In the name of beauty

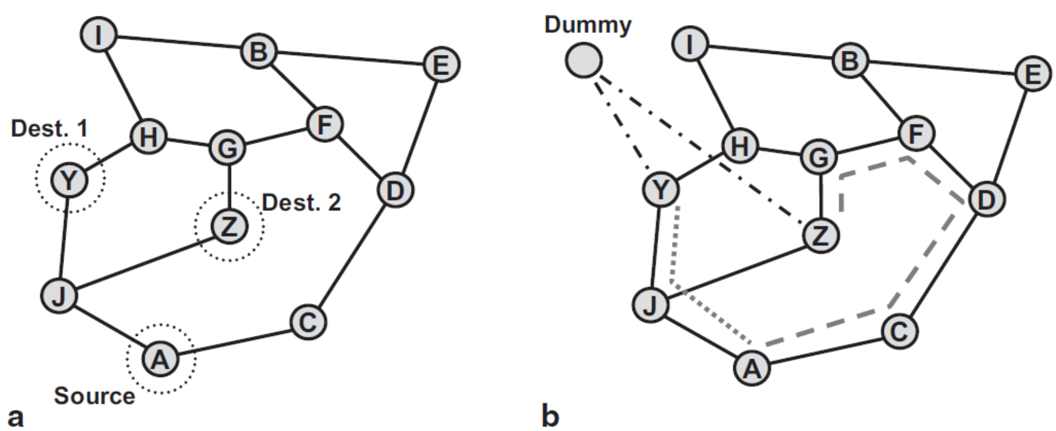
The 6th problem set solution of Optical Networks course

Question 1)

a. Both fixed path and alternative-path routing are performed prior to any demands added to the network. However in fixed-path routing, a single path is chosen from a set of M possible paths between source and destination, which is used in all the demands betwenn that source and destination. In alternative-path routing, more than one path is used (maybe even one for each demand).

Dynamic-path routing is quite different. Paths and resources are allocated once the demand arrives at the network. The order of elapsed time is tremendously less than that in the two previous schemes.

b. It is possible to add a "dummy" node to the physical topology and connect it to the two destinations via links that are assigned a metric of zero. An SPDP algorithm is run between Node A and the dummy node to implicitly generate the desired disjoint paths, as shown by the *dotted line* and the *dashed line*.



Question 2)

a. The following table shows the Dijkstra's algorithm steps:

Dijkstra's algorithm					
Iterations	b	c	d	e	f
a	4,a	5,a	2,a	inf	inf
a,d	4,a	5,a		inf	inf
a,d,b		5,a		1,b	inf
a,d,b,e		2,e			7,e
a,d,b,e,c					3,c

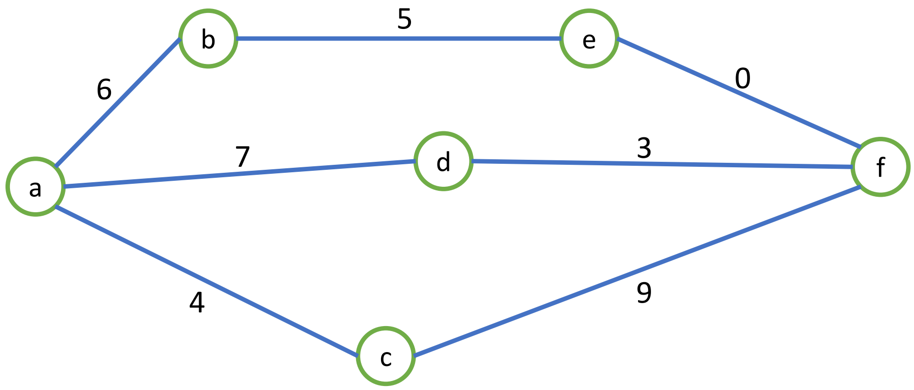
As seen, the shortest path between nodes a and b is not optimum as the path $a-c-e-b$ has less cost.

b. The following table shows the Bellman-Ford's algorithm steps:

Bellman-Ford's algorithm					
	b	c	d	e	f
Iterations	inf	inf	inf	inf	inf
1	4,a	5,a	2,a	inf	inf
2	4,a	5,a	2,a	1,b	6,c
3	4,a	2,e	2,a	1,b	6,c
4	4,a	2,e	2,a	1,b	3,c
5	4,a	2,e	2,a	1,b	3,c

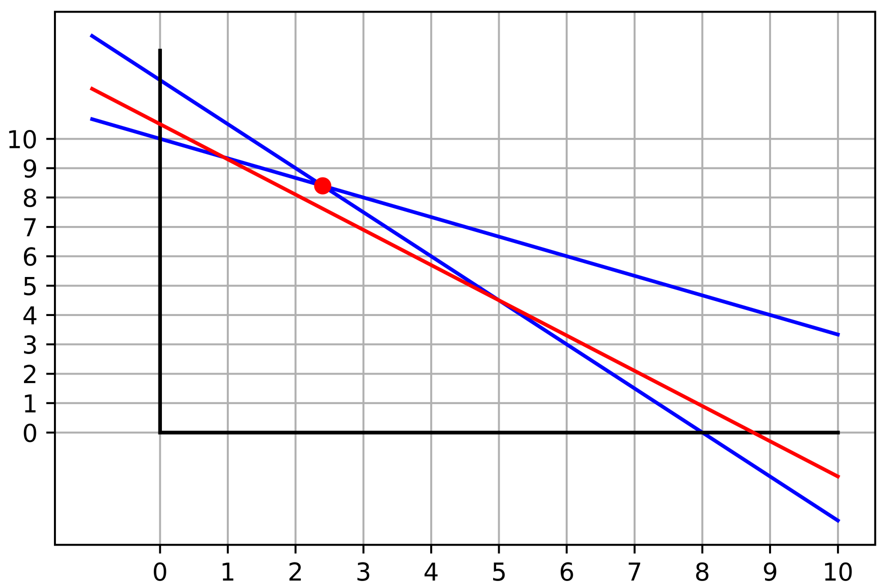
Even still, the shortest path between nodes a and b is not optimum since the Bellman-Ford's algorithm does not yield to correct answer **when there is a negative cycle in the network.** The negative cycle in this case is the cycle $b-e-b$, as once a packet ariiving there, can be bounced back and forth between these two nodes two reduce the path cost to any arbitrarily low value.

c. The minimum-cost path is $a-b-e-f$, however by elevating the link costs by 2, we have the following altered version of the network:



with the new shortest path being $a-d-f$, which is actually incorrect.

Question 3)
We first sketch the feasible region of the optimization problem: Considering



the gradient vector of the objective function, the red-marked point (where the two linear constraints meet), is the optimum solution to the optimization problem **when the variables are not constrained to be integers.** The point is $(2.4, 8.4)$, so the ILP optimum solution can be (a, b) with $a \geq 2$

or $b \leq 9$ or both; i.e. one of the following:

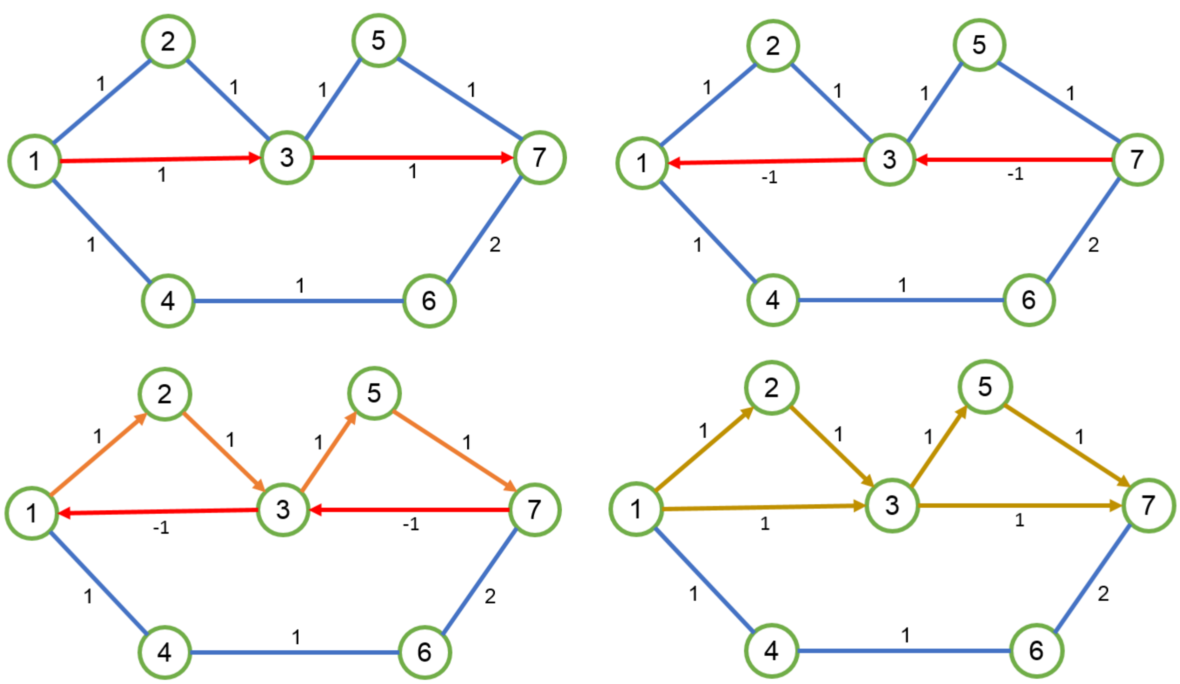
$$(2,8)$$
$$(3,7)$$
$$(4,6)$$
$$(5,4)$$
$$(6,3)$$

Hence the optimal point is $(4,6)$ with the optimum value of 108.

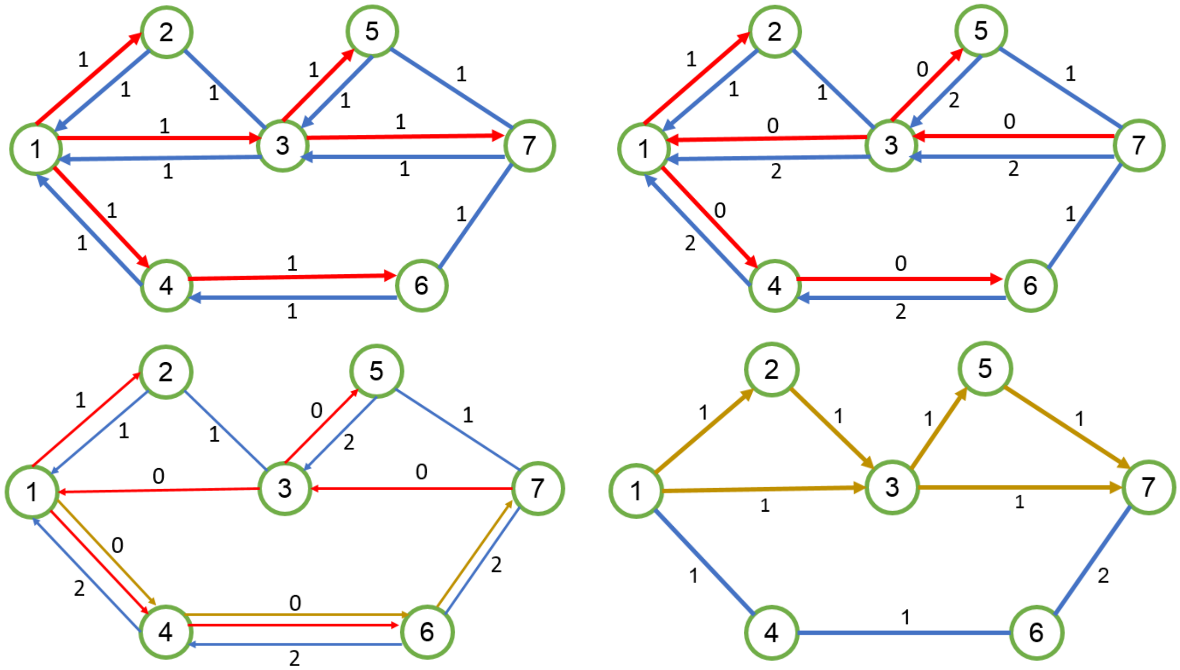
Question 4)
The shortest paths can be calculated without considering the link capacities.

$$\text{Shortest path between 1 and 6} : 1-4-3-5-6$$
$$\text{Shortest path between 2 and 6} : 2-3-5-6$$

Question 5)
a. Bhandari's algorithm:



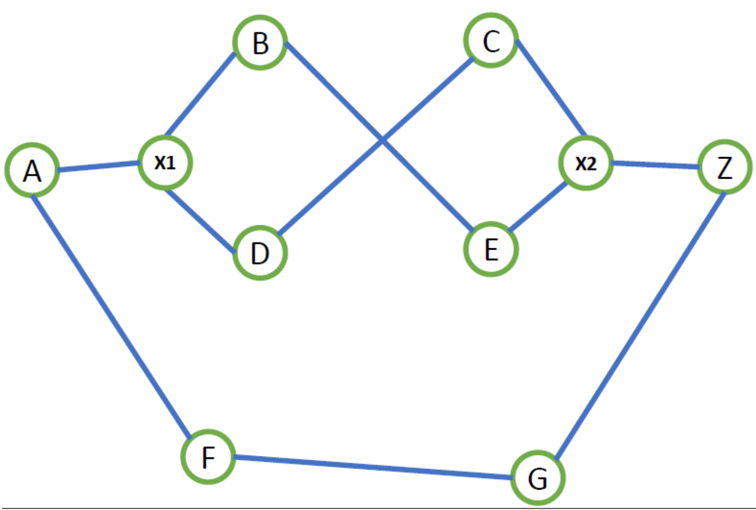
b. Suurballe's algorithm:



There two differences between Suurballe's and Bhandari's algorithm. Suurballe's algorithm relies on non-negative link costs and hence exploits the Dijkstra's algorithm while Bhandari's algorithm relies on generally valued link costs and hence exploits the Bellman-Ford's algorithm. Also Suurballe's algorithm leads to node disjoint paths while Bhandari's algorithm yields a link disjoint one.

Question 6)

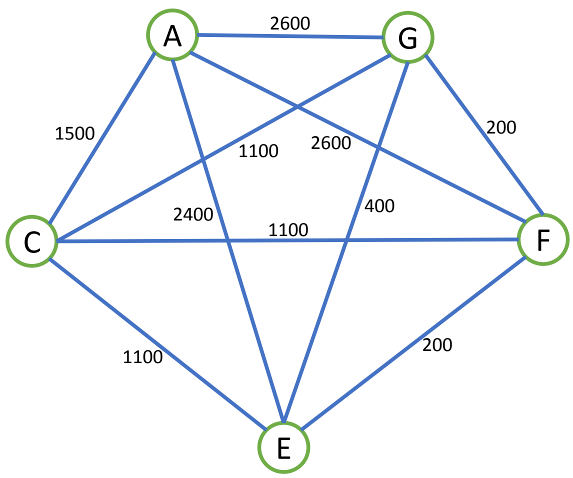
By considering the following topology, we have included the SRLG in routing:



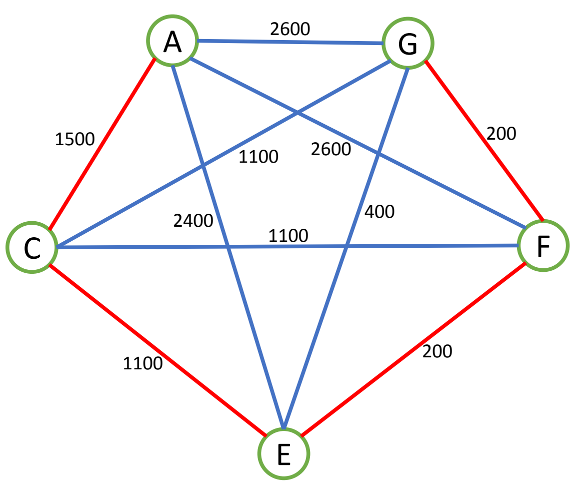
The advantage of the new topology is that running a SPDP algorithm on the old network (where SRLG was not considered) may lead to link-disjoint paths A-B-E-Z and A-D-C-Z, however, in reality these two paths are not actually disjoint (a failure in the amplifer hut can bring down both the paths). In the new physical topology however, the possibility of co-existence of such paths is removed since the links A-X1 and X2-Z are now shared between such paths and as a consequence, the SPDP algorithms do not yield both the paths simultaneously.

Question 7)

In the first step, we extract a complete graph wose nodes are multicast group nodes and links denote the shortest paths between the corresponding nodes.



Next, a minimum spanning tree is generated as:



Since all the shortest paths on the minimum spanning tree contain only multicast nodes (no non-multicast node such as B is included), the algorithm is terminated here and the found spanning tree is our final result.