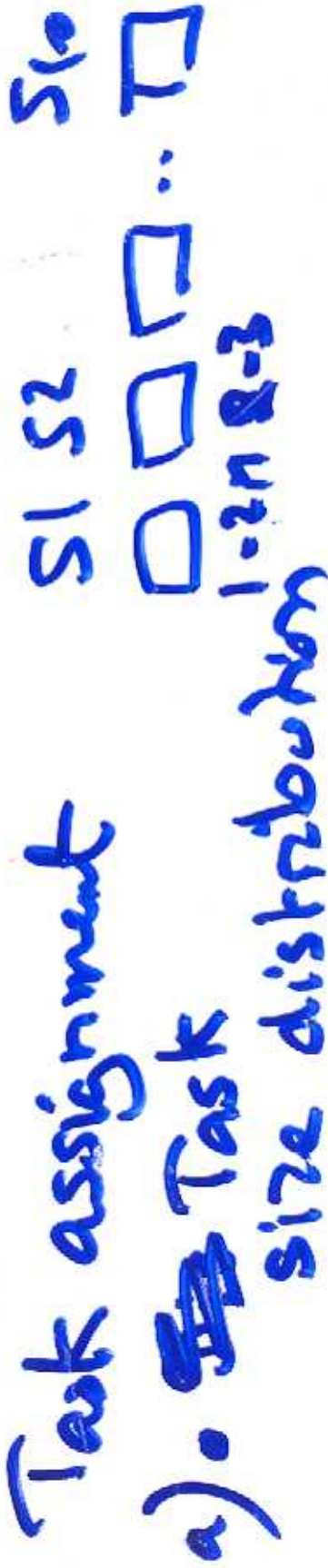


# Load balancing

Load index. (saw some models.)

Ref. lecture notes  
for papers)

Task assignment



know / not  
know task  
size



LLF is optimal (but) load balancing

technique when you have a ~~normal~~ distribution (of task size), exponential.

BUT LLF is not good

when there is high variability

in task size distribution

(e.g. Pareto distribution)

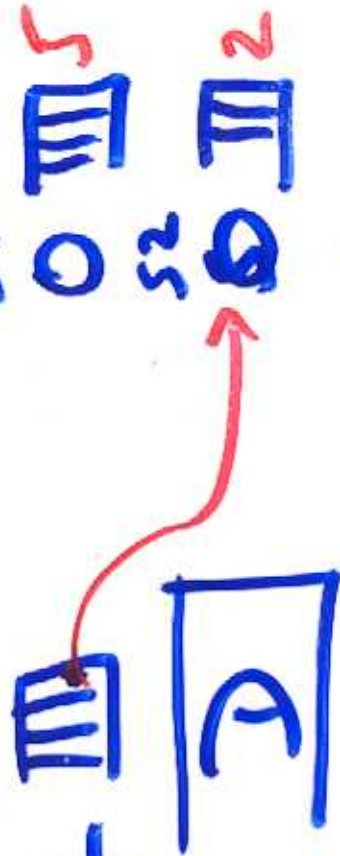
# CENTRAL QUEUE



S4

Least Work Remaining

t1



(least loaded First LWF)

Load = number

of tasks in the queue of a given server.

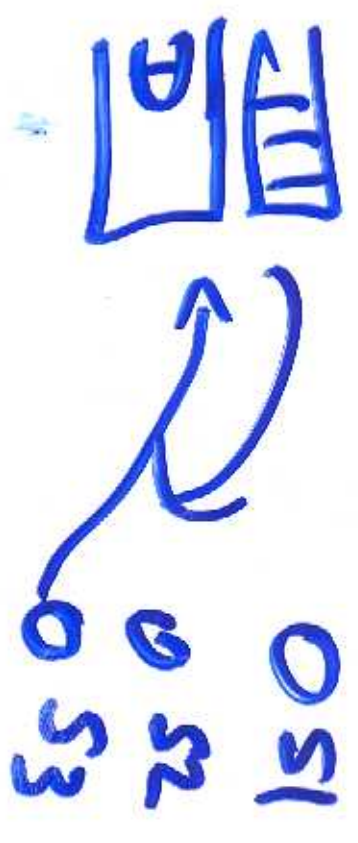
$$L(S2) < L(S1) < L(S3)$$

$$L(S2) < L(S3) < L(S1)$$

$$L(S2) < L(S4) < L(S1)$$

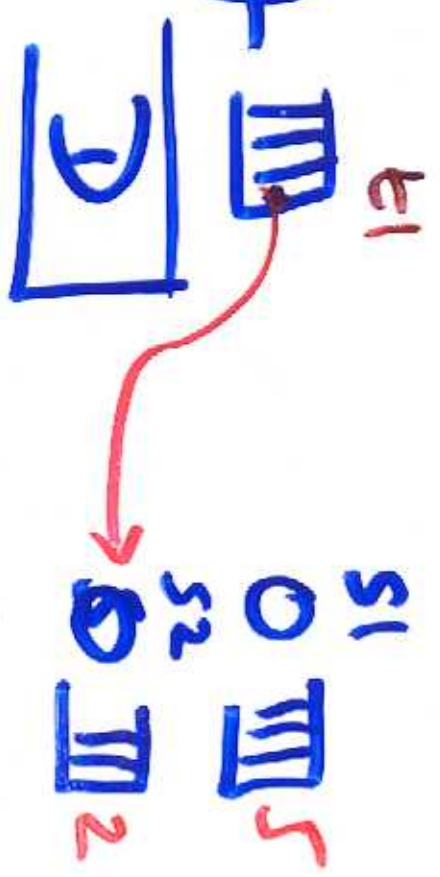
S4

# CENTRAL QUEUE



Last Work Remaining

(Least loaded First LWF)



Load = number

of tasks in the queue of a

given server.

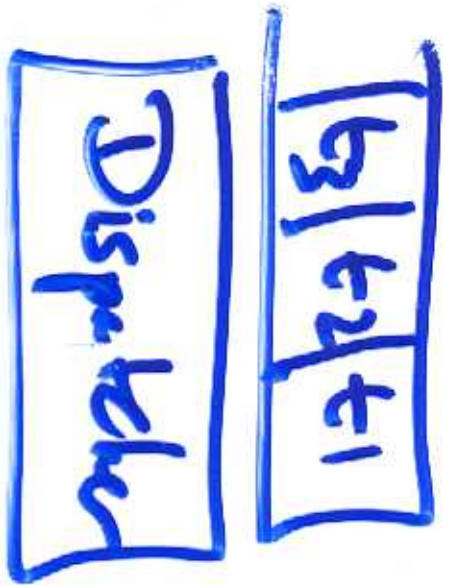
$$L(S_2) < L(S_1)$$

$$L(S_2) < L(S_3)$$

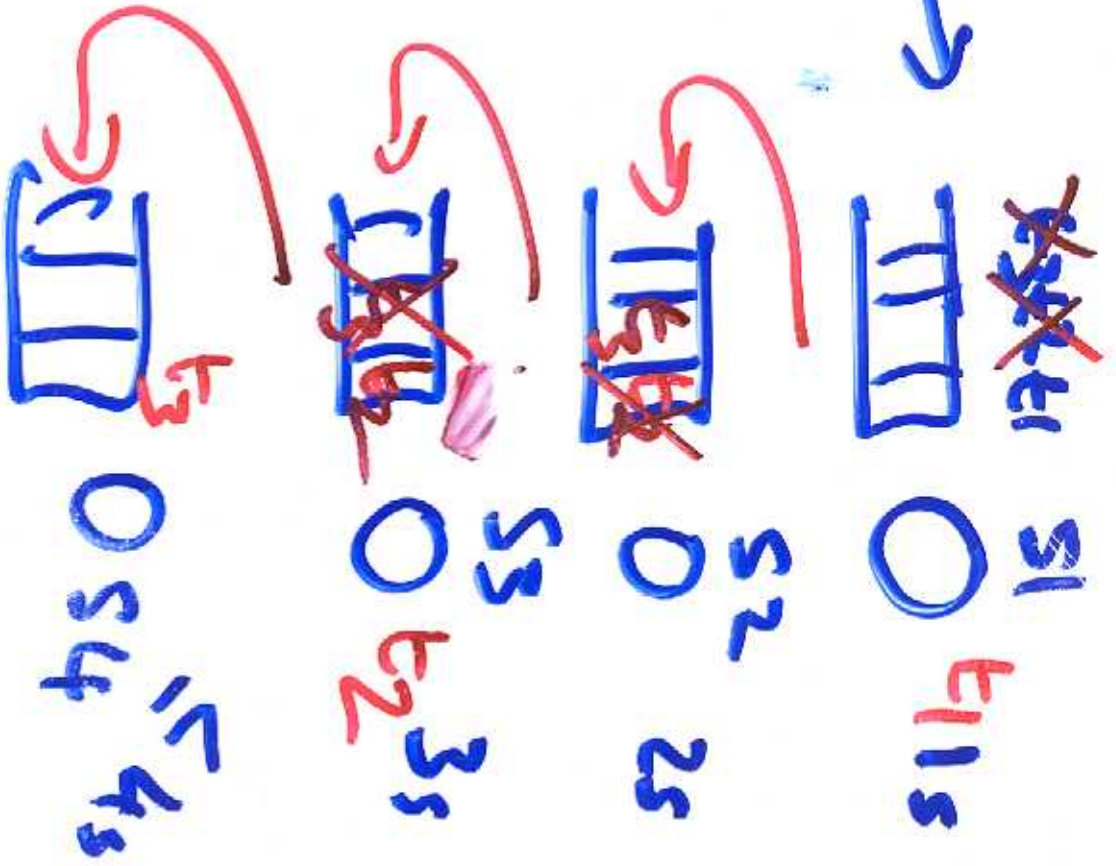
$$L(S_2) < L(S_4)$$



# TACS



$0.5s \leq \text{size}(k_1) < 1$   
 $2.5s = \text{size}(k_2)$   
 $4s = \text{size}(k_3)$



$E(x) =$  task size

$E(x^2) =$  density

⋮

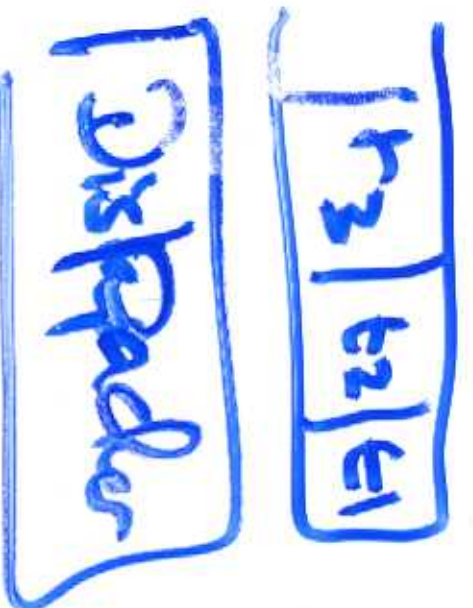
→ need to estimate

• number of tasks  
in each queue

• Size of tasks in  
queue.

⑤-6

# SITAV



$$E(S) = P_1 E(S_1) +$$

$$P_2 E(S_2)$$

buf



Small

cut-off (size)

S1



Large

S2

> 25

$E(S) =$  slow down

6-6

Given a task  $t_1$

$$\text{Slowdown of } t_1 \equiv \frac{\text{waiting } t_1}{\text{size } t_1}$$

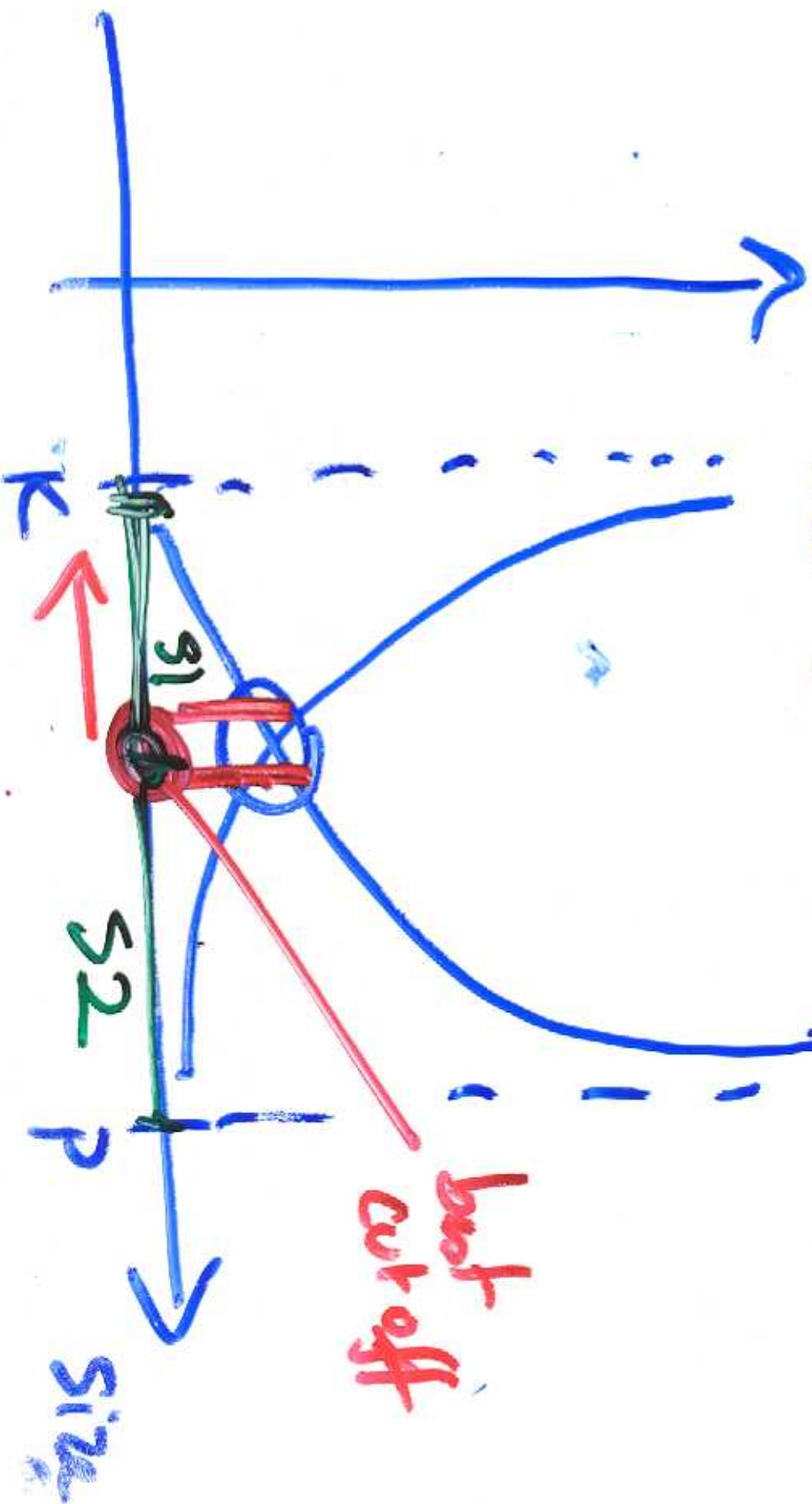
$P_1$  : fraction of tasks that  
go to  $S_1$

$P_2$  : fraction of tasks that  
go to  $S_2$

$$E(S) = P_1 E(S_1) + E(S_2) P_2$$

$E(S_2)$

$E(S_1)$



cut off  $N k$

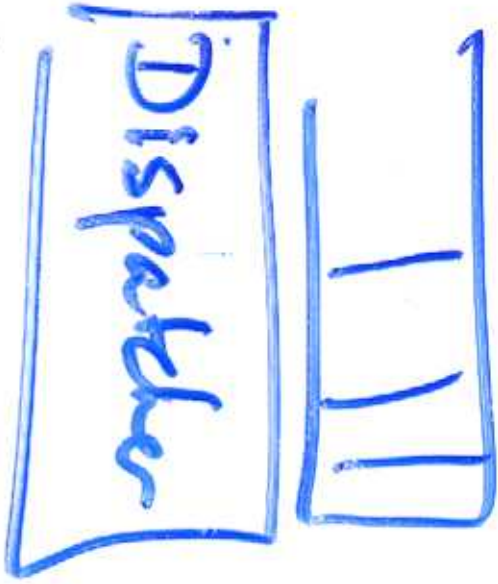
cut off

best cut off

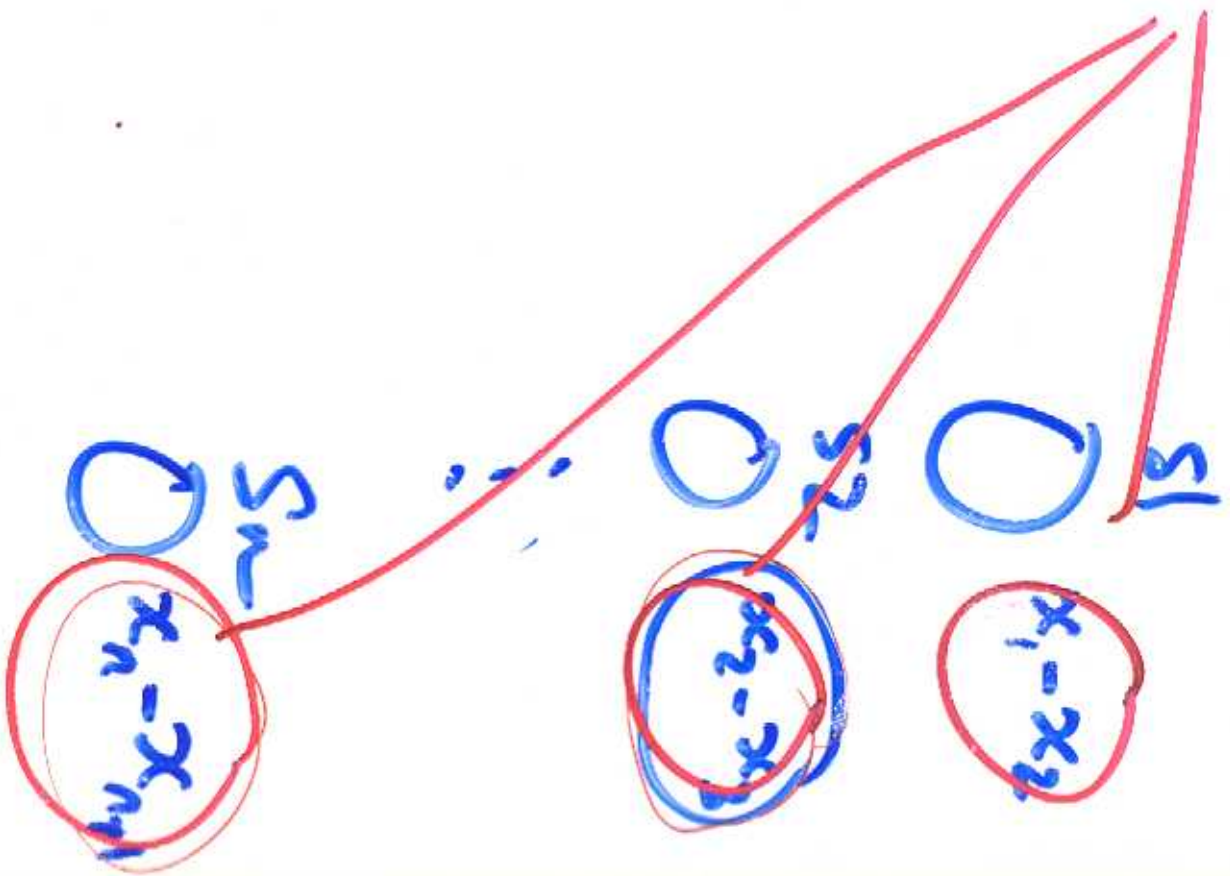
size

⑧-6

SIN-E

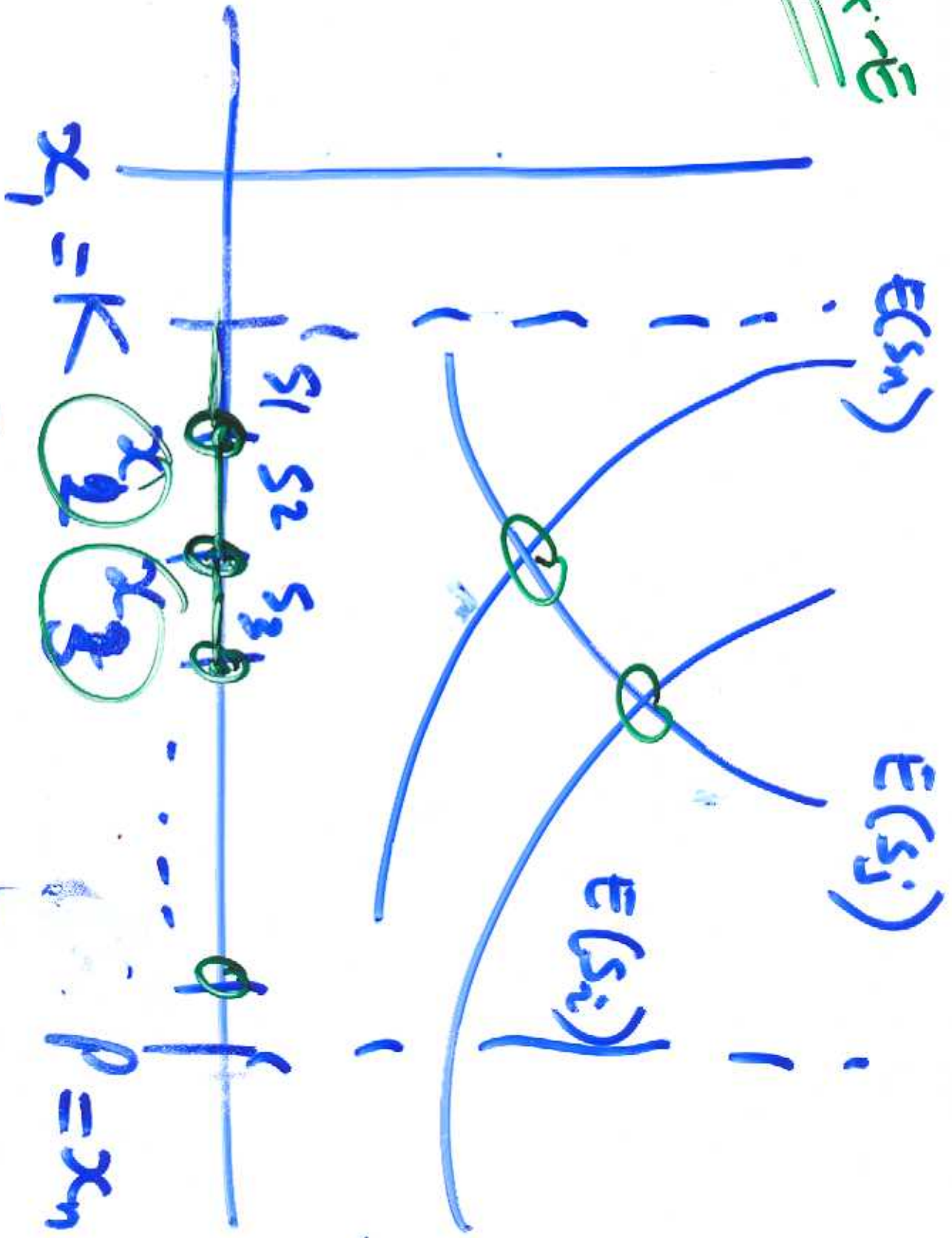


fast size  
range



5-6

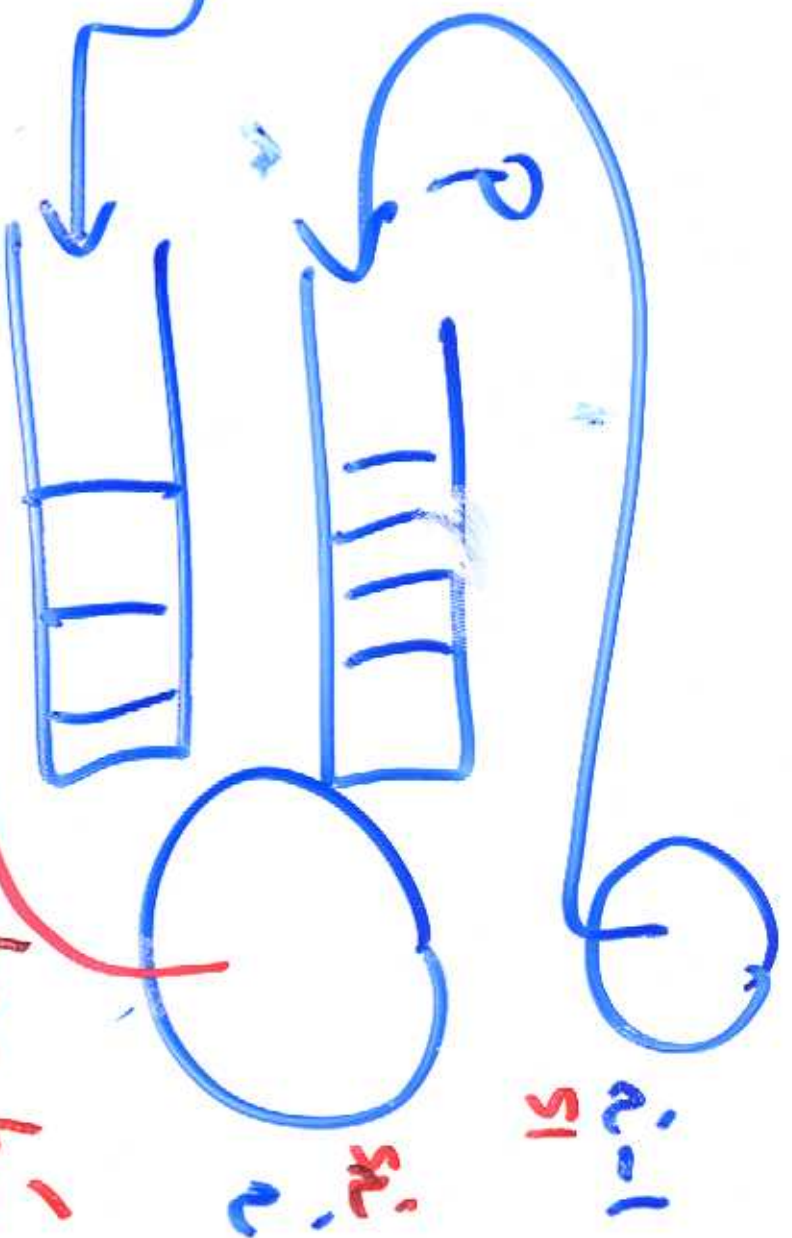
SICK-E



FIND THE BEST CUTOFF  
for  $s_1, s_2, s_3, \dots, s_n$

⑩-6

Dispatcher



$h_{i+1} - h_i =$  visit the priority queue of  $n$   
 $h_i - h_{i-1} =$  processed by server  $i$   
 $h_{i+1} - h_i =$  number of tasks that are moved to  $s_{i+1}$