

# 时延范围&时延差算法

Dijkstra-lab

[www.huawei.com](http://www.huawei.com)

Author/ Email: Author's name/Author's email

Version: V1.0 (20YYMMDD)

HUAWEI TECHNOLOGIES CO., LTD.



# 难题1: [新路由]网络时延保障&高可靠路由算法

出题组织: 迪杰斯特拉实验室 接口专家: 徐慧颖 xuhuiying@huawei.com 、赵凤华 zhaofenghua@huawei.com

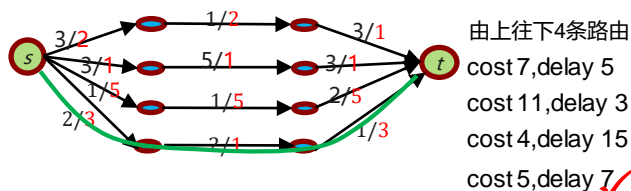
➢ 算力网络需要品质连接, 带宽、时延和成本是影响客户选择云池的关键指标。在Cloud VR, 远程自动驾驶等众多智能场景下都共同需要一类连接, 要求低时延、高可靠, 时延分级定价, 实现差分服务, 最大化收益。

**对网络路由算法的需求:** 给定网络拓扑, 根据业务带宽、时延、可靠性要求, 综合考虑网络和云池成本, 计算满足时延范围、时延差、分离约束成本最低的路由。

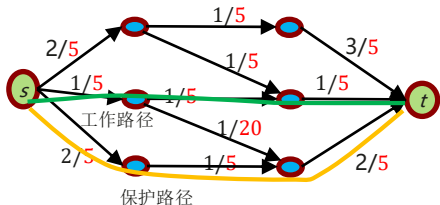


## 场景1: 时延范围约束路由:

为s到t的业务计算, 满足时延范围[5,7]约束, 成本最低路由



## 场景2: 时延差约束分离路由: 为s到t的业务计算满足链路分离, 时延差小于5(ms), 工作路径成本最低路由



## • 时延范围约束最短路

Variables

- $X_p$ : decides whether path  $p \in P_k$  is used by demand  $k$ .

ILP model

$$\min \sum_{p \in P_k} c_p X_p \quad (1)$$

$$\sum_{p \in P_k} X_p = 1 \quad (2)$$

$$X_p \in \{0, 1\} \quad (3)$$

Note that  $P_k$  is defined as the set of path that meet end-to-end latency constraints: their latency is between  $D_{min}$  and  $D_{max}$ .

## 技术挑战

## • 时延差约束分离最短路

Variables

- $X_p$ : decides whether working path  $p \in P_k$  is used by tunnel  $k$ .
- $Y_p$ : decides whether protection path  $p \in P_k$  is used by tunnel  $k$ .

ILP model

$$\min \sum_{p \in P_k} c_p X_p \quad (1)$$

$$\sum_{p \in P_k} X_p = 1 \quad (2)$$

$$\sum_{p \in P_k} Y_p = 1 \quad (3)$$

$$|D(X_p) - D(X_{p'})| \leq \Delta \quad \forall p, p' \in P_k$$

Note that  $P_k$  is defined as the set of path that meet end-to-end latency constraints: their latency is between  $D_{min}$  and  $D_{max}$ .

The working path and the protection path need to support nodes/links/SRLGs disjoint

时延差、时延范围等多种复杂约束组合寻优是一个NPC图论问题, 实时 (20ms) 内计算最优路由是一个业界难题。

## 当前结果

**暴力搜索:** 采用KSP暴力搜索可以求得最优解, 但在复杂网络下, 算法搜索空间巨大, 时间性能无法接受

Compute k1 work path(KSP)  
compute k2 protect path(KSP)  
if (check constraints):  
record the best solution.  
复杂网络,  $k1 = k2 = 40K$ , 极限搜索次数高达16亿

**时延范围约束最短路:** 7K个测试例, 已有算法平均有路通过率99.25%, 平均优度值71%

**时延差约束分离最短路:** 1K个测试例, 已有算法平均有路通过率98.20%, 平均优度值65%

## 技术诉求

**网络拓扑类型:** 典型组网, 规模: 5k节点, 20k链路

**计算性能要求:** Intel Core i7-8700 CPU @3.2GHZ, 单条业务路由耗时<20 ms, 内存占用<512MB。

- 时延范围约束路由:** 为业务计算一条时延范围为[L,U], 满足带宽约束, 成本最小的路由。
- 时延差约束路由:** 为业务计算满足指定分离要求 (链路分离/结点分离/SRLG分离) 的主备两条路由, 且2条路由满足时延范围和带宽约束, 保证两条路由的时延差在一定范围内, 工作路径成本最小。
- 评价标准:** 满足约束下看优度值, 优度值定义为: 最优工作cost / 输出工作cost。
- 挑战目标:** 所有用例算法有路通过率99.99%及以上, 平均优度值95%及以上。

备注: 可以提供测试例进行算法的验证

## 参考文献:

- [1] Rostami M J, Khorsandi S, Khodaparast A A. CoSE: A SRLG-disjoint routing algorithm[C]//Fourth European Conference on Universal Multiservice Networks (ECUMN'07). IEEE, 2007: 86-92.
- [2] Handler G Y, Zang I. A dual algorithm for the constrained shortest path problem[J]. Networks, 1980, 10(4): 293-309.
- [3] Cabrera N, Medaglia A L, Lozano L, et al. An exact bidirectional pulse algorithm for the constrained shortest path[J]. Networks, 2020, 76(2): 128-146.

# 时延范围约束

## Variables

- $X_p$ : decides whether path  $p \in P_k$  is used by demand  $k$ .

## ILP model

$$\min \sum_{p \in P_k} c_p X_p \quad (1)$$

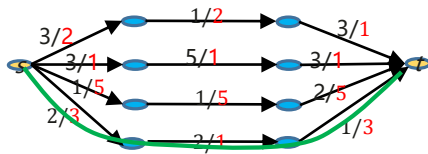
$$\sum_{p \in P_k} X_p = 1 \quad (2)$$

$$X_p \in \{0, 1\} \quad (3)$$

Note that  $P_k$  is defined as the set of path that meet end-to-end latency constraints: their latency is between  $D_{min}$  and  $D_{max}$ .

时延范围约束:

为s到t的业务计算, 满足时延范围[5,7]  
约束, 成本最低路由



由上往下4条路由

cost 7, delay 5

cost 11, delay 3

cost 4, delay 15

cost 5, delay 7 ✓

资源约束最短路问题的变种问题 (资源约束最短路RCSP—the resource constrained shortest path problem)

# 时延差约束分离

## Variables

$X_p$ : decides whether working path  $p \in P_k$  is used by tunnel  $k$ .

$Y_p$ : decides whether protection path  $p \in P_k$  is used by tunnel  $k$ .

## ILP model

$$\begin{aligned} \min \quad & \sum_{p \in P_k} c_p X_p \\ & \sum_{p \in P_k} X_p = 1 \\ & \sum_{p \in P_k} Y_p = 1 \\ & |D(Y_p) - D(X_p)| \leq \Delta \quad \forall p \in P_k \\ & Y_p, X_p \in \{0, 1\} \end{aligned}$$

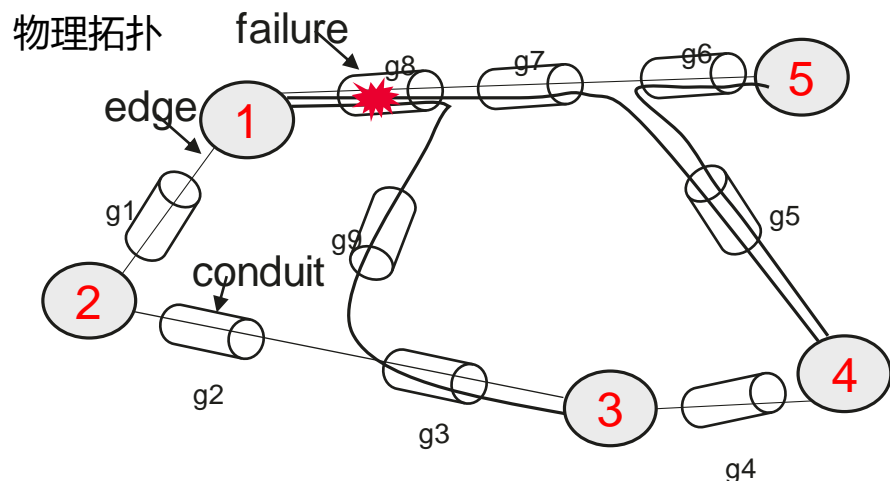
Note that  $P_k$  is defined as the set of path that meet end-to-end latency constraints  $D_{max}$ .

The working path and the protection path need to support nodes/links/SRLGs disjoint

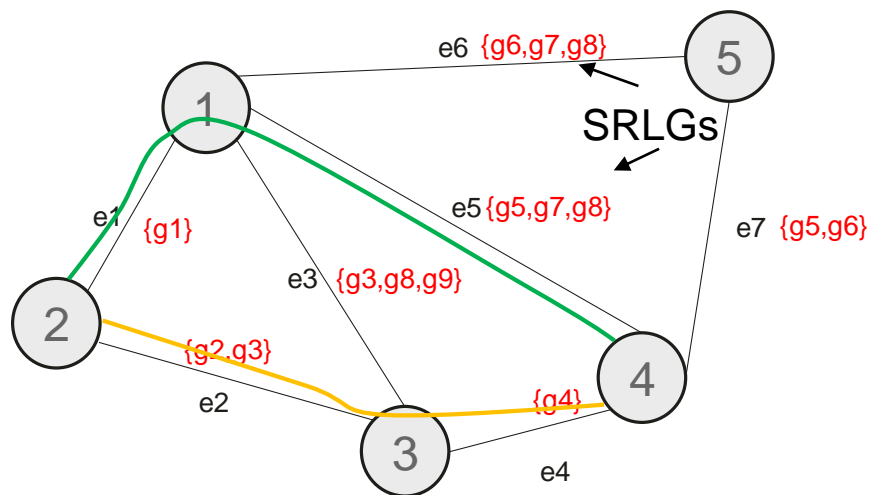
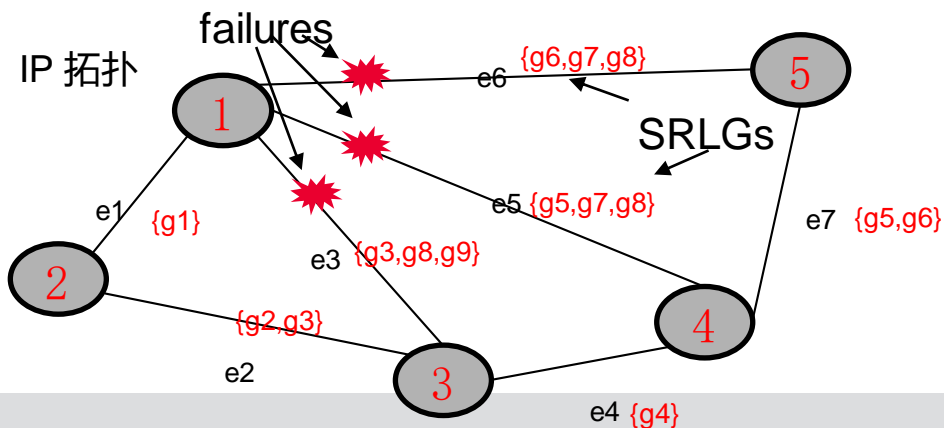
分离类型:

- 1) 链路分离: 工作, 保护路径经过的链路不重合
- 2) 结点分离: 除源宿结点外, 工作保护路径不经过重复结点
- 3) SRLG分离: 链路分离 & 工作保护路径经过的SRLG集合不相交

# SRLG分离约束类型



物理管道g8的中断故障会引起IP 链路e3,e5,e6同时中断  
SRLG  $g8 = \{e3, e5, e6\}$



1) 工作路径: 结点序列【2, 1, 4】,  
链路序列【e1,e5】, SRLG序列【g1,g5,g7,g8】

2) 保护路径: 结点序列【2, 3, 4】,  
链路序列【e2,e4】,  
SRLG序列【g2,g3,g4】

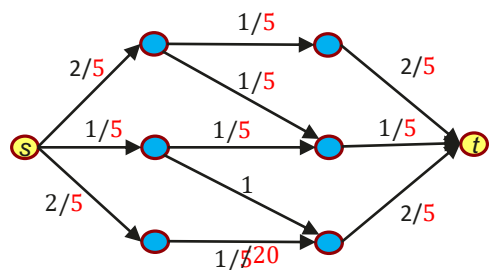
链路分离 ✓

结点分离 ✓

SRLG分离 ✓

# 时延差约束分离最小成本选路

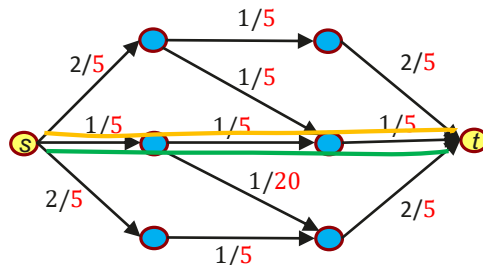
Cost/Delay



问题：S到T计算链路完全分离的两条路径，

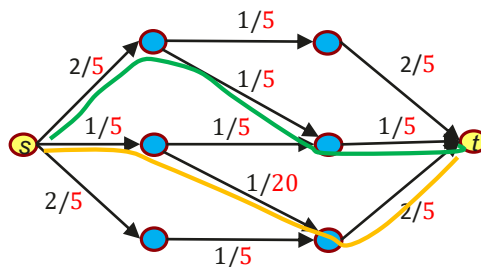
且两条路径的时延差小于等于5，

且工作路径成本最低



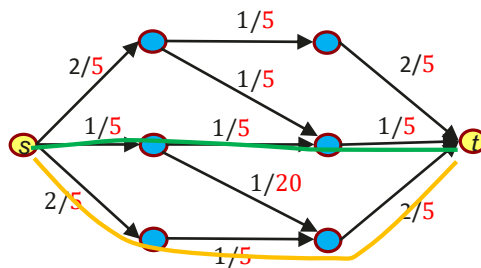
- 工作路径cost 3,时延15
- 保护路径cost 3,时延15
- 工作路径成本最小
- 满足时延差约束
- 不满足链路分离要求

X



- 工作路径cost 4,时延15
- 保护路径cost 4,时延30
- 不满足时延差约束
- 满足链路分离要求

X



- 工作路径cost 3,时延15
- 保护路径cost 5,时延15
- 满足时延差约束
- 满足链路分离要求
- 工作成本最低

最优解

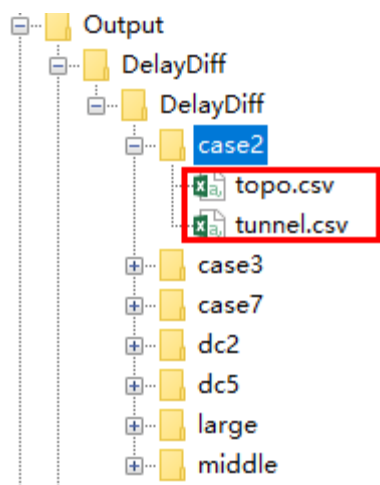


# 网例说明

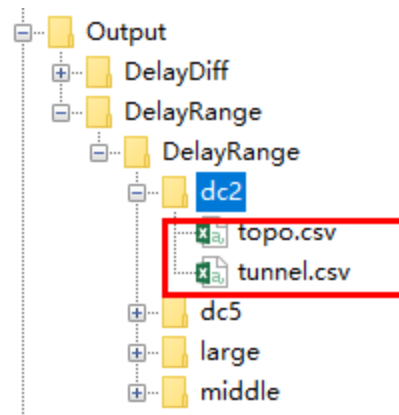
测试场景：时延范围，时延差各对应一个文件  
时延范围对应文件夹DelayRange，时延差对应DelayDiff



每个主文件夹对应的最低层子目录是一个测试网例



时延差



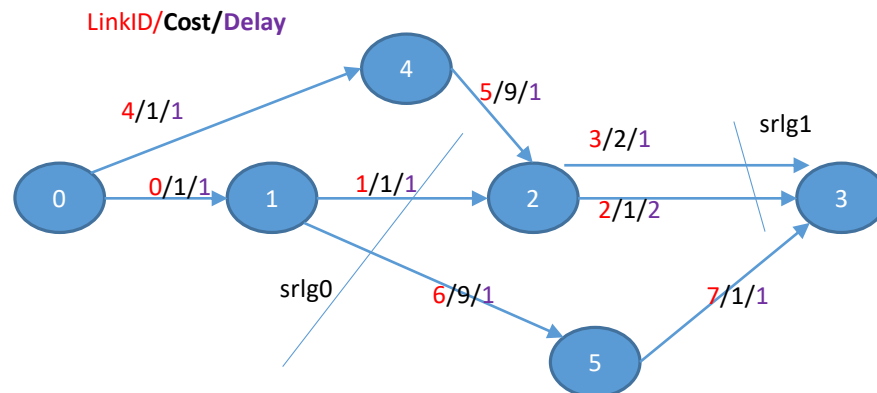
时延范围

topo.csv tunnel.csv 对应网例格式文件;  
topo.csv 包含网络拓扑信息, tunnel.csv包含网络业务信息

# topo.csv

LinkID	SourceID	DestinationID	PeerID	Cost	Bandwidth	Delay	SRLGNum	SRLGs
0	0	1	0	1	10000	1	0	
1	1	2	0	1	10000	1	1	0
2	2	3	0	1	10000	2	1	1
3	2	3	0	2	10000	1	1	1
4	0	4	0	1	10000	1	0	
5	4	2	0	9	10000	1	0	
6	1	5	0	9	10000	1	1	0
7	5	3	0	1	10000	1	0	

- 1) LinkID: 链路的ID
- 2) SourceID: 链路对应的源点
- 3) DestinationID: 链路对应的宿点
- 4) PeerID: 链路对应的对等链路ID, 目前测试场景未涉及请忽略
- 5) Cost: 链路的成本
- 6) Bandwidth: 链路可用带宽
- 7) Delay: 链路对应的时延值
- 8) SRLGNum: 链路上配置的SRLG个数
- 9) SRLGs: 链路上配置详细 SRLG 标签值



topo.csv对应的算法抽象有向图  
链路2和3都有一个SRLG标签1  
因此SRLG 1 包含链路组{2, 3}



# tunnel.csv

demandID	SourceID	DestinationID	MinDelay	MaxDelay	Bandwidth	Ishotstandby	DelayDiff	DisjointType	WorkOptCost
0	0	3	2	9	10	0	0	-1	3
1	0	3	2	9	10	1	1	0	3
2	0	3	2	9	10	1	1	1	11
3	0	3	2	9	10	1	1	2	11

1) demandID: 业务ID

2) SourceID: 业务源点

3) DestinationID: 业务宿点

4) MinDelay: 业务最小时延约束

5) MaxDelay: 业务最大时延约束

6) Bandwidth: 业务带宽需求

7) Ishotstandby: 可选值[0,1],

0时表示非可靠算路场景，对应单纯时延范围约束网例；

1表示可靠算法场景，对应时延差保护分离约束网例，在此场景下工作&保护路径均受[MinDelay, MaxDelay]限制

8) DelayDifference: 在Ishotstandby = 0时，该值无意义默认为0； Ishotstandby = 1时，该值表示工作&保护路径的时延差阈值约束

9) DisjointType: 在Ishotstandby = 0时，该值无意义默认为-1； Ishotstandby = 1时，有3个可选值{0, 1, 2}，

DisjointType = 0时，表示工作&保护路径要求链路分离； DisjointType = 1是，表示工作&保护路径要求结点分离；

DisjointType = 2时，表示工作&保护路径要求SRLG分离

10) WorkOptCost: 最优工作路径cost

总结：

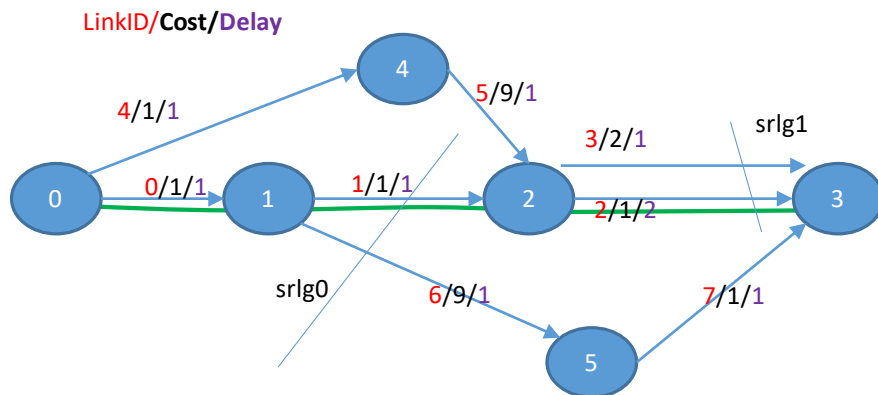
1) 业务0，时延范围约束算路场景，其对应的时延范围约束是[MinDelay, MaxDelay] = [2,9]

2) 业务1，时延差保护分离约束场景，要求满足链路分离，工作&保护路径的时延范围约束[MinDelay, MaxDelay] = [2,9]且工作&保护路径的时延差小于等于1

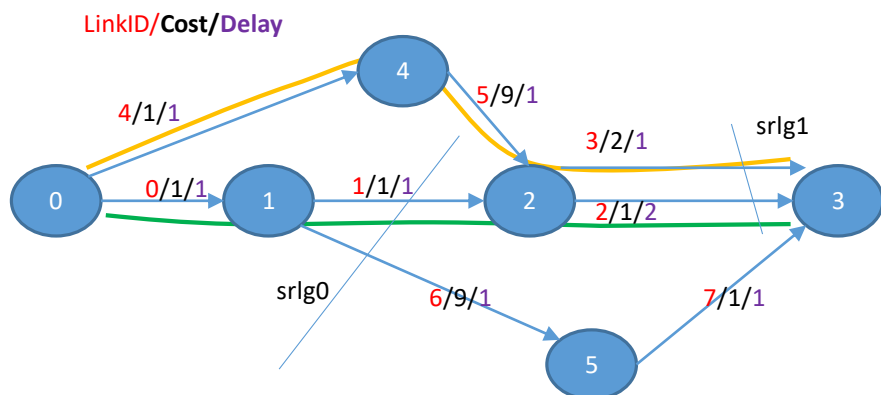
3) 业务2，时延差保护分离约束场景，要求满足结点分离，工作&保护路径的时延范围约束[MinDelay, MaxDelay] = [2,9]且工作&保护路径的时延差小于等于1

4) 业务3，时延差保护分离约束场景，要求满足SRLG分离，工作&保护路径的时延范围约束[MinDelay, MaxDelay] = [2,9]且工作&保护路径的时延差小于等于1

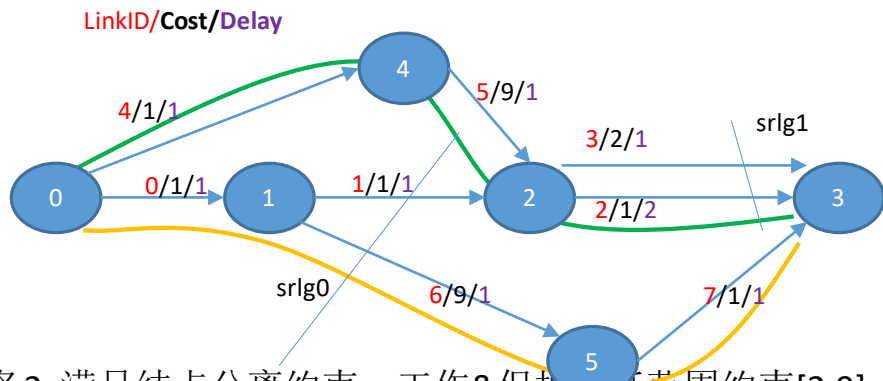
# 测试结果



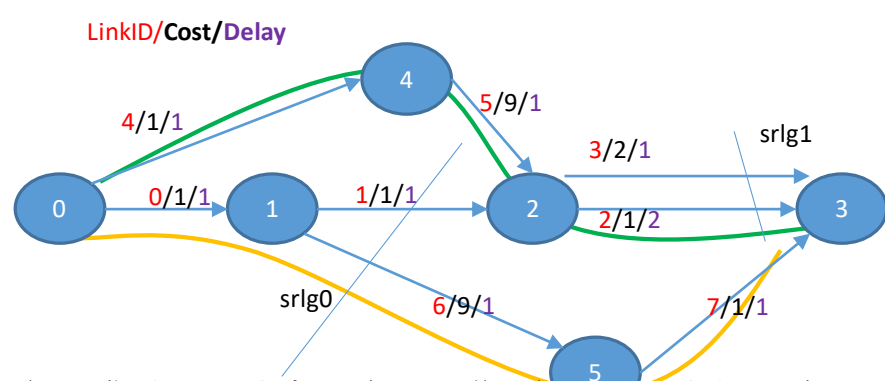
业务0,时延范围约束[2,9],最优路径如绿线所示[0,1,2], 路径时延4,路径成本3



业务1, 满足链路分离约束, 工作&保护时延范围约束[2,9], 工作&保护路径时延差小于等于1。最优工作路径如绿线所示路径[0,1,2], 路径时延4, 路径成本3  
保护路径如橙线所示路径[4,5,3],路径时延3, 路径成本12



业务2, 满足结点分离约束, 工作&保护时延范围约束[2,9], 工作&保护路径时延差小于等于1。最优工作路径如绿线所示路径[4,5,2], 路径时延4, 路径成本11  
保护路径如橙线所示路径[0,6,7], 路径时延3, 路径成本11



业务3, 满足SRLG分离约束, 工作&保护时延范围约束[2,9], 工作&保护路径时延差小于等于1。最优工作路径如绿线所示路径[4,5,2], 路径时延4, 路径成本11  
保护路径如橙线所示路径[0,6,7], 路径时延3, 路径成本11

# Thank you

[www.huawei.com](http://www.huawei.com)

**Copyright©2011 Huawei Technologies Co., Ltd. All Rights Reserved.**

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.