第二章 道路与回路Ⅱ

计算机系网络所：张小平

主要内容

Tsinghua University

旅行商问题

- 哈密顿道路(回路)问题，是“点遍历”问题，不涉及边的长度(权值)。

- 给定一个特定的正权图G，可能存在许多H道路(回路)，如何从中选出最短路径(总权值最小)?

旅行商问题

- 旅行商问题(Traveling Salesman Problem, TSP)，又译为旅行推销员问题，货郎担问题，简称为TS问题：

  有若干个城市，任何城市之间的距离都是确定的，旅行商从某城市出发，必须经过每一个城市且只经过一次，最后回到出发城市。问如何事先确定好一条最短的路线，使其旅行的费用最少。

- 给定一个正权完全图，求其总长最短的哈密顿回路，就是TSP问题。

旅行商问题

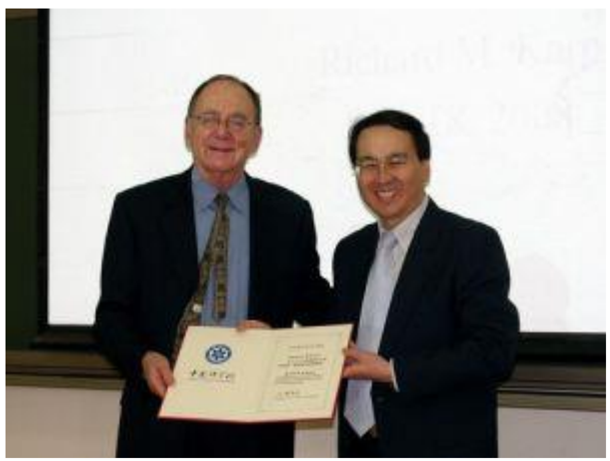
- 问题复杂度分析：
  - 对n个结点的完全图，存在$\frac{1}{2}(n-1)!$个不同的H回路
  - Stirling公式：$n! \approx \sqrt{2\pi \cdot n} \cdot \left(\frac{n}{e}\right)^n$
  - 如果采用枚举方式，在n增大时，计算量将急剧膨胀！

旅行商问题

- 若用枚举搜索法对N=26的TSP问题进行求解，即使采用每秒钟计算33.86千万亿次的计算机，需要104 年。
- 1948年，由美国兰德(Rand)公司推动，TSP成为近代组合优化领域的一个典型难题，而且兰德公司三位专家采用手工和计算机相结合的办法，创造了周游49个城市的纪录
- 60年代，理查德·卡普（Richard Manning Karp）把纪录提高到65个城市

旅行商问题

- 分枝与界法：
  - branch-and-bound method
  - 分枝定界法/分枝限界法

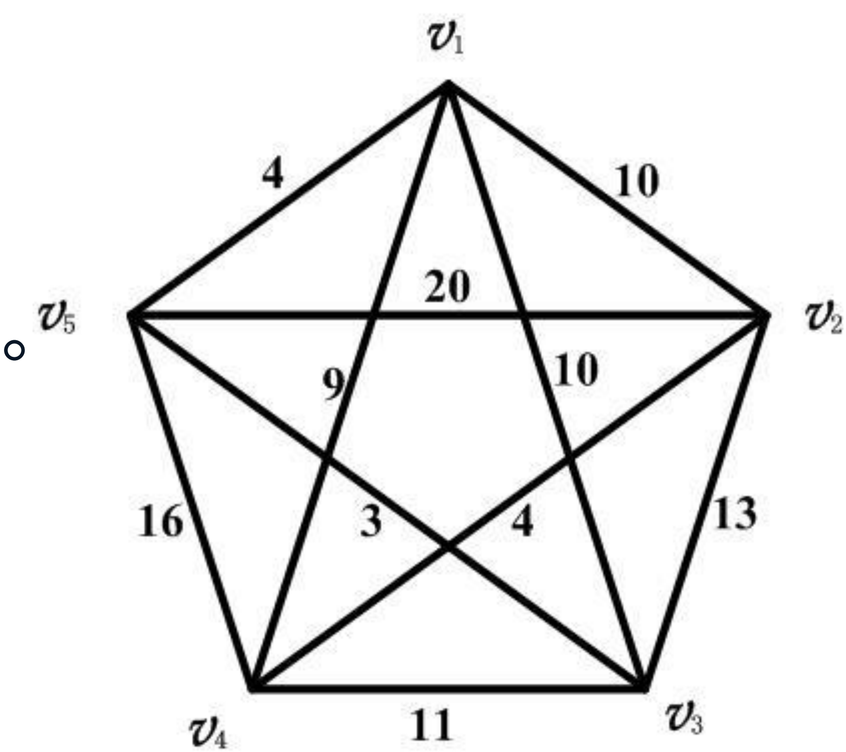


旅行商问题

- 分枝与界法基本思想：
  - 边按照权值排序
  - 按照该顺序，搜索H回路
  - 搜索过程中，记录发现的最短回路，并随时更新界值（取最小值）
  - 搜索过程中，如果路径长度已经超过界值，则剪枝

旅行商问题

- 例：该图表示5个城市间的铁路线，各边的值表示该线路的旅途费用。求从v_1出发经各城市一次且仅一次最后返回v_1总费用最省的一条路径



旅行商问题

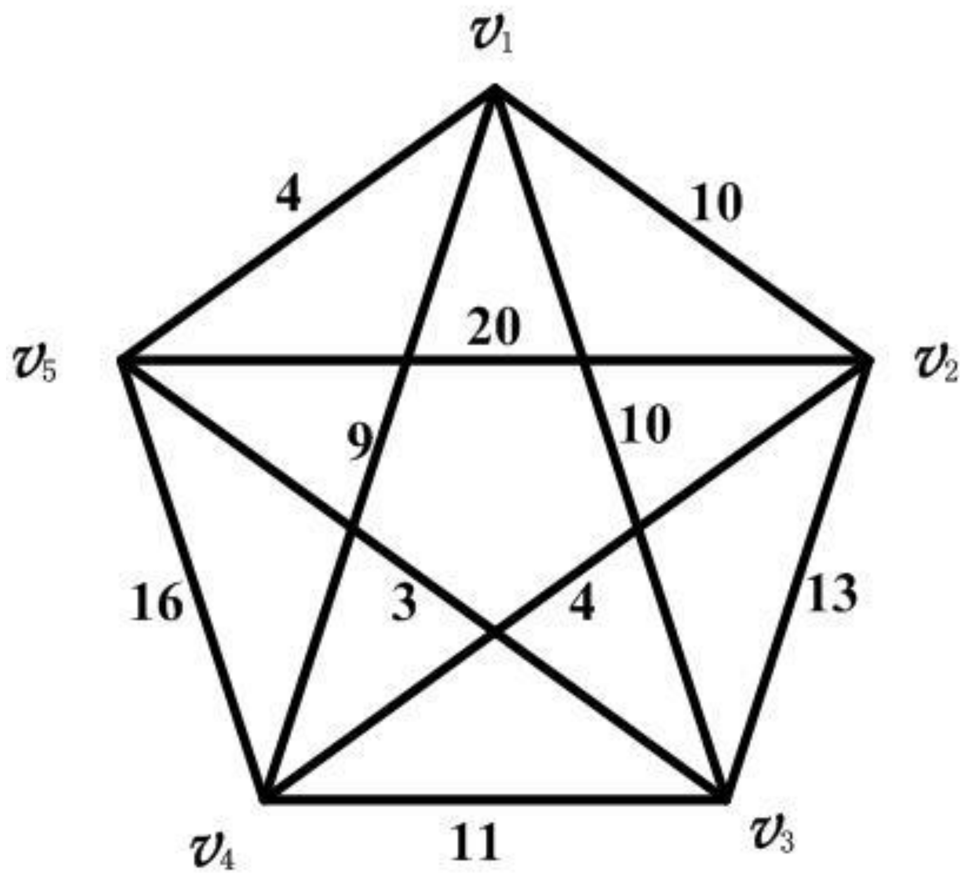
- 解：该问题就是求G的一条最短的H回路。采用分枝与界法的基本步骤是：
  - (1) 将边按权值由小到大排序，初始界为$d_0 \leftarrow \infty$。
  - (2) 在边权序列中依次选边进行深探，直到选取n条边，记为s，判断是否构成H回路，若是，$d_0 \leftarrow d(s)$。

    (思考，如何判断n条边可构成H回路？)。

旅行商问题

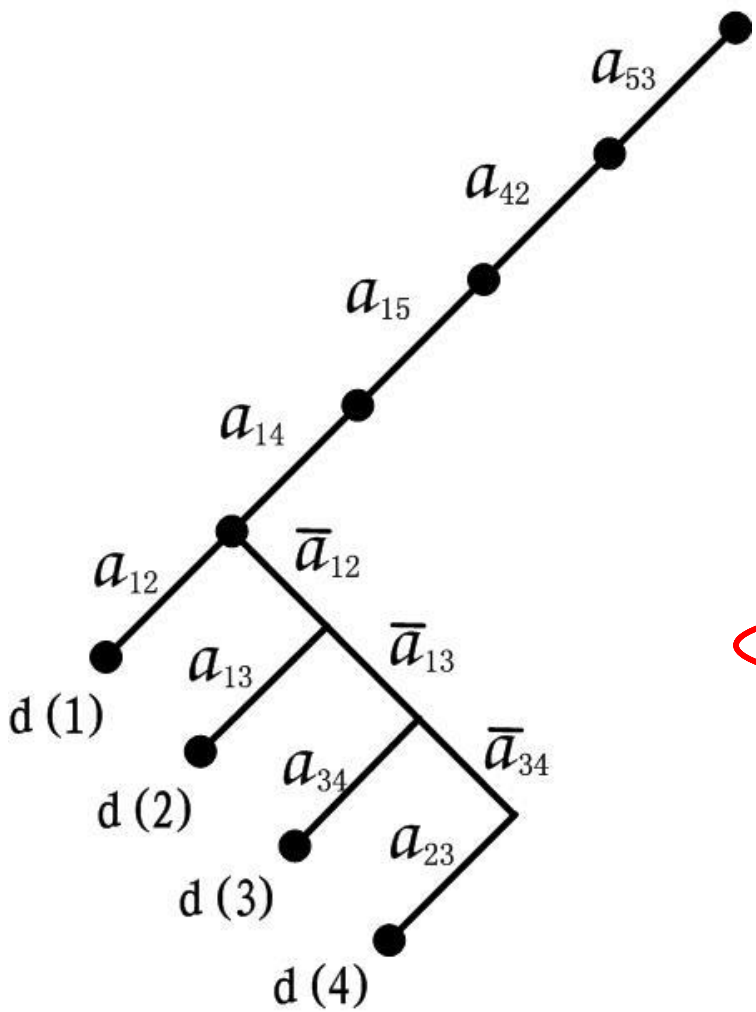
- (3) (继续深探)依次删除当前s中最长的边，加入后面第一条待选边，进行深探，若它是H回路，且$d(s) < d_0$，则用$d(s)$替换d_0作为界。

- (4) (退栈过程)不能再深探时需要退栈，如果栈空，结束，其最佳值为d_0；否则，如果新分支的$d(s) >= d_0$,继续退栈；若$d(s) < d_0$,转(3)。



v_1
4
10
20
v_5
v_2
9
10
16
3
4
13
11
v_4
v_3

a_{ij}:	$\underline{a_{53}}$	$\underline{a_{42}}$	$\underline{a_{15}}$	$\underline{a_{14}}$	$\underline{a_{12}}$	$\underline{a_{13}}$	$\underline{a_{34}}$	$\underline{a_{23}}$	a_{45}	a_{25}
l_{ij}:	3	4	4	9	10	10	11	13	16	20



$$\boldsymbol{d_0} \leftarrow \infty$$

$$d(1) = d(a_{53}, a_{42}, a_{15}, a_{14}, a_{12}) = 30$$

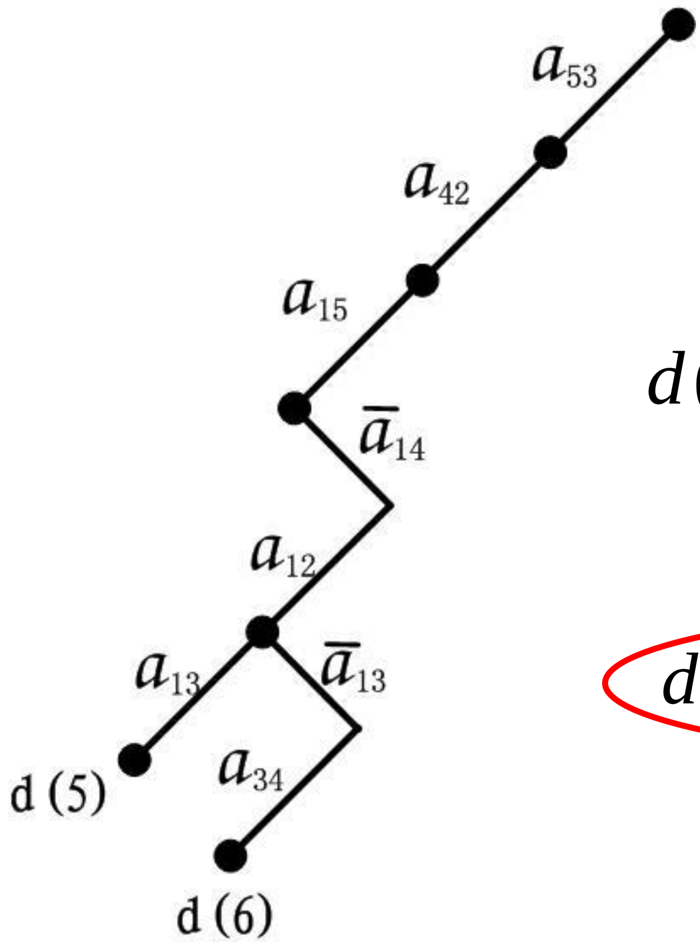
$$d(2) = d(a_{53}, a_{42}, a_{15}, a_{14}, a_{13}) = 30$$

$$d(3) = d(a_{53}, a_{42}, a_{15}, a_{14}, a_{34}) = 31$$

$$d(4) = d(a_{53}, a_{42}, a_{15}, a_{14}, a_{23}) = 33$$

$$d_0 = 33$$

a_{ij}:	$\underline{a_{53}}$	$\underline{a_{42}}$	$\underline{a_{15}}$	$\underline{a_{14}}$	$\underline{a_{12}}$	$\underline{a_{13}}$	$\underline{a_{34}}$	$\underline{a_{23}}$	a_{45}	a_{25}
l_{ij}:	3	4	4	9	10	10	11	13	16	20

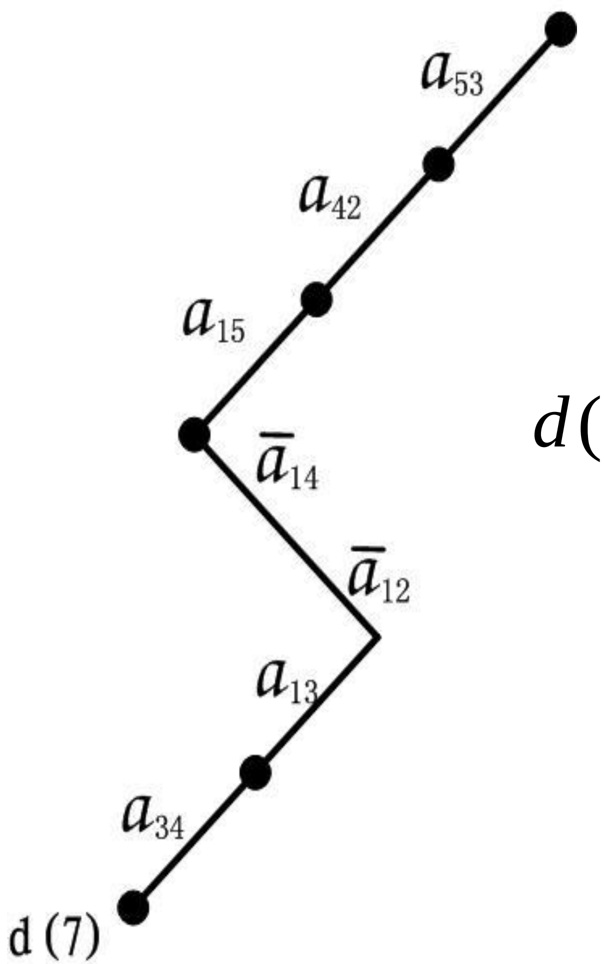


$$d(5) = d(a_{53}, a_{42}, a_{15}, a_{12}, a_{13}) = 31$$

$$d(6) = d(a_{53}, a_{42}, a_{15}, a_{12}, a_{34}) = 32$$

$$d_0 = 33 \rightarrow d_0 = 32$$

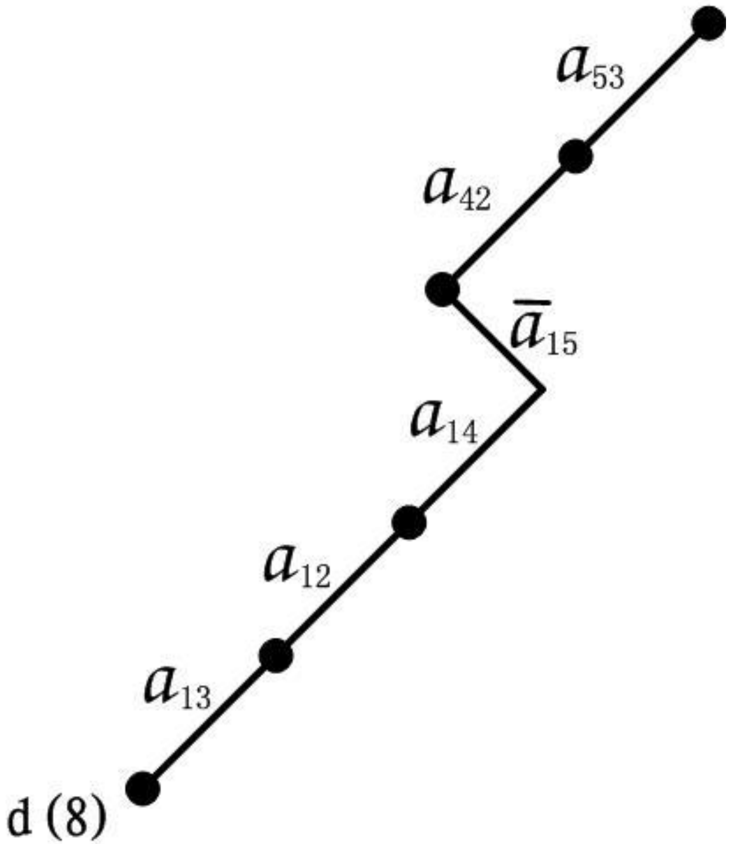
a_{ij}:	$\underline{a_{53}}$	$\underline{a_{42}}$	$\underline{a_{15}}$	a_{14}	$\underline{a_{12}}$	$\underline{a_{13}}$	$\underline{a_{34}}$	a_{23}	a_{45}	a_{25}
l_{ij}:	3	4	4	9	10	10	11	13	16	20



$$d(7) = d(a_{53}, a_{42}, a_{15}, a_{13}, a_{34}) = 32$$

$$d(7) \geq d_0$$

a_{ij}:	$\underline{a_{53}}$	$\underline{a_{42}}$	$\underline{a_{15}}$	$\underline{a_{14}}$	$\underline{a_{12}}$	$\underline{a_{13}}$	$\underline{a_{34}}$	a_{23}	a_{45}	a_{25}
l_{ij}:	3	4	4	9	10	10	11	13	16	20



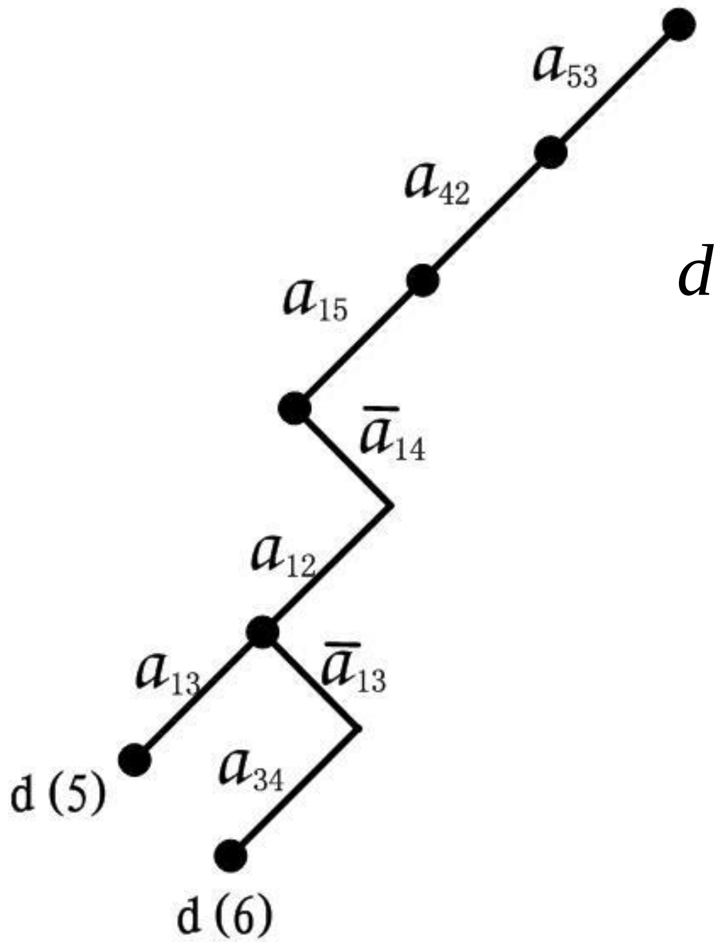
$$d(8)=d(a_{53},a_{42},a_{14},a_{12},a_{13})=36$$

$$d(8)\geq d_0$$

容易推算，之后深探得到的分枝，其总长都大于界值

因此，算法可以结束

a_{ij}:	a_{53}	a_{42}	a_{15}	a_{14}	a_{12}	a_{13}	a_{34}	a_{23}	a_{45}	a_{25}
l_{ij}:	3	4	4	9	10	10	11	13	16	20



$$d(6) = d(a_{53}, a_{42}, a_{15}, a_{12}, a_{34}) = 32$$

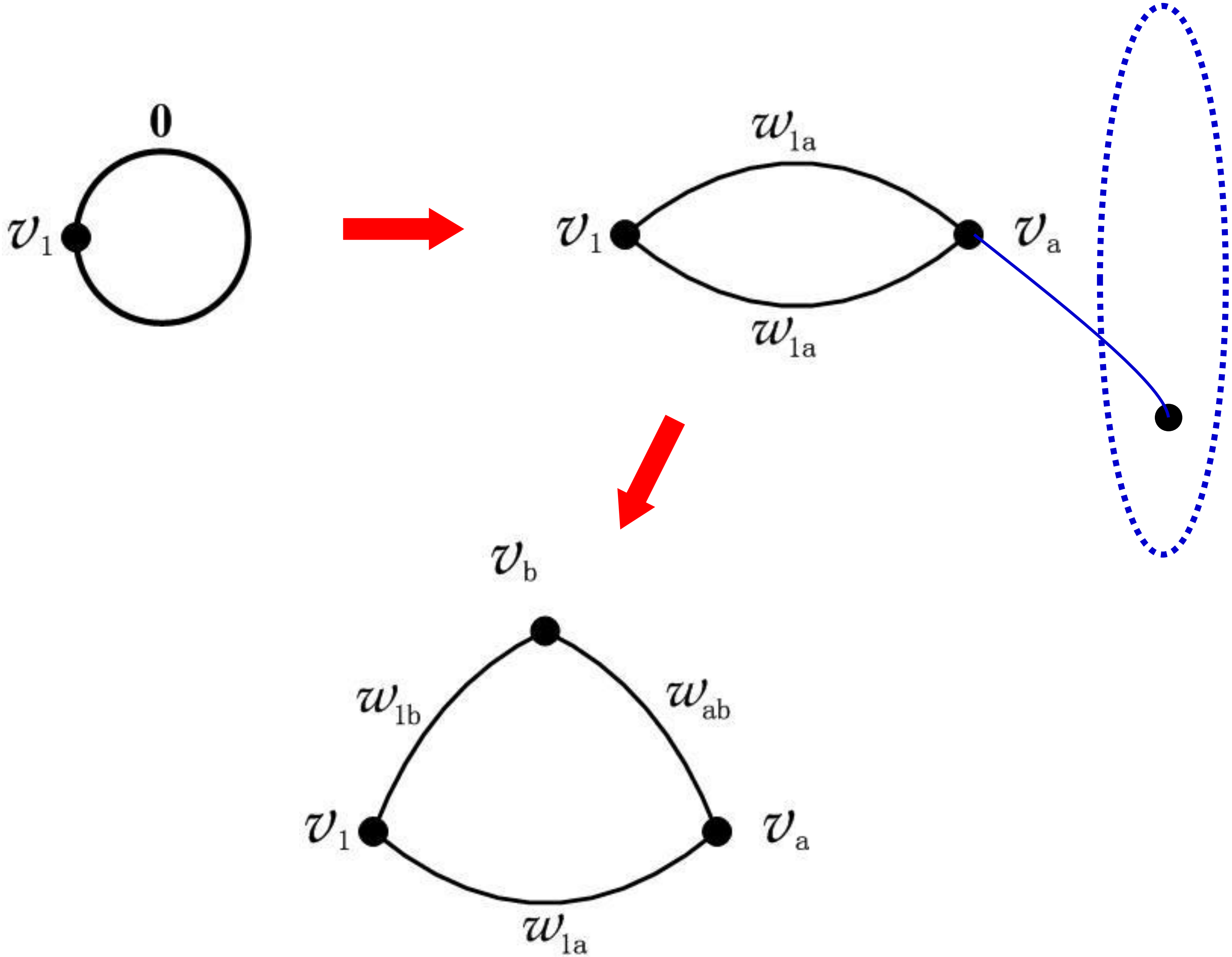
$$d_0 = 32$$

旅行商问题

- 分枝与界法小结：
  - 分枝与界法就是在不断地构造分枝与确定界值。
  - 一旦确定了界值，大于等于界值的分枝将不再继续搜索。
  - 最后一次得到的界值就是最优解
  - 一般情况下，分枝与界法通过“剪枝”，可以大量减少计算，因此优于枚举法
  - 但该方法的计算复杂度仍为$O(n!)$

旅行商问题

- 实际情况中，常采用近似算法，从而避免浩瀚的计算量
- 采用近似算法，往往也需要根据实际情况对原始问题增加一定的限制
- 旅行商问题可以增加如下限制：
  - G为无向正权图
  - 任意三结点之间权值符合三角不等式
- 在这些条件下，介绍“便宜”算法

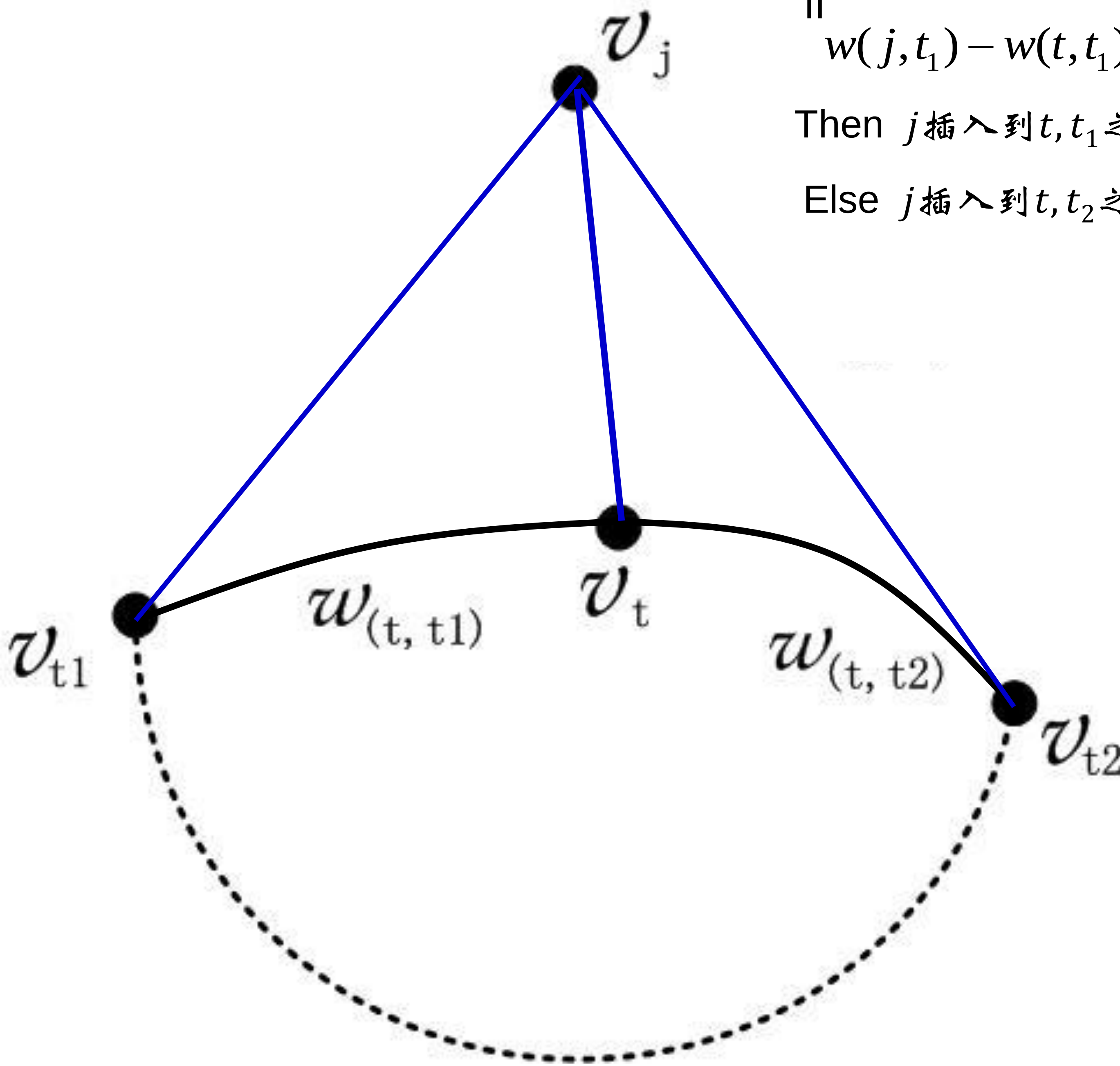


0
v_1
w_{1a}
w_{1a}
v_1
v_a
v_b
w_{1b}
w_{ab}
v_1
v_a
w_{1a}

if
$$w(j,t_1)-w(t,t_1)\le w(j,t_2)-w(t,t_2)$$

Then j插入到t,t_1之间

Else j插入到t,t_2之间



旅行商问题

- 定理2.5.1 设正权完全图的边权满足三角不等式，其旅行商问题的最佳解是O_n，便宜算法的解是T_n，则

$$\frac{T_n}{O_n} < 2$$

证明：（自学）

旅行商问题－小结

- 问题引出，及计算复杂度分析
- 分枝与界法
- “便宜”算法

主要内容

- 2.1 道路与回路
- 2.2 道路与回路的判定
- 2.3 欧拉道路与回路
- 2.4 哈密顿道路与回路
- 2.5 旅行商问题
- **2.6 最短路径**
- 2.7 关键路径
- 2.8 中国邮路问题

最短路径

- 最短路径问题按照实际问题的模型，包括三类模型：
  - (1) 某两结点之间的最短路径
  - (2) 某结点到其他各结点的最短路径
  - (3) 任意两结点之间的最短路径
- 容易分析，模型(2)如果能够解决，模型(1)和(3)自然可以解决。
- 本节将重点就模型(2)进行讨论

Tsinghua University

最短路径

- 对于研究对象－图G，依据边权值的特点，可以有如下几种情况：
  - (1) 均大于零（正权图）
  - (2) 均等于1
  - (3) 为任意实数
- 本节将就这三种情况分别进行分析

最短路径

- 相关概念及前提：
  - 不失一般性，我们研究v_1到其他各结点的最短路径
  - v_1到v_i的一条路径$P(i)$的长度记为$\pi(i)$，且

$$\pi(i)=\sum_{e\in P(i)} w(e)$$

其中，$w(e)$ 表示$e=(v_j, v_k)$ 的权，也记为w_{jk}

  - v_1到v_i的最短路径就是求$\pi(i)$的最小值

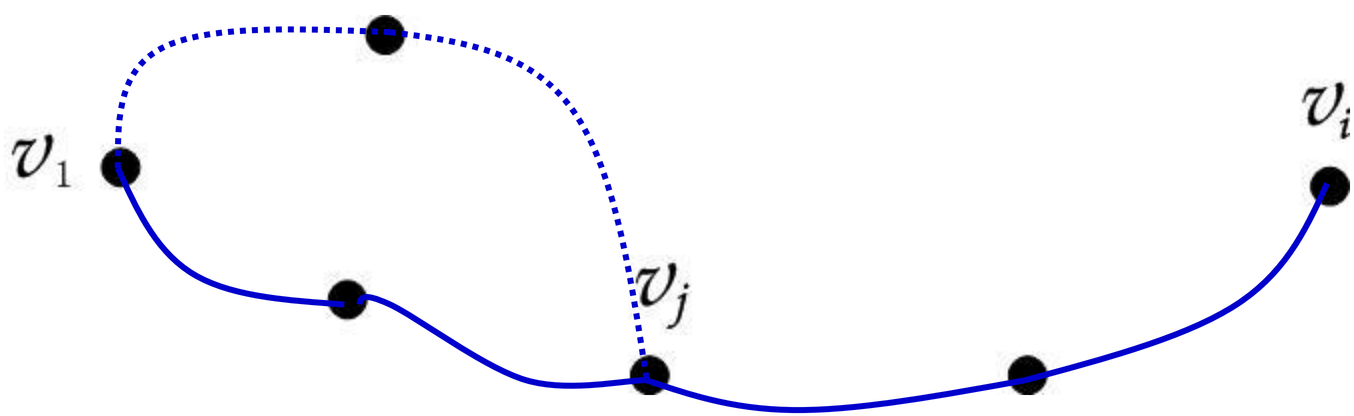
最短路径

- 思考：
  - v_1到v_i的最短路径中，是否会出现回路？
    - 若回路长度为正，则删掉回路可以得到更短路径
    - 若回路长度为负，则不存在最短路径
  - 本节只讨论无负长回路的图

最短路径 – 正权图

- 引理2.6.1 正权图G中，如果$P(i)$是v_1到v_i的一条最短路，且$v_j \in P(i)$，则$P(j)$是v_1到v_j的一条最短路。

证明：



最短路径－正权图

- 引理2.6.2 正权图中任意一条最短路径的长度大于其局部路径长度

证明：结论显然。

最短路径 – 正权图

- *Dijkstra*(狄克斯特拉)算法
  - 每个结点用从源结点沿已知最佳路径到本结点的距离来标注，标注分为临时性标注和永久性标注，最新永久标注节点为工作节点。
  - 初始时，所有结点都为临时性标注，标注值为无穷大；
  - 将源结点标注为0，且为永久性标注，并令其为工作结点；

最短路径－正权图

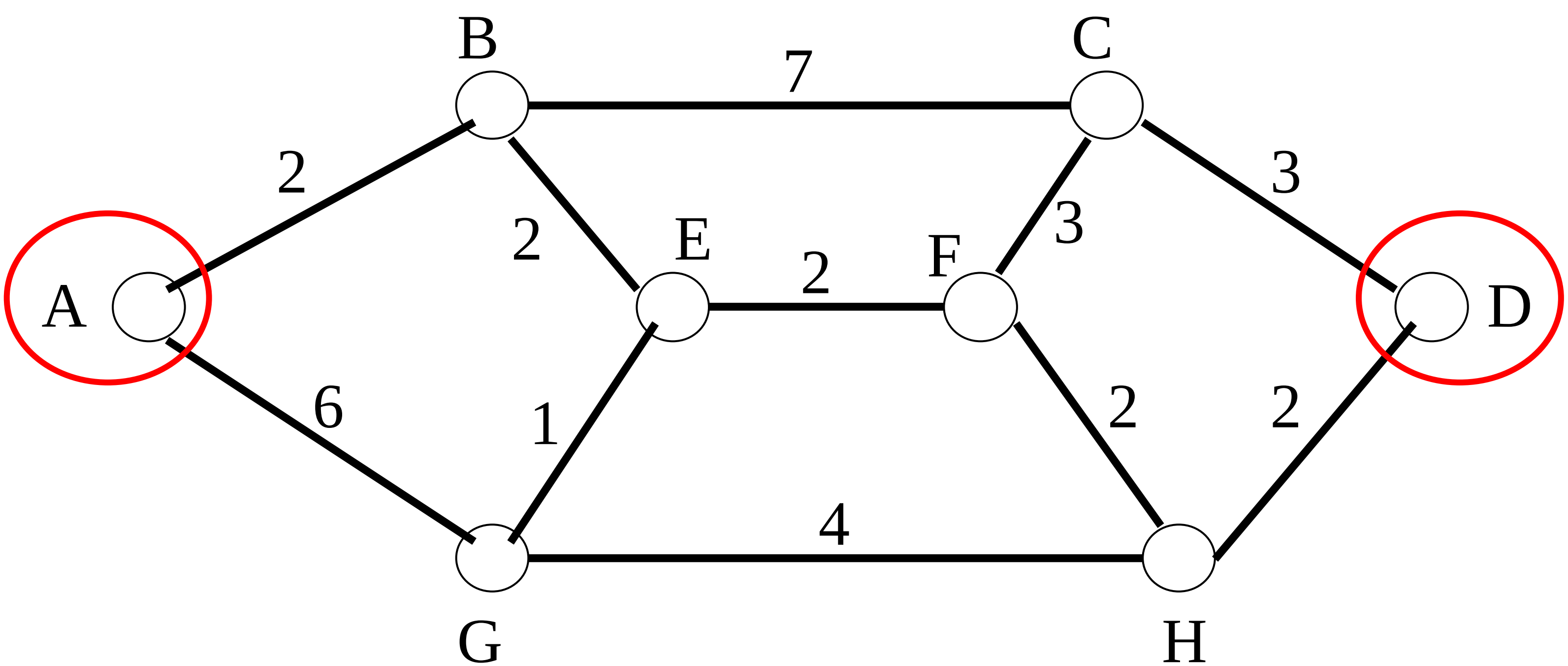
4) 检查与工作结点相邻的临时性结点，若该结点到工作结点的距离与工作结点的标注之和小于该结点的临时标注，则用新计算得到的和重新标注该结点

5) 在整个图中查找具有最小值的临时性标注结点，将其变为永久性结点，并成为下一轮检查的工作结点；

6) 重复第四、五步，直到目的结点成为工作结点。

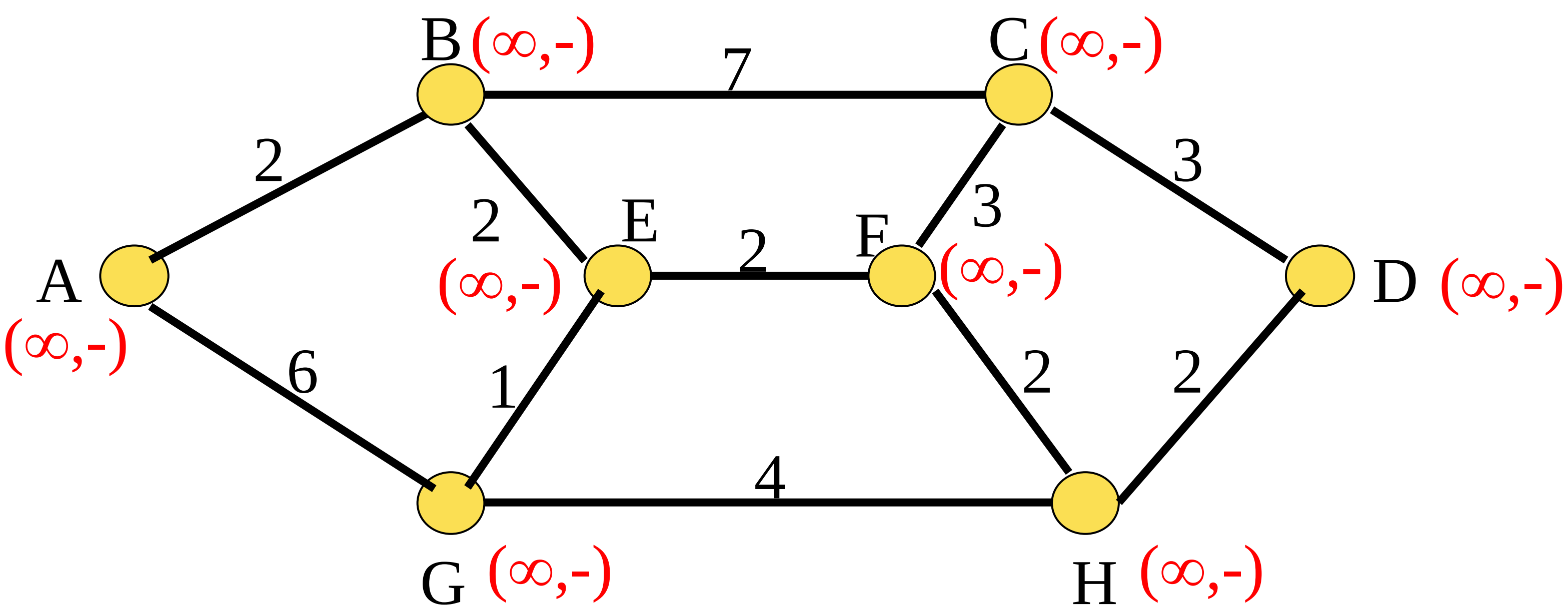
Dijkstra算法应用举例：找出从A到D的最短路径

1) 每个结点用从源结点沿已知最佳路径到本结点的距离来标注，标注分为临时性标注和永久性标注



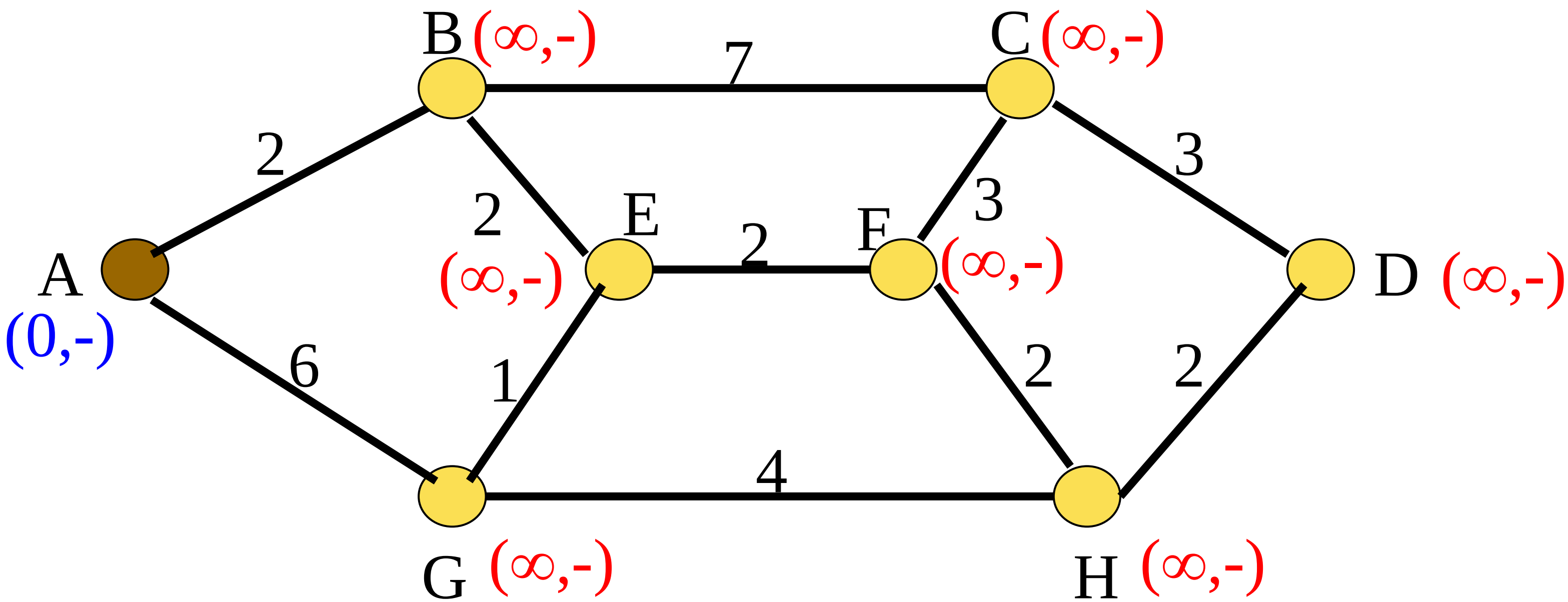
$Dijkstra$算法应用举例：找出从A到D的最短路径

2) 初始时，所有结点都为临时性标注，标注为无穷大；



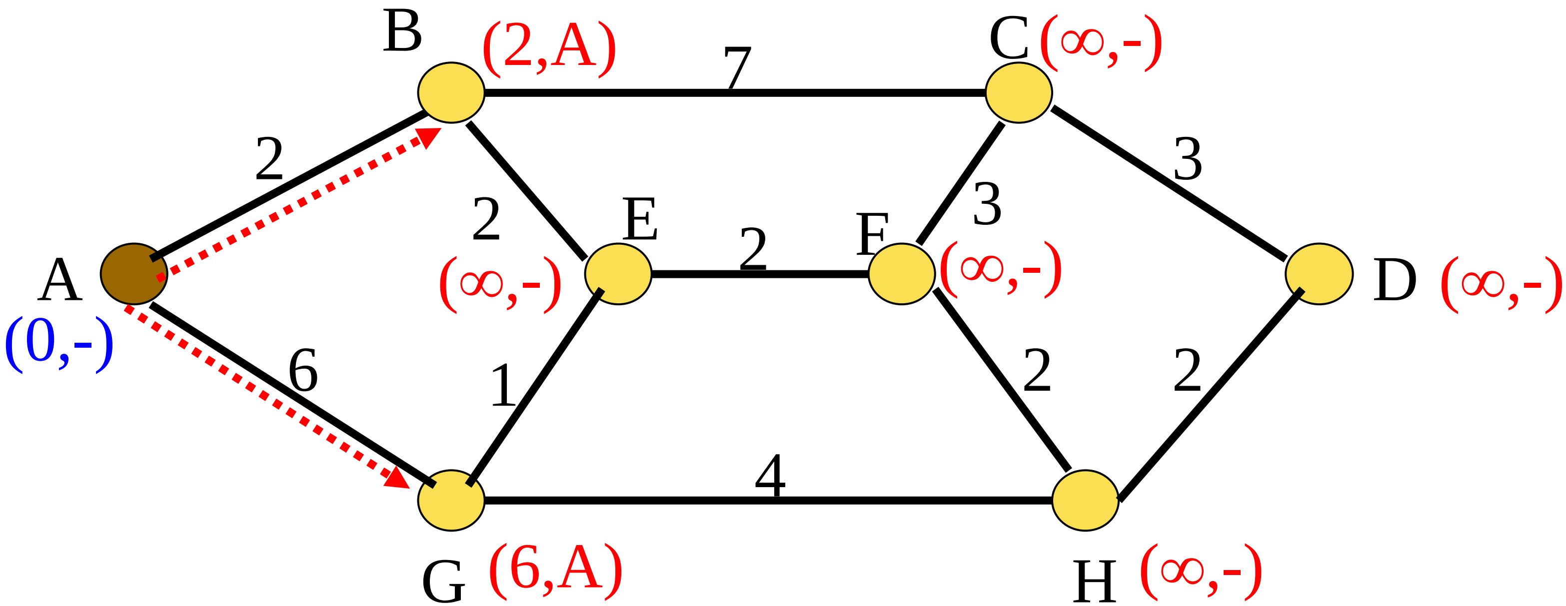
Dijkstra算法应用举例：找出从A到D的最短路径

3) 将源结点标注为0，且为永久性标注，并令其为工作结点；



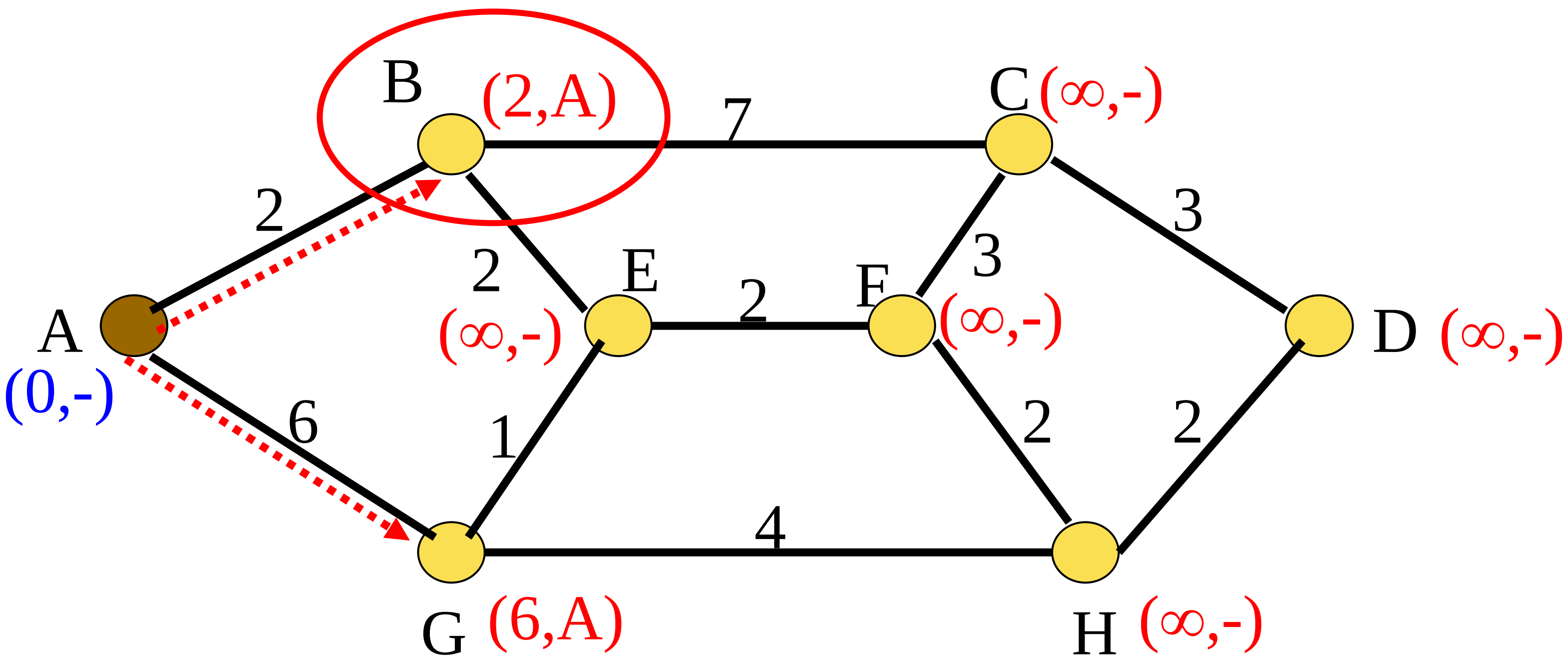
Dijkstra算法应用举例：找出从A到D的最短路径

4) 检查与工作结点相邻的临时性结点，若该结点到工作结点的距离与工作结点的标注之和小于该结点的临时标注，则用新计算得到的和重新标注该结点



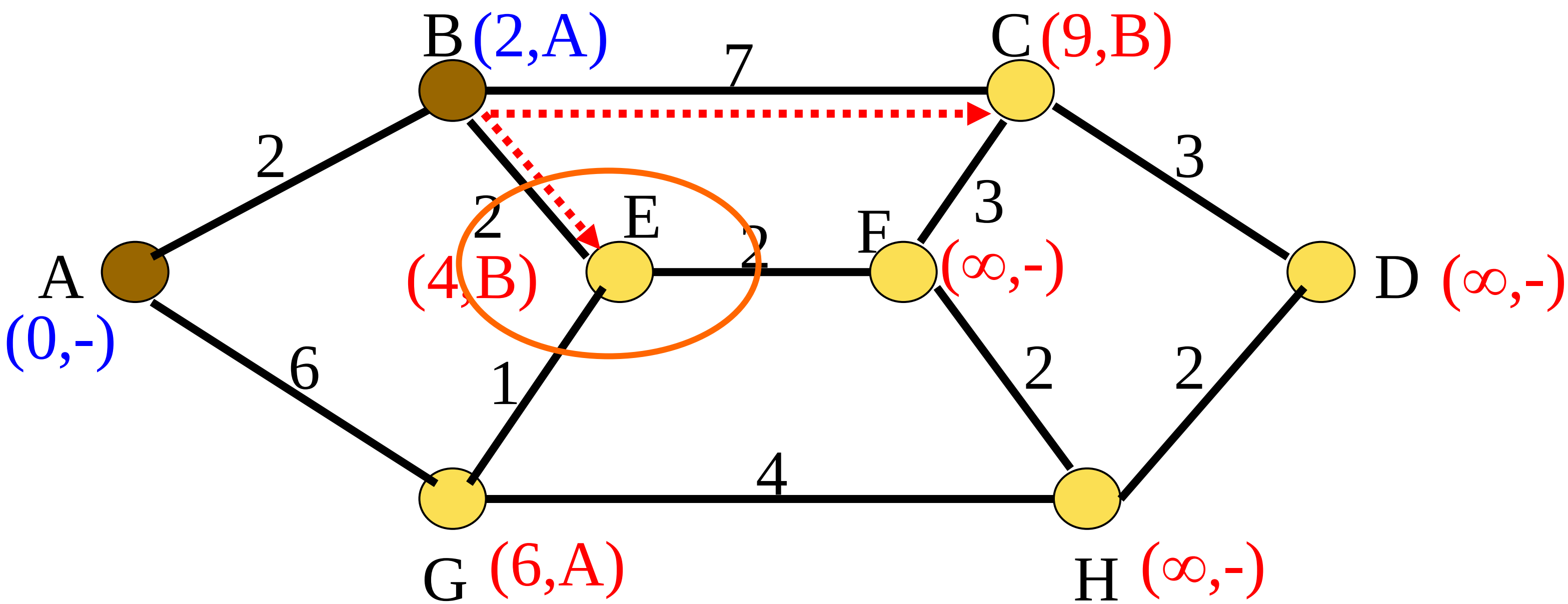
Dijkstra算法应用举例：找出从A到D的最短路径

5) 在整个图中查找具有最小值的临时性标注结点，将其变为永久性结点，并成为下一轮检查的工作结点；

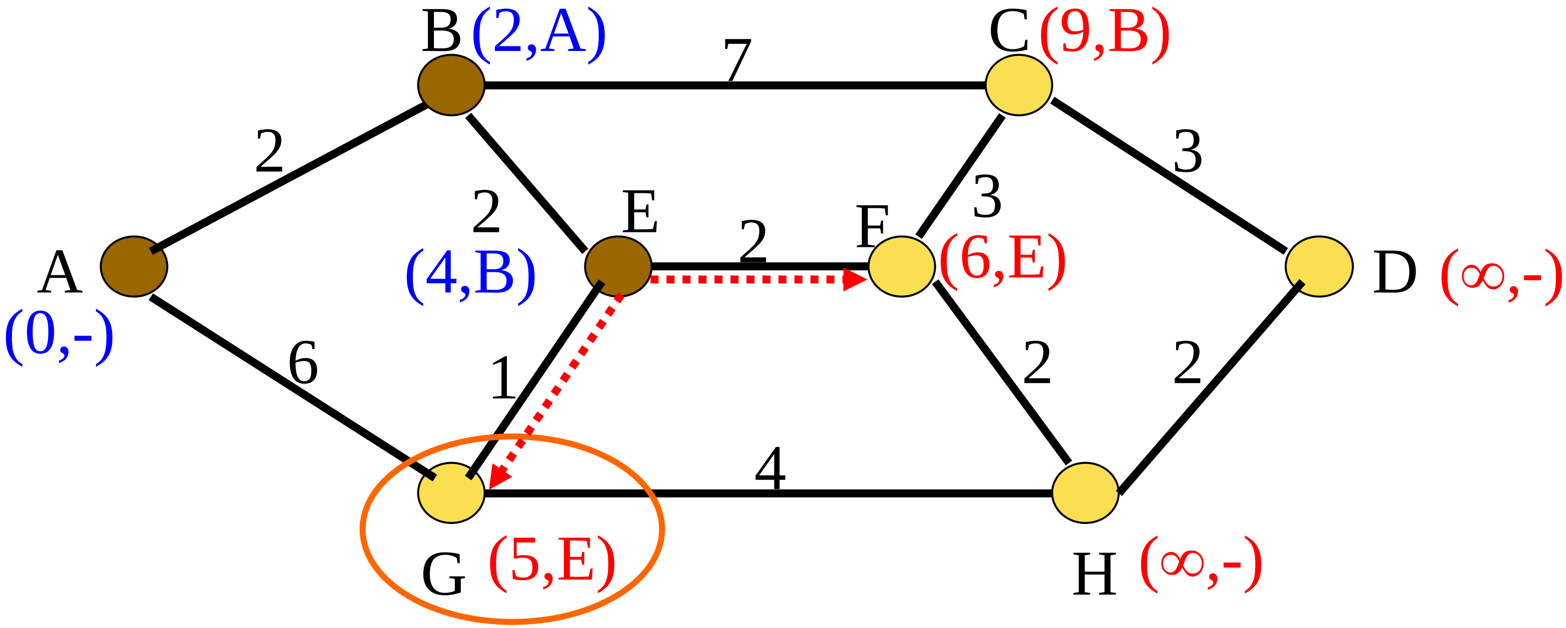


*Dijkstra*算法应用举例：找出从A到D的最短路径

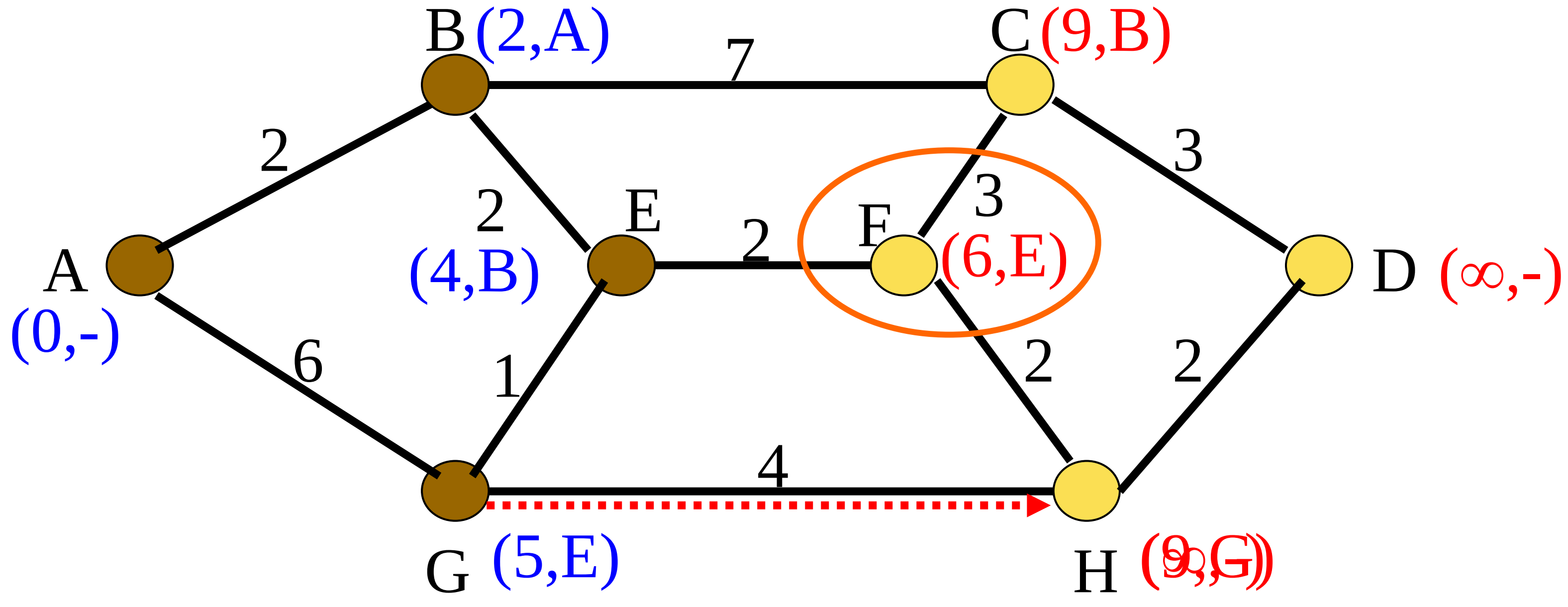
6) 重复第四、五步，直到目的结点成为工作结点。



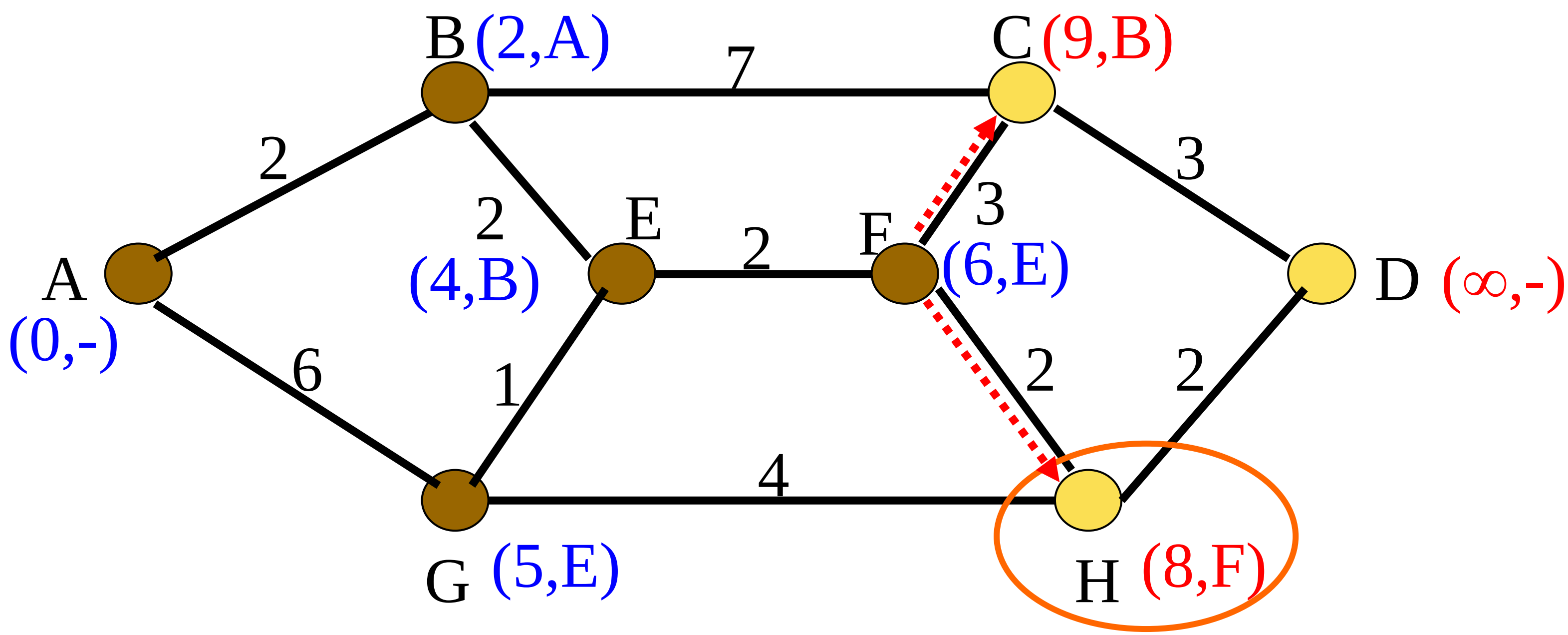
*Dijkstra*算法应用举例：找出从A到D的最短路径



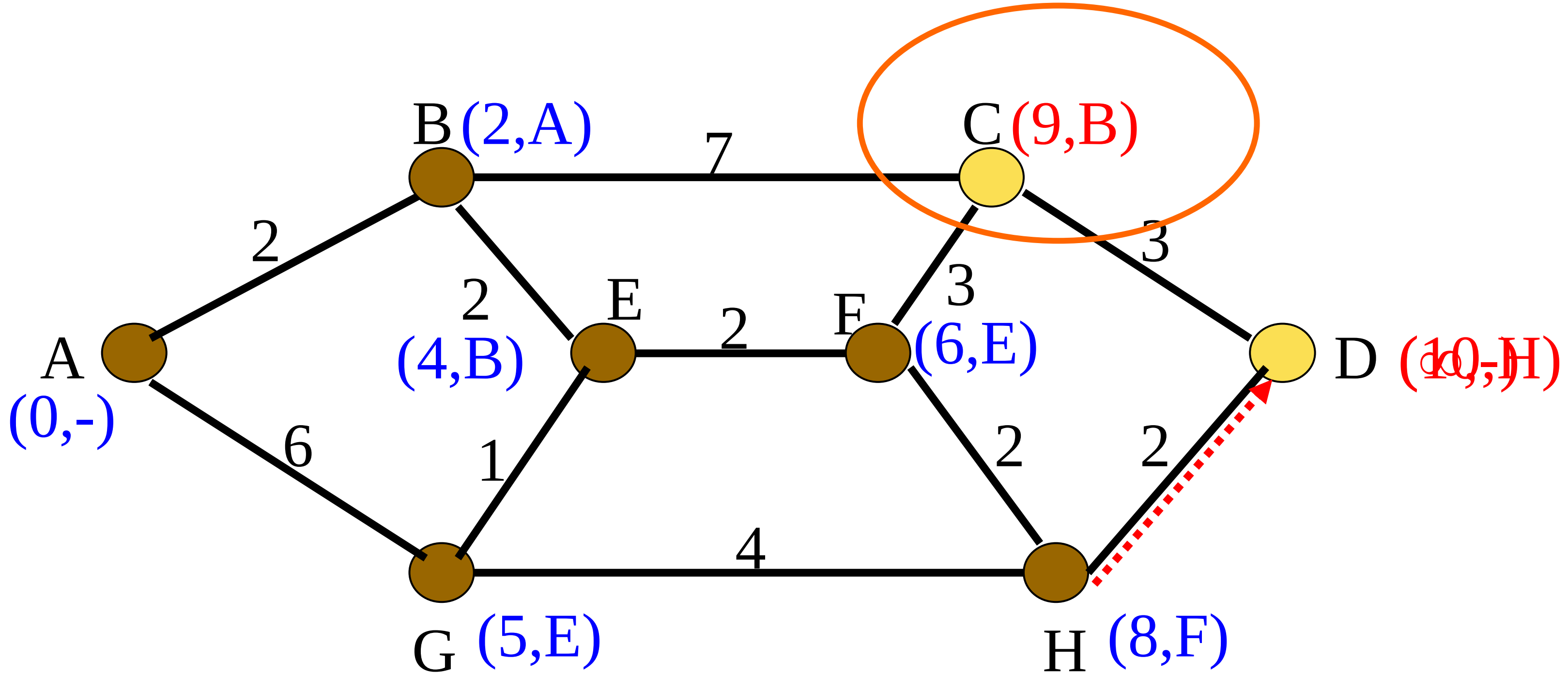
*Dijkstra*算法应用举例：找出从A到D的最短路径



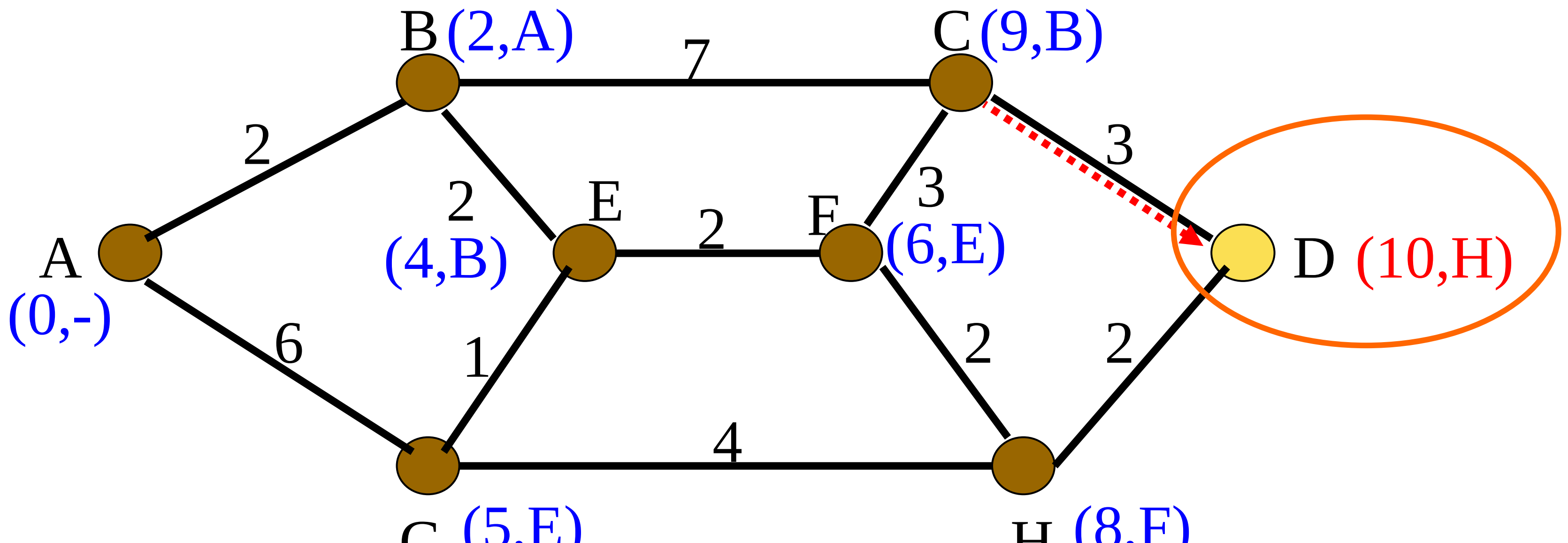
Dijkstra算法应用举例：找出从A到D的最短路径



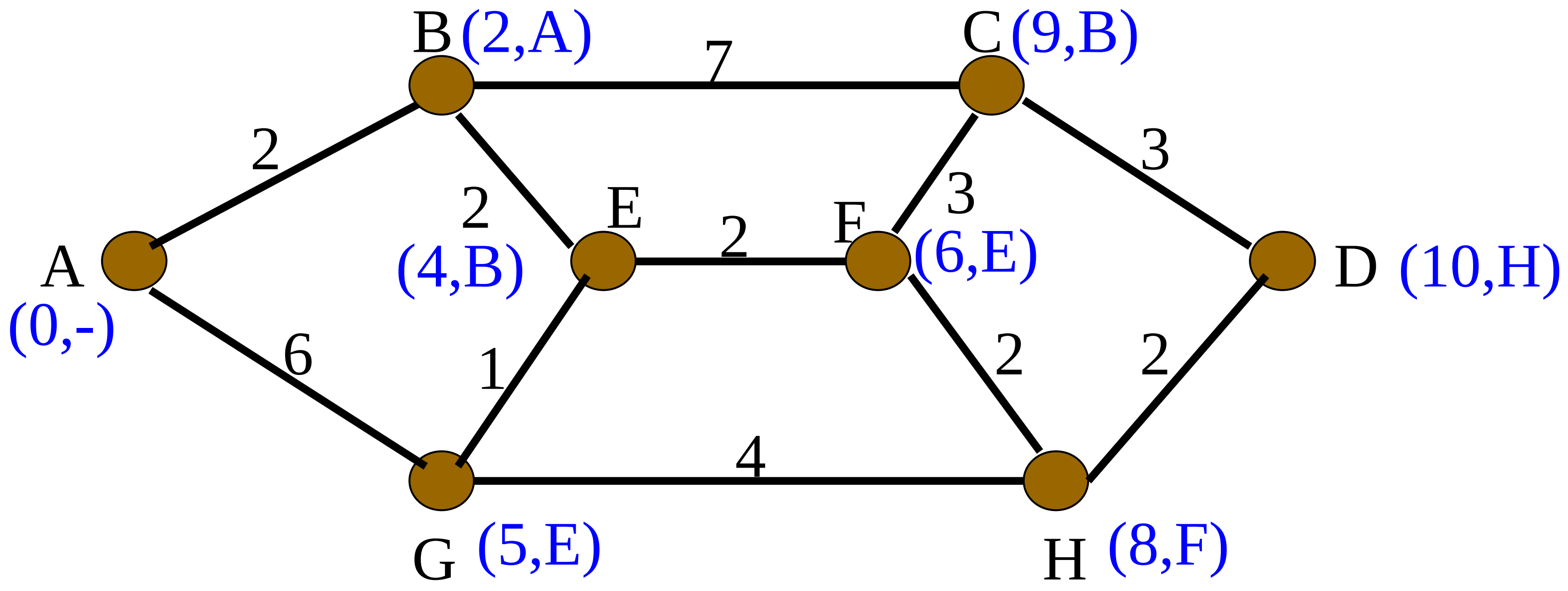
*Dijkstra*算法应用举例：找出从A到D的最短路径



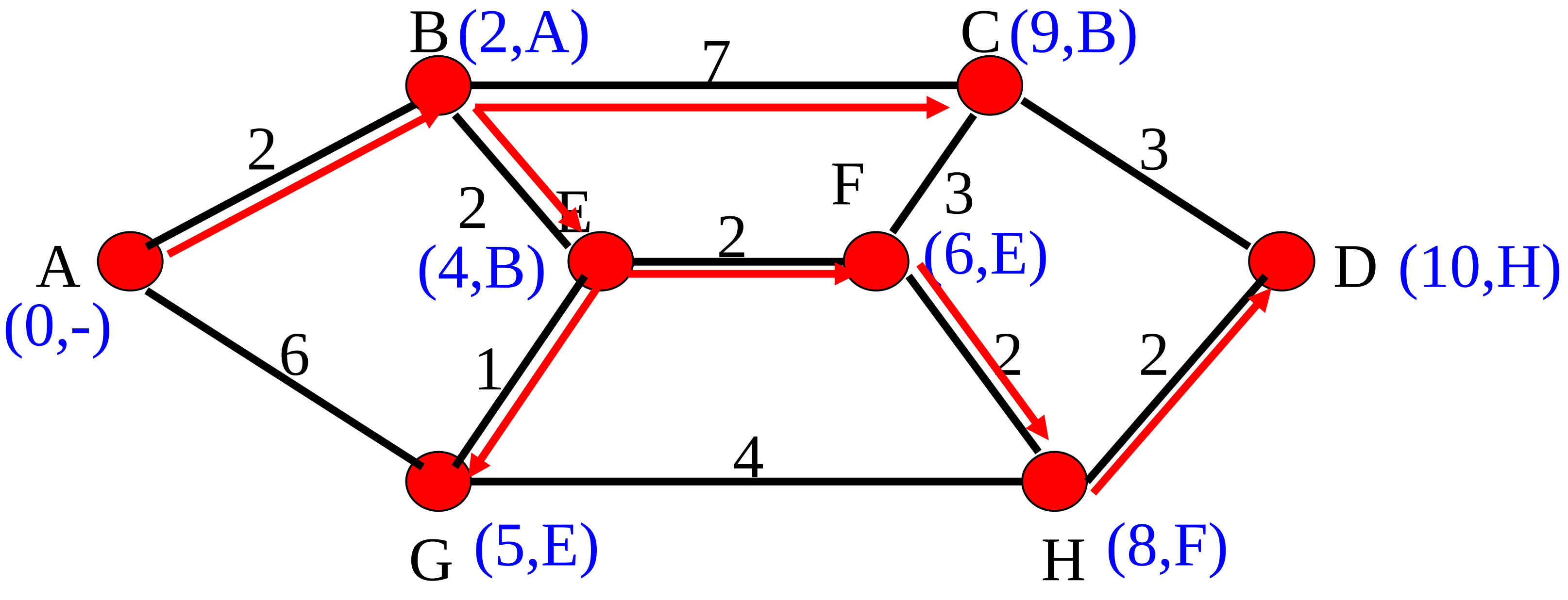
*Dijkstra*算法应用举例：找出从A到D的最短路径



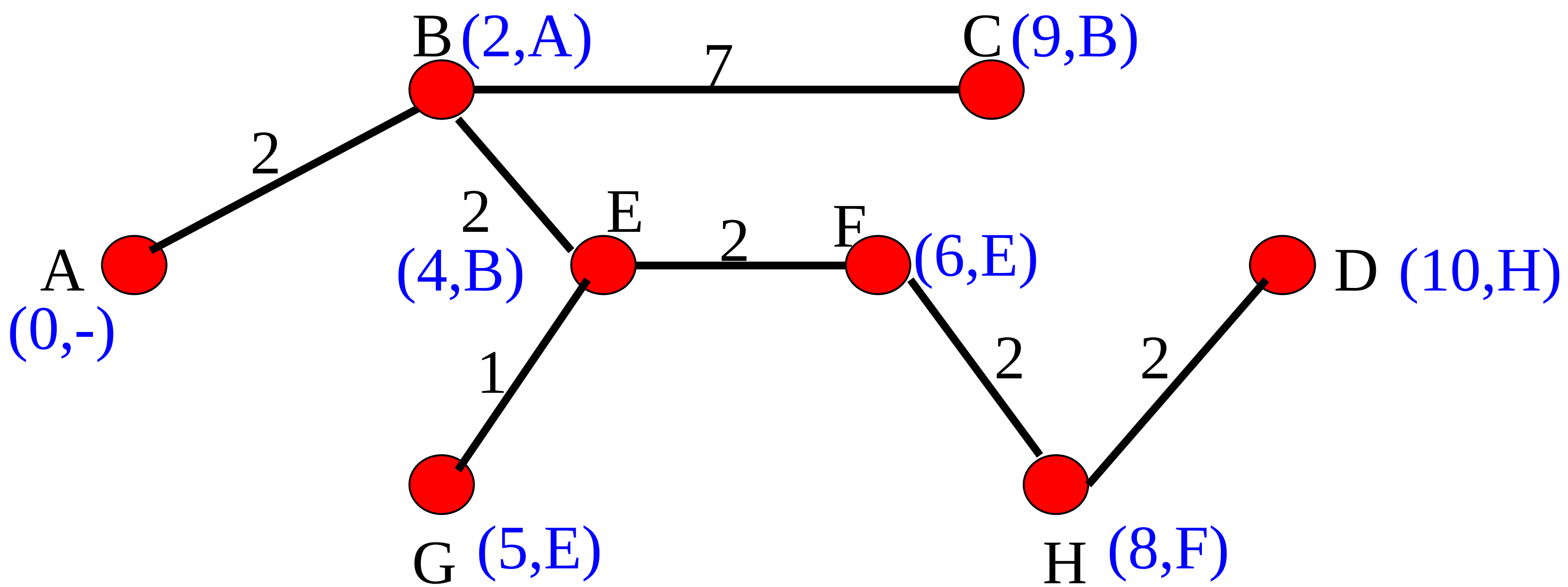
*Dijkstra*算法应用举例：找出从A到D的最短路径



*Dijkstra*算法应用举例：找出从A到D的最短路径



*Dijkstra*算法应用举例：找出从A到D的最短路径



最短路径 – 正权图

- 小结：
  - “临时标注结点”集合对应教材中所描述算法中的 $\overline{S}$，请大家复习时注意参照
  - $Dijkstra$算法的计算复杂度为$O(n^2)$

Tsinghua University

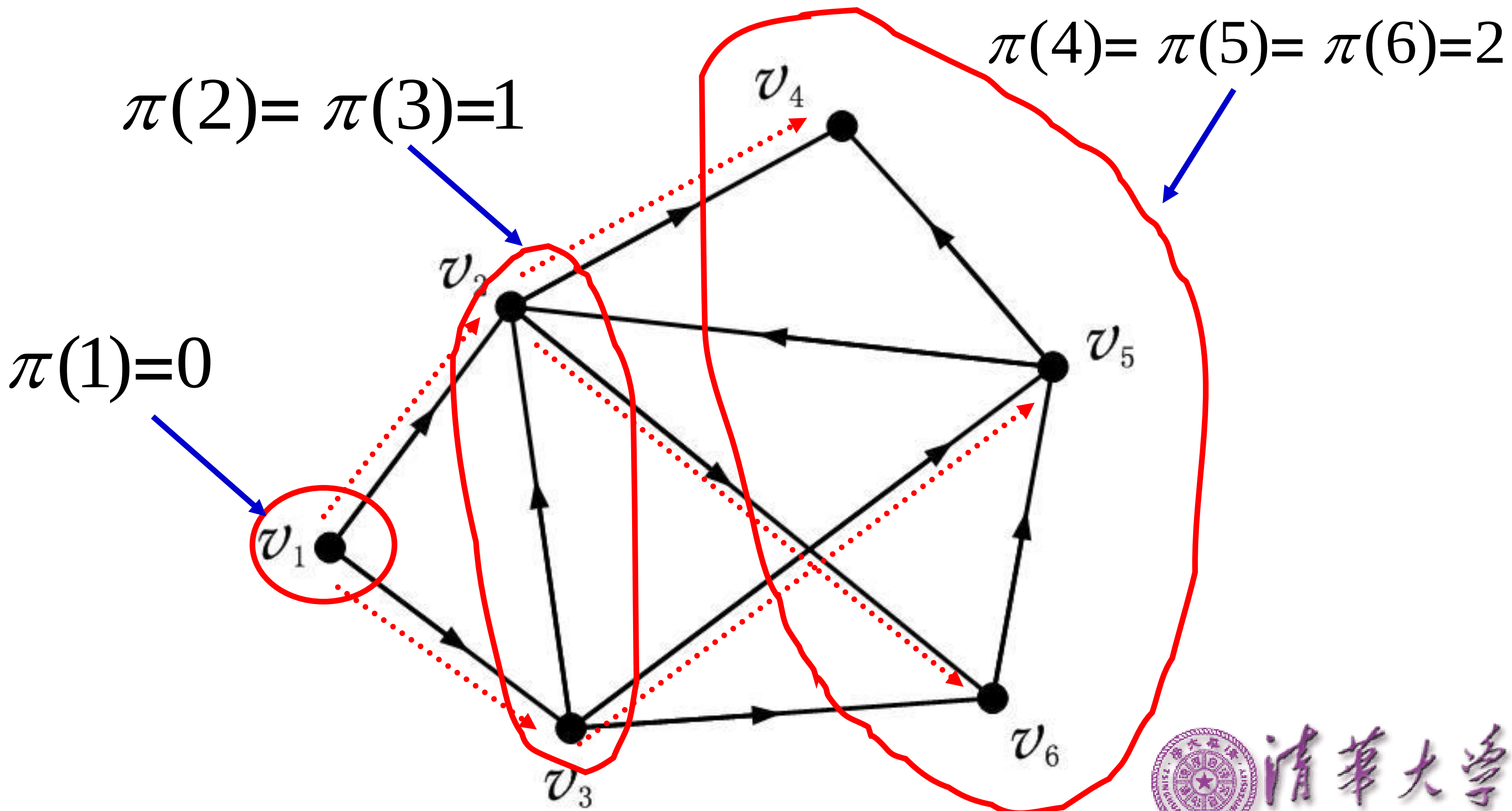
最短路径－正权图

- 课后思考：

Dijkstra算法为什么是正确的？

最短路径－权为1

- 边权值均为1的图，其最短路径算法可以用

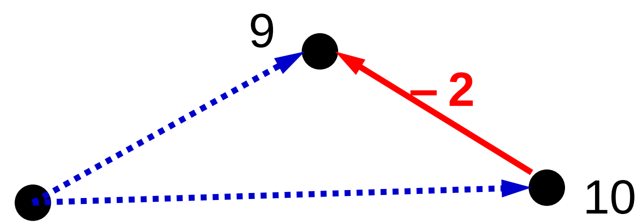


最短路径－权为1

- 小结：
  - 如果图G是以正向表或邻接表的数据结构表示，则本算法的计算复杂性为$O(m)$

最短路径－边权任意

- 当图中边出现负权时，情况将比较复杂。
- *Dijkstra* 算法的过程是由近及远、逐点扩展，得到某点的最短路径后不会再变。
- 但有负权边时，*Dijkstra*算法可能会失效。



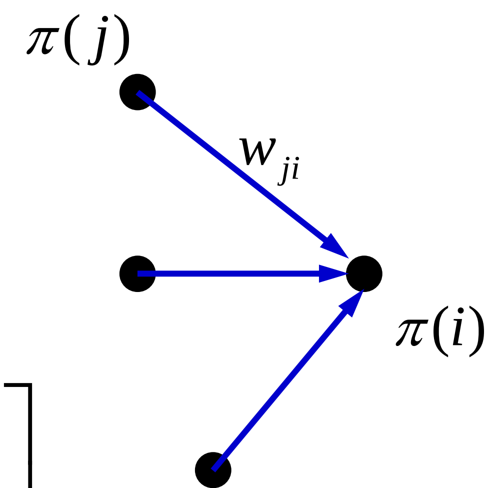
- 注意，有负权边，不一定存在负长回路。

最短路径－边权任意

- $Ford$算法给出问题的解决方案，算法描述如下：
  - (1) 置 $\pi(1)=0, \pi(i)=\infty, \qquad i=2,3,\cdots,n$
  - (2) $\boldsymbol{i}$ 从2到$\boldsymbol{n}$，令

$$\pi(i) \leftarrow \min\left[\pi(i), \min_{j\in\Gamma_i^-}(\pi(j)+w_{ji})\right]$$

  - (3) 若全部$\pi(i)$都没变化，结束。否则转(2)



Tsinghua University
清華大學

最短路径－边权任意

- 从算法的特点来看：算法每进行一步计算得到的$\pi(i)$，都是从v_1到v_i之间最短路径长度的上界。
- 图中不存在负长回路，因此v_1到v_i之间最短路径长度就是$\pi(i)$的下界。
- 以下证明算法收敛后得到的值就是从v_1到v_i的最短路径长度。

最短路径－边权任意

证明：

设算法结束时，对某个结点v_s有$\pi(s)$。

设v_1到v_s之间存在某条路径如下：

$$\mu = \left(1, t_h, t_{h-1}, \cdots, t_1, s\right)$$

根据算法步骤(2)，我们可以得到如下等式：

$$\pi(s) = \min\left(\pi(s), \min\left(\left(\pi(t_1) + w(t_1, s)\right), \cdots\right)\right)$$

Tsinghua University

最短路径－边权任意 $\mu=(1,t_h,t_{h-1},\cdots,t_1,s)$

证明（续）：显然，我们可以得到不等式：

$$\pi(s)\le\pi(t_1)+w(t_1,s)$$

$$\Rightarrow\quad \pi(s)-\pi(t_1)\le w(t_1,s)$$

同理，

$$\pi(t_1)-\pi(t_2)\le w(t_2,t_1)$$

$$\cdots\cdots$$

$$\pi(t_h)-\pi(1)\le w(1,t_h)$$

即：$\pi(s)\le w(t_1,s)+w(t_2,t_1)+\cdots+w(1,t_h)=w(\mu)$

证毕！

最短路径 – 边权任意

- 思考：*FORD*算法为什么一定会收敛，会不会发生一直在递减但永不收敛的情况？
- 思考：随意给定一张图，用那种算法会比较合适？
- 思考：在正权图中采用*FORD*算法，同*Dijkstra*算法会有什么区别？

最短路径－小结

- 三类模型：
  - (1) 某两结点之间的最短路径
  - (2) 某结点到其他各结点的最短路径
  - (3) 任意两结点之间的最短路径
- 三种情况：
  - (1) 均大于零（正权图）：*Dijkstra*算法
  - (2) 均等于1：广探法
  - (3) 为任意实数：*Ford*算法

Tsinghua University

主要内容

- 2.1 道路与回路
- 2.2 道路与回路的判定
- 2.3 欧拉道路与回路
- 2.4 哈密顿道路与回路
- 2.5 旅行商问题
- 2.6 最短路径
- **2.7 关键路径**
- 2.8 中国邮路问题

清華大學 Tsinghua University

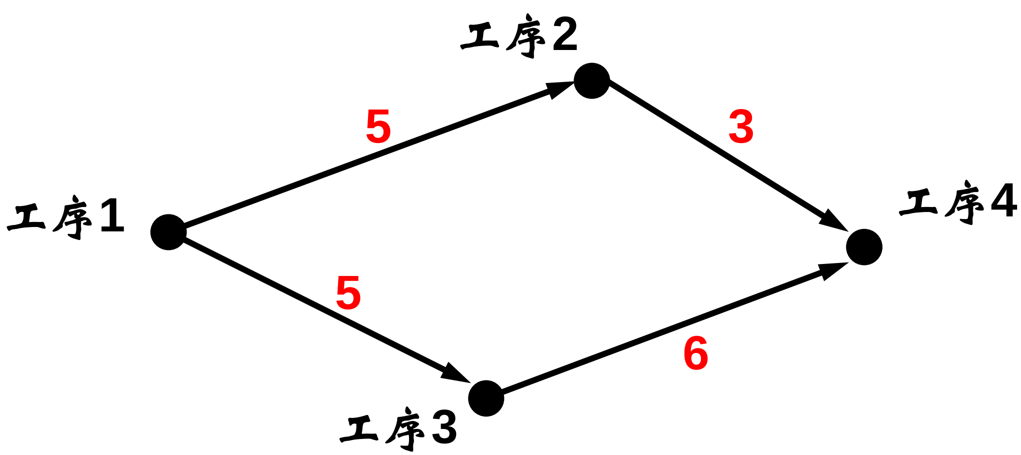
关键路径

- 现实中有很多工程问题，建水坝、造飞机、组装机床、软件开发...，都会包含很多工序。
- 工序与工序之间很多都存在先后次序关系，一般这些次序关系是预知的。
- 对于工程领导人员来说：
  - 了解工程最少需要多少时间
  - 要害工序是哪些
- 此类问题，如何转化为图论问题解决？

关键路径 – PT图

- PT (Potentialtask graph)图：
  - 用结点表示工序
  - 用有向边表示工序间的次序关系
  - 用边权值表示工序所需要的时间

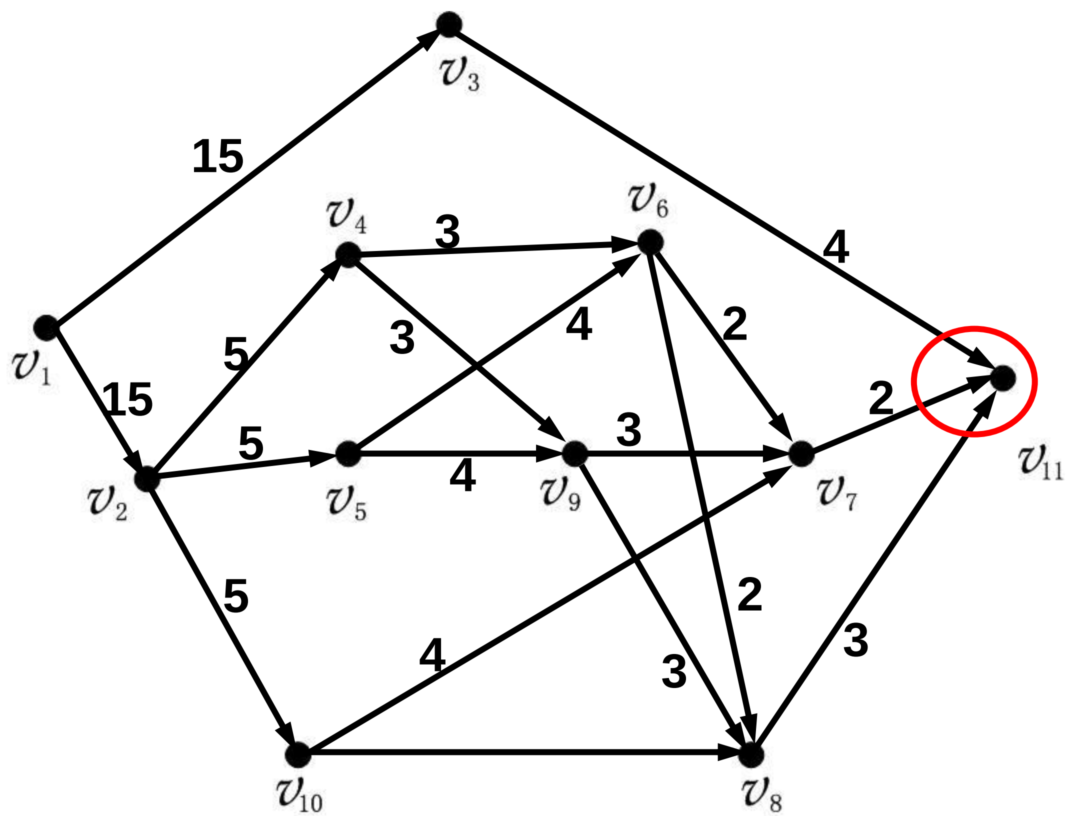
关键路径 – PT图



- PT图的特点：
  – 从某结点出发的边，权值均相等
  – 必定不存在有向回路
  – 存在没有入度和没有出度的结点

例：

序号	名称	所需时间	先序工序
1	基础设施	15	
2	下部砌砖	5	1
3	电线安装	4	1
4	圈梁支模	3	2
5	水暖管道	4	2
6	大梁安装	2	4,5
7	楼板吊装	2	6,9,10
8	楼板浇模	3	6,9,10
9	吊装楼梯	3	4,5
10	上部砌砖	4	2



整个工程最短完成时间应该是：

从v_1到v_{11}的最长路径！

该路径也是工程的关键路径！

关键路径

- 引理2.7.1 不存在有向回路的图G中，一定存在负度及正度为零的结点。

证明（构造法）：

- 在G中构造一条极长的有向道路P，并设P的起点为v_i，终点为v_j，则一定有$d^-(v_i)=0, d^+(v_j)=0$
- 否则，假定 $d^-(v_i)\neq 0$，则一定有边$(v_k, v_i)\in E(G)$

  若 $v_k\in P$，则G存在有向回路；

  若 $v_k\notin P$，则P不是极长道路。因此 $d^-(v_i)$=0
- 同理，可证 $d^+(v_j)=0$ 　　**证毕！**

清華大學 Tsinghua University

关键路径

- 引理2.7.2 设G不存在有向回路，可以将G的结点重新编号为$v_1', v_2', \cdots, v_n'$，使得对任意的边$(v_i', v_j') \in E(G)$，都有$i < j$

证明：

- 根据引理2.7.1，G中存在v_i，满足$d^-(v_i) = 0$，对之重新编号为v_1'。
- 在G中删掉v_i，得到$G' = G - v_1'$，可知，G'为G的导出子图，因此没有有向回路，因此必然存在负度为零的结点

关键路径

- 将G'中负度为零的结点编号为v_2'，再做$G' - v_2'$
- 依次类推，可以将G的全部结点重新编号。
- 此时，G中所有边的编号均为从编号小的结点指向编号大的结点，否则与编号原则相悖

证毕！

关键路径

- PT图中最长路径算法：
  - (1) 对结点重新编号为 $v_1', v_2', \cdots, v_n'$
  - (2) $\pi(v_1') \leftarrow 0$
  - (3) 对于 j 从 2 到 n，令

$$\pi(v_j') = \max_{v_i' \in \Gamma^-(v_j')} (\pi(v_i') + w(v_i', v_j'))$$

  - (4) End!

该算法计算复杂性为 $O(m)$

清華大學 Tsinghua University

关键路径

- 上述算法将得到最长路径就是工程的关键路径。
  - 路径长度即为工程的最早完成时间。
- 思考：
  - 对于非关键路径上的工序，是否可以延误，如果可以，最多可延误多长时间？

关键路径

- 设$\pi(v_n)$是工程完工的最早时间，设工序i到工程完工最少需要的时间为$\pi(v_i, v_n)$，则工序i的最晚启动时间应该是

$$\tau(v_i) = \pi(v_n) - \pi(v_i, v_n)$$

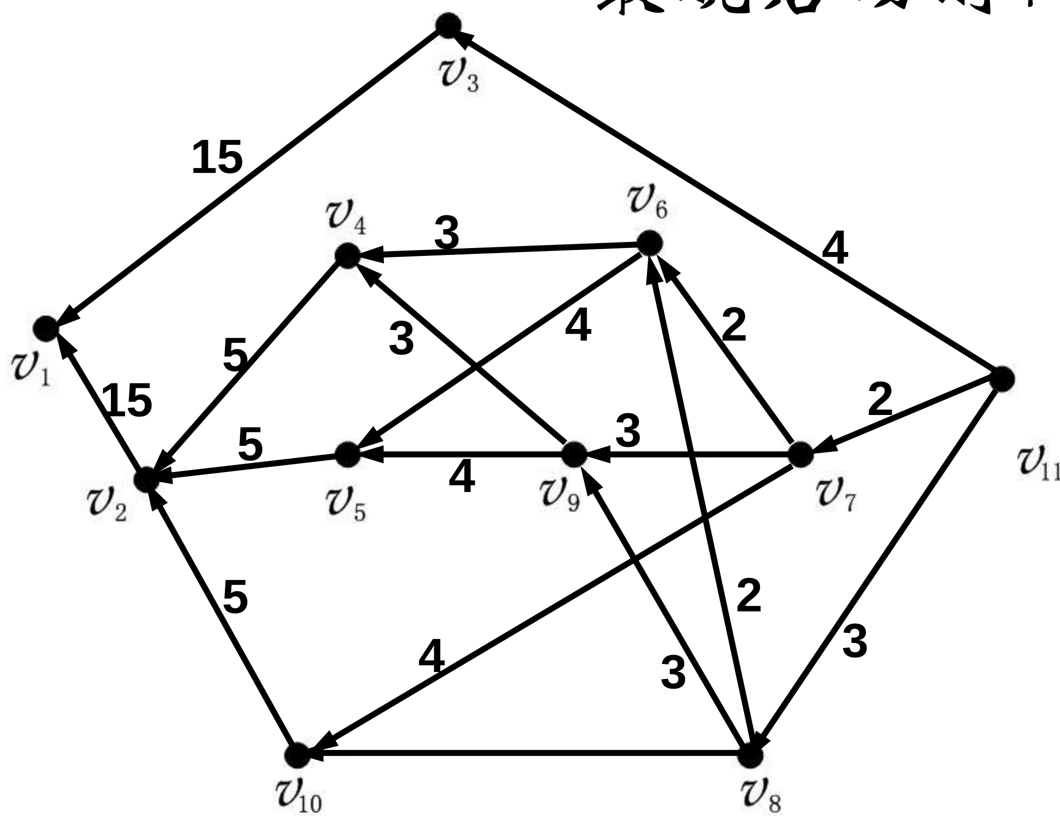
$\pi(v_i, v_n)$? $\longrightarrow$ v_i到v_n的最长路长度

Tsinghua University

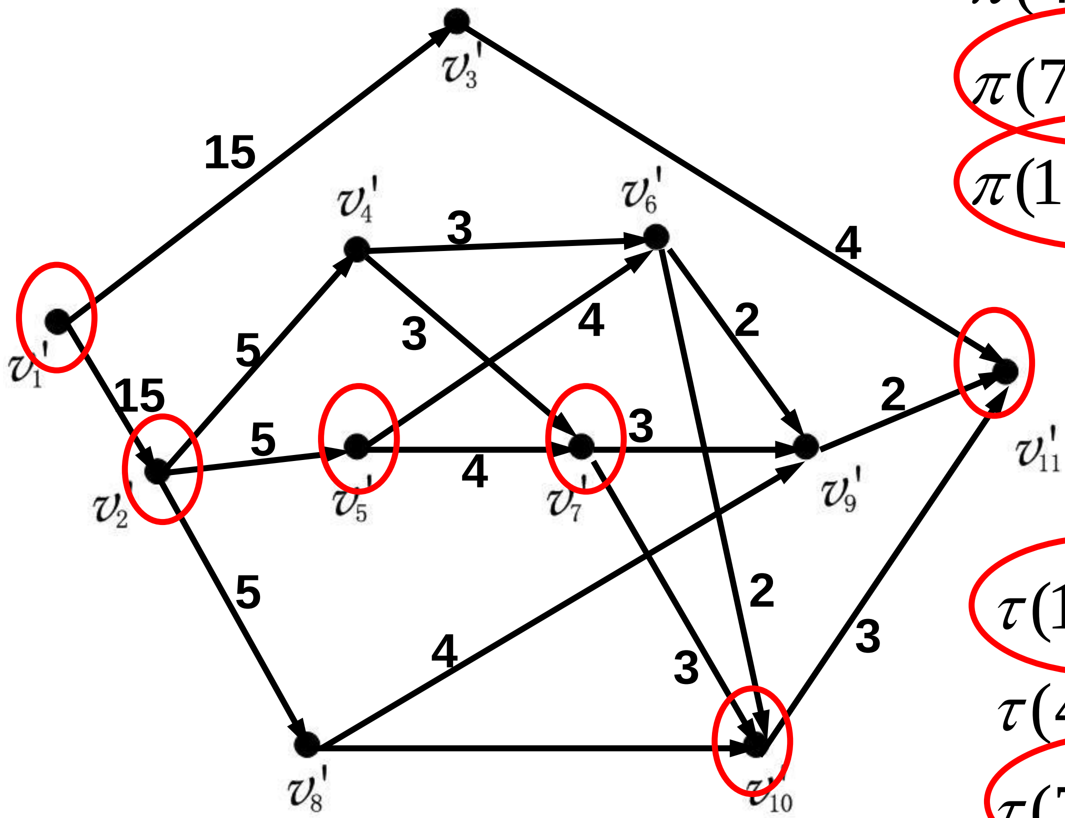
这样每个结点v_i有两个值：

最早启动时间 $\pi(v_i)$

最晚启动时间 $\tau(v_i)$



允许延误时间为： $t(v_i) = \pi(v_i) - \tau(v_i)$



$\pi(1')=0, \pi(2')=15, \pi(3')=15,$
$\pi(4')=20, \pi(5')=20, \pi(6')=24,$
$\pi(7')=24, \pi(8')=20, \pi(9')=27,$
$\pi(10')=27, \pi(11')=30$

$\tau(1')=0, \tau(2')=15, \tau(3')=26,$
$\tau(4')=21, \tau(5')=20, \tau(6')=25,$
$\tau(7')=24, \tau(8')=23, \tau(9')=28,$
$\tau(10')=27, \tau(11')=30$

从上面可以看出，最长路径即关键路径上各工序是不允许延误的，否则会延误整个工程进度

关键路径 – PERT图

- PERT (Programme evaluation and review technique) 图：
  - 采用有向边表示工序，权值表示工序所需要的时间
  - 关键路径算法与PT图相同
  - 工序最晚启动时间算法略有不同

关键路径－小结

- PT图
  - 如何用PT图表示“工程”
  - 如何求工程进度的关键路径
  - 如何求每个工序的最晚启动时间
- PERT图
  - 自学
  - 要求同PT图

主要内容－总结

- 2.1 道路与回路
- 2.2 道路与回路的判定
- 2.3 欧拉道路与回路
- 2.4 哈密顿道路与回路
- 2.5 旅行商问题
- 2.6 最短路径
- 2.7 关键路径
- 2.8 中国邮路问题

作业

- 课后 13，17
- 第一章课后第6题
- 练习题：求下图中各顶点的最早完成时间、最晚完成时间、缓冲时间，并求关键路径。