首先将整体的网络表示为G(V, E)

V代表网络拓扑中的节点集合

E代表网络拓扑中边的集合

假设业务的表示为B(source, dest, st, ed, w, endure)

source代表业务的源节点

dest代表业务的目的节点

st代表业务的到达时间

ed代表业务的结束时间

w代表业务的需求带宽

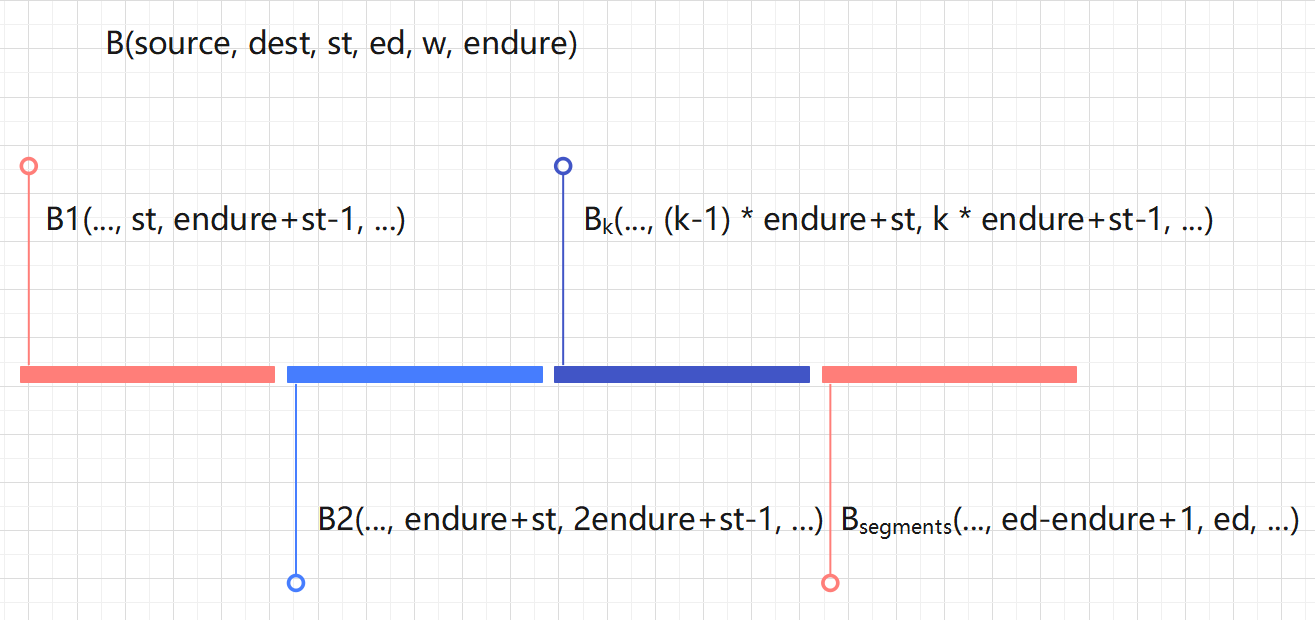
endure代表业务的每条光路上的最大传输时间

具体的算法经过如下:

Step1: 初始化G(V, E)，将每个链路对应的频隙状态向量置为0，代表未使用，将每个链路对应的权重置为1。

Step2: 等待新业务B(source, dest, st, ed, w, endure)来临，进入Step3，否则一直等待直到结束。

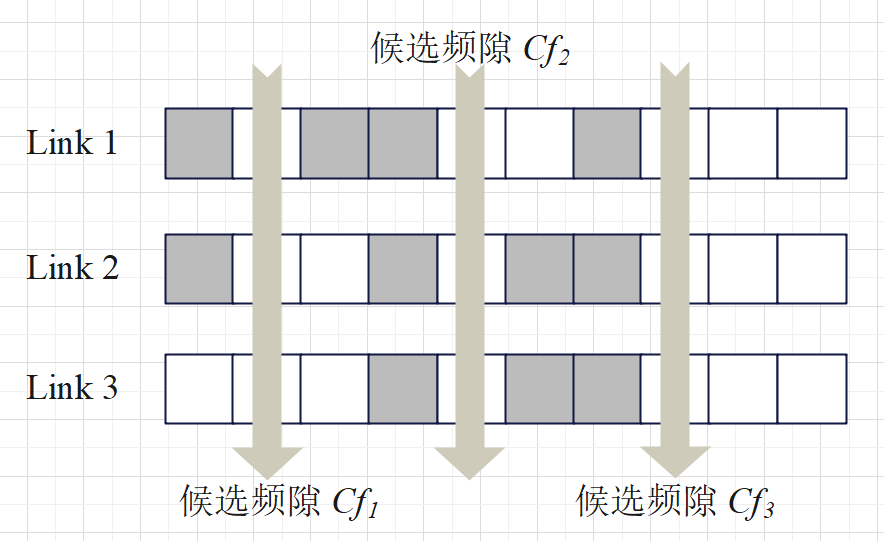
Step3: 拆分新业务B，计算最少需要几段（**业务段**），segments = ceiling((ed - st + 1) / endure)。如果segments＞source节点的度数，则直接阻塞。



Step4: 根据 单源最短路 算法找到待分配链路*e*（每条链路的权重，即，weight(Vx) = 这条链路的所占FS最大值）。找不到则宣告此业务阻塞，返回Step2。

Step5: 遍历链路 *e* 对应的频隙状态，找到可以容纳业务段的候选频隙集Cf。如果len(Cf)==1的话，将候选频隙分配给当前业务段，进入Step13。如果候选频隙集为空返回Step4。或者同一个业务的业务段有节点相交也要返回Step4。否则进入Step6。

Step6: 利用公式计算候选频隙集Cf上每个候选频隙的价值price(Cfi)。



Step7: 对任意的Cfi∈Cf，计算将此候选频隙作为最终分配频隙后价值的减少量d(Cfi)，并保存。

Step8：找到d(Cf)中最小的值min{d(Cf)}，如果len(min{d(Cf)})==1，将对应的候选频隙Cfi分配给当前业务，进入Step13。否则进入Step9。

Step9: 找到链路的所有邻接链路Ae。

Step10: 对任意的d(Cfi)∈min{d(Cf)}，计算将d(Cfi)对应的候选频隙Cfi作为最终分配频隙后和邻接链路Ae频隙对齐量的减少u(Cfi)。

Step11: 找到u(Cf)中最小的值min{u(Cf)}，如果len(min{u(Cf)})==1，将这个唯一的值所对应的候选频隙Cfi分配给当前业务，进入Step13，否则进入Step12。

Step12: 对任意的u(Cfi)∈min{u(Cf)}，查看其i值大小（i为候选频隙Cfi在fe上的起始下标），找到i值最小的u(Cfi)所对应的候选频隙Cfi，将其分配给当前业务，进入Step13。

Step13: 依次将最短路径组合链路上的频隙状态更为已占用状态。并对每一个业务B，都要记录每个业务段的频谱分配方案，包括经过的节点，所用的FSs的起止序号。

Step14: 更新每条链路的权重，返回Step2。