Traveling Repairman Problem with Multiple Trucks in Multiple Bases

Chen Ma

Jan. 30, 2015

Scenario

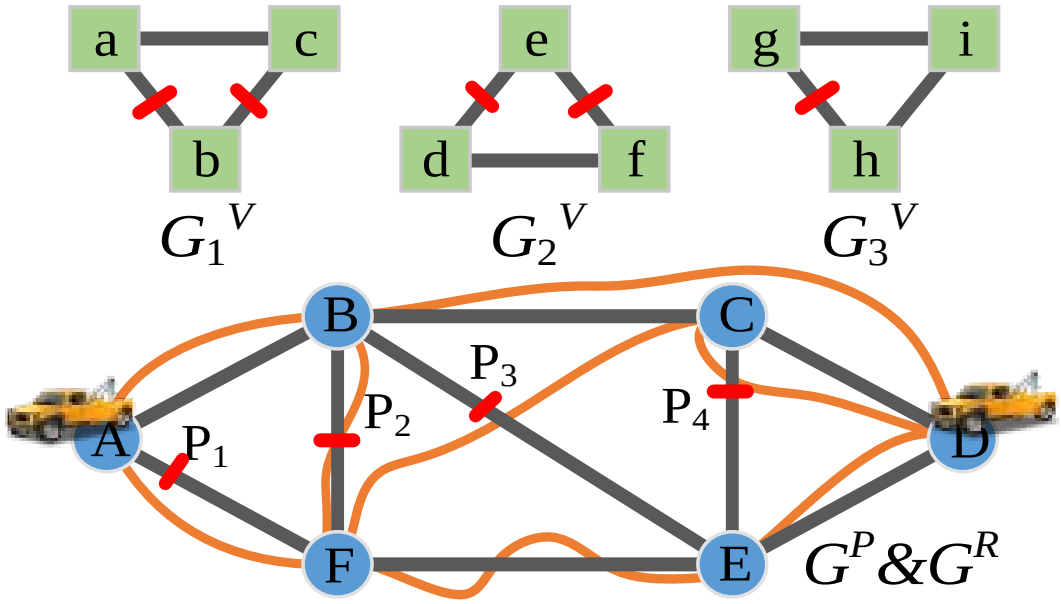


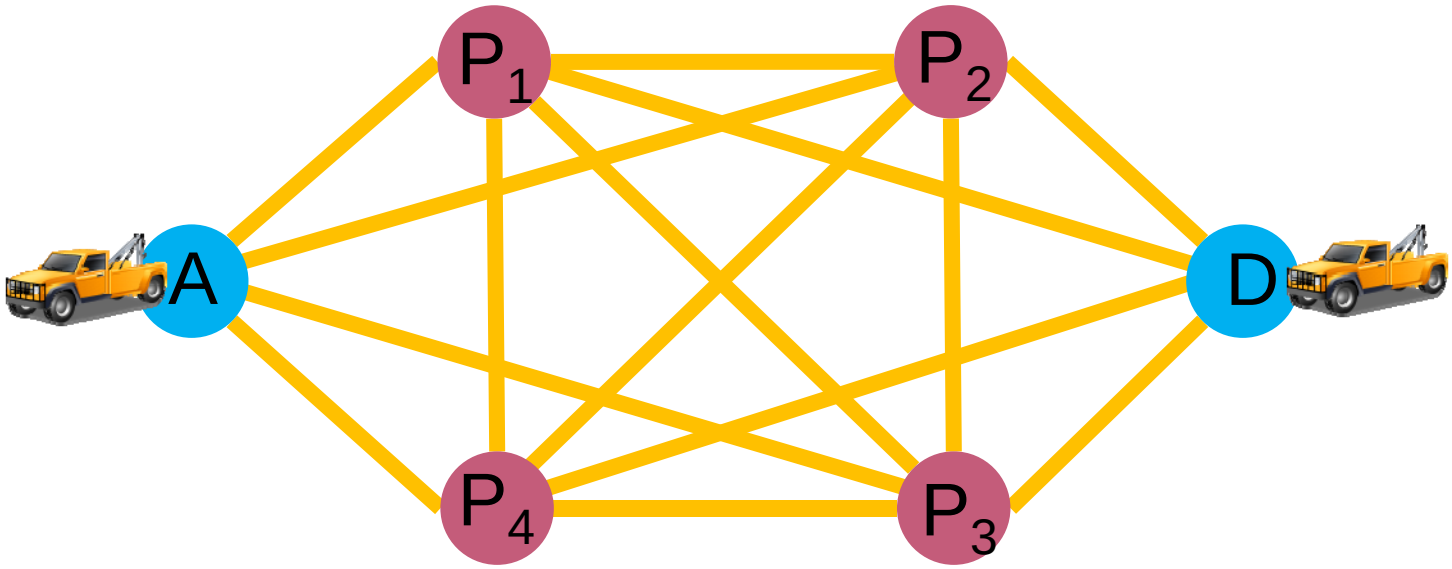
TABLE I
VIRTUAL MAPPING OF ILLUSTRATION

Virtual Networks	Virtual Link	Physical Link
G_1^V	a-b	F-B
	b-c	B-E
	c-a	E-F
G_2^V	d-e	B-E
	e-f	E-C
	f-d	C-B
G_3^V	g-h	C-E
	h-i	E-D
	i-g	D-C

Pre-conditions:

- Repairmen know the exactly location and type of each failure, which means they know the traveling time between any two failures and repair time of each failure.
- At least 1 truck in a base, and at least 1 base in the network.
- Repair time of each truck is different.

Auxiliary Graph

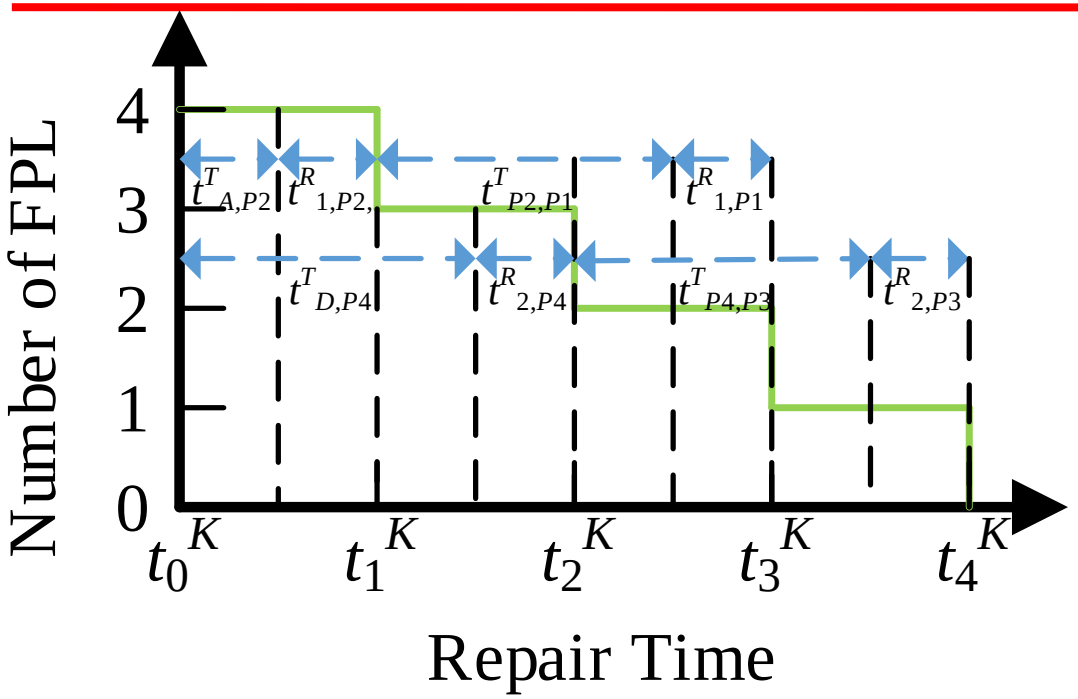


- Weight of each link is the traveling time.

Problem Statement

- Input:
- G^P: Physical topology.
- G^A: Auxiliary graph.
- M^{AV}: Mapping of G^P and G^A.
- J: Number of trucks in each base.
- V^B: Set of base nodes.
- $T_{J,P}$: Repair time of each truck for each failure.
- Output:
- Optimal repair schedule for the disaster
- Objective:
- (1) Minimizing the damage of disconnected virtual networks (DVN).
- (2) Minimizing the damage of failed virtual links (FVL).
- (3) Minimizing the damage of failed physical links (FPL).

Repair Process



Repair Sequence:
P_2-P_4-P_1-P_3
Truck 1:
P_2-P_1
Truck 2:
P_4-P_3

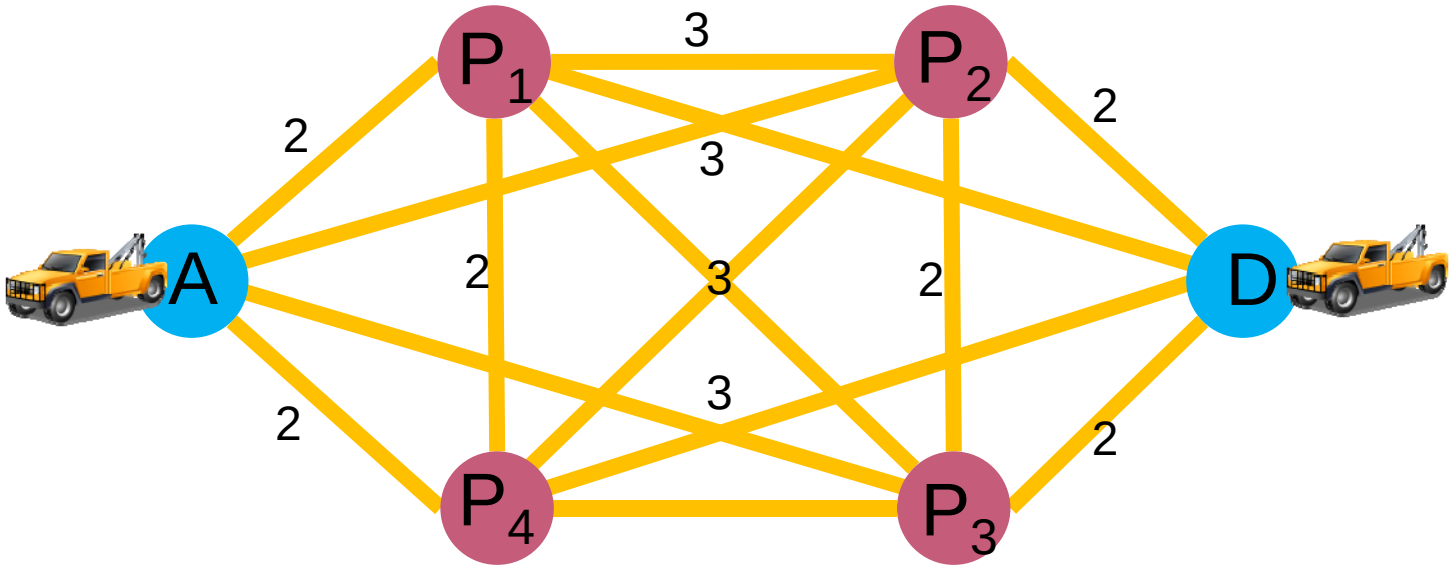
Compared with single truck by complexity (*K* failures):

- Single truck: *K*!
- M trucks:
  - Sequences of failures: *K*!
  - Candidate trucks for each failure: *M*
  - Complexity: $K!*M^K$

ILP

- I have finished the constraints of ILP, and will do the coding in the next month.

Greedy Algorithm----A bad algorithm



- Main idea:
- ➢ Each truck repairs the nearest failure until all the failures repaired
- Shortage:
- ➢ It neglects the relationship of trucks, and it is not a global optimization strategy.

Help!

- 1. Improvement for the problem
- 2. Heuristic algorithms for the problem

Thank you!