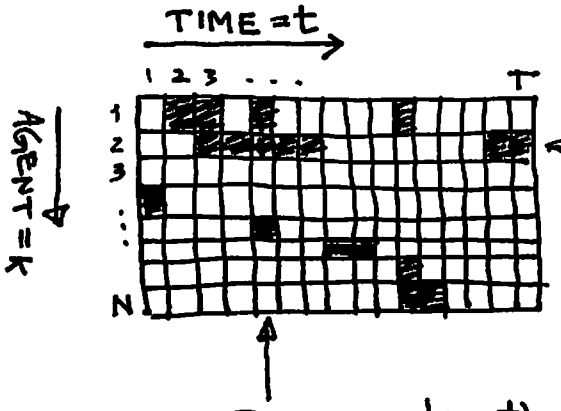
QUEUING (LITTLE'S LAW)
MINIMAL PATH ALGORITHMS (DIJKSTRA)

6.02/Nov. 16, 2010



$\sum_t$ = total time in the system for agent k

$\sum_k$ = queue length at time t

"agent k present at time t"

$$\sum_t \sum_k p(k,t) = \sum_k \sum_t p(k,t)$$

$\sum_k p(k,t)$: queue (t) ("Length")

$\sum_t p(k,t)$: time (k) ("in the system")

$\sum_t$ queue (t) = T · avg(queue)

$\sum_k$ time (k) = N · avg(time)

$$\text{avg(queue)} = \frac{N}{T} \cdot \text{avg (time)}$$

$\frac{N}{T}$ ← "agent rate"

LITTLE'S LAW

Ex. 1 "Little's Little Gym" statistics: every day (24 hrs)

$$\boxed{\text{avg. \# of minutes spent in the gym by a customer}} = 2 + \boxed{\text{avg. \# of customers at a time}}^3$$

(a) What is the maximal # of customers that the gym can have?

(b) AT THIS "MAXIMAL CAPACITY", WHAT IS THE AVERAGE # OF MINUTES A CUSTOMER SPENDS IN THE GYM?

SOLUTION: TIME UNITS = MINUTES

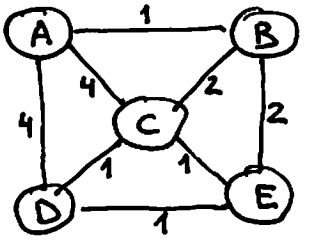
LITTLE'S LAW:

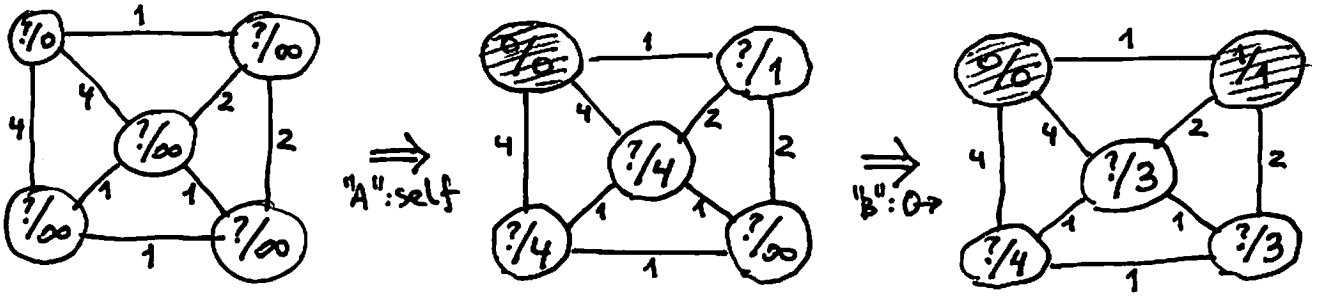
$$\frac{N}{24 \cdot 60} \cdot T_A = Q_A$$

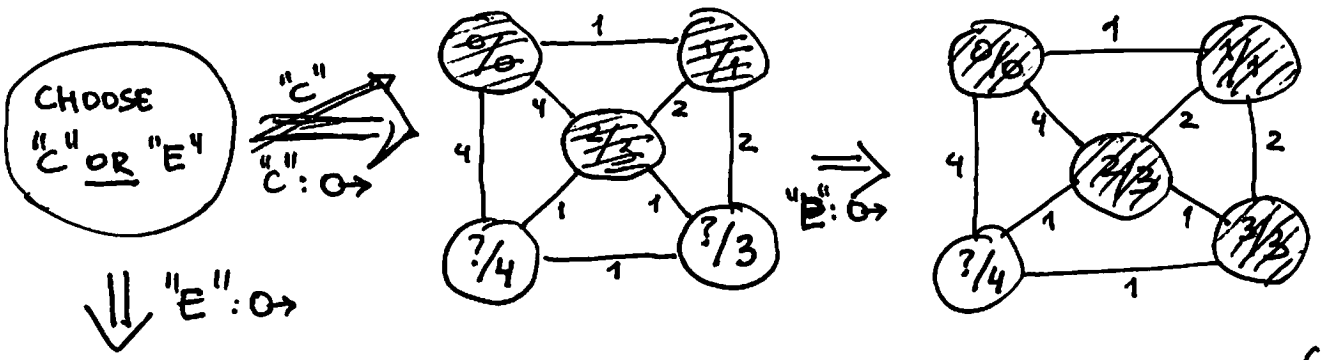
$T_A = 2 + Q_A^3$

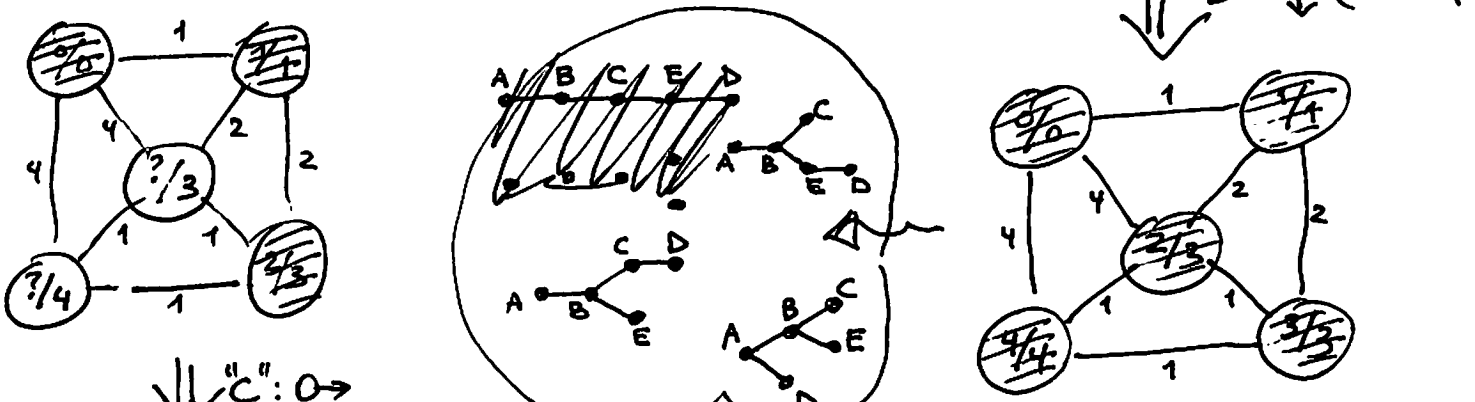
DIJKSTRA'S ALGORITHM

Ex.2 shortest paths from "A" in

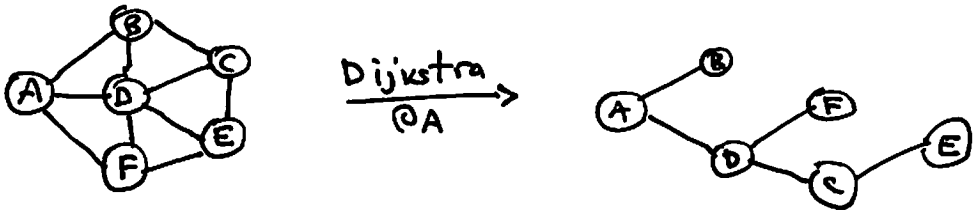






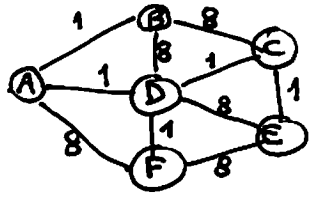


Ex.3 (Lec.18, pr. 5)

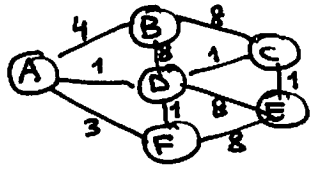


(a) $|AD| + |DF| \leq |AF| \Rightarrow |AD| < |AF| \Rightarrow$ "not AD"

"Af" possible:

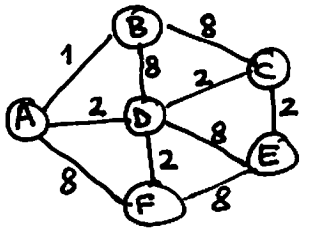


"AB" possible:

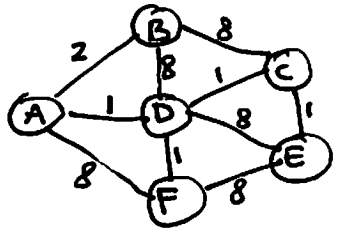


(b) $|AD| < |AF| \Rightarrow$ "not AF"

"AB" possible:



"AD" possible:



When "designing" costs, you can put arbitrary costs onto the optimal path tree, as long as the other links are assigned high enough costs

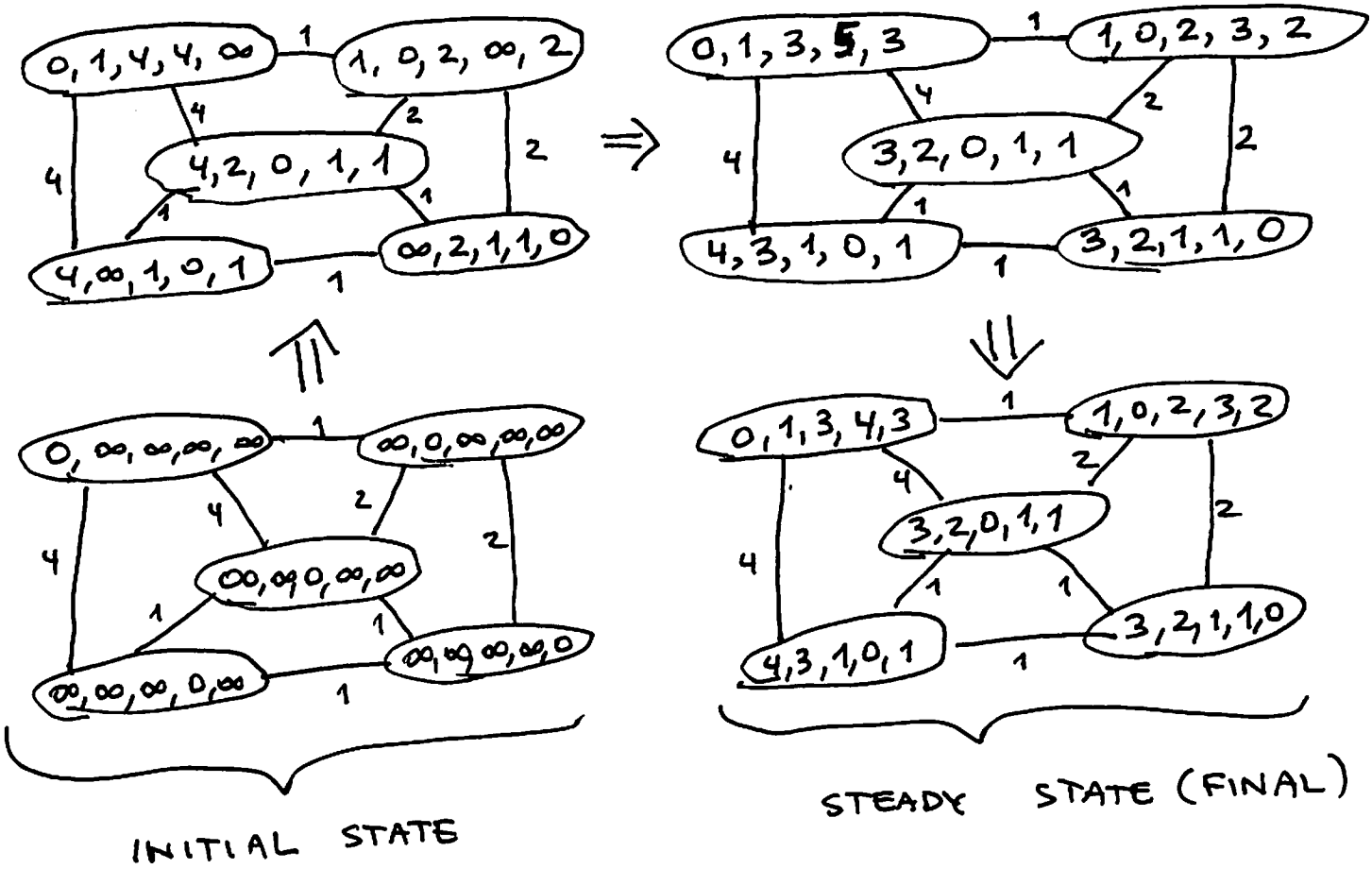
(c) A PART OF THE OPTIMAL TREE IS ALWAYS AN OPTIMAL PATH
BUT IT IS NOT NECESSARILY A PART OF A GIVEN OPTIMAL TREE!

Ex. 3 (continued)

(d) BD, DE – otherwise it's not a "tree"

BELLMAN/FORD ALGORITHM

Ex. 4 (same network as in Ex. 2, different algorithm)



INITIAL STATE

STEADY STATE (FINAL)