图论与代数结构

最优道路与回路

崔 勇
清华大学计算机系
网络技术研究所

主要内容 道路与回路

- 欧拉、哈密顿、旅行商……
- 最短路径
- 关键路径
- 中国邮路

假如我是欧拉，如何写书？
特殊道路，存在/唯一性，最短回路
然后呢？

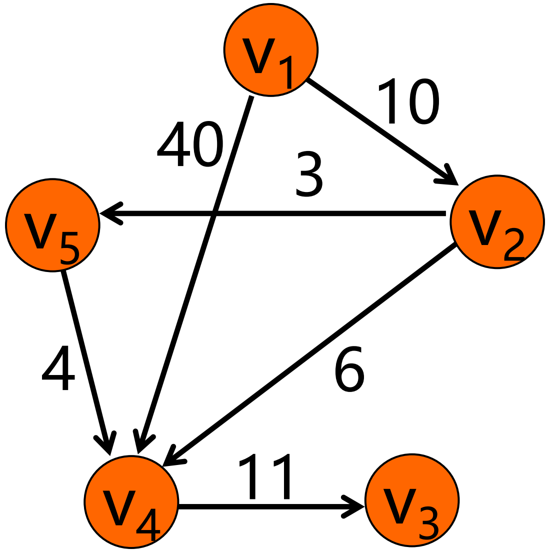
最短/最长？回路/道路？重复呢？

问题提出-定义定理-求解算法-特色分析
基本数学方法和创新思维

最短路径

- 最短路径可研究的情况包括
  - 某两点之间的最短路
  - 某点到其它各个结点的最短路
  - 任意两点之间的最短路
- 边的权值可分为
  - 均为1
  - 正数
  - 任意实数

广探法
深探法



分析认识
创造理论

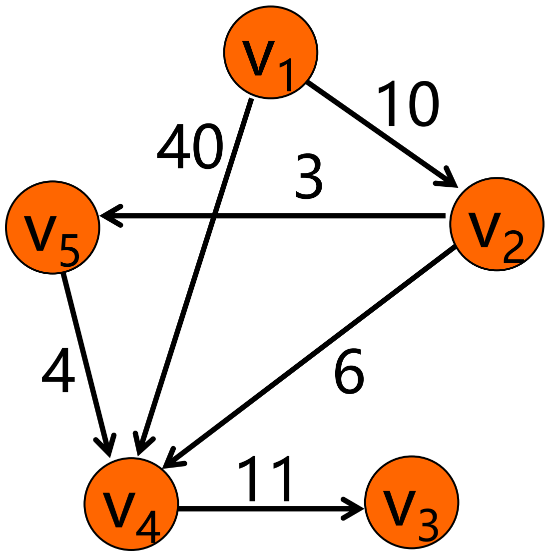
- 这里仅讨论权值为正的情况下某点到其他各个结点最短路径的问题

最短路径

- 定义：路径长度
  - v_1到v_i的一条路径P(i)长度记为π(i)

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

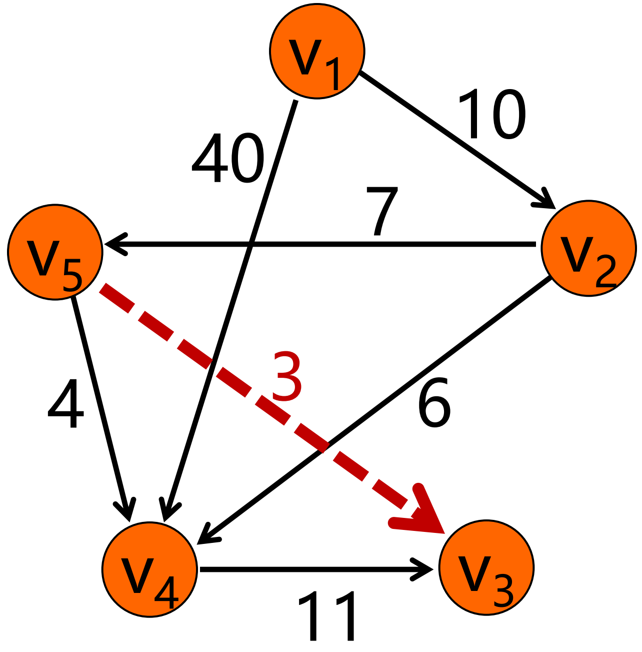
v_1->v_3最短路?子路径?



- 引理1
  - 若P(i)是v_1到v_i的最短路，$v_j\in p(i)$，则**最短路P(i)的子路径P(j)也是v_1到v_j最短路**。
  - 证明:
    - 若P(j)不是最短路，即存在最短路P'(j),使得 $\pi'(j)<\pi(j)$
      这样 $\pi'(i)=\pi'(j)+\pi(j,i)<\pi(i)=\pi(j)+\pi(j,i)$
      与P(i)是最短路矛盾。

最短路径

- 引理2.6.2
  - 对正权图中任一条最短路径，其总长度大于其局部路径长度。
  - 证：由于正权图中每一条边的权值为正，结论显然成立。
- 从两个引理可知
  - 最短路的子路径必然是最短路
  - 最短路不断增长构成新的最短路时，权值不断增大
- 算法设计核心
  - v_1到v_3的最短路依赖哪些节点呢？



如果？ v_4依赖谁？

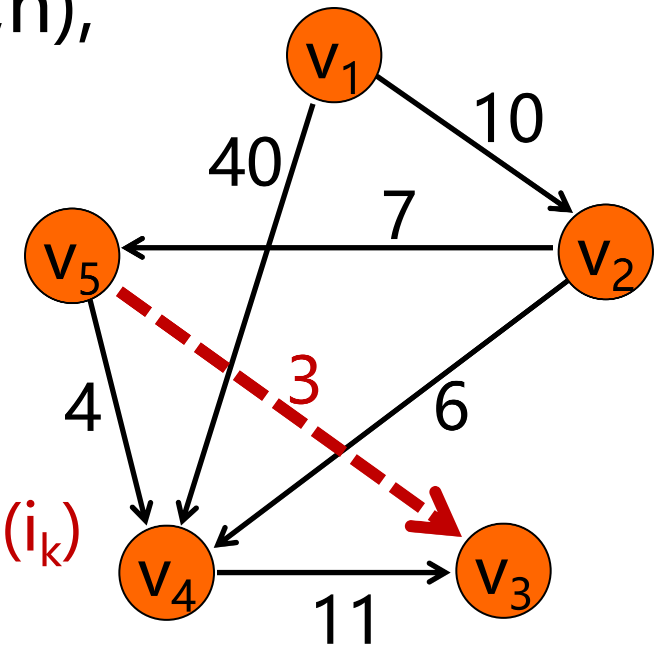
最短路径

- 最短路径算法

核心是先算谁?

  - 若已知从v_1到各个结点的最短路径$P(i_k)(k=1,2,\ldots,n)$，满足 $\pi(1)=\pi(i_1)\le\pi(i_2)\le\cdots\le\pi(i_n)$
  - 由引理可知，若$l<k$，则：
    - $P(i_l)$可能是$P(i_k)$的子路径
    - $P(i_k)$不可能是$P(i_l)$的一部分，即计算$P(i_l)$ 时不用考虑$P(i_k)$
  - 思路：不断找距v_1最近的节点l（通过更近的点j）

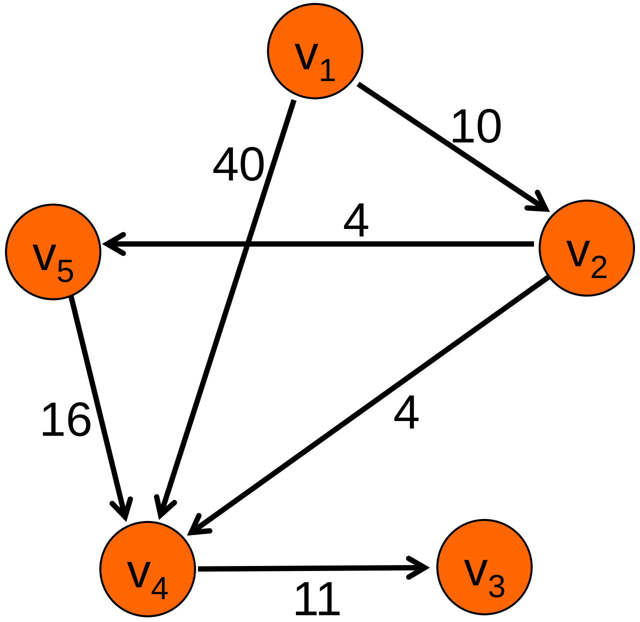
$$\pi(i_l)=\min_{i\le j<l}(\pi(i_j)+W_{i_j i_l})$$



先算的节点影响后面节点

最短路径

- 例
  - 使用我们AI算法求下图中v_1到其余各点的最短路径



π(1)	π(2)	π(3)	π(4)	π(5)	-S	访问
0	∞	∞	∞	∞	1,2,3,4,5	1
0	10	∞	40	∞	2,3,4,5	2
0	10	∞	14	14	3,4,5	4
0	10	25	14	14	3, 5	5
0	10	25	14	14	3	3
0	10	25	14	14	Φ	-

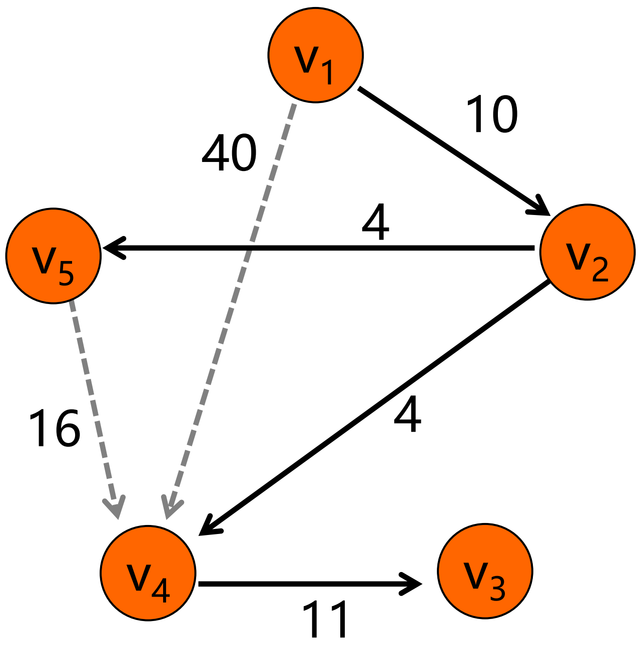
10会变吗？其他数呢？

struct {flag, π, prior} v[n]

特点：每个节点仅被访问一次而不迭代
需要记录：每个节点的前驱

最短路径

- 例
  - 使用我们AI算法求下图中v_1到其余各点的最短路径



π(1)	π(2)	π(3)	π(4)	π(5)	-S	访问
0	∞	∞	∞	∞	1,2,3,4,5	1
0	10	∞	40	∞	2,3,4,5	2
0	10	∞	14	14	3,4,5	4
0	10	25	14	14	3, 5	5
0	10	25	14	14	3	3
0	10	25	14	14	Φ	-

构成最短路径树：

沿着树到达任何节点均为最短路径（从v_1开始）

最短路径

- Dijkstra算法（1959）

1. 置 $\overline{S}=\{2,3,\cdots,n\},\pi(1)=0,\pi(i)=\begin{cases} l_{1i} & i\in\Gamma_1^+ \\ \infty & other \end{cases}$

   $\overline{S}$ 为尚未找到最短路径的节点集

2. 在 $\overline{S}$ 中寻找j满足 $\pi(j)=\min\limits_{i\in\overline{S}}\pi(i),\overline{S}\leftarrow\overline{S}-\{j\}$

   若 $\overline{S}=\Phi$，结束；否则转3。

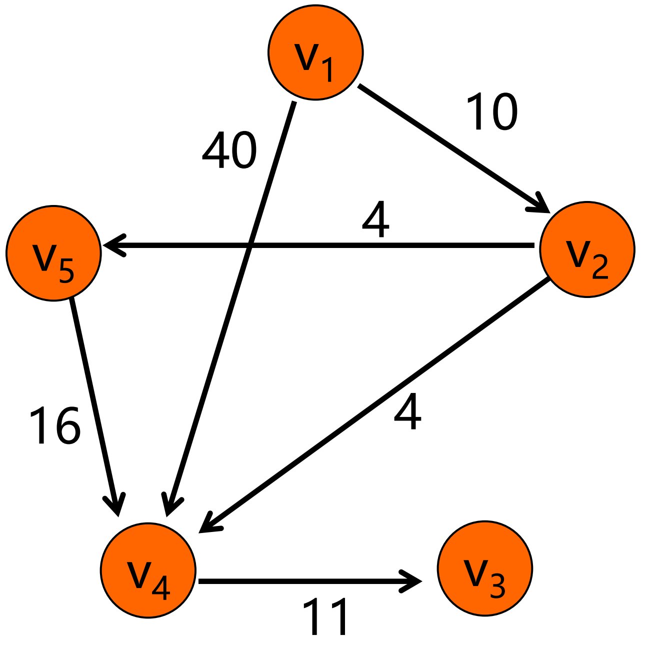
3. 对全部 $i\in\overline{S}\cap\Gamma_j^+$ 置：

   $\pi(i)\leftarrow\min(\pi(i),\pi(j)+l_{ji})$

   转2

访问最近节点j
每个节点仅访问一次
访问后绝不再改距离

更新j的直接后继(各i)
的距离并记录前驱



Dijkstra算法复杂度? O(n²)

单选题 1分

设置

Dijkstra算法按照什么顺序求解源到各点的最短路径？

A 路径长度递减

B 路径长度递增

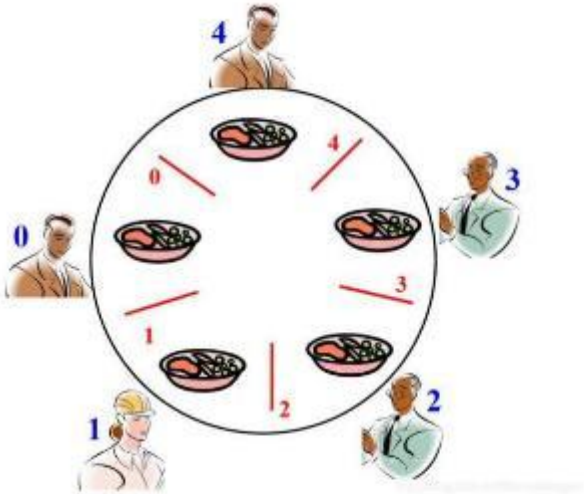
C 顶点编号递减

D 顶点编号递增

提交

最短路径

- 荷兰人Edsger Wybe Dijkstra
  - 提出信号量和PV原语
  - 最短路径算法(Dijkstra 算法)
  - 解决了有趣的“哲学家聚餐”问题
  - 提出“goto有害论”
  - 第一个Algol 60编译器的设计者和实现者
  - THE操作系统的设计者和开发者
- 最伟大的计算机科学家
  - 程序设计大师Donald E. Knuth
  - 《计算机程序设计的艺术》
- 1972年获得图灵奖
- 数学v.s.计算机，真理v.s.导师



最短路径

- 边权为1时v_1到各点的最短路径
  - 使用广度优先算法

1. 置 $\pi(1)=0, \pi(i)=\infty, i\geq 2$

   $k=0, S=\{1\}, S_0=\{1\}$

   第0步访问根节点

2. 第k+1步

   置 $S_{k+1}=\Gamma^+_{S_k}\cap\overline{S}$

   第k+1层子节点

   置 $\pi(i)=k+1, i\in S_{k+1},$

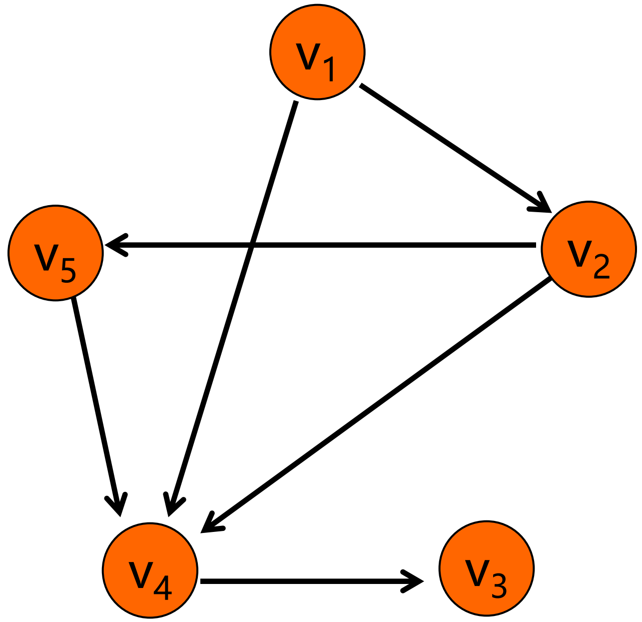
   更新节点的距离

   置 $S=S\cup S_{k+1}$

3. 若 $|S|=|V(G)|$，结束；否则 $k\leftarrow k+1$，转2

最短路径

- 例
  - 求下图中v_1到其余各点的最短路径长度

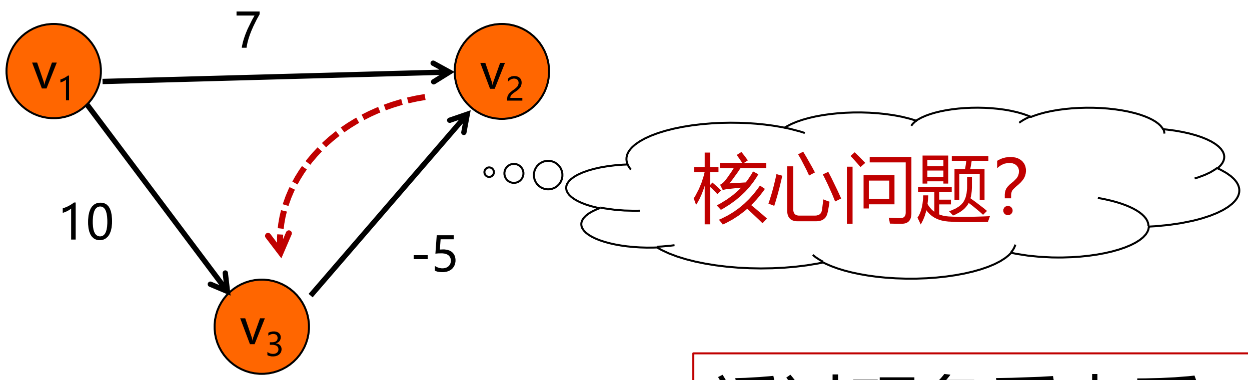


π(2)	π(3)	π(4)	π(5)	k	S	S_{k+1}
1	∞	1	∞	0	1,2,4	2,4
1	2	1	2	1	1,2,3,4,5	3,5

Dijkstra与BFS适用性区别?
假如我是欧拉，如何写书?
Dijkstra算法不足之处?

最短路径

- 边权任意时最短路
  - 边权存在负数时，Dijkstra算法的正确性？



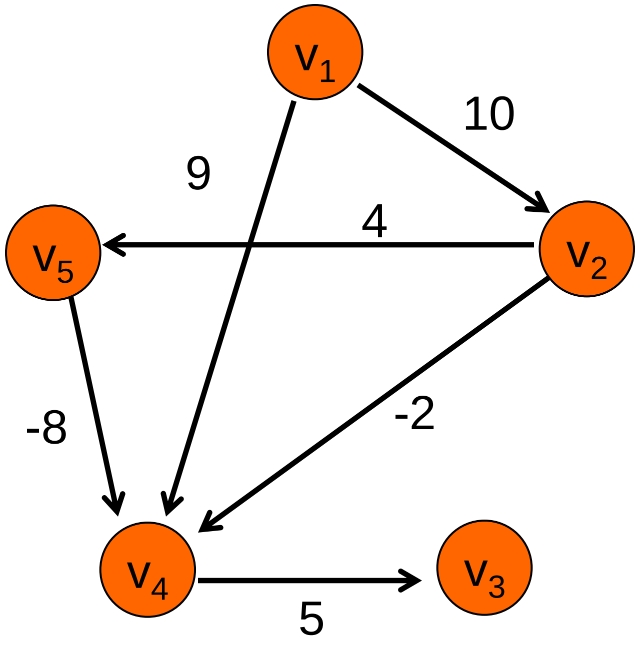
- Dijkstra算法中，π(2)=7
- 正确解是π(2)=5

  - 负权破坏了依赖关系！
  - 如何解决这一问题？

透过现象看本质
连问三个为什么！
1. 为什么只能算出7？
2. 为什么不能先算v_3？
3. 为什么不能用v_3更新？

最短路径

- 例: 使用Ford算法计算下图中v_1到各点的最短路径长度



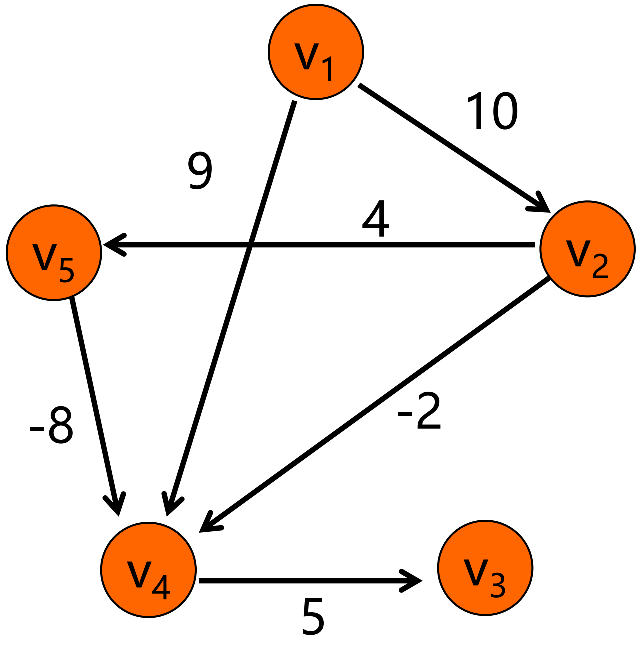
π(1)	π(2)	π(3)	π(4)	π(5)	k
0	∞	∞	∞	∞	0
0	10	∞	8	14	1
0	10	13	6	14	2
0	10	11	6	14	3
0	10	11	6	14	4

潜在问题与改进方法?

节点序和初始化：当负边权数较少时，可先忽略负边而使用Dijkstra算法得到一个结点序，将所得的结果取代Ford的步骤1，再进行迭代。

最短路径

- 例:使用改进的算法计算v_1到各点的最短路径



使用Dijkstra算法得到的结果(忽略负边):

$\pi(2)=10$, $\pi(3)=14$, $\pi(4)=9$, $\pi(5)=14$

结点序为v_4, v_2, v_3, v_5

$\pi(1)$	$\pi(4)$	$\pi(2)$	$\pi(3)$	$\pi(5)$	k
0	9	10	14	14	0
0	6	10	11	14	1

潜在问题与改进方法?

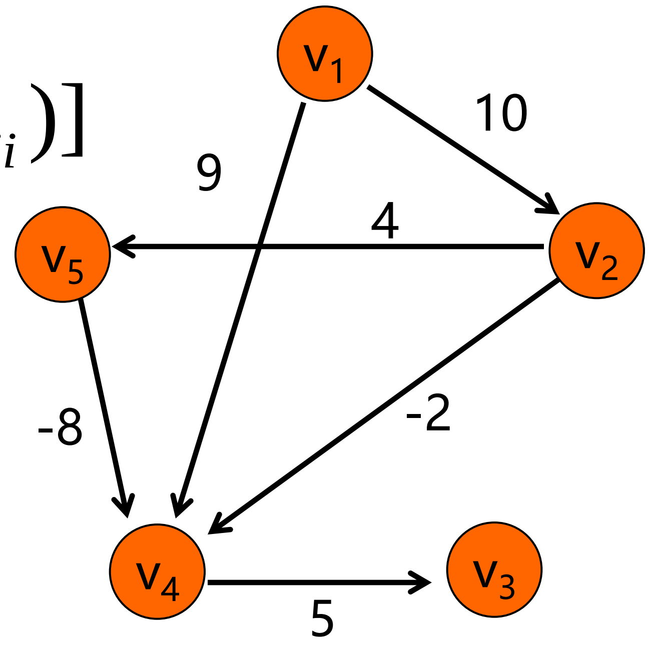
非连通图则采用权值为0使用D算法

进一步考虑依赖关系优化节点序!

最短路径

- 回顾：Ford算法

1. 置 $\pi(1)=0, \pi(i)=\infty, i=2,3,\cdots,n$
2. i从2到n，令 循环n次吗?
   $$\pi(i) \leftarrow \min[\pi(i), \min_{j\in\Gamma_i^-}(\pi(j)+w_{ji})]$$
3. 若 全部π(i)都没变化，结束；
   否则转2



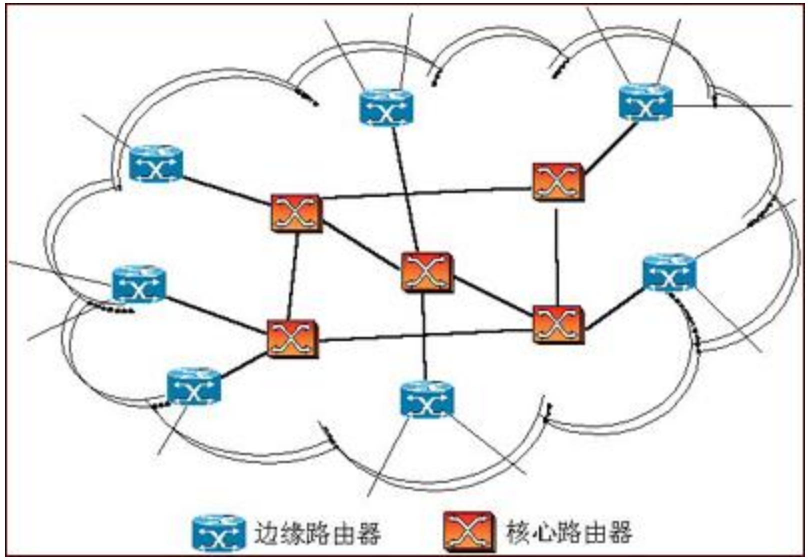
能分布式协作运行吗?

v_4告诉v_3什么内容？v_3如何计算?

D算法呢?

最短路径

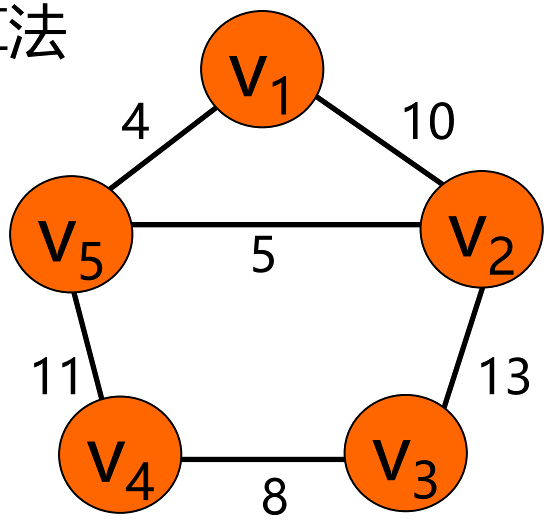
- 互联网路由器Router & 路由协议
  - 功能目标：寻找“合适”的路径
  - 域内路由协议：OSPF、RIP
  - 域间路由协议：BGP
- 协议 v.s. 算法



最短路径

- 路由协议分类（技术特点）
  - 链路状态协议：OSPF
    将自己的信息通过邻居告诉世界
    “集中式”算法 “分布式”计算
  - 距离向量协议：RIP
    将世界的(最佳)信息告诉邻居
    采用“分布式”算法

可能的问题：
v_2-v_5断开，
但v_1不知道，
于是v_1->v_2？



华为“天才少年”

最短路径算法(SPF)
(Dijkstra 算法)

Ford算法

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

路由协议研究前沿探讨：
1、计算复杂度
2、多路径与服务质量
3、路由一致性与回路
4、路由收敛性与抖动
5、特定网络场景(无线)

第二章 道路与回路

- 道路与回路的定义
- 道路与回路的判定
- 欧拉道路与回路
- 哈密顿道路与回路
- 旅行商问题与分支定界法
- 最短路径
- 关键路径
- 中国邮路

搞定最短路径后，假如我是欧拉，如何写书？
最短路径v.s.最长路径？

关键路径

- 帝国大厦
  - 纽约downtown
  - 102层，仅410天(1930年)

关键路径(0)

- 盖大楼啦
  - 钢筋、水泥、混凝土
  - 油漆、玻璃、瓷砖
  - 农民工住房
  - 建筑工人
  - 装修工人
  - 挖地基
  - 载重汽车运输
  - 大楼选址和工程设计
- 面临的问题
  - 工程时间与预算
    - 各部分需要多少人力
    - 各部分需要多少设备
  - 不要过早开始?
    - 占用资金
    - 占用场地
    - 产品保质
  - 不要过晚开始
    - 影响工程进度

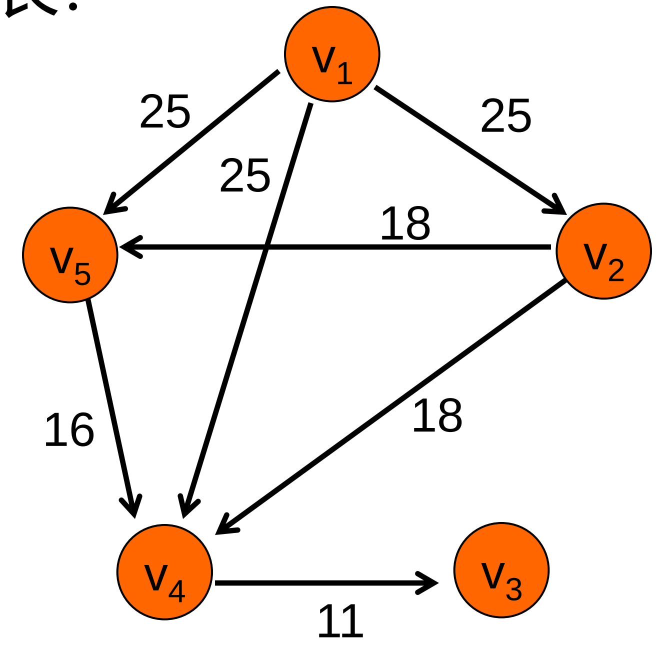
图论建模&设计计算法?

关键路径(1)

- 关键路径
  - 一项工程都要由很多工序组成
    - 每个工序的执行时长是可预知的
    - 这些工序相互约束，只有在某些工序完成之后，一个新的工序才能开始，这种关系也是预知的
  - 需要求解
    - 完成整个工程任务最少需要多少时间?
    - 影响工程进度的关键工序是哪几个?
    - 每项工序的最早/最晚启动时间?
  - 这就是这里所讨论的关键路径问题

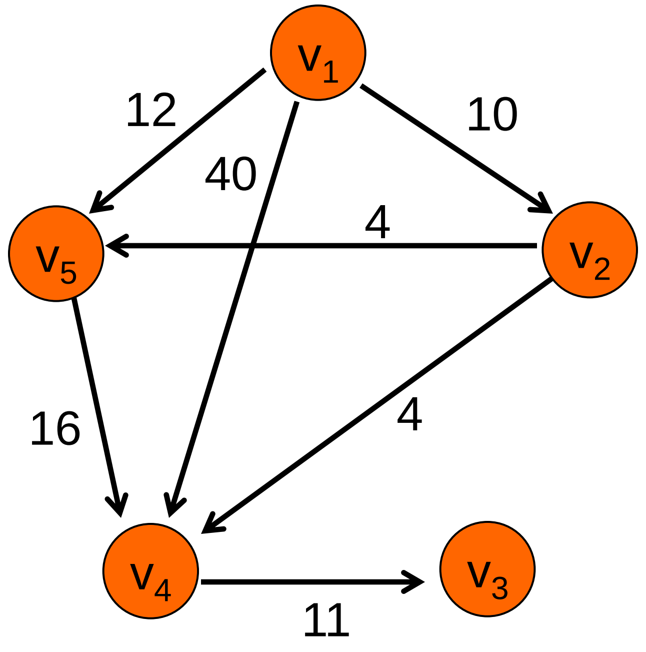
关键路径(2)

- 图论建模
  - 图的要素：点，边，边权
  - 实际问题的要素
    - 工作(工序)，工序之间的依赖关系，工序的时长?
- PT图(Potential task graph)
  - 用结点i表示工序i
  - 用有向边e_{ij}表示工序i和工序j之间的依赖关系
  - 边权l_{ij}表示该工序i的时长
- 存在的缺点及其他表示方法?



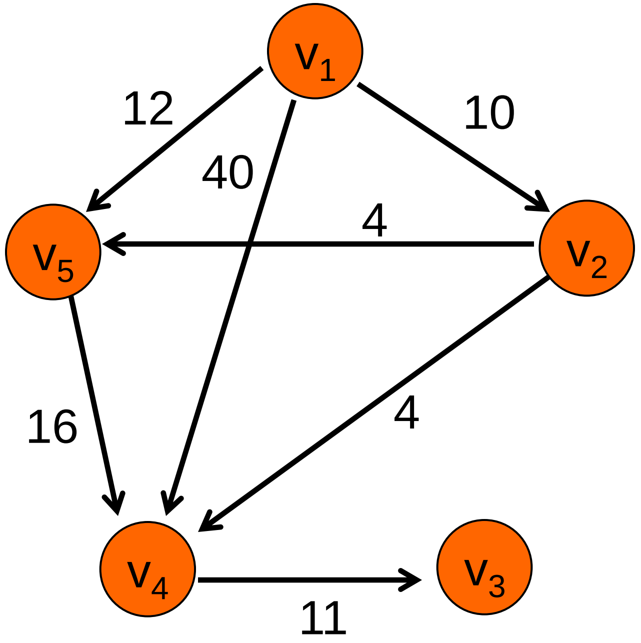
关键路径(3)

- PERT图建模(Program evaluation and review technique)
  - 用有向边e_{ij}表示工序(i,j)
  - 边权l_{ij}表示该工序花费的时间
  - 结点为工序之间的关系
- PERT图举例
  - v1是整个工程的起点
  - v3是终点
  - 含有7项工序



关键路径(4)

- 工序(5,4)的最早开始时间?
  - 12 v.s. (10+4)?
  - 工序(5,4)必须在工序(1,5)和(2,5)完成之后才能开始
  - (5,4)最早开始时间是14?
- 工程的最短完成时间
  - 从起点到终点的最长路径长度
  - 这个最长路径就是关键路径



含7项工序的PERT图

想写程序了?　　是否存在开始结点和终了结点?

关键路径(5)

- 引理2.7.1
  - PERT图是否存在回路?
  - 若有向图G无有向回路，则存在负度为0的顶点
  - 证明:
    - 在G中构造一条极长的有向初级道路P($v_1,v_2,\ldots,v_l$)，考虑 $d^-(v_1)=0$?
    - 若$d^-(v_1)\neq 0$，即v_1有直接前趋v_i
      - 假设该前趋v_i在P之外，那么(v_i ,$v_1,v_2,\ldots,v_l$)构成更长的道路，与P是极长道路矛盾
      - 若v_i属于P，那么($v_1,v_2,\ldots, v_i$)构成有向回路，与已知矛盾
    - 因此有$d^-(v_1)=0$
    - 存在开始和结束结点

是否存在开始节点?

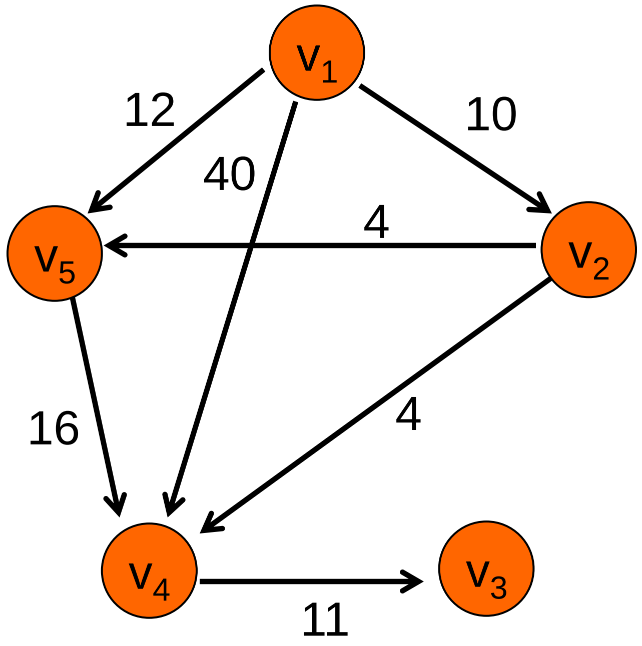
•是否存在多个开始结点?
•增设虚拟的超结点!

关键路径(6)

- 如何求解关键路径?
  - 已知存在开始结点和终了结点
  - 从起点到终点的最长路径

设计算法?　找找巨人?

最长路径子路径是什么?
最长路径依赖哪些节点?
先算哪些节点呢?



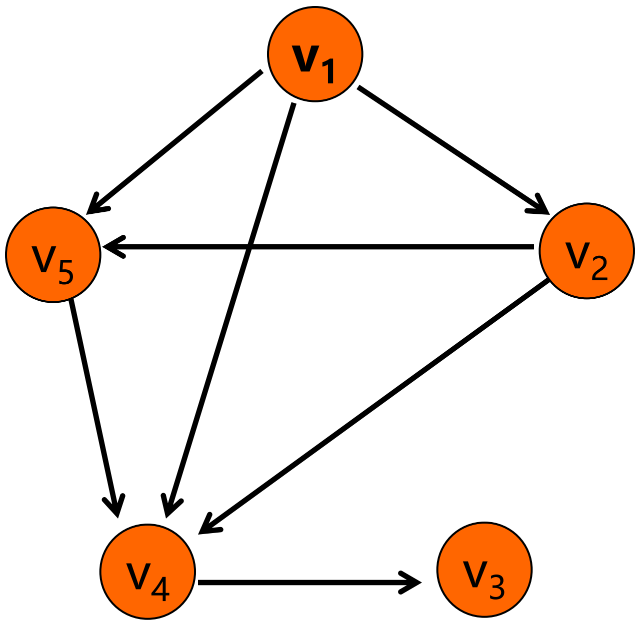
含有7项工序的PERT图

关键路径(7)

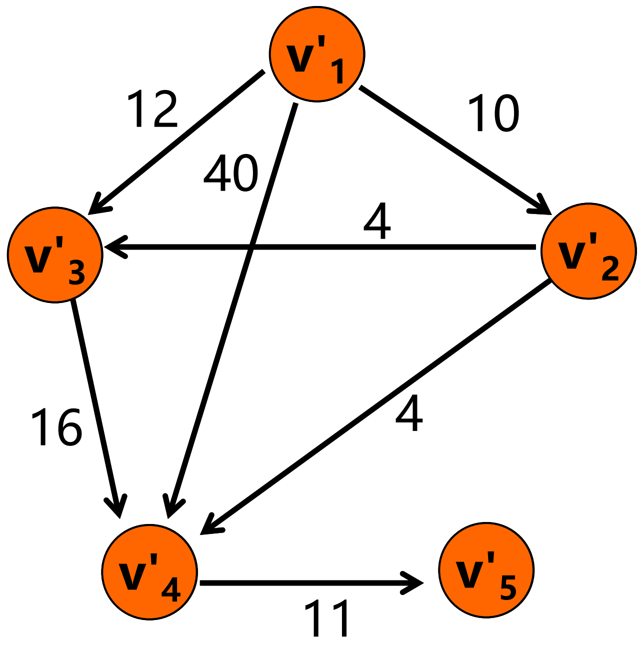
- 定理
  - 设有向图G没有有向回路，则可以将其顶点重新编号：$v_1', v_2', \ldots, v_n'$，使得对G的任意边(v_i', v_j')，都有$i<j$。
  - 证明（思路？）
    - 构造法：构造结点序列
    - 由引理2.7.1，G中存在负度为0的点
    - 取一个这样的点编号为v_1'，令$G \leftarrow G - v_1'$,得到的子图仍然没有回路
    - 再取一个负度为0的点编号为v_2'
    - 重复这个过程，直到所有点都编号为止
    - 这时所有的编号都满足定理的条件

关键路径(8)

- 例
  - 将下图中结点重新编号为$v_1'\sim v_5'$，使得对任意边(v_i',v_j')，都有$i<j$



- $v_1'=v_1$
- $v_2'=v_2$
- $v_3'=v_5$
- $v_4'=v_4$
- $v_5'=v_3$



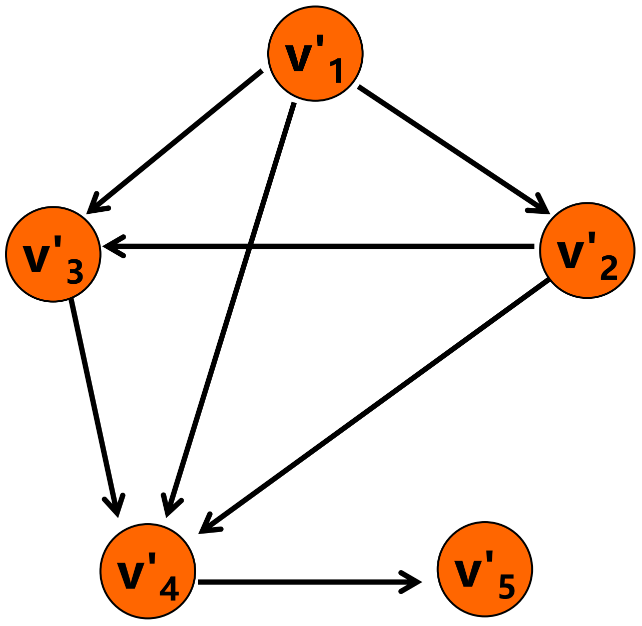
关键路径(9)

- 关键路径算法

1. 对顶点重新编号为 $v_1', v_2', \cdots, v_n'$
2. $\pi(1) \leftarrow 0$
3. 对 j'从2到n令 $\pi(j') = \max\limits_{i' \in \Gamma_{j'}^{-}} (\pi(i') + l_{i'j'})$
4. 结束

重新编号的重要性?

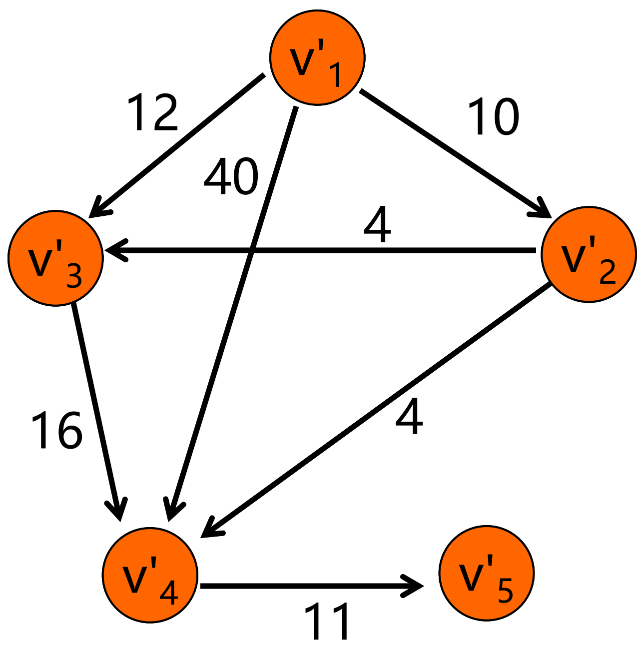
最长路径



其中π(i)就是工序(i,k)的最早启动时间

关键路径(10)

- 例
  - 求下图所表示的工序的关键路径



- 对结点重新编号
- $\pi(2')=10$
- $\pi(3')=\max(12,\ 10+4)=14$
- $\pi(4')=\max(14+16,\ 40,\ 10+4)=40$
- $\pi(5')=40+11=51$
- 关键路径是 (v_1', v_4', v_5')，其长度是51

关键路径的物理意义？非关键路径呢？

关键路径(11)

- 最大允许延误时间
  - 显然关键路径上的工序是不允许延误的，否则不可能按时完成工程项目
  - 而对非关键工序的允许延误时间将可以给工程规划人员带来工作上的灵活性?
  - 设工序(i,j)：
    最早启动时间为$\pi(i)$
    最晚启动时间为$\tau(i,j)$
  - 则最大允许延误时间为
    $\Delta_{ij} = \tau(i,j) - \pi(i)$

关键路径(12)

- 计算最晚启动时间（v_3依赖谁呢）

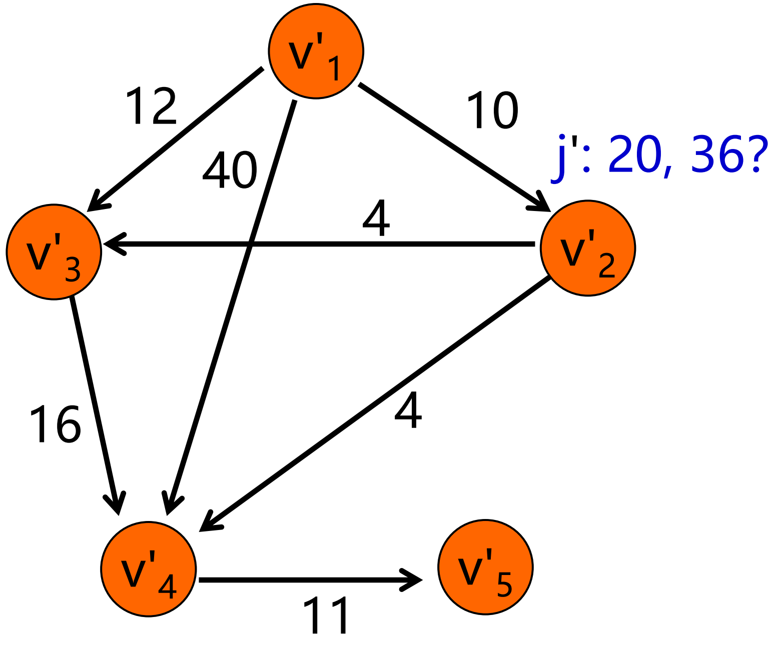
$$\tau(n') = \pi(n')$$

1. j'从(n-1)'到1'，结点j' 最晚完工时间依赖于后续节点

$$\tau(j') = \min_{i' \in \Gamma^+_{j'}} (\tau(i') - l_{j'i'})$$

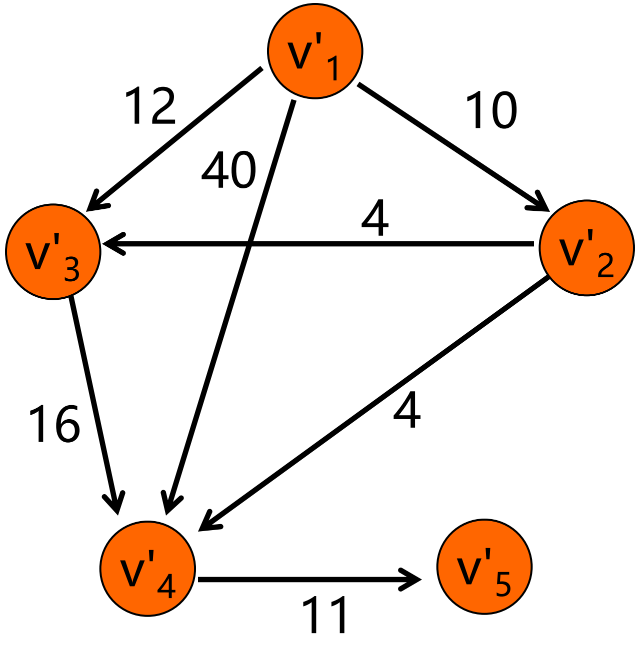
2. 对每一条边(i',j')，则工序(i',j')的最晚启动时间

$$\tau(i', j') = \tau(j') - l_{ij}$$



关键路径(14)

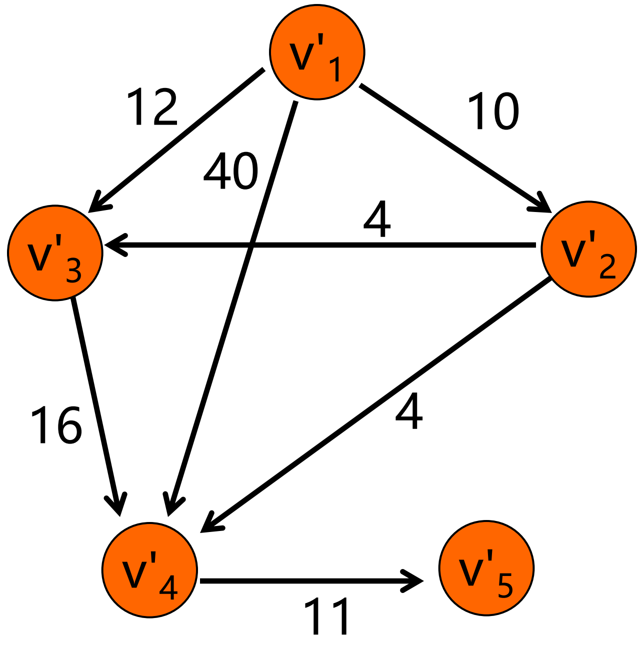
- 例
  - 求下图中各工序的最晚启动时间和最大允许延误时间



- 已知关键路径和最早完工时间
- 计算“结点”最晚完工时间
  - $\tau(5')=51$
  - $\tau(4')=51-11=40$
  - $\tau(3')=40-16=24$
  - $\tau(2')=\min(24-4, 40-4)=20$
  - $\tau(1')=\min(20-10, 24-12, 40-40)=0$

关键路径(15)

- 例(续)



- 已知结点最晚完工时间

  $\tau(1')=0$; $\tau(2')=20$; $\tau(3')=24$;

  $\tau(4')=40$; $\tau(5')=51$

- 计算工序最晚启动时间

  $\tau(1', 2')= \tau(2')-10= 10$

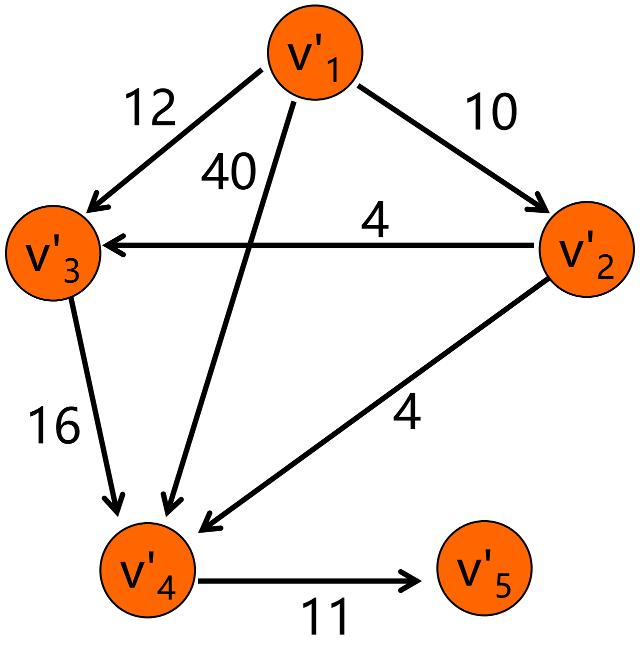
  $\tau(1', 3')=12$ $\tau(1', 4')=0$

  $\tau(2', 3')=20$ $\tau(2', 4')=36$

  $\tau(3', 4')=24$ $\tau(4', 5')=40$

关键路径(16)

- 例(续)



- 已知工序最晚启动时间

$\tau(1', 2')=10$ $\tau(1', 3')=12$
$\tau(1', 4')=0$ $\tau(2', 3')=20$
$\tau(2', 4')=36$ $\tau(3', 4')=24$
$\tau(4', 5')=40$

- 最早启动时间

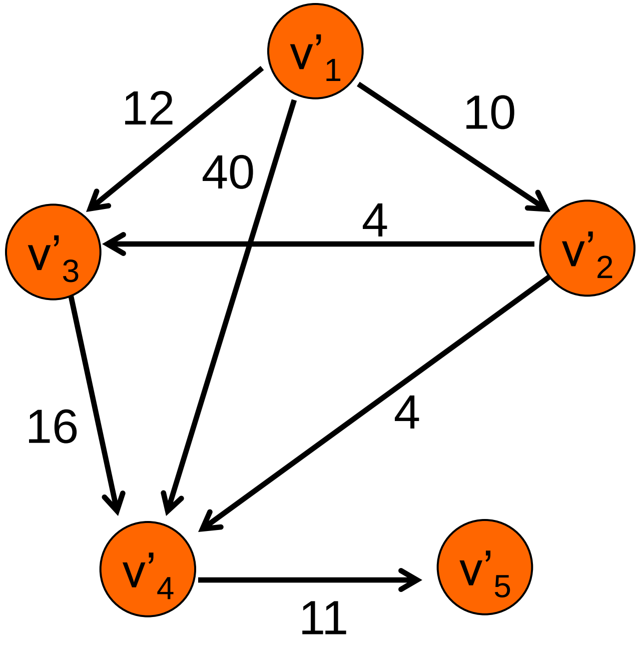
$\pi(1')=0, \pi(2')=10, \pi(3')=14, \pi(4')=40, \pi(5')=51$

- 计算最大允许延误时间

$\Delta_{1'2'}=10-0=10$ $\Delta_{1'3'}=12-0=12$
$\Delta_{1'4'}=0-0=0$ $\Delta_{2'3'}=20-10=10$
$\Delta_{2'4'}=36-10=26$ $\Delta_{3'4'}=24-14=10$
$\Delta_{4'5'}=40-40=0$

关键路径(17)

- 关键路径小结
  - 工程问题图论建模
    - PERT图
  - 关键路径
    - 从前向后计算的最长路径
    - 完工时间(工序最早启动时间)
  - 工序最晚启动时间
    - 结点/工序最晚完工时间
    - 从后向前倒推回来
  - 工序最大允许延误时间



能当包工头了吗?

单选题 1分

设置

一项工程可能（）

A 可能没有关键路径

B 有多条关键路径

C 有一条关键路径

D 有一条或多条关键路径

提交

第二章 道路与回路

- 道路与回路的定义
- 道路与回路的判定
- 欧拉道路与回路
- 哈密顿道路与回路
- 旅行商问题与分支定界法
- 最短路径
- 关键路径
- 中国邮路

旅行商问题：最短H回路
最短欧拉回路是什么？
再创新一步？

中国邮路(1)

- 从欧拉回路到——最佳邮路（中国邮路）
  - 在一个正权连通图G中，从某点出发经过每条边至少一次最后返回出发点的最短回路称为最佳邮路(中国邮路)
  - 欧拉回路?
    - 当G所有结点度都是偶数时，即G有欧拉回路，该回路就是最佳邮路
  - 当G只有1个节点的度为奇时……
  - 当G只有2个节点的度为奇时……
    - 则存在一条欧拉道路，该道路加上从起点到终点的最短路径组成的回路就是最佳邮路
  - 当奇结点的个数大于2时，如何确定最佳邮路?

管梅谷教授

开小灶的

“笨”学生

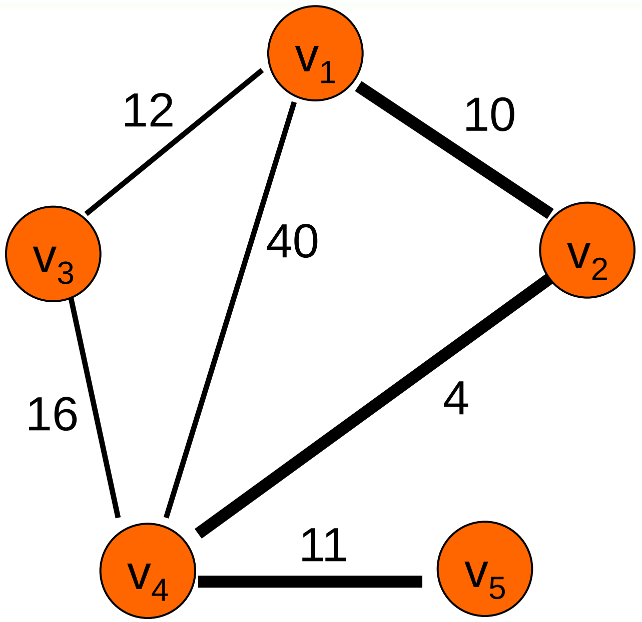
上大学的

两件事

学习&合作

中国邮路(2)

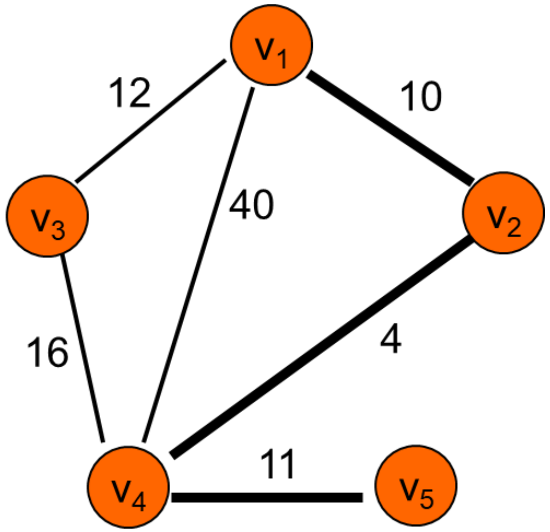
- 最佳邮路中的边重复走
  - 加上重边后，是否为欧拉回路?
  - 什么样的边会被重复走?
  - 某边最多被重复多少次?
- 呼唤个定理?
  - L是无向连通图G的最佳邮路的条件：
    1. G中的每条边最多重复一次；
    2. 在G的任一个回路上，重复边的长度之和不超过该回路长度的一半
  - 证明（分必要性和充分性两部分）



充要条件!

中国邮路(3)

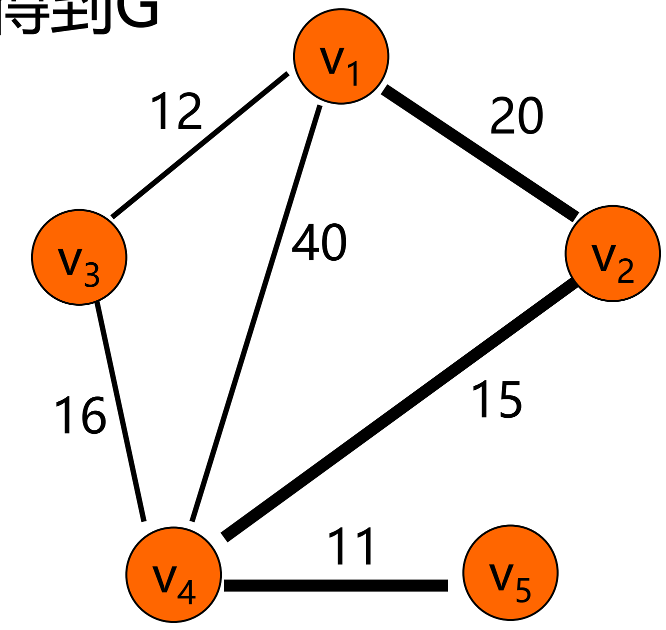
- 证明（续）
  - 必要性（ 1.最佳邮路G中每条边最多重复一次）
    - 若一条最佳邮路重复经过图的某些边，将G中k次重复的边画k次，得到G'
    - 设最佳邮路L'使G中边e_{ij}重复n(n>1)次，这时G′中有欧拉回路L',即G'各点度均为偶数
    - 若使e_{ij}重复n-2次，得到G''，G''各点度仍是偶数
    - G''的欧拉回路L''也是G的一条中国邮路，且L''长度小于L'，与L'是最佳邮路矛盾
    - 因此边e_{ij}最多重复一次。



凡是邮路，将重复的边画为重边后均是欧拉图

中国邮路(4)

- 证明（续）
  - 必要性(2. 任一回路重复边长不超过回路一半)
    - 假设G中某个回路C的重复边的长度，大于C总长度的一半
    - 令C中重复的边不重复，不重复的边重复，得到G''
    - G''仍是欧拉图，且L''长度小于L'，与L'是最佳邮路矛盾
    - 因此，在任意一个回路上，重复边的长度之和不会超过回路的一半
    - 必要性证毕



中国邮路(5)

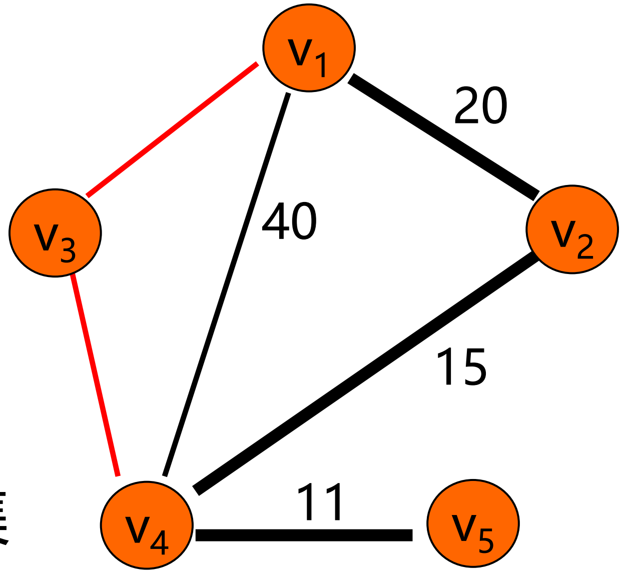
- 证明（续）

凡是邮路，加上重复的边后均是欧拉图

  - 充分性：（最多重复一次；重复边和不超过回路一半）
    - 假设任意两个不同邮路L_1和L_2满足定理的两个条件，我们将证明$\pi(L_1)=\pi(L_2)$
      - （最佳邮路L'满足定理条件，但最佳邮路不唯一）
    - 设$L_1=E(G)+Q+Q_1$, $L_2=E(G)+Q+Q_2$
      - 其中Q是L_1和L_2共同的重复边集，Q_1和Q_2是分别只属于L_1和L_2的重复边集
      - L_1和L_2的结点度数均为偶数
    - 设L_1和L_2的对称差为$E'(G)=Q_1+Q_2$，结点度数均为偶数
    - 若$E'(G)=\Phi$，显然$\pi(L_1)=\pi(L_2)$

中国邮路(6)

- 证明（续）
  - 充分性：（最多重复一次；重复边和不超过回路一半）
    - 否则（L_1和L_2的对称差$E'(G)=Q_1+Q_2\neq\Phi$ ）
    - 构造$G'=(V(G),E'(G))$
    - 因为G'是简单图，各结点度是偶数
    - 可以将G'划分成若干简单回路C，对任一回路C，设C_1和C_2分别是L_1和L_2的重复边集
    - 由已知条件(2. 任一回路重复边长不超过回路一半) $\pi(C_1)\le\pi(C_2)$, $\pi(C_2)\le\pi(C_1)$
    - 因此$\pi(C_1)=\pi(C_2)$，$\pi(Q_1)=\pi(Q_2)$，$\pi(L_1)=\pi(L_2)$，得证

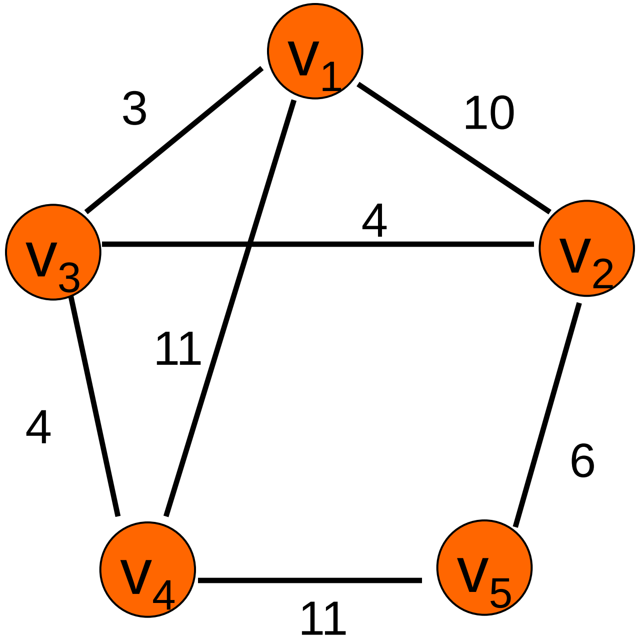


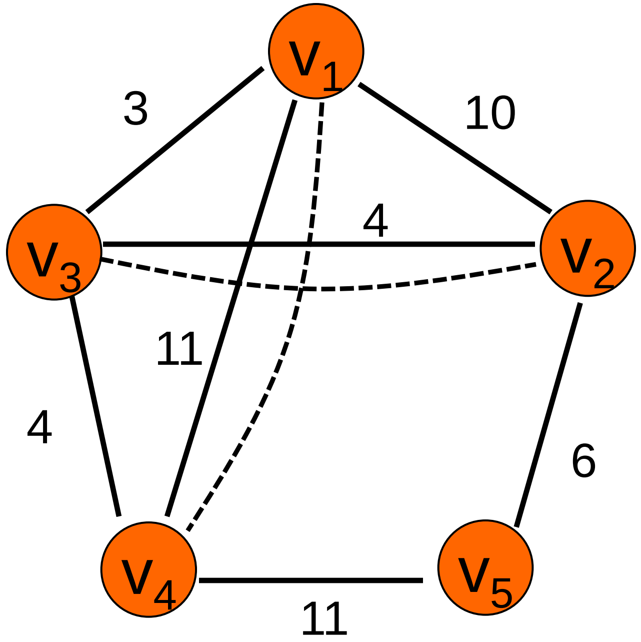
中国邮路(7)

- 回顾所证明的定理
  - L是无向连通图G的最佳邮路的充要条件是：
    1.G中的每条边最多重复一次；
    2.在G的任一个回路上，重复边的长度之和不超过该回路长度的一半
- 构造中国邮路
  - 找出度为奇的点
  - 依据条件1构造邮路，保证计算重复边之后度都是偶数
  - 由条件2对所有回路进行判断，若不满足条件，则令回路中的重复边不重复，不重复边变为重复

中国邮路(8)

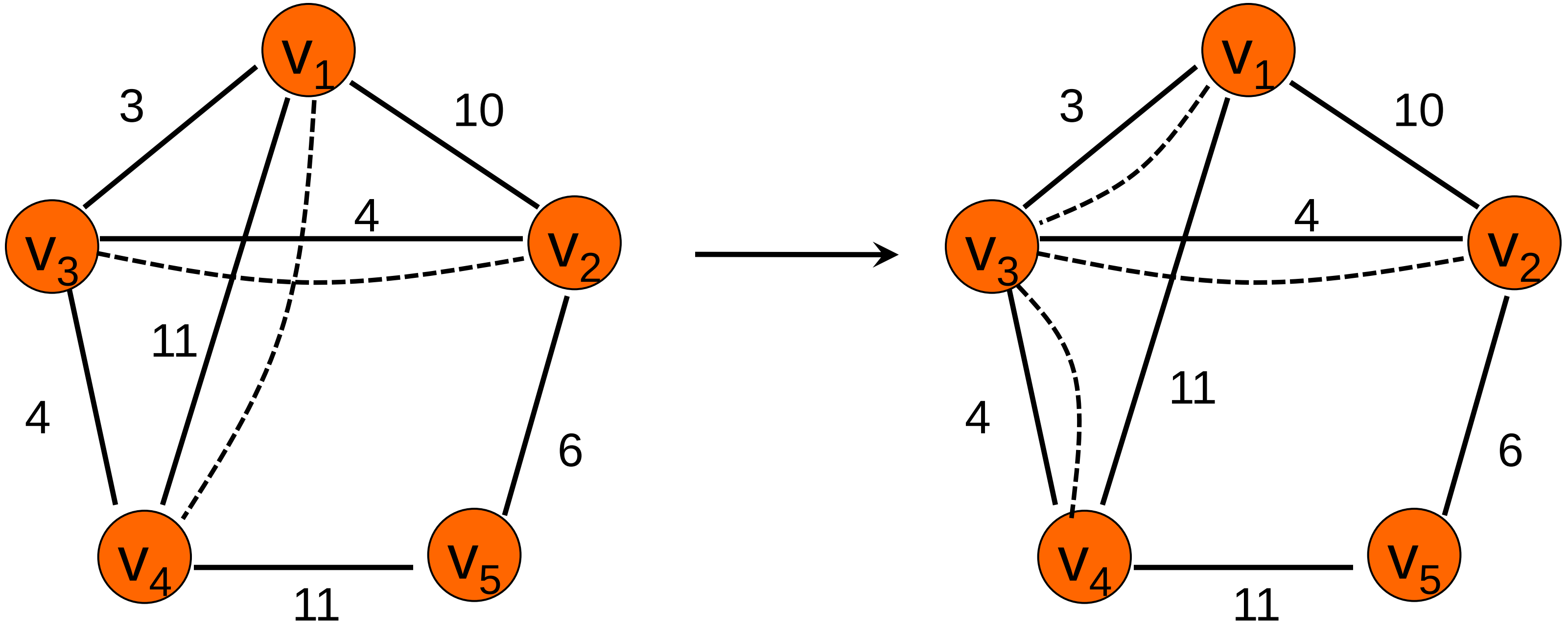
- 例
  - 使用构造法找出下图中的中国邮路





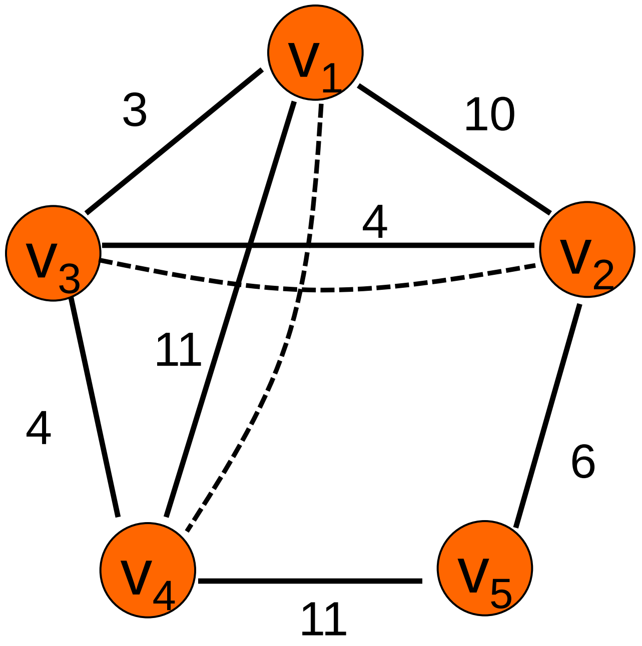
中国邮路(9)

- 例（续）



中国邮路(10)

- 求解中国邮路
  - 上述构造法求解中国邮路
    - 由于回路数量很多，所以计算量很大
  - Edmonds提出的最小权匹配算法
    - 基本思路为构造欧拉回路
    - 中国邮路的好算法



第二章 道路与回路

- 道(回)路的定义、判定
  - 邻接矩阵$P=A+A^2+\dots+A^n$
  - 道路存在性：深度优先、广度优先搜索道路
- 欧拉、哈密顿道(回)路
  - 一笔画问题、偶数度、构造法(回路、极长初级道路)
- 旅行商问题
  - 分支定界法、便宜算法
- 最短路径
  - Dijkstra (正权)、BFS (权为1)、Ford (无负回路)
- 关键路径
  - PERT图：最早、最晚启动时间和最大允许延误时间
- 中国邮路
  - 无向图：充要条件与构造法

讲课太快/太慢?
找问题，定义、定理、算法
设计而不仅仅学习

透过现象看本质
连问三个为什么!

看书复习和做题

思维拓展

- 严格的一次且仅一次——存在性问题
  - 欧拉回路：经过每边一次且仅一次
  - 哈密顿回路：经过每点一次且仅一次
- 优化问题
  - 旅行商：过所有点（点不重复），求最短
  - 中国邮路：过所有边（边可重复），求最短
  - 关键路径：求最长路径，从后向前算
- 更普适的优化问题
  - 过所有点（点可重复）？
  - 经过给定的点？经过给定的边?
  - 给定边和点的集合？经过给定集合中不少于k个元素?
  - 权值在节点上?

期待发掘新问题和求解

作业-最短路径、PERT图和邮路

- 习题二（P50）
  - PERT图：P56:45

    仅画PERT图并求解（灵活应用）
  - 中国邮路：P57:48