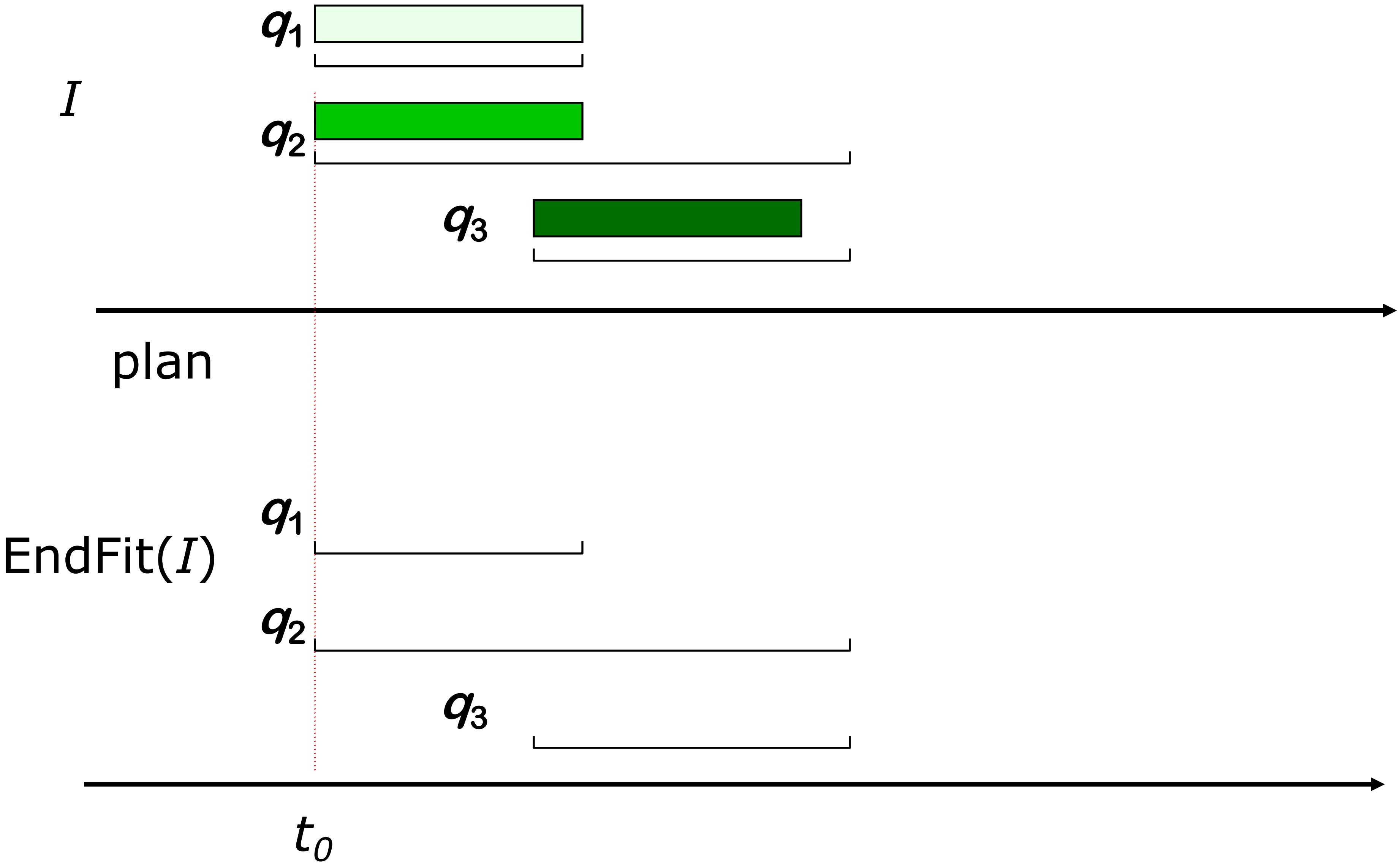
Theorem 1

FirstFit is 2-competitive.

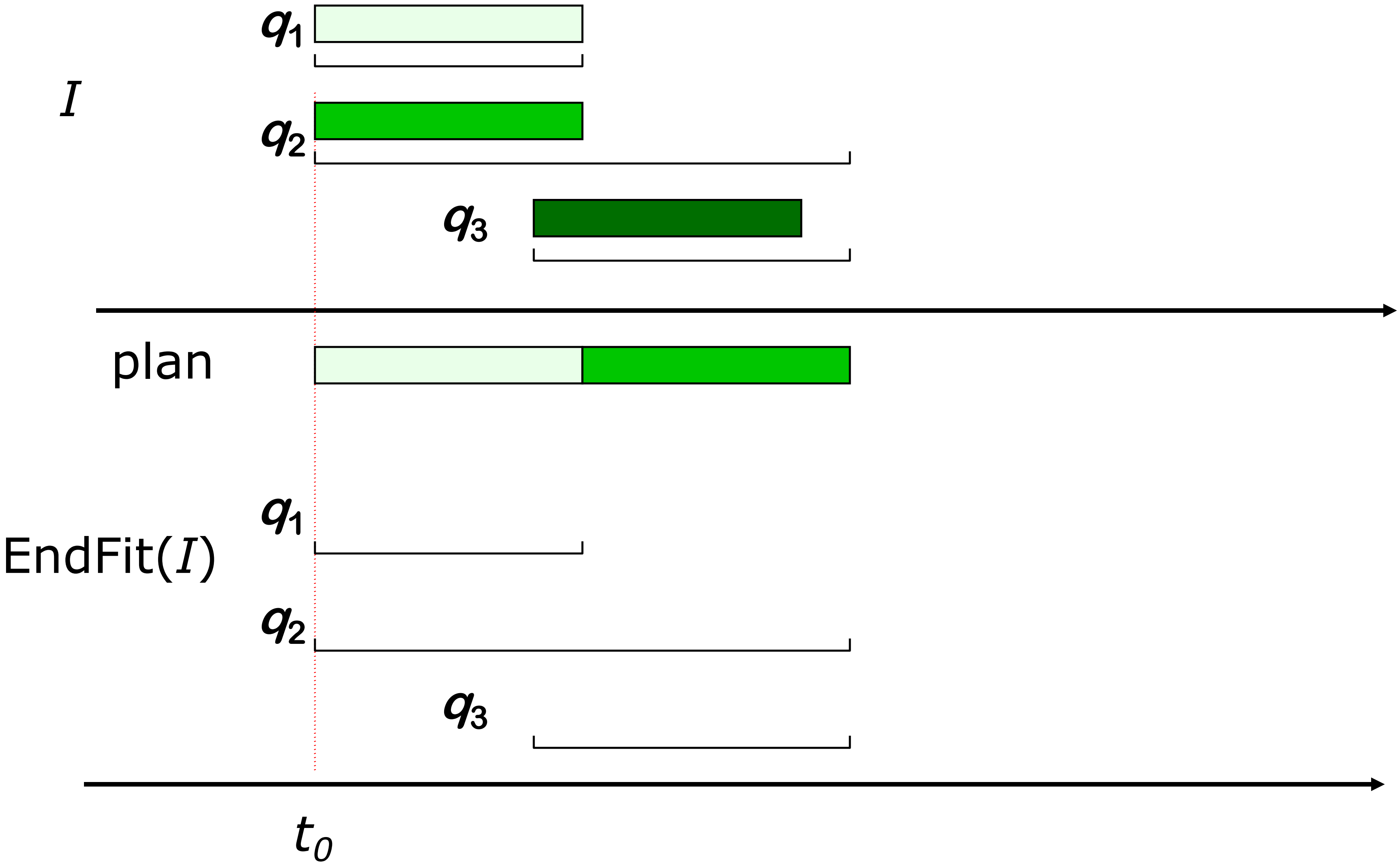
For any instance I and any schedule H for I

$$2\, \mathit{merit}\,(\,\text{FirstFit}\,(I)\,)\ \geqslant\ \mathit{merit}\,(\,H\,).$$

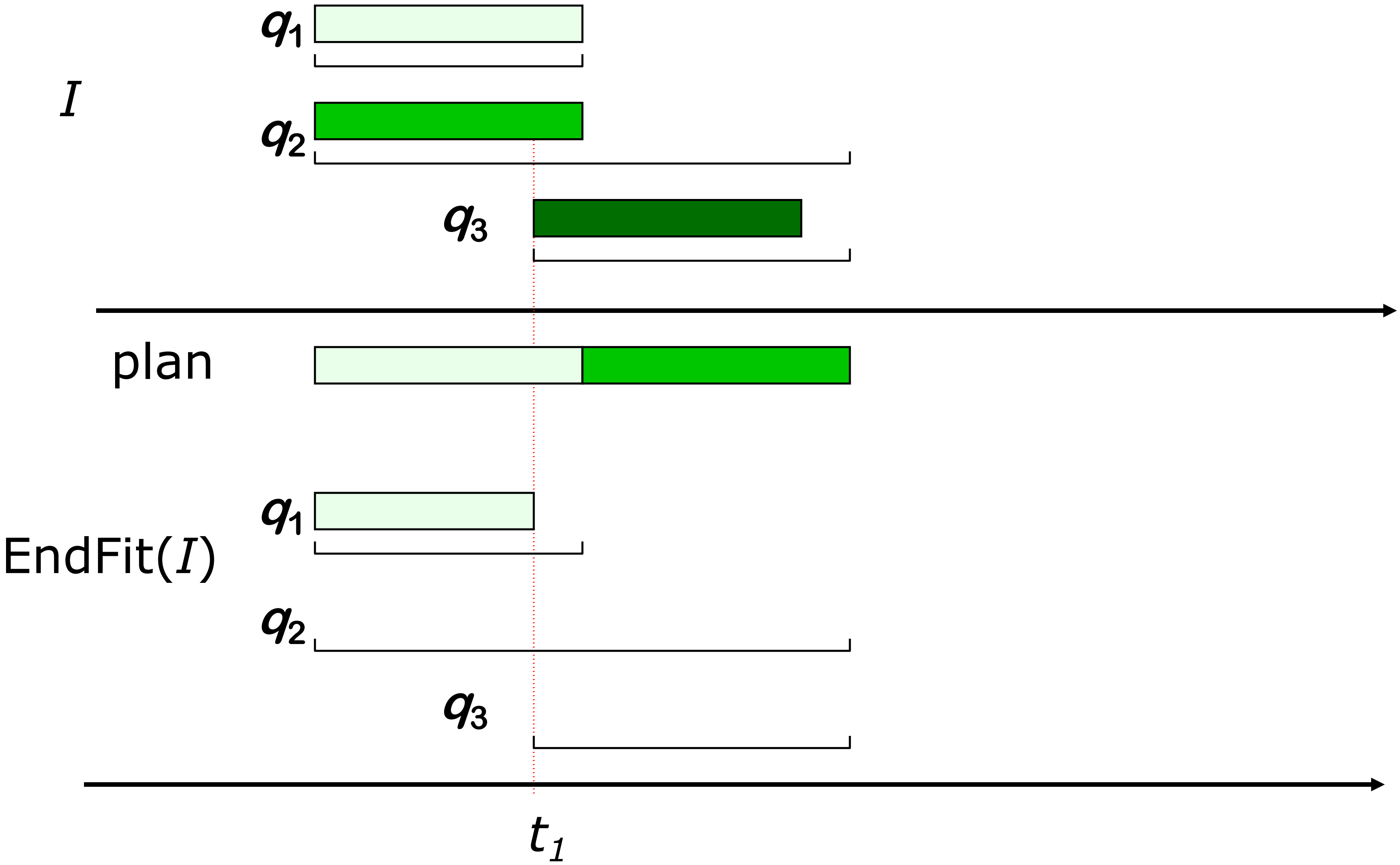
2. EndFit: always serves according to the off-line optimal schedule of the residual requests.



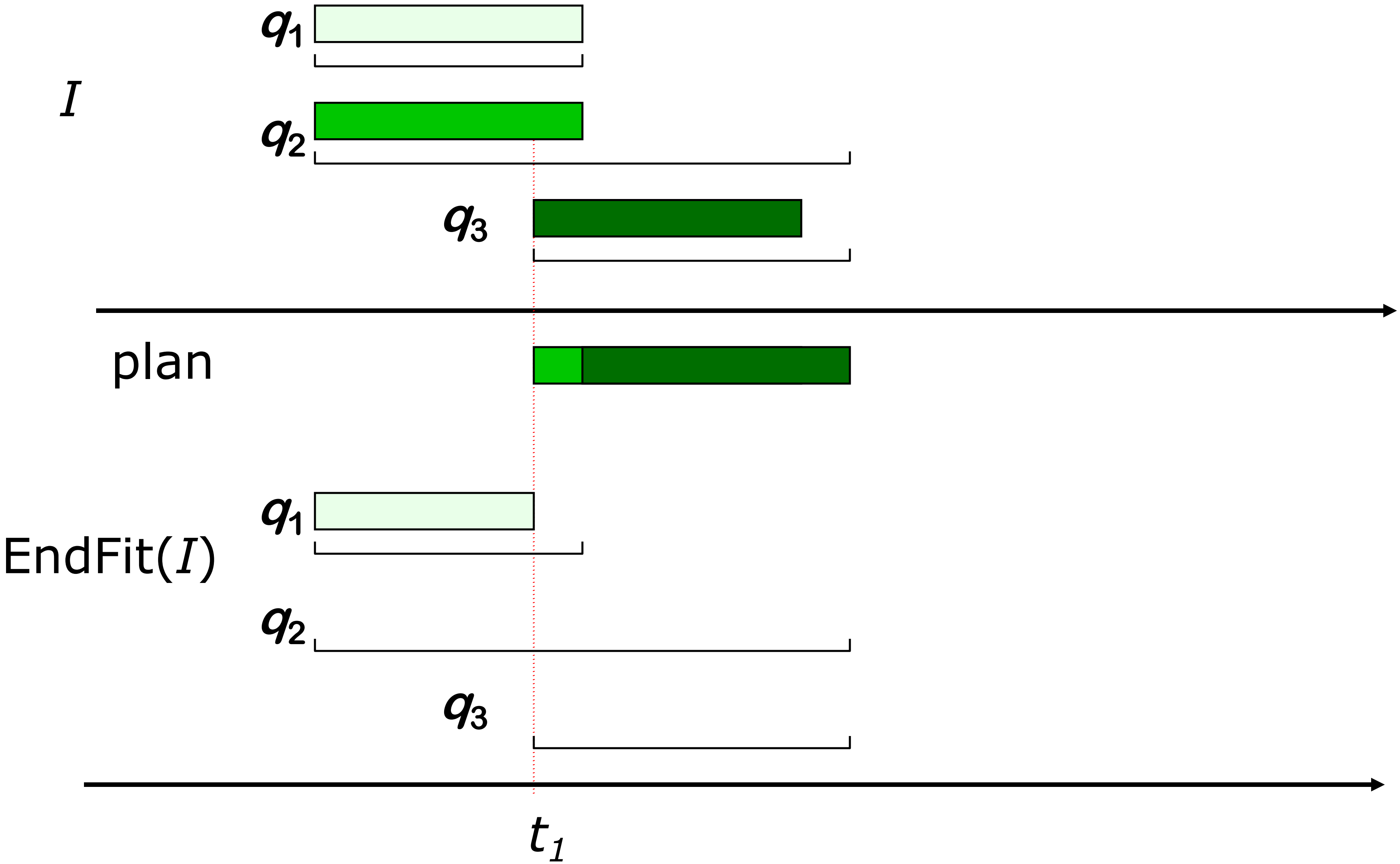
2. EndFit: always serves according to the off-line optimal schedule of the residual requests.



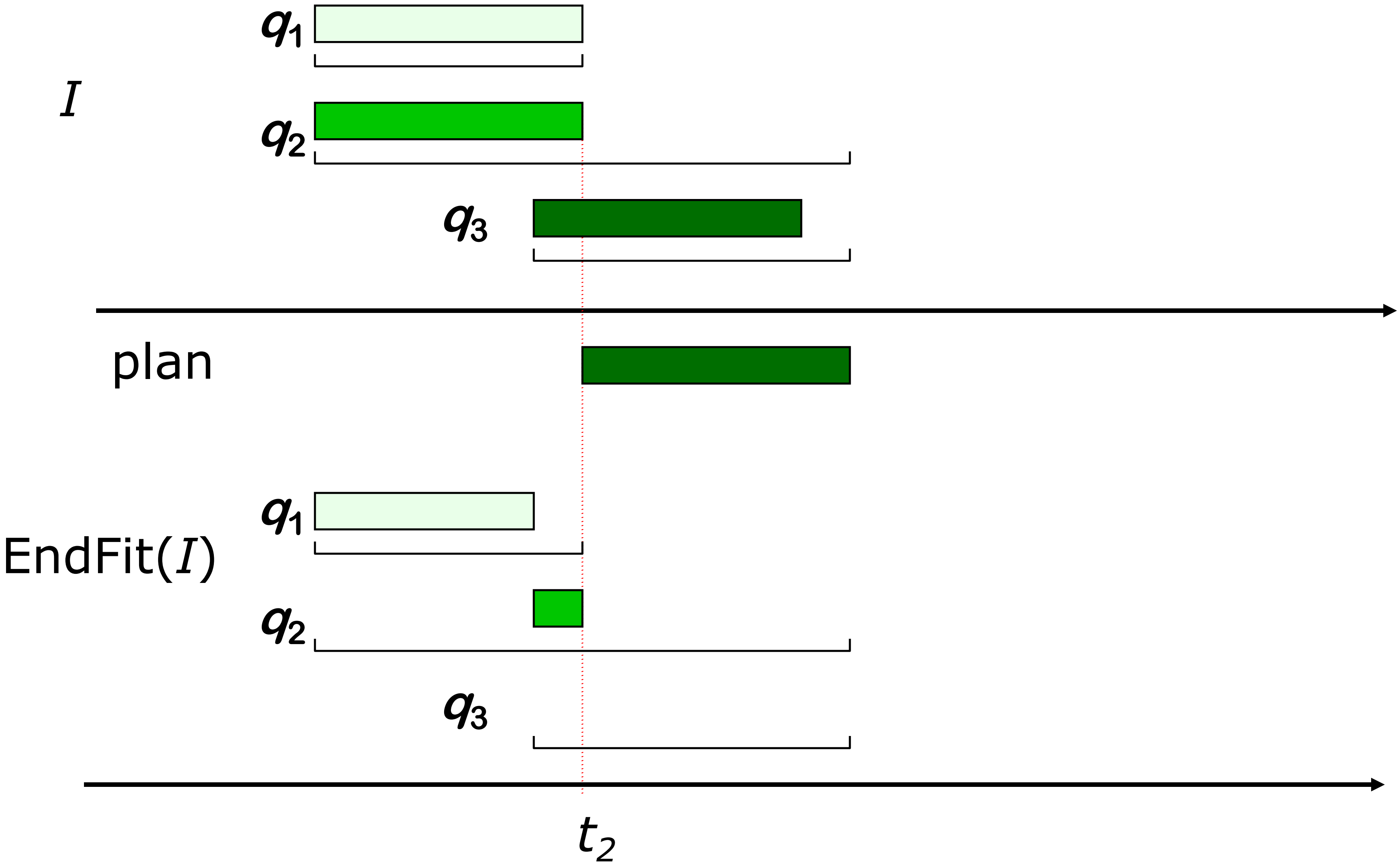
2. EndFit: always serves according to the off-line optimal schedule of the residual requests.



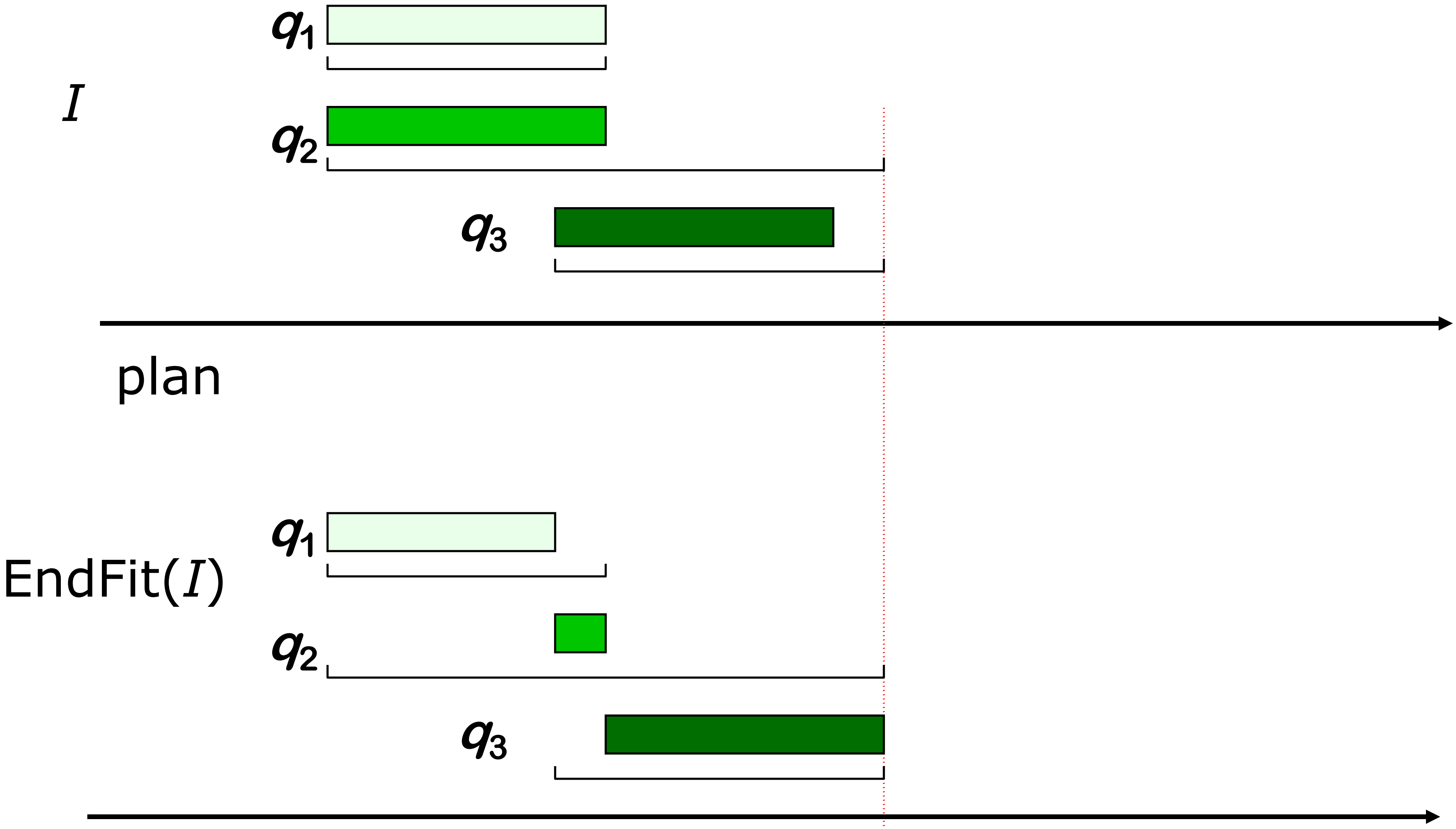
2. EndFit: always serves according to the off-line optimal schedule of the residual requests.



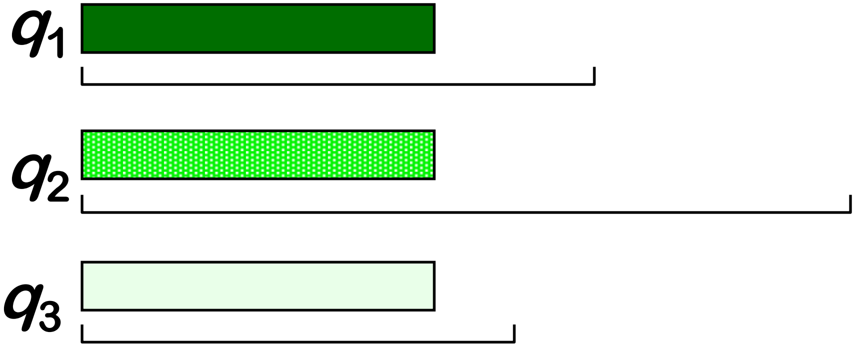
2. EndFit: always serves according to the off-line optimal schedule of the residual requests.



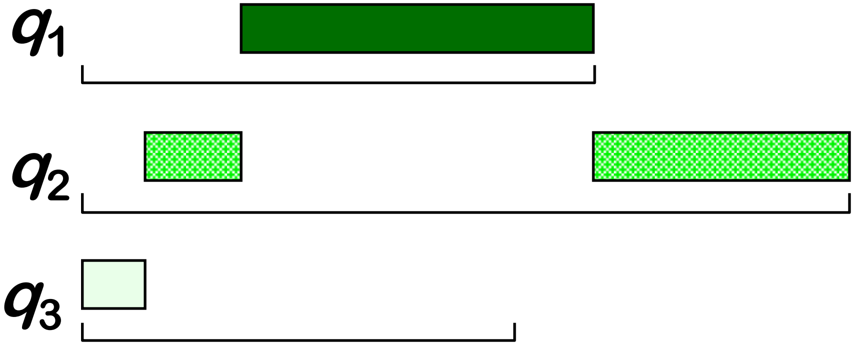
2. EndFit: always serves according to the off-line optimal schedule of the residual requests.



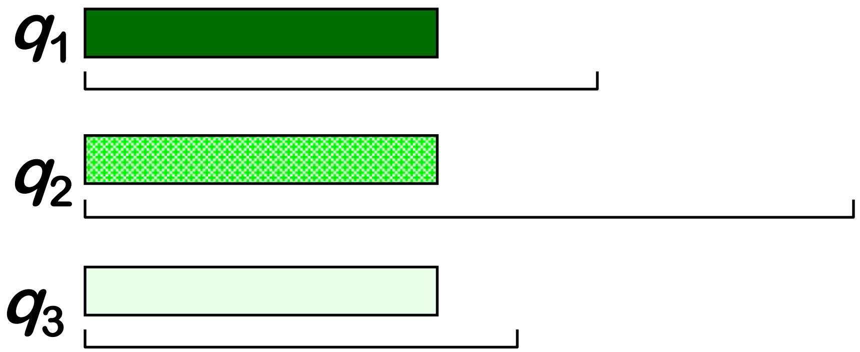
EndFit always delays the service of a heavier request to the latest possible time slot.



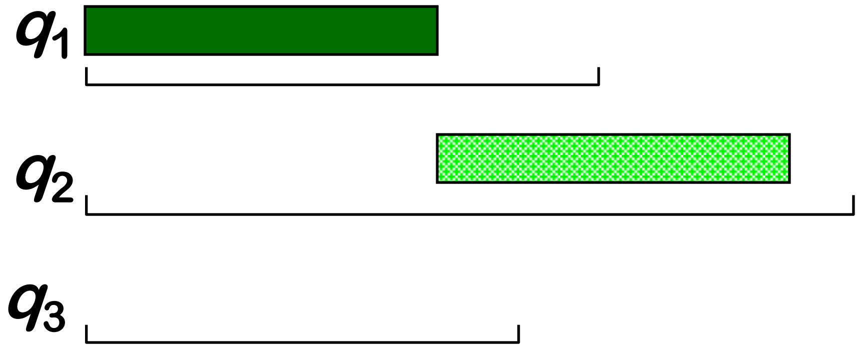
offline optimal schedule



FirstFit always serve a heavier request in the earliest possible time slot.



offline optimal schedule



Theorem 2

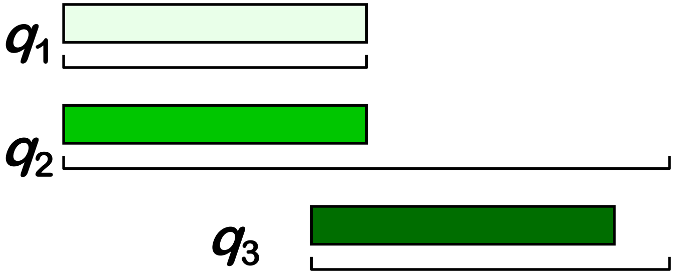
EndFit is 2-competitive.

Lemma

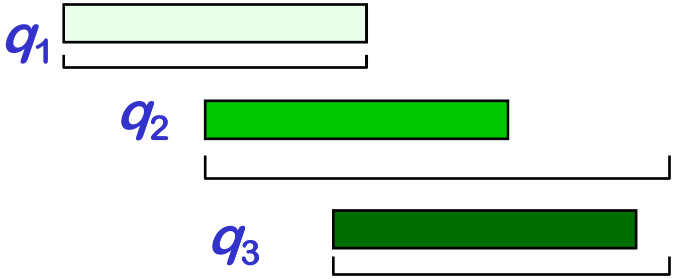
For any instance I

merit(EndFit (I)) ⩾ merit(EndFit(trim(I))).

I

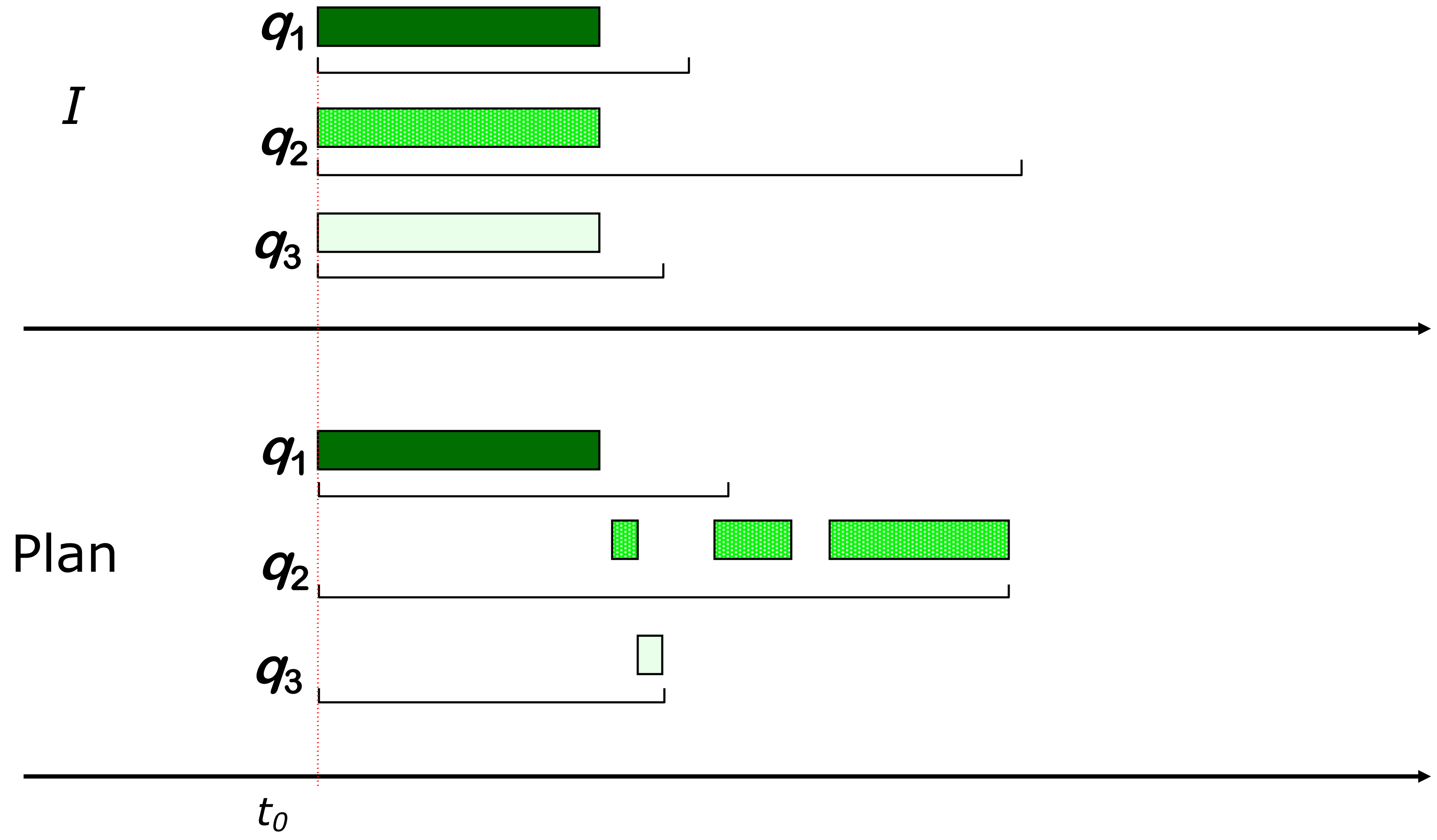


trim(I)



Greedy Schedulers

Computes a plan for the residual requests.
Serves according to the plan until a new request arrive.



EndFit and FirstFit are greedy schedulers.

Theorem 3

Any greedy scheduler is 3-competitive.

We can find a greedy scheduler that is not better than 3-competitive.

With additional constraints, we can show that any greedy scheduler is not better than 2-competitive.

All online scheduler are not better than 1.17-competitive.

FirstEndFit: Toss a fair coin. If the outcome is *head*, then simulates FirstFit. If *tail*, then simulates EndFit.