|  |
| --- |
| **Assignment #4**  **ELG 5383 – Survivable Optical Networks** |
| Winter 2015  Submitted To    Professor Hussein T. Mouftah  By  **Ferhan Jamal (100 953 487-Carleton University)** |

**Q 1: For the network given below; assume dedicated mesh protection. Demand *r* requires *b*=1 unit of bandwidth. The working path for demand *r* is already computed by Dijkstra algorithm, *SrW* = {1, 2}. The backup bandwidth reserved on every link (*Bj*) is given by the last row in the table given below. *Bj* is also shown in the figure below. Assume that all links have large available capacity. Determine what path will be chosen by Dijkstra algorithm to be the backup path for demand *r*? Is this path the shortest path, why?**

**B3=11**

**B8=17**

**B5=6**

**B6=15**

**B7=2**

**B4=10**

**B9=10**

**B1=7**

**B2 = 9**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Link | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | 0 | **9** | **11** | 6 | **6** | 7 | 0 | 0 | **10** |
| 2 | 3 | 0 | 5 | 4 | **6** | 1 | **2** | 4 | 6 |
| 3 | 2 | 2 | 0 | **10** | 3 | 9 | 1 | 9 | 8 |
| 4 | 0 | 0 | 7 | 0 | 0 | **15** | 0 | 8 | 10 |
| 5 | 1 | **9** | 6 | 7 | 0 | 4 | 0 | 5 | 3 |
| 6 | **7** | 5 | 9 | 8 | 2 | 0 | 0 | 13 | 3 |
| 7 | 4 | 7 | 0 | 1 | 1 | 7 | 0 | 15 | 0 |
| 8 | 5 | 8 | 2 | **10** | 4 | 8 | 1 | 0 | 9 |
| 9 | 2 | 3 | 3 | 4 | 4 | 11 | 1 | **17** | 0 |
| **Bj** | 7 | 9 | 11 | 10 | 6 | 15 | 2 | 17 | 10 |

**Answer:**

It is given in the question:

b=1 unit of bandwidth

SrW = {1,2} [Working path for demand r]

Now, we have to find the backup path for the demand r.

Total amount of backup bandwidth reserved on link j is the maximum of all elements in column j of matrix K:

Bj = max kij

Ɐ i

Also, the maximum amount of backup bandwidth required on link j if a link in the working path of r fails will be:

Tj = b + max kij

Ɐ i ∈ SrW

As per question b=1 unit of bandwidth, therefore we have:

For j not in **SrW,** Tj is calculated as follows:

T3 = 1+max[k13 + k23] = 1+max[11,5]

T3 = 12

T5 = 1+max[k15 + k25] = 1+max[6,6]

T5 = 7

T6 = 1+max[k16 + k26] = 1+max[7,1]

T6 = 8

T7 = 1+max[k17 + k27] = 1+max[0,2]

T7 = 3

T8 = 1+max[k18 + k28] = 1+max[0,4]

T8 = 5

T9 = 1+max[k19 + k29] = 1+max[10,6]

T9 = 11

The values of k13, k23, k15,  k25, k16,  k26, k17,  k27, k18,  k28, k19,  k29 are given in the matrix above.

**Computation of Link Cost**

θj : Cost of using link *j* to be on the backup path of demand *r*:

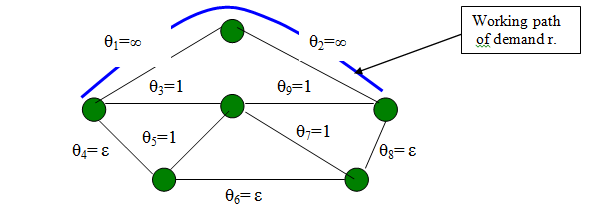
∞ if j ∈ SrW

ε else if Tj <= Bj

θj =

Tj - Bj else if Tj - Bj <= Rj

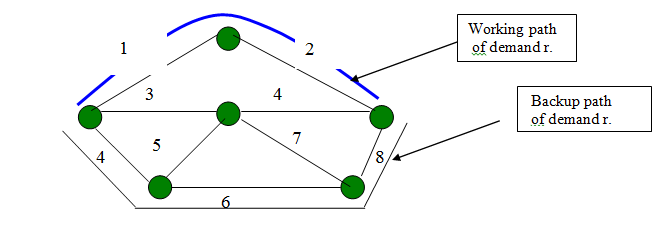
∞ otherwise



θ1  and θ2  = ∞ [ We have put the value of θ1  and θ2  equal to infinite as working path can never be backup path ]

θ3, θ5, θ7 and θ9  = 1 [ As Tj is greater than Bj, therefore we have subtract the two ]

θ4, θ6, and θ8  = ε [ As Tj is lesser than Bj, therefore we have put epsilon ]



Therefore, the backup-path for b=1 unit of bandwidth will be { 4,6,8 } because the sum of the cost is minimum in this path. The no’s above in the networks denote the link.

SrW = {1, 2} [Working path]

SrB = {4, 6, 8} [Backup path]

***In the question, it is asked that the calculated backup path obtained is the shortest backup path or not***. The answer of this question is:

“Yes it is, because it is most effective in terms of epsilon. “

**Q 2: If problem1 assume pool sharing with the same working and backup paths computed for demand *r (assume that the value of vwf is ignored)*:**

* **Determine the set of troublesome links (*Sr, jD*) for demand *r* on backup links *j =* 4, 6, and 8**
* **Determine *lr, jD* and *gr, jD* for each *j* above**
* **Assume, threshold *G* is given as 0, 0.5, or 1**
* **Using Enhanced Pool Sharing algorithm, and for each G, determine if extra backup =bandwidth is needed on link *j***

***Let l0 be 500 (failures per mile per 109 hours) and let the lengths of links be given***

***below****:*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Link*** | ***1*** | ***2*** | ***3*** | ***4*** | ***5*** | ***6*** | ***7*** | ***8*** | ***9*** |
| ***Distances in miles*** | ***450*** | ***400*** | ***650*** | ***300*** | ***250*** | ***270*** | ***290*** | ***150*** | ***650*** |

**Answer:**

Ɐj ∈ SrB

Ɐw ∈ SrW

Ɐf s.t. f ∈ SrW and f≠j

f=3, 5, 7, 9

Mj = kwj + kfj – vwf

M4 = k14 + k34 – 0 =16

M4 = k24 + k34 – 0 = 14

M4 = k14 + k54 – 0 = 13

M4 = k24 + k54 – 0 = 11

M4 = k14 + k74 – 0 = 7

M4 = k24 + k74 – 0 = 5

M4 = k14 + k94 – 0 = 10

M4 = k24 + k94 – 0 = 8

*From above, 3 and 5 are the troublesome links. (M4>B4 in some equations above where B4 equals 10)*

M6 = k16 + k36 – 0 = 16

M6 = k26 + k36 – 0 = 10

M6 = k16 + k56 – 0 = 11

M6 = k26 + k56 – 0 = 5

M6 = k16 + k76 – 0 = 14

M6 = k26 + k76 – 0 = 8

M6 = k16 + k96 – 0 =18

M6 = k26 + k96 – 0 = 12

*From above, 3 and 9 are the troublesome links. (M6>B6 in some equations abovewhere*

*B6 = 15)*

M8 = k18 + k38 – 0 = 9

M8 = k28 + k38 – 0 = 13

M8 = k18 + k58 – 0 = 5

M8 = k28 + k58 – 0 = 9

M8 = k18 + k78 – 0 = 15

M8 = k28 + k78 – 0 = 19

M8 = k18 + k98 – 0 = 17

M8 = k28 + k98 – 0 = 21

*From above, 7 and 9 are the troublesome links. (M8>B8 in some equations abovewhere*

*B8 = 17)*

Therefore overall 3, 5, 7 and 9 are the troublesome links.

**λΔr,j = ∑ λ0lj**

**Ɐ∈ SΔ r,j**

where λ r,j Δ  is the sum of the failure rate of links in SΔ r,j  and SΔ r,j set of troublesome links along the working paths of interfering demands on link *j* therefore we have:

λΔr,j = 500[650+650+290+ 250] = **9,20,000 FITS**

**γΔ r.j  = λΔr,j / ∑ λ0lj , (0 <= γΔ r.j <=1)**

j<=J

γΔ r.j = 9,20,000 / [ 500(450+400+650+300+250+270+290+150+650)]

γΔ r.j = 9,20,000/[500\*(3410)] = 9,20,000/1,705,000

γΔ r.j = **0.5395**

It is given in the question that the value of threshold G is given as 0, 0.5,1.

When G=0, then γΔ r.j (0.5395) > G which means **one unit of extra bandwidth is**

**added to link j.**

When G=0.5, then γΔ r.j (0.5395) > G which means **one unit of extra bandwidth is**

**added to link j.**

When G=1, then γΔ r.j (0.5395) < G which means **there is no need to add any bandwidth** to link j.