Hard Cases:

1. Maintenance:
   - APX-hard
   - 2-APX algorithm

2. Fixed Costs: NP-hard

$x_{ij} \Rightarrow c_{ij} \cdot x_{ij} + d_{ij}$

APX-**hardness**

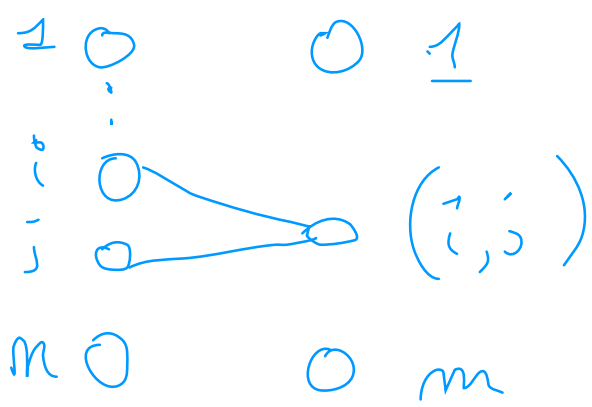
Vertex Cover $\leq_{AP}$ RSR-Maintenance

(CUBIC GRAPHS)

⇓

NP-HARDNESS PROOF BECOMES APX-PRESERVING REDUCTION

Main Ideas

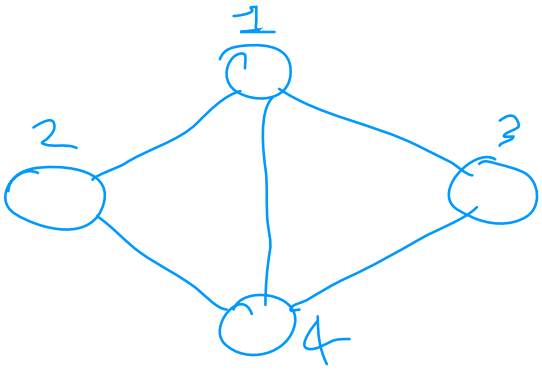
1) 

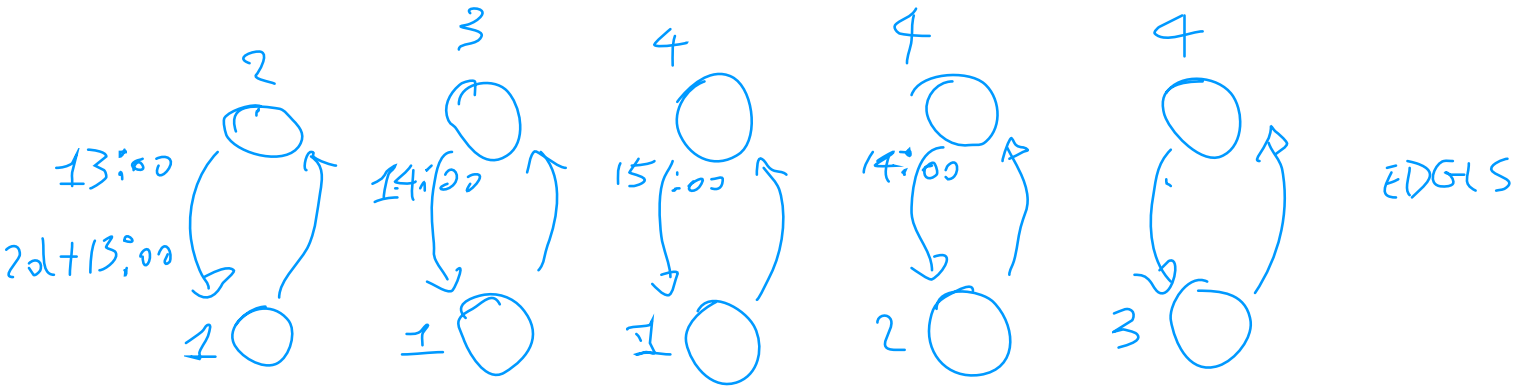
2) VERTEX COVER $\leq_P$ RSR-M

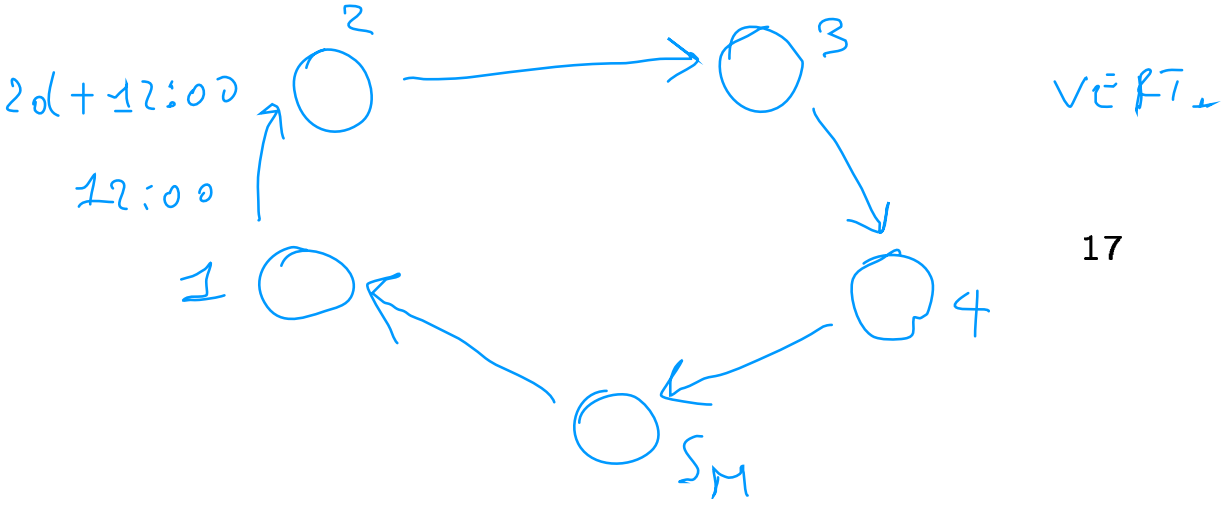
V.C. OF SIZE k $\Leftrightarrow$ $C_0 + k$

3) G CUBIC $\Rightarrow$ IGNORE C_0

Main Ideas 2


1
2
3
4


2
13:00
2d+13:00
1
3
14:00
1
4
15:00
1
4
14:00
2
4
3
EDGES


2
2d+12:00
12:00
1
3
4
SM
VERT.

Main Ideas 3

1) IGNORE MAINTENANCE ⇒ TRIVIAL SOLUTION OF COST C_0

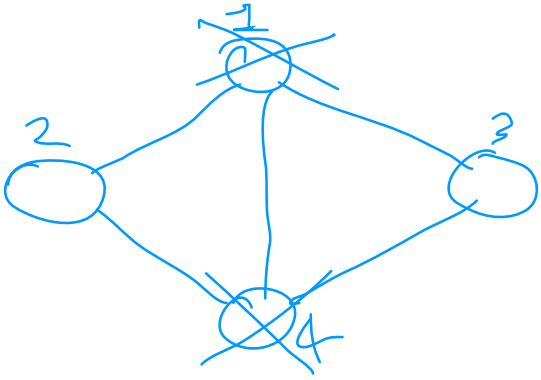
2) MAINTENANCE ⇒ ONLY ONE CYCLE

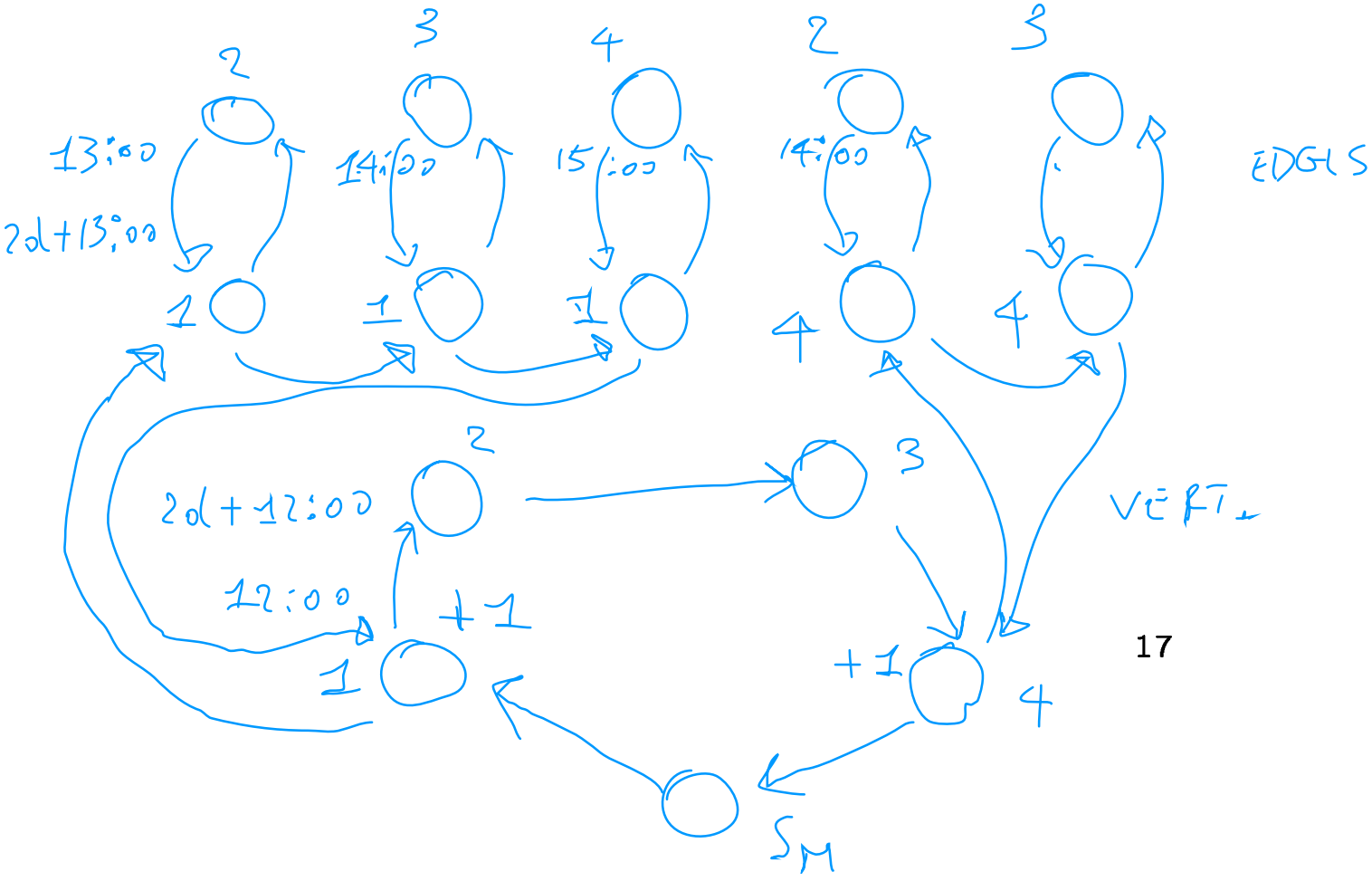
3) EVERY SINGLE CHANGE ⇒ ONE EXTRA TRAIN

4) # OF CHANGES = SIZE OF VERTEX COVER

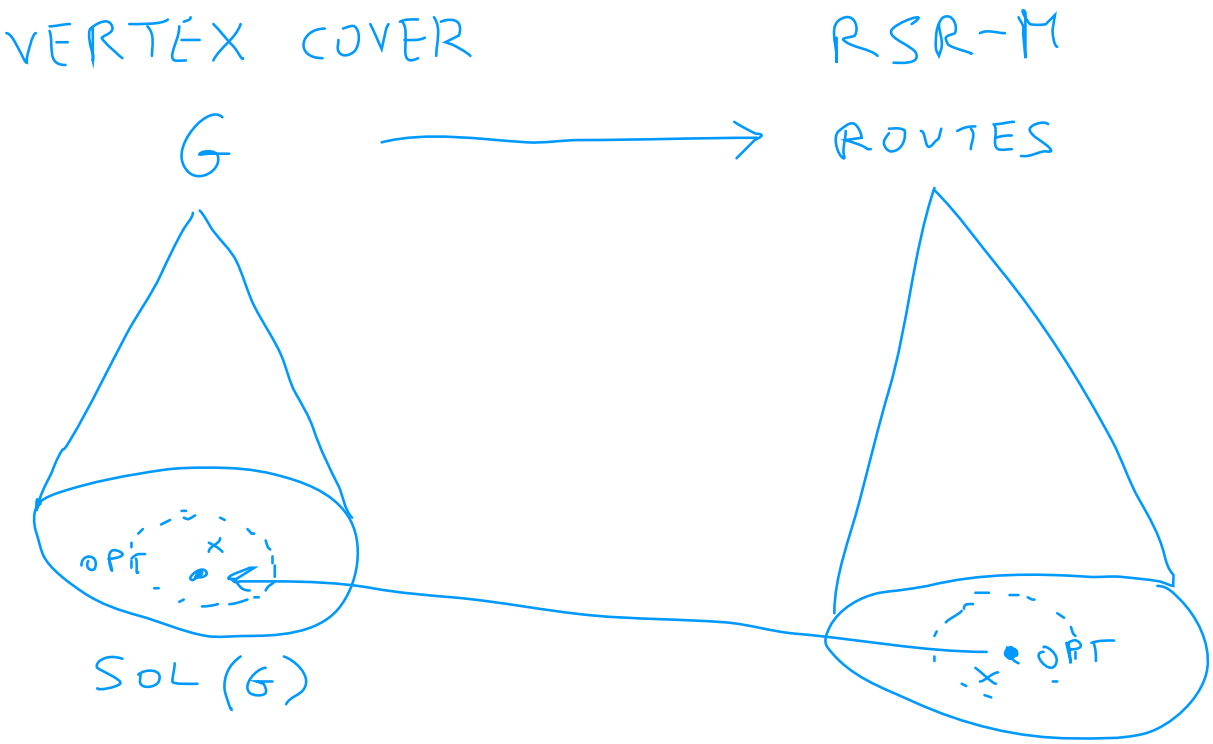
$K \Leftrightarrow$ ⇒ EASY ⇐

Main Ideas


1
2
3
4


2
3
4
2
3
13:00
14:00
15:00
14:00
EDGES
2d+13:00
1
1
1
4
4
2
3
2d+12:00
VERT.
12:00
+1
+1
1
4
S_M

APX. REDUCTION



$$k \iff C_0 + k$$

$$\frac{k_{APX}}{k_{OPT}} \iff \frac{C_0 + k_{APX}}{C_0 + k_{OPT}}$$

$$r = \frac{C_0 + k_{APX}}{C_0 + k_{OPT}} = 1 + \underbrace{\frac{k_{APX} - k_{OPT}}{C_0 + k_{OPT}}}_{\varepsilon}$$

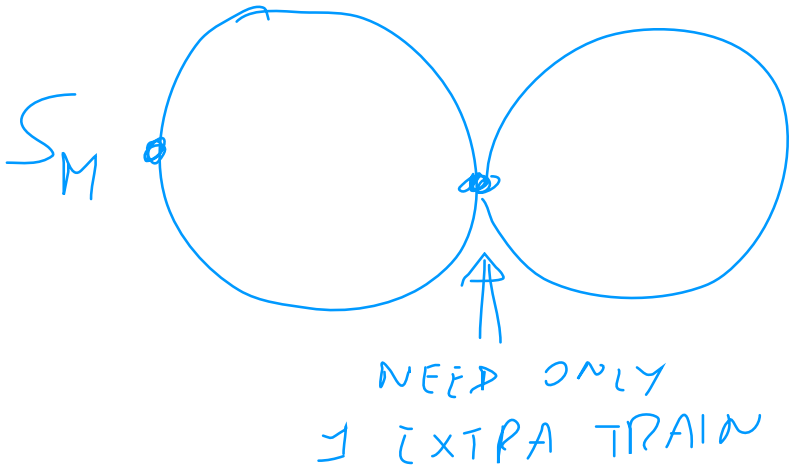
$$\left.\begin{array}{l} C_0 = 2m + 4m + 2 \\ m \geq n - 1 \\ k_{OPT} \geq m/3 \\ \text{(CUBIC GRAPH)} \end{array}\right\} \Rightarrow \begin{array}{c} C_0 \approx \alpha \cdot k_{OPT} \\ \Downarrow \\ \dfrac{k_{APX} - k_{OPT}}{(\alpha+1)\, k_{OPT}} \end{array}$$

$$r = 1 + \underbrace{\frac{k_{APX}}{(\alpha+1)k_{OPT}}}_{\varepsilon'} - \frac{1}{\alpha+1}$$

2-APX algorithm

1) IGNORE MAINTENANCE

2) MERGE CYCLES

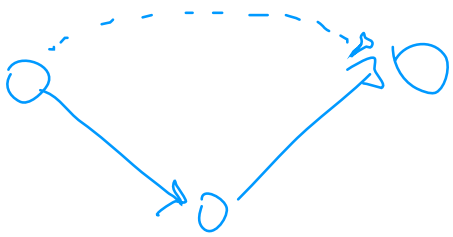


Empty Movements + Maintenance

- Still APX-hard
- 4-APX algorithm

~ SIMMETRIC

— △ INEQUALITY



EMPTY MOVEMENTS + MANTEN.

1) APX HARDNESS:

ADD ALL POSSIBLE EMPTY MOVEMENTS WITH COST 2 DAYS EACH

⇓

TOO EXPENSIVE TO BE USED ⇒
THE REDUCTION WORKS AS BEFORE

2) APX ALGO:

- MERGE CYCLES WITH A COMMON STATION
- MERGE CYCLES USING EMPTY MOVEMENTS

NO MATTER HOW LONG THESE EMPTY MOVEMENTS ARE, TWO EXTRA TRAINS ARE ENOUGH

Fixed Costs

For example:

- Crew
- Locomotive
- Occupying tracks

(Some) Open Questions

- Station capacity, station topology, length of the tracks

- Train types: locomotives, cars

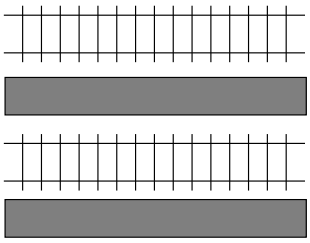
Station Capacity

Problem: We cannot send too many trains into a station!!

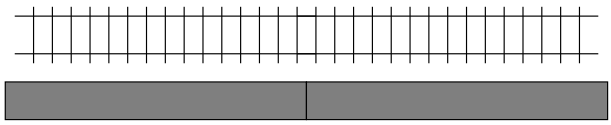
Solution: Count how many train units can get into the station.

Station Topology (Platforms)

Station A



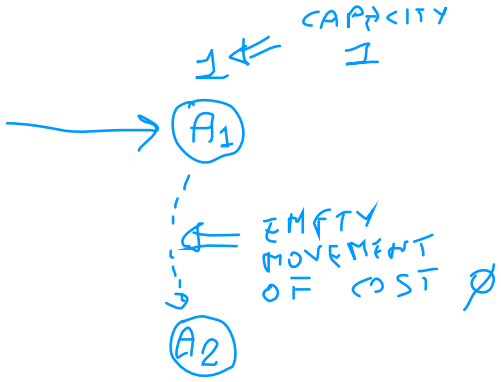
Station B



THESE TWO STATIONS ARE MODELLED IN THE SAME WAY ⇒ WRONG!!

STATION A

2 TRACKS
OF UNIT LENGTH



EX: