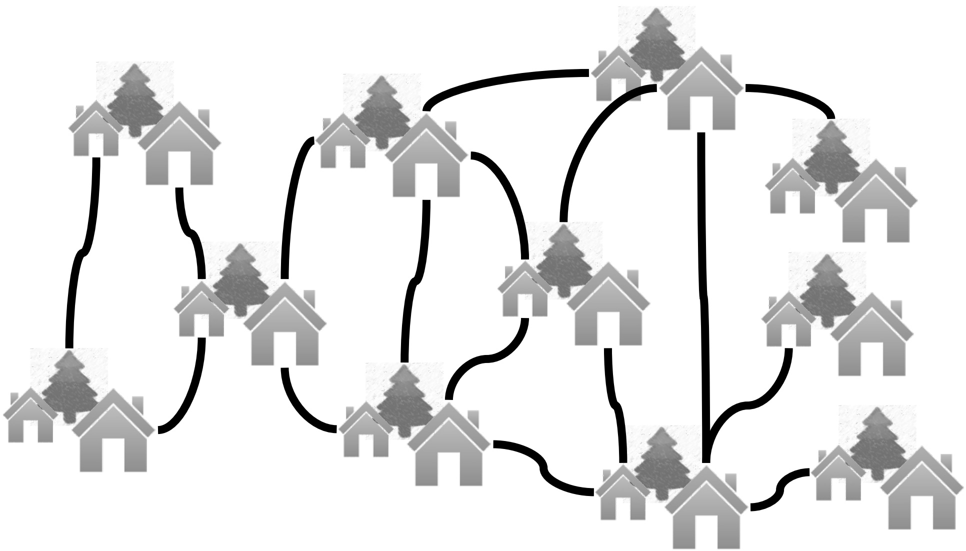
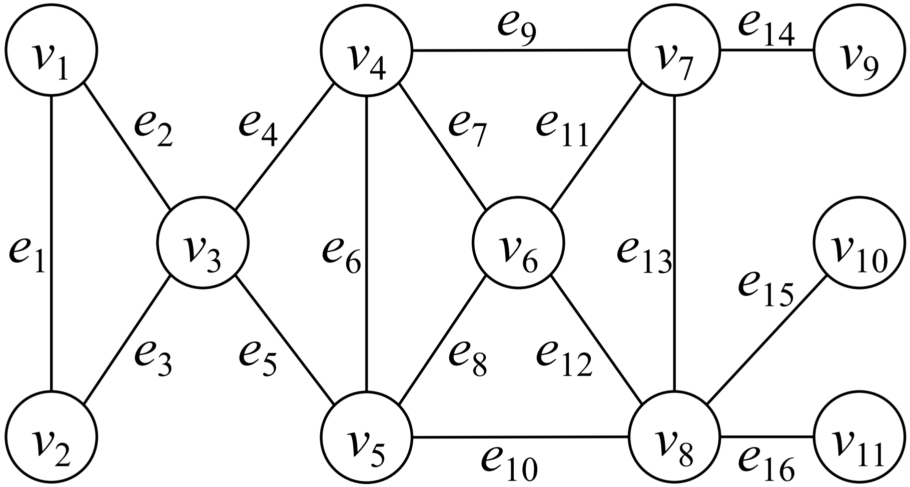
# 第6.1节 边的独立、覆盖和支配

程龚

# 如何选择道路建设加油站更合理？

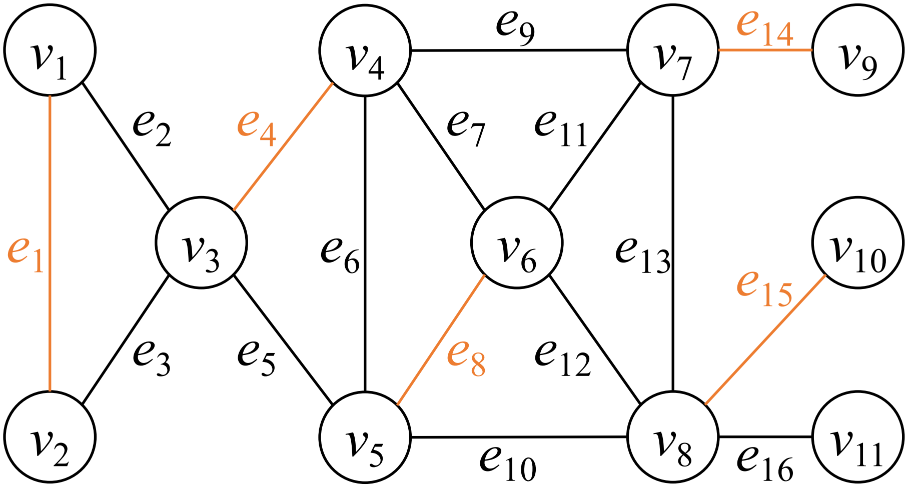


# 如何选择道路建设加油站更合理？



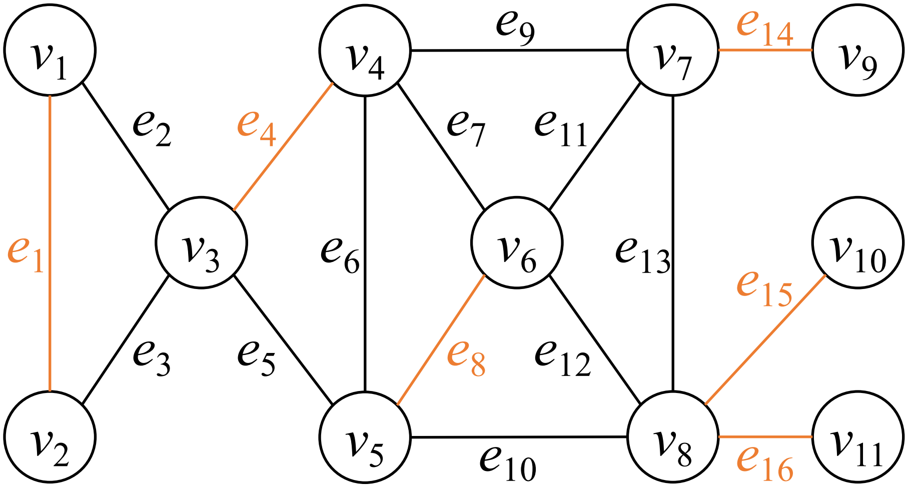
# 如何选择道路建设加油站更合理？

- 赵专家：尽可能多选道路，但为避免浪费，相邻的道路原则上至多选一条。



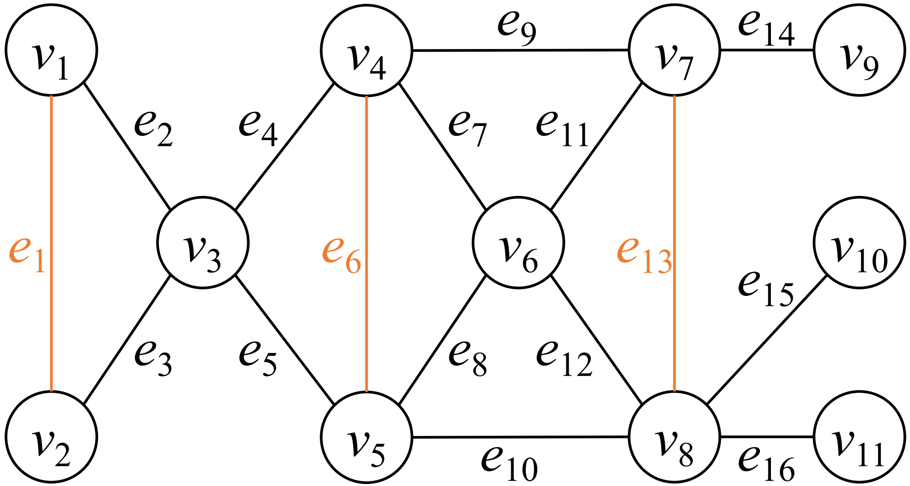
# 如何选择道路建设加油站更合理？

- 钱专家：尽可能少选道路，但为确保够用，每座城镇关联的道路原则上至少选一条。



# 如何选择道路建设加油站更合理？

- 孙专家：尽可能少选道路，但每条未被选中的道路原则上至少与一条被选中的道路相邻。

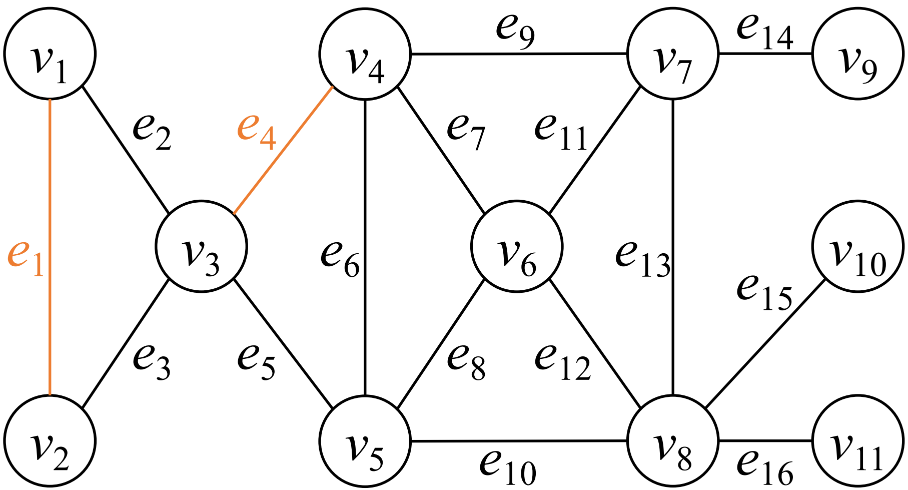


# 本次课的主要内容

6.1 边的独立、覆盖和支配

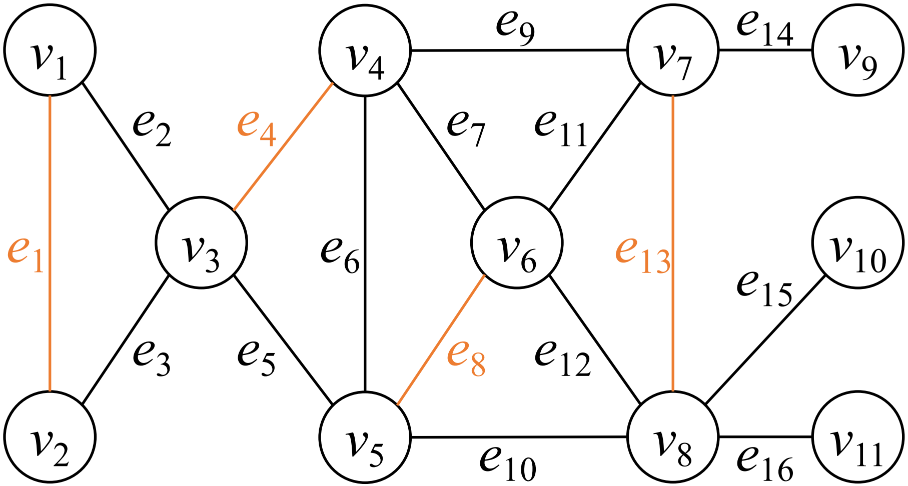
# 边的独立、覆盖和支配

- **边独立集**：匹配



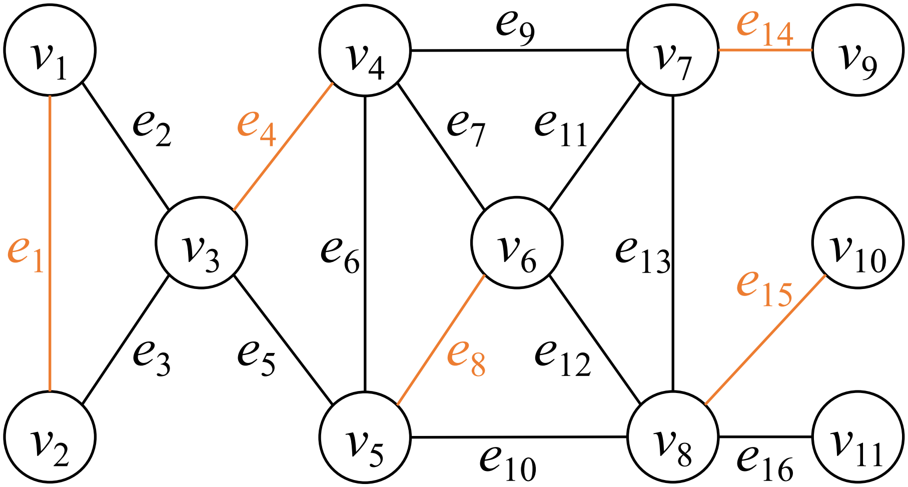
# 边的独立、覆盖和支配

- 边独立集：匹配
- **极大边独立集**：极大匹配



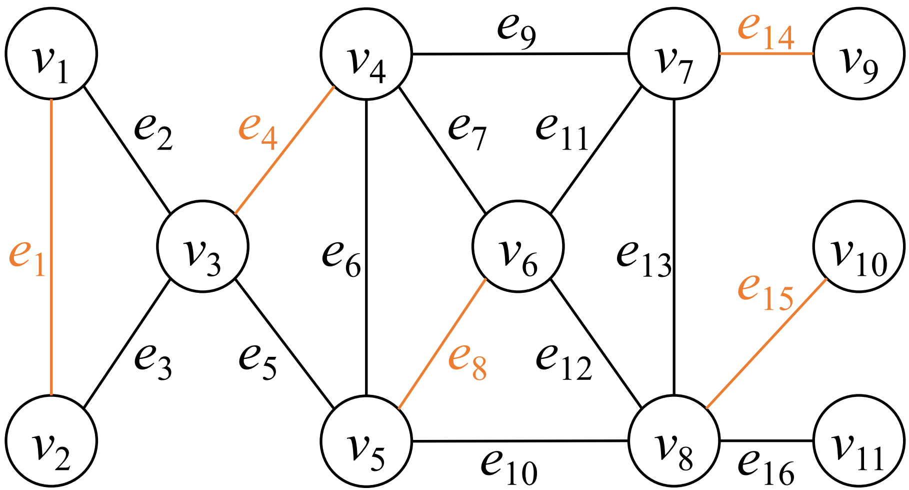
# 边的独立、覆盖和支配

- 边独立集：匹配
- 极大边独立集：极大匹配
- **最大边独立集**：最大匹配



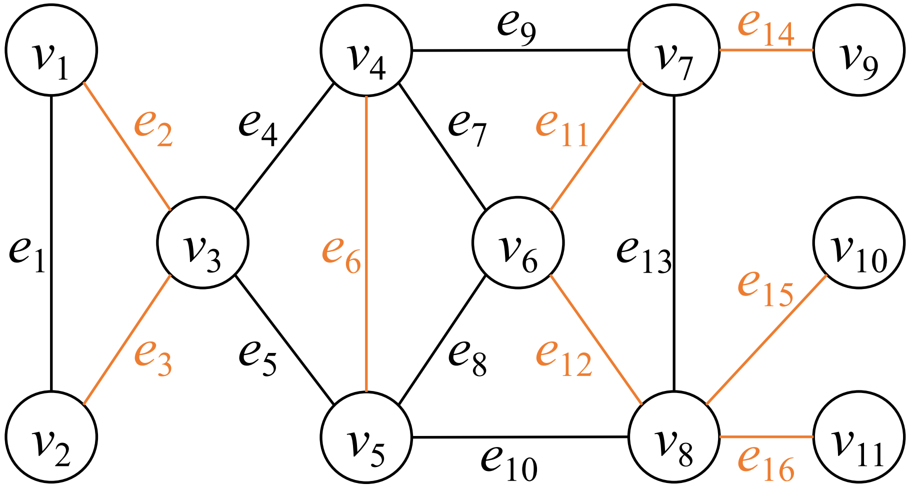
# 边的独立、覆盖和支配

- 边独立集：匹配
- 极大边独立集：极大匹配
- 最大边独立集：最大匹配
- **边独立数**：最大边独立集的大小，记作α’(G)



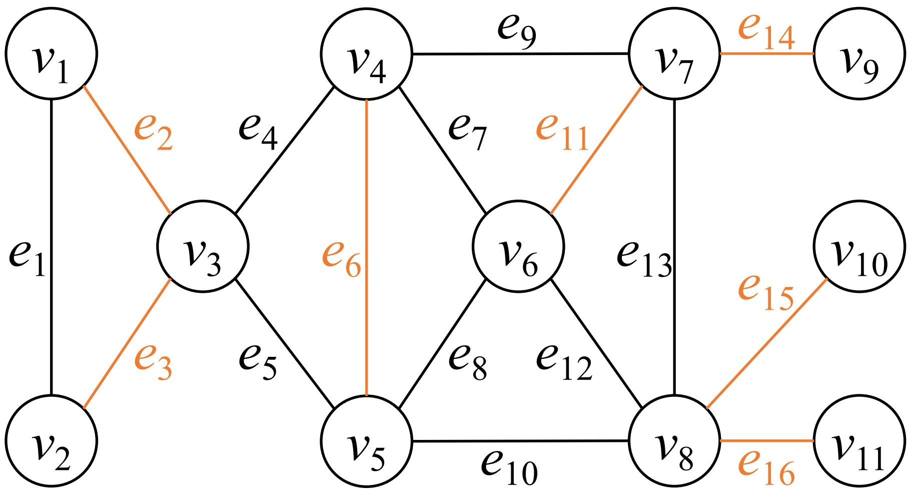
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$C' \subseteq E$
  - **边覆盖集**：$C'$中所有边的端点形成的集合为$V$



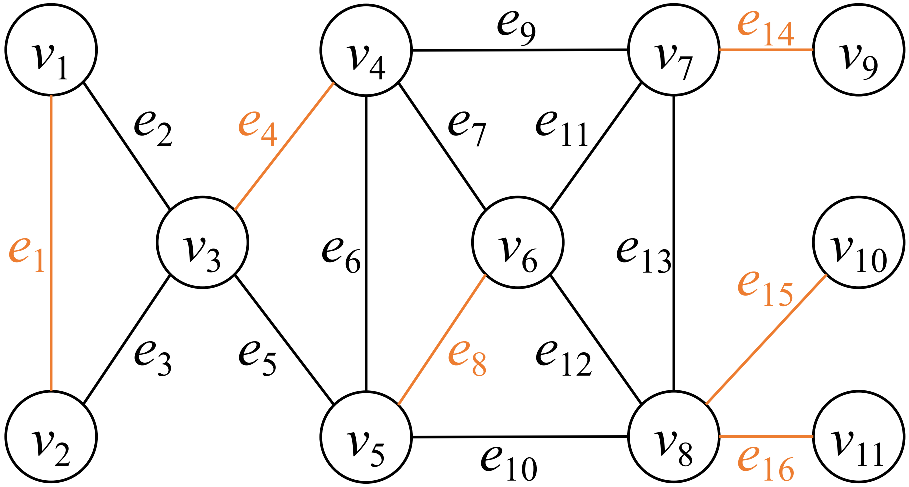
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$C' \subseteq E$
  - 边覆盖集：$C'$中所有边的端点形成的集合为$V$
  - **极小边覆盖集**：边覆盖集，且$G$的任何边覆盖集都不是$C'$的真子集



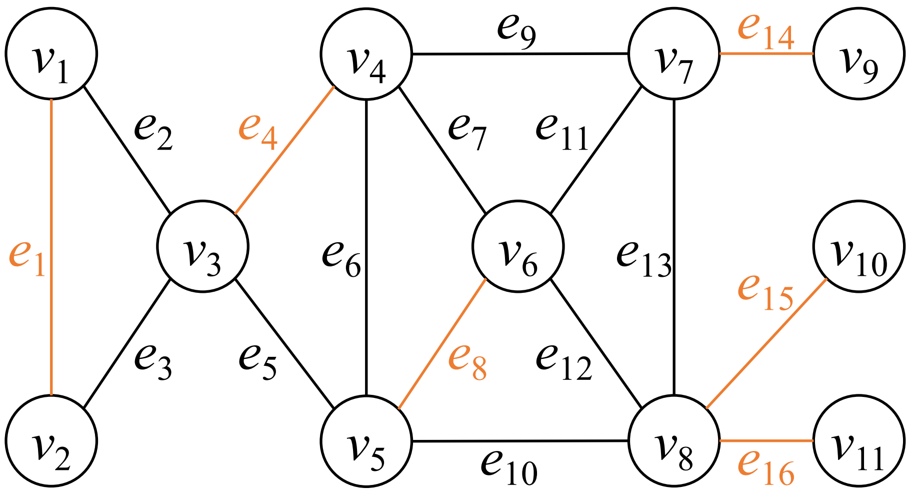
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$C' \subseteq E$
  - 边覆盖集：$C'$中所有边的端点形成的集合为$V$
  - 极小边覆盖集：边覆盖集，且$G$的任何边覆盖集都不是$C'$的真子集
  - **最小边覆盖集**：边的数量最少的边覆盖集



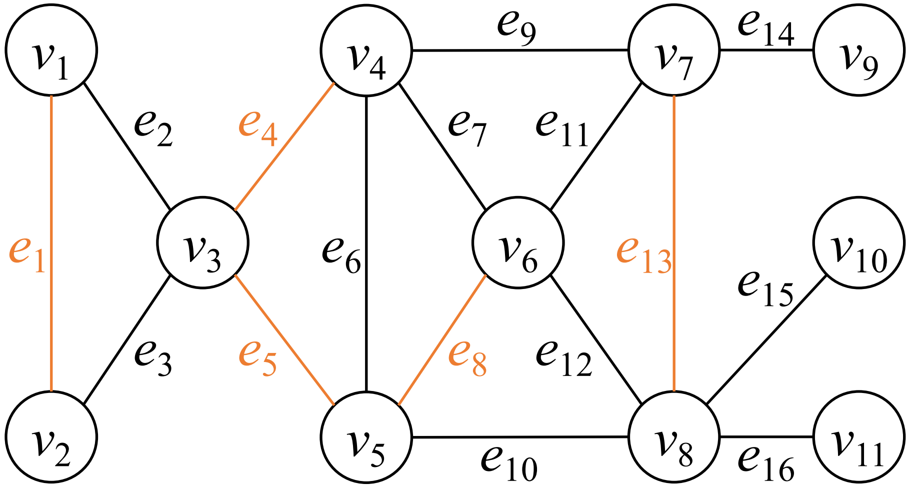
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$C' \subseteq E$
  - 边覆盖集：$C'$中所有边的端点形成的集合为$V$
  - 极小边覆盖集：边覆盖集，且$G$的任何边覆盖集都不是$C'$的真子集
  - 最小边覆盖集：边的数量最少的边覆盖集
  - **边覆盖数**：最小边覆盖集的大小，记作$\beta'(G)$



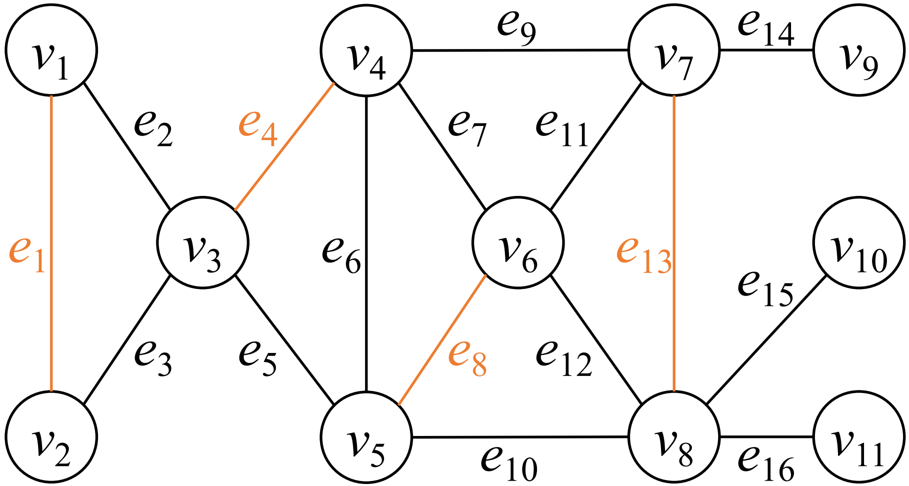
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$D' \subseteq E$
  - **边支配集**：$E \setminus D'$中的每条边都与$D'$中的至少一条边相邻



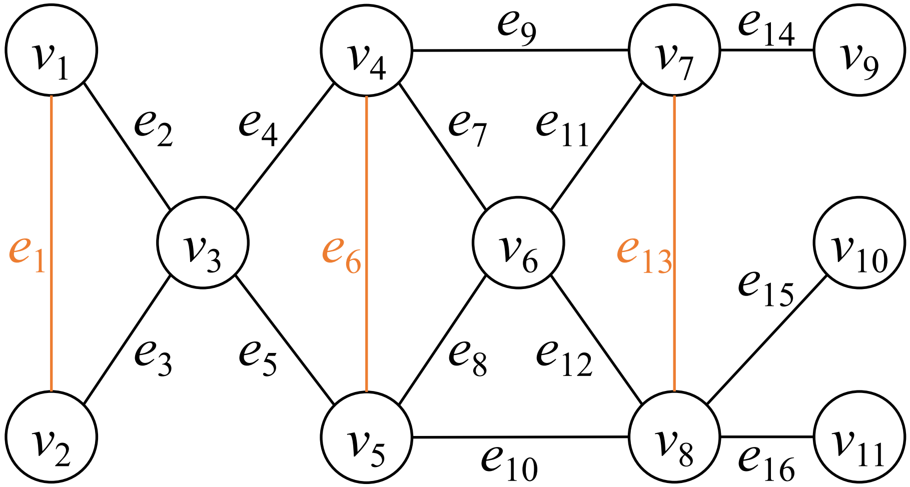
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$D' \subseteq E$
  - 边支配集：$E \setminus D'$中的每条边都与$D'$中的至少一条边相邻
  - **极小边支配集**：边支配集，且$G$的任何边支配集都不是$D'$的真子集



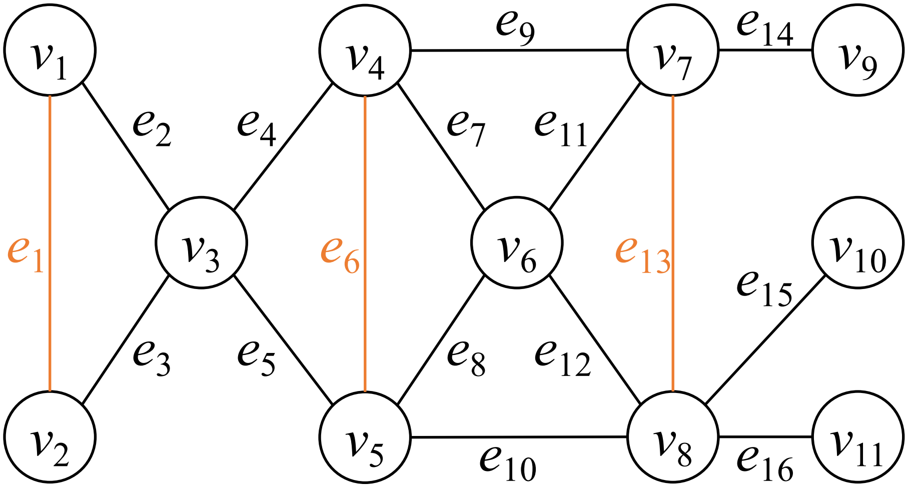
# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$D' \subseteq E$
  - 边支配集：$E \setminus D'$中的每条边都与$D'$中的至少一条边相邻
  - 极小边支配集：边支配集，且$G$的任何边支配集都不是$D'$的真子集
  - **最小边支配集**：边的数量最少的边支配集



# 边的独立、覆盖和支配

- 对于图$G = <V, E>$和边子集$D' \subseteq E$
  - 边支配集：$E \setminus D'$中的每条边都与$D'$中的至少一条边相邻
  - 极小边支配集：边支配集，且$G$的任何边支配集都不是$D'$的真子集
  - 最小边支配集：边的数量最少的边支配集
  - **边支配数**：最小边支配集的大小，记作$\gamma'(G)$



# 边的独立、覆盖和支配

- 每个图都有边独立集、边覆盖集和边支配集吗？

# 边的独立、覆盖和支配

- 每个图都有边独立集、边覆盖集和边支配集吗？
- 阶为$n$的图的边独立数的上界是多少？边覆盖数的下界是多少？边支配数的下界是多少？

# 边的独立、覆盖和支配

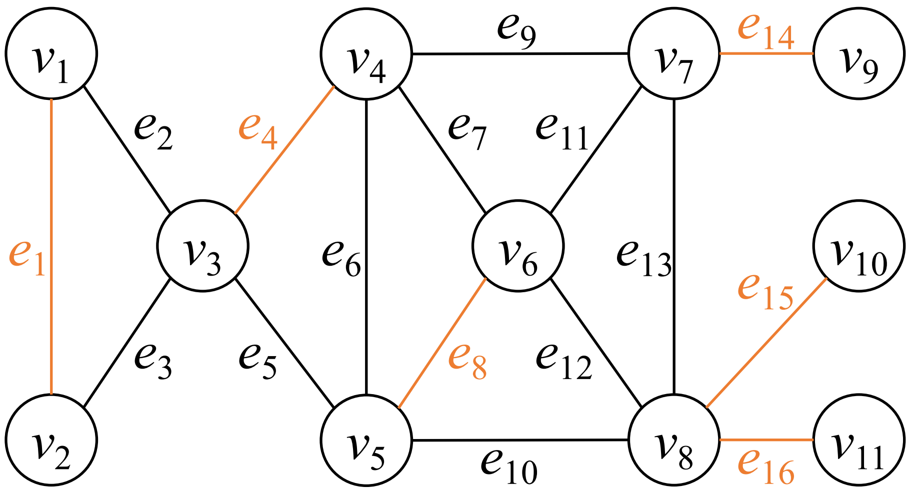
- 每个图都有边独立集、边覆盖集和边支配集吗？
- 阶为$n$的图的边独立数的上界是多少？边覆盖数的下界是多少？边支配数的下界是多少？

- 完全图$K_n$的边独立数、边覆盖数和边支配数分别是多少？
- 完全二分图$K_{m,n}$的边独立数、边覆盖数和边支配数分别是多少？

**随堂小测**

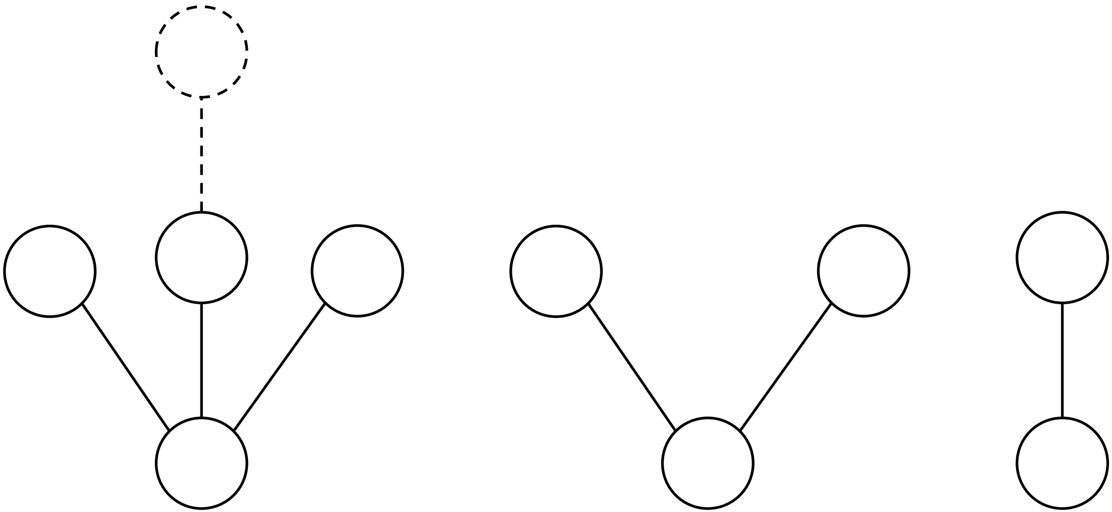
# 边的独立、覆盖和支配

- 极小边覆盖集的边导出子图的每个连通分支有什么特征？



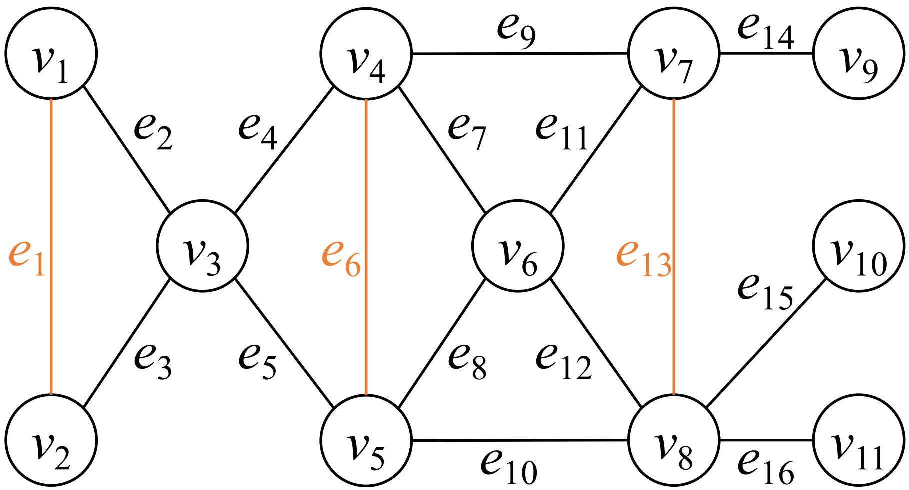
# 边的独立、覆盖和支配

- 极小边覆盖集的边导出子图的每个连通分支有什么特征？



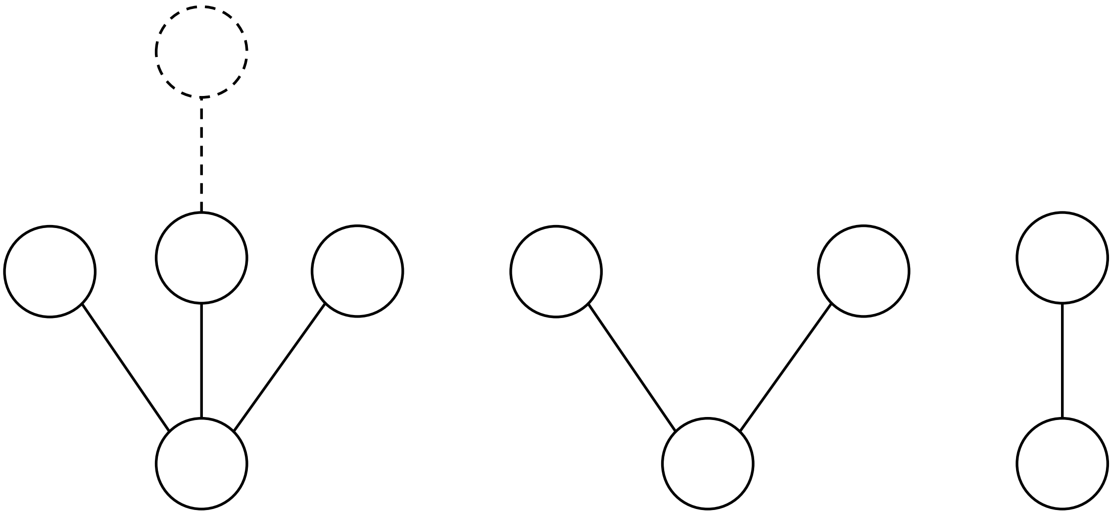
# 边的独立、覆盖和支配

- 极小边支配集的边导出子图的每个连通分支有什么特征？



# 边的独立、覆盖和支配

- 极小边支配集的边导出子图的每个连通分支有什么特征？

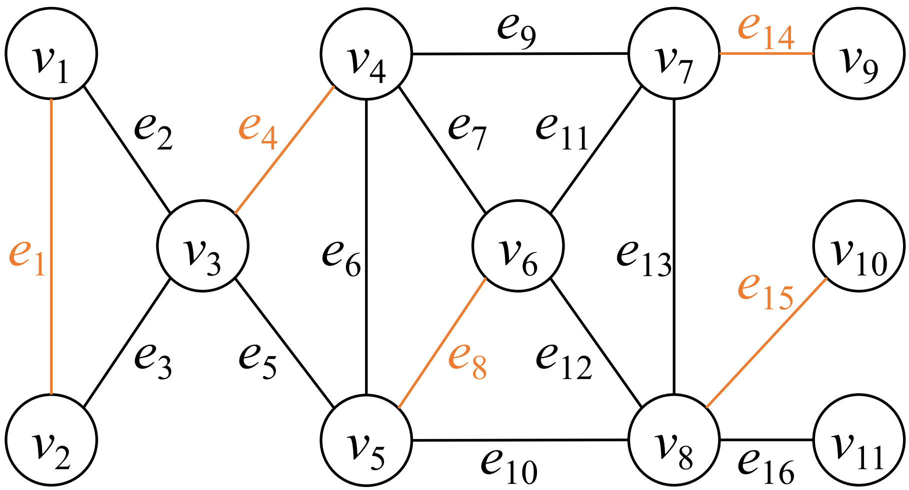


# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) \le \beta'(G)$

# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) \le \beta'(G)$
- 你能自己证明吗？
  （基于最大边独立集，分析边覆盖数。）



# 边的独立、覆盖和支配

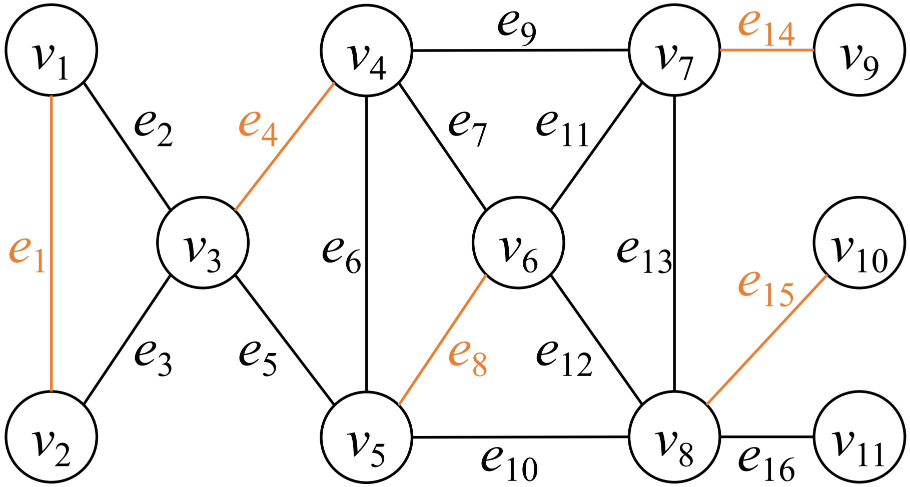
- 对于任意一个不含孤立点的图$G$：$\alpha'(G) \le \beta'(G)$
- 若图$G$的一个边独立集和一个边覆盖集的大小相等，你能得出什么结论？

# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$

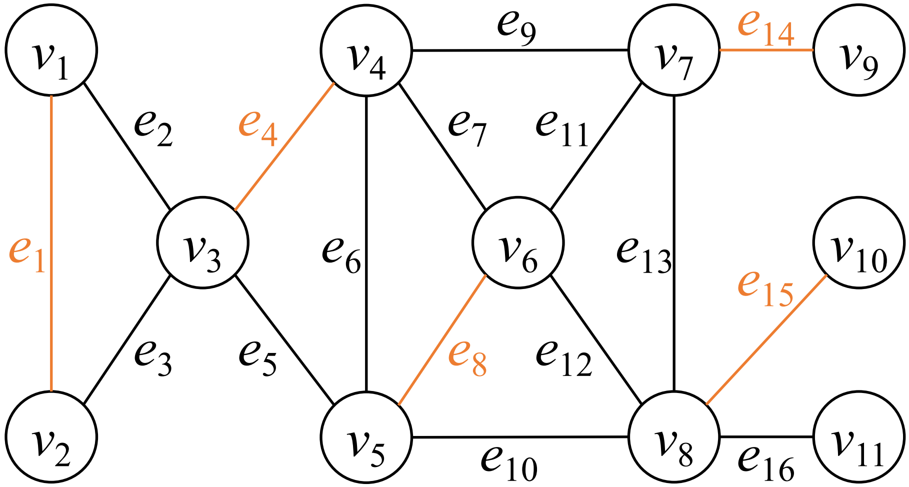
# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$
- $\alpha'(G) + \beta'(G) \le \nu(G)$，即$\beta'(G) \le \nu(G) - \alpha'(G)$，你能自己证明吗？（基于最大边独立集构造边覆盖集）



# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$
- $\alpha'(G) + \beta'(G) \leq \nu(G)$，即$\beta'(G) \leq \nu(G) - \alpha'(G)$，你能自己证明吗？（基于最大边独立集构造边覆盖集）
  - 构造的边覆盖集的大小：$\alpha'(G) + (\nu(G) - 2\alpha'(G)) = \nu(G) - \alpha'(G)$

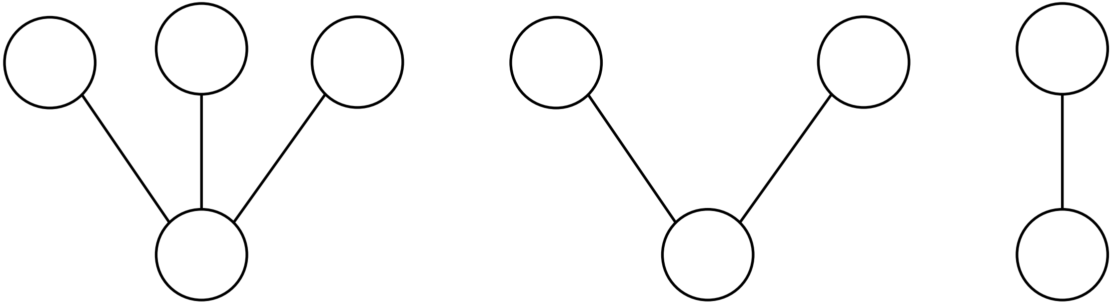


# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$
- $\alpha'(G) + \beta'(G) \geq \nu(G)$，即$\alpha'(G) \geq \nu(G) - \beta'(G)$
  （基于最小边覆盖集构造边独立集）

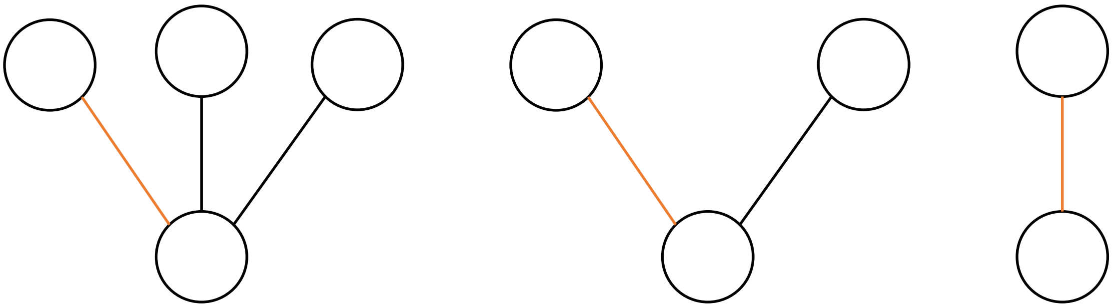
# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$
- $\alpha'(G) + \beta'(G) \geq \nu(G)$，即$\alpha'(G) \geq \nu(G) - \beta'(G)$（基于最小边覆盖集构造边独立集）
  - 最小边覆盖集$C'$的边导出子图$G[C']$的每个连通分支是星
  - $G[C']$含$\nu(G) - \beta'(G)$个连通分支，为什么？



# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$
- $\alpha'(G) + \beta'(G) \geq \nu(G)$，即$\alpha'(G) \geq \nu(G) - \beta'(G)$（基于最小边覆盖集构造边独立集）
  - 最小边覆盖集$C'$的边导出子图$G[C']$的每个连通分支是星
  - $G[C']$含$\nu(G) - \beta'(G)$个连通分支，为什么？
  - 构造的边独立集的大小：$\nu(G) - \beta'(G)$

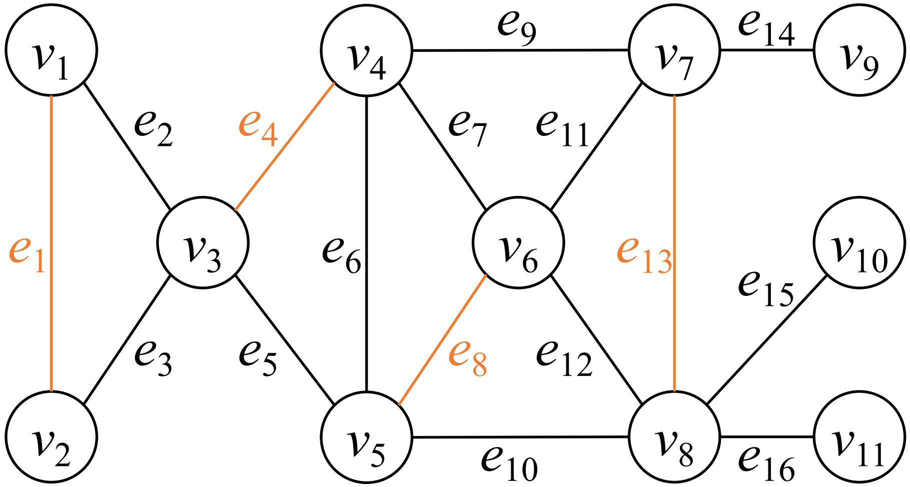


# 边的独立、覆盖和支配

- 对于任意一个不含孤立点的图$G$：$\alpha'(G) + \beta'(G) = \nu(G)$
- 对于任意一个不含孤立点的图$G$，$\alpha'(G) = \beta'(G)$当且仅当$G$有完美匹配。
  - 你能自己证明吗？

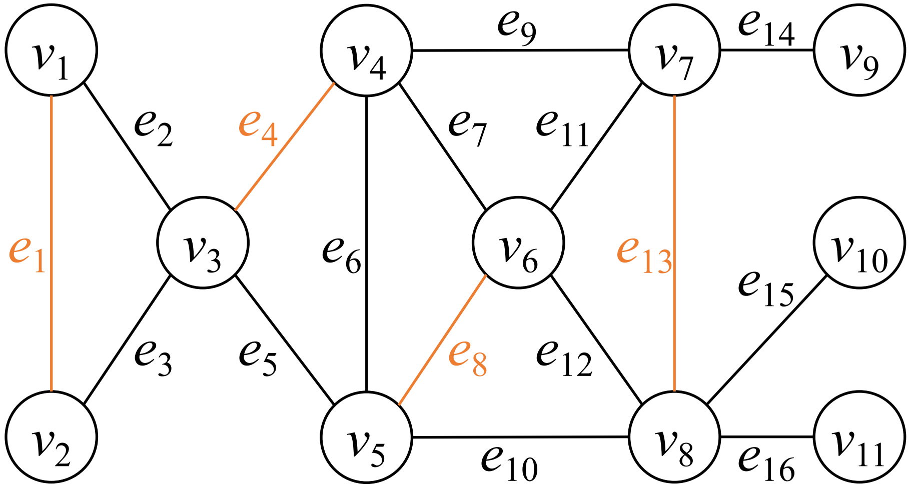
# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？



# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？
- 对于任意一个图$G$：$\alpha'(G) \geq \gamma'(G)$
  - 你能自己证明吗？

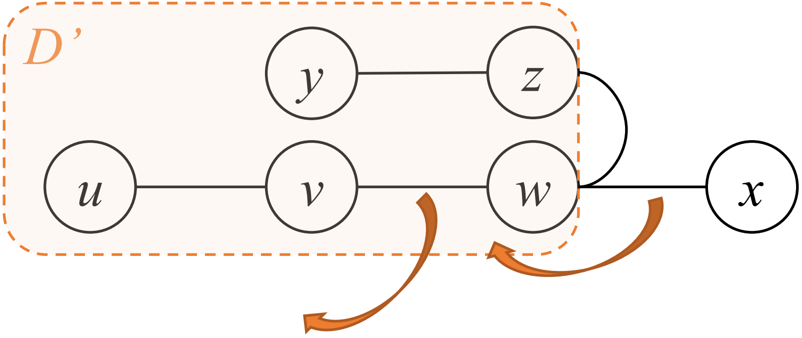


# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？
- 对于任意一个图$G$：$\alpha'(G) \geq \gamma'(G)$
- 对于任意一个图，存在一个最小边支配集是极大边独立集。

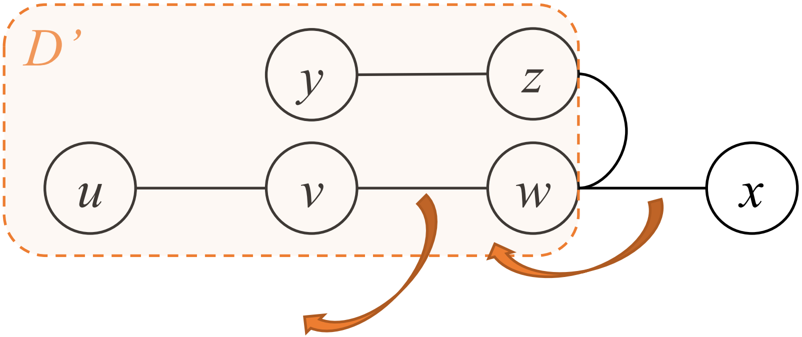
# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？
- 对于任意一个图$G$：$\alpha'(G) \geq \gamma'(G)$
- 对于任意一个图，存在一个最小边支配集是极大边独立集。
  - 思路：若非边独立集，则通过反复替换边可减少相邻的边的对数，同时保持最小边支配集，直至得到边独立集，也是极大边独立集（为什么？）



# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？
- 对于任意一个图$G$：$\alpha'(G) \geq \gamma'(G)$
- 对于任意一个图，存在一个最小边支配集是极大边独立集。
  - 思路：若非边独立集，则通过反复替换边可减少相邻的边的对数，同时保持最小边支配集，直至得到边独立集，也是极大边独立集（为什么？）
  - 关键：$w$关联一条边$(w, x)$满足$x \neq v$且与$D' \setminus \{(w, v)\}$中的每条边都不相邻（否则，会导致什么矛盾？）



# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？
- 对于任意一个图$G$：$\alpha'(G) \geq \gamma'(G)$
- 对于任意一个图，存在一个最小边支配集是极大边独立集。
- 对于任意一个图，边支配数是最小的极大边独立集包含的边数。

# 边的独立、覆盖和支配

- 边独立集是边支配集吗？极大边独立集是边支配集吗？
- 对于任意一个图$G$：$\alpha'(G) \geq \gamma'(G)$
- 对于任意一个图，存在一个最小边支配集是极大边独立集。
- 对于任意一个图，边支配数是最小的极大边独立集包含的边数。
  - 否则，会导致什么矛盾？

# 边的独立、覆盖和支配

- 如何找出图中的最大边独立集？
- 如何找出图中的最小边覆盖集？
- 如何找出图中的最小边支配集？

# 边的独立、覆盖和支配

- 如何找出图中的最大边独立集？
- 如何找出图中的最小边覆盖集？
- 如何找出图中的最小边支配集？

# 边的独立、覆盖和支配

- 如何找出图中的最大边独立集？
- 如何找出图中的最小边覆盖集？
- 如何找出图中的最小边支配集？

# 边的独立、覆盖和支配

- Robert Z. Norman
- Michael O. Rabin，1931-，出生于德国，1976年获图灵奖

提出了非确定性自动机

https://math.dartmouth.edu/gallery/picture.php?/161/category/7
https://en.wikipedia.org/wiki/Michael_O._Rabin

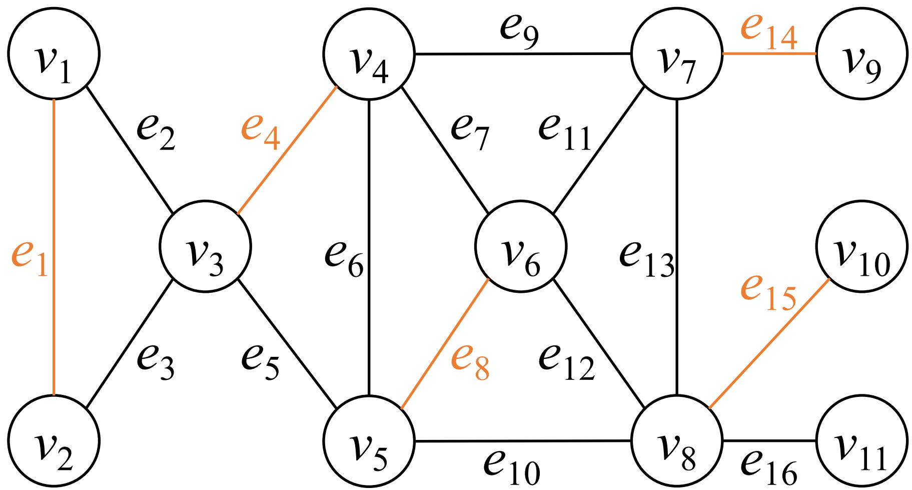
# 边的独立、覆盖和支配

- 从最大边独立集$I'$开始

**算法 13:** 计算最小边覆盖集的算法

**输入:** 不含孤立点的图 $G = \langle V, E\rangle$

```
I′ ← G 的最大边独立集;
C′ ← I′;
foreach v ∈ V do
    if v 未被 I′ 饱和 then
        e ← v 关联的任意一条边;
        C′ ← C′ ∪ {e};
输出 (C′);
```



# 边的独立、覆盖和支配

- 对于未被$I'$饱和的每个顶点$v$

---

**算法 13:** 计算最小边覆盖集的算法

---

**输入:** 不含孤立点的图 $G=\langle V,E\rangle$

$I' \leftarrow G$ 的最大边独立集;

$C' \leftarrow I'$;

**foreach** $v \in V$ **do**

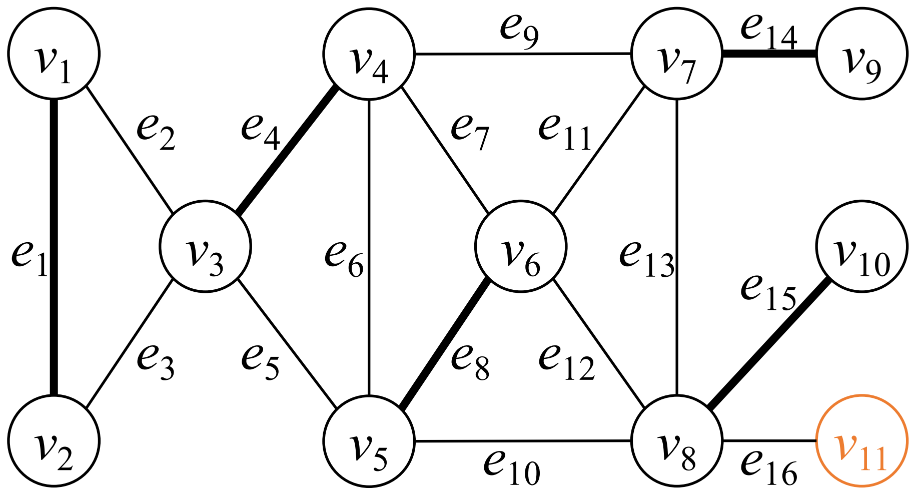
  **if** $v$ 未被 $I'$ 饱和 **then**

    $e \leftarrow v$ 关联的任意一条边;

    $C' \leftarrow C' \cup \{e\}$;

输出 $(C')$;

---



# 边的独立、覆盖和支配

- 对于未被$I'$饱和的每个顶点$v$，向$C'$中增加$v$关联的任意一条边

**算法 13:** 计算最小边覆盖集的算法

**输入:** 不含孤立点的图 $G=\langle V,E\rangle$

$I' \leftarrow G$ 的最大边独立集;

$C' \leftarrow I'$;

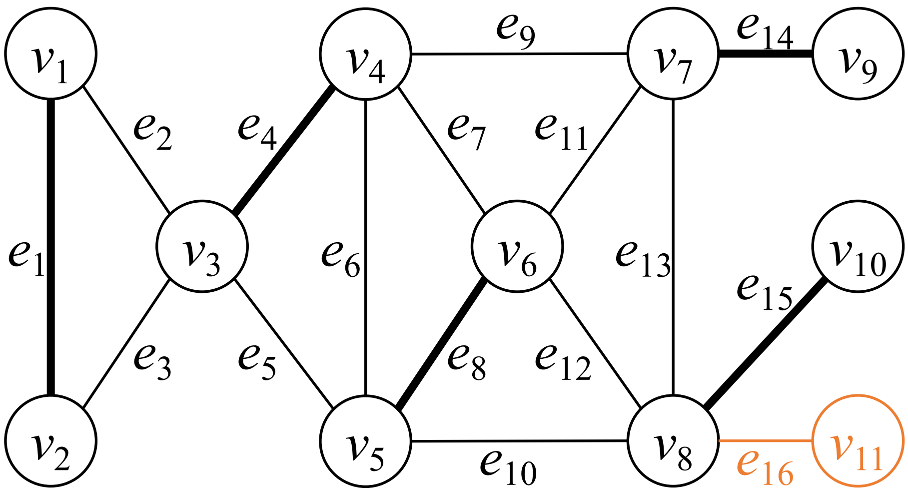
**foreach** $v \in V$ **do**

  **if** $v$ 未被 $I'$ 饱和 **then**

    $e \leftarrow v$ 关联的任意一条边;

    $C' \leftarrow C' \cup \{e\}$;

输出 $(C')$;



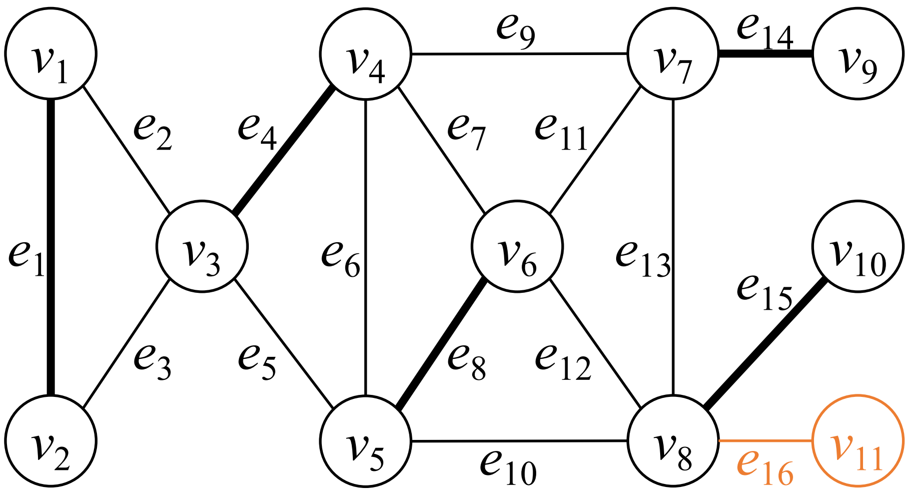
# 边的独立、覆盖和支配

- 对于未被$I'$饱和的每个顶点$v$，向$C'$中增加$v$关联的任意一条边
  - $v$一定有关联的边吗？

**算法 13:** 计算最小边覆盖集的算法

**输入:** 不含孤立点的图 $G = \langle V, E \rangle$

```
I′ ← G 的最大边独立集;
C′ ← I′;
foreach v ∈ V do
    if v 未被 I′ 饱和 then
        e ← v 关联的任意一条边;
        C′ ← C′ ∪ {e};
输出 (C′);
```



# 边的独立、覆盖和支配

- 输出$C'$

---

**算法 13:** 计算最小边覆盖集的算法

---

**输入:** 不含孤立点的图 $G = \langle V, E\rangle$

$I' \leftarrow G$ 的最大边独立集;

$C' \leftarrow I'$;

**foreach** $v \in V$ **do**

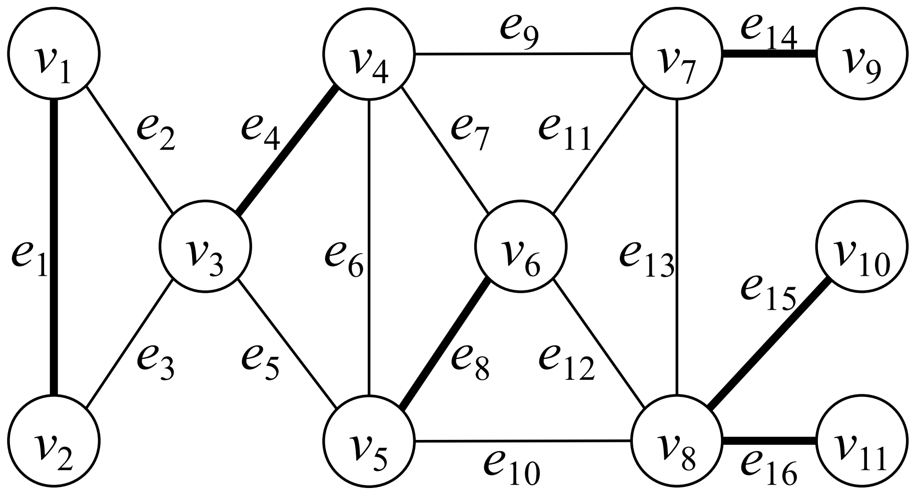
  **if** $v$ 未被 $I'$ 饱和 **then**

    $e \leftarrow v$ 关联的任意一条边;

    $C' \leftarrow C' \cup \{e\}$;

输出 ($C'$);

---



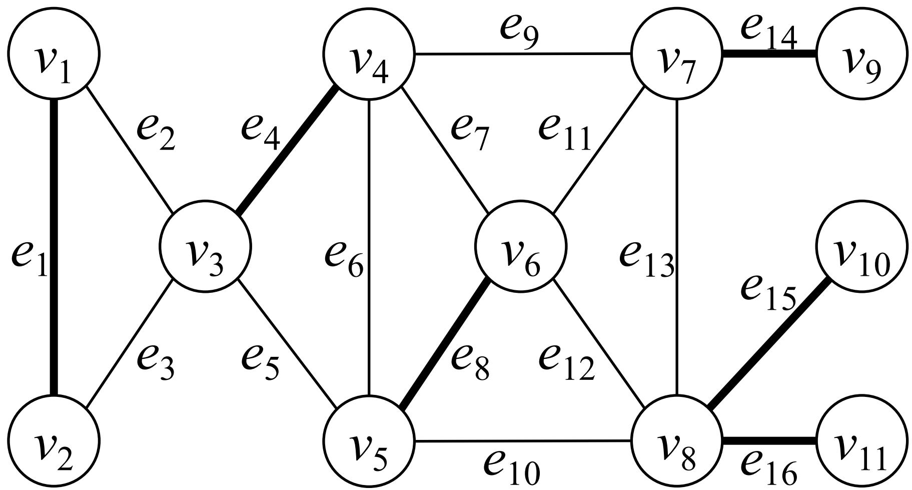
# 边的独立、覆盖和支配

- 输出$C'$
  - $C'$为什么是边覆盖集？

**算法 13:** 计算最小边覆盖集的算法

**输入:** 不含孤立点的图 $G = \langle V, E \rangle$

```
I′ ← G 的最大边独立集;
C′ ← I′;
foreach v ∈ V do
    if v 未被 I′ 饱和 then
        e ← v 关联的任意一条边;
        C′ ← C′ ∪ {e};
输出 (C′);
```



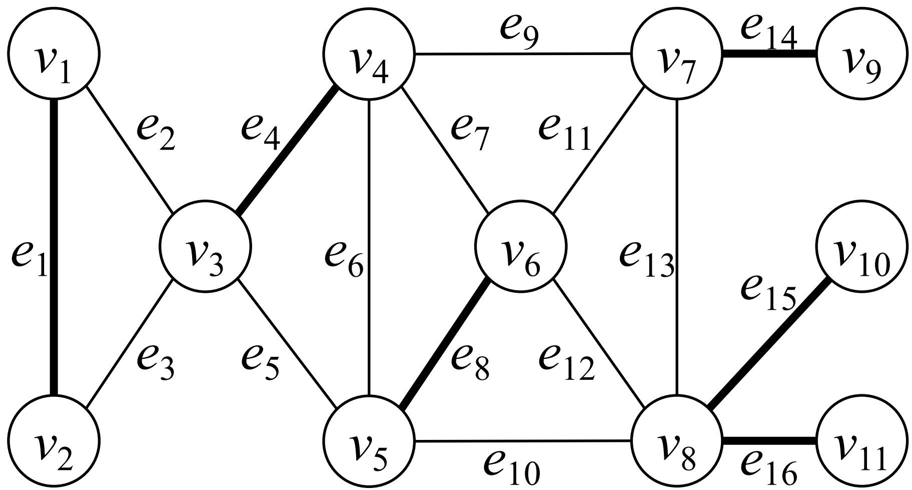
# 边的独立、覆盖和支配

- 输出$C'$
  - $C'$为什么是边覆盖集？
  - $C'$为什么是最小边覆盖集？
    - $|C'| = ?$

**算法 13:** 计算最小边覆盖集的算法

**输入:** 不含孤立点的图 $G = \langle V, E \rangle$

```
I′ ← G 的最大边独立集;
C′ ← I′;
foreach v ∈ V do
    if v 未被 I′ 饱和 then
        e ← v 关联的任意一条边;
        C′ ← C′ ∪ {e};
输出 (C′);
```



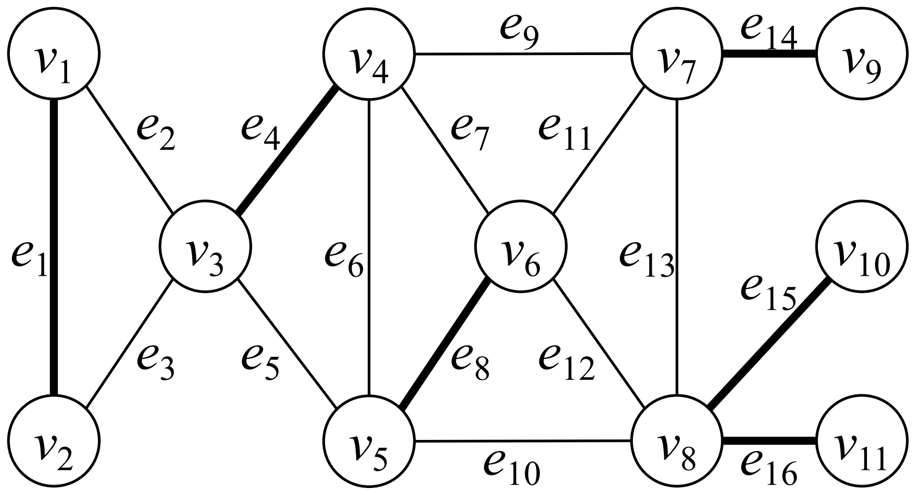
# 边的独立、覆盖和支配

- 输出$C'$
  - $C'$为什么是边覆盖集？
  - $C'$为什么是最小边覆盖集？
    - $|C'| = \alpha'(G) + (n - 2\alpha'(G)) = n - \alpha'(G) = \beta'(G)$

**算法 13:** 计算最小边覆盖集的算法

**输入:** 不含孤立点的图 $G = \langle V, E\rangle$

```
I′ ← G 的最大边独立集;
C′ ← I′;
foreach v ∈ V do
    if v 未被 I′ 饱和 then
        e ← v 关联的任意一条边;
        C′ ← C′ ∪ {e};
输出 (C′);
```

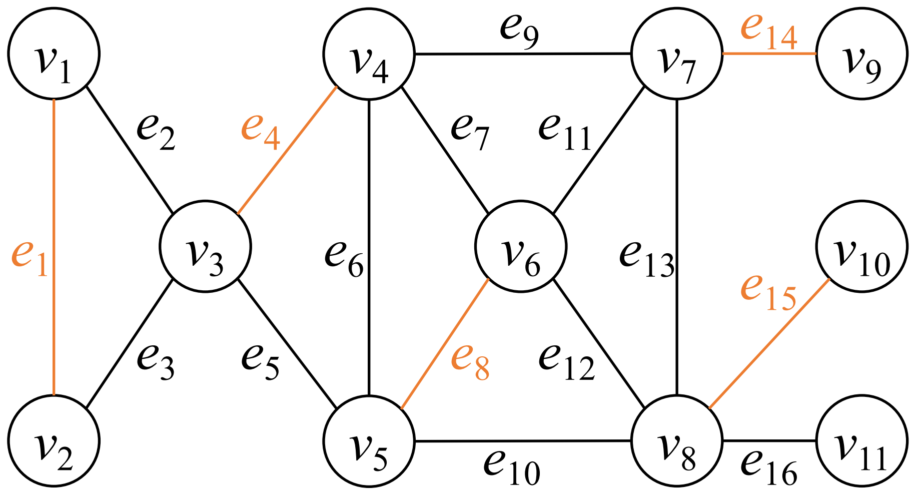


# 边的独立、覆盖和支配

- 时间复杂度：与计算最大边独立集的算法的时间复杂度相同

**算法 13:** 计算最小边覆盖集的算法

```
输入: 不含孤立点的图 G = ⟨V, E⟩
I′ ← G 的最大边独立集;
C′ ← I′;
foreach v ∈ V do
    if v 未被 I′ 饱和 then
        e ← v 关联的任意一条边;
        C′ ← C′ ∪ {e};
输出 (C′);
```



# 边的独立、覆盖和支配

- 如何找出图中的最大边独立集？
- 如何找出图中的最小边覆盖集？
- 如何找出图中的最小边支配集？

# 边的独立、覆盖和支配

- 最小边支配集的计算是一个NP难的优化问题，不存在多项式时间算法（除非P=NP）

# 边的独立、覆盖和支配

- 最小边支配集的计算是一个NP难的优化问题，不存在多项式时间算法（除非P=NP）
- **近似算法**：在多项式时间内计算一个较优但未必最优的解

# 边的独立、覆盖和支配

- 极大边独立集是边支配集

# 边的独立、覆盖和支配

- 极大边独立集是边支配集
- 近似比 = 2
  - 极大边独立集包含的边的数量不超过最小边支配集包含的边的数量的2倍（你能自己证明吗？）

# 边的独立、覆盖和支配

- 极大边独立集是边支配集
- 近似比 = 2
  - 极大边独立集包含的边的数量不超过最小边支配集包含的边的数量的2倍（你能自己证明吗？）
    - 对于任意一个图，边支配数是最小的极大边独立集包含的边数。
    - |任意一个极大边独立集| ≤ 2 × |最小的极大边独立集|，为什么？

# 边的独立、覆盖和支配

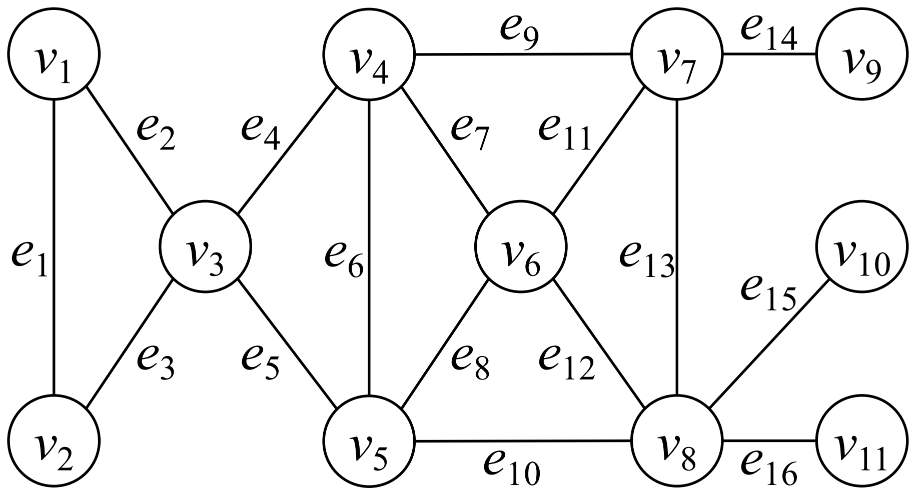
- 计算极大边独立集

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I' ← I' ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I');
```



# 边的独立、覆盖和支配

- 计算极大边独立集
  - 从初值为空集的边独立集$I'$开始

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**

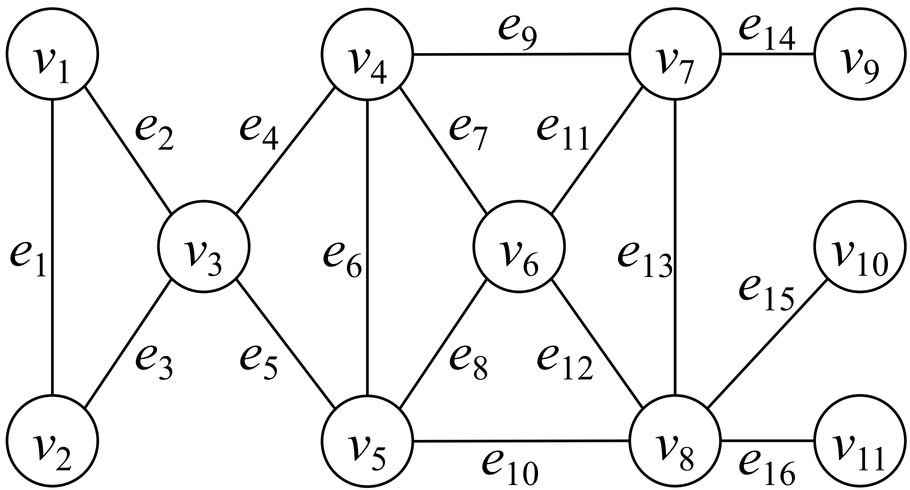
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**

    $I' \leftarrow I' \cup \{(u, v)\}$;

    $u$.saturated $\leftarrow$ true;

    $v$.saturated $\leftarrow$ true;

输出 $(I')$;



# 边的独立、覆盖和支配

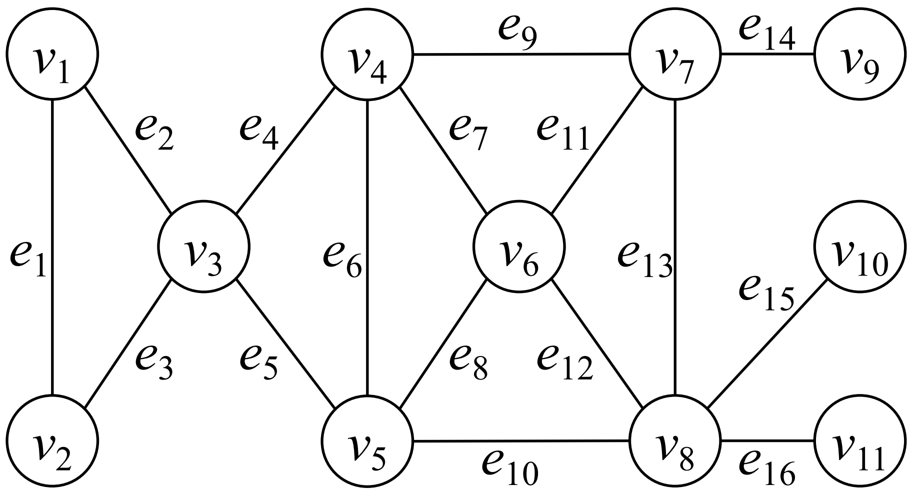
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
    **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
        $I' \leftarrow I' \cup \{(u, v)\}$;
        $u$.saturated $\leftarrow$ true;
        $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**

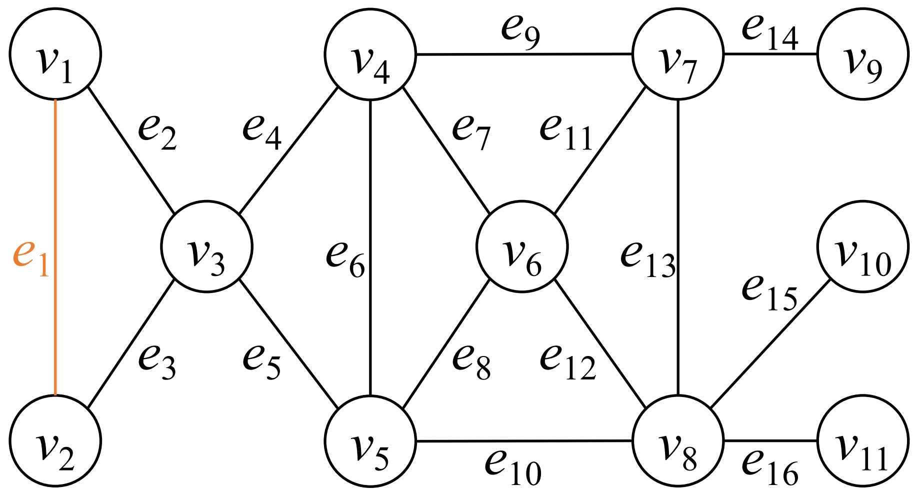
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**

    $I' \leftarrow I' \cup \{(u, v)\}$;

    $u$.saturated $\leftarrow$ true;

    $v$.saturated $\leftarrow$ true;

输出 ($I'$);



# 边的独立、覆盖和支配

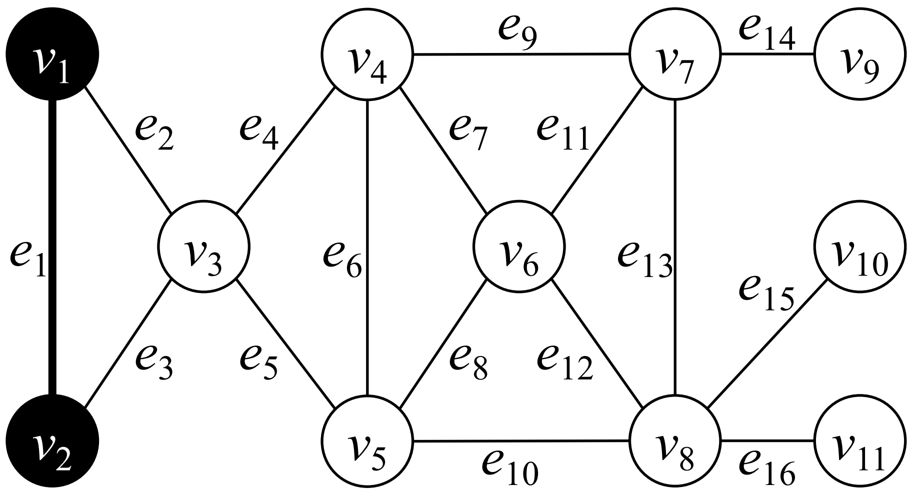
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

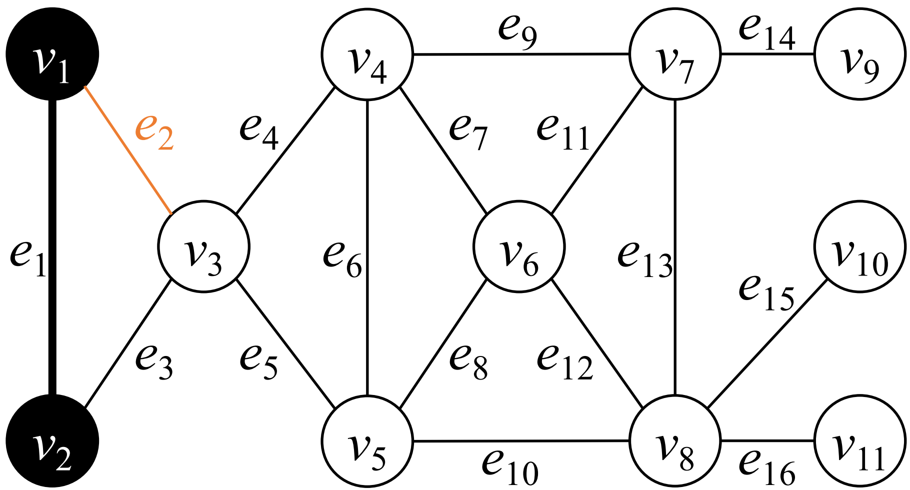
**foreach** $(u, v) \in E$ **do**
    **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
        $I' \leftarrow I' \cup \{(u, v)\}$;
        $u$.saturated $\leftarrow$ true;
        $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

```
算法 14: 计算极大边独立集的算法
  输入: 图 G = ⟨V, E⟩
  初值: I′ 初值为 ∅; V 中所有顶点的 saturated 初值为 false
  foreach (u, v) ∈ E do
      if u.saturated = false 且 v.saturated = false then
          I′ ← I′ ∪ {(u, v)};
          u.saturated ← true;
          v.saturated ← true;
  输出 (I′);
```



# 边的独立、覆盖和支配

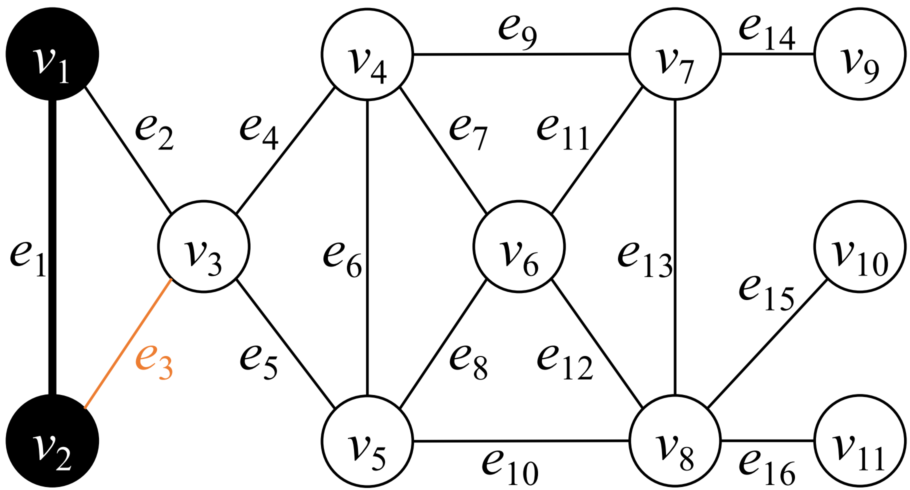
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E\rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I' ← I' ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I');
```



# 边的独立、覆盖和支配

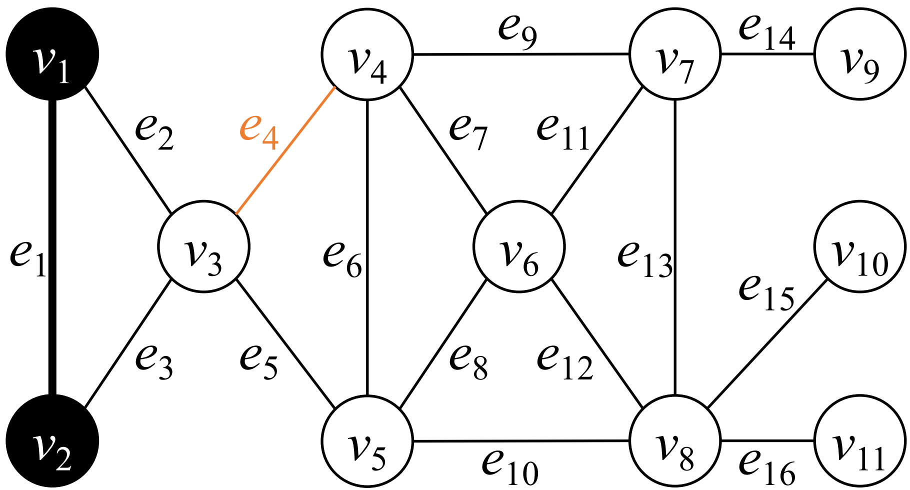
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
    **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
        $I' \leftarrow I' \cup \{(u, v)\}$;
        $u$.saturated $\leftarrow$ true;
        $v$.saturated $\leftarrow$ true;
输出 ($I'$);



# 边的独立、覆盖和支配

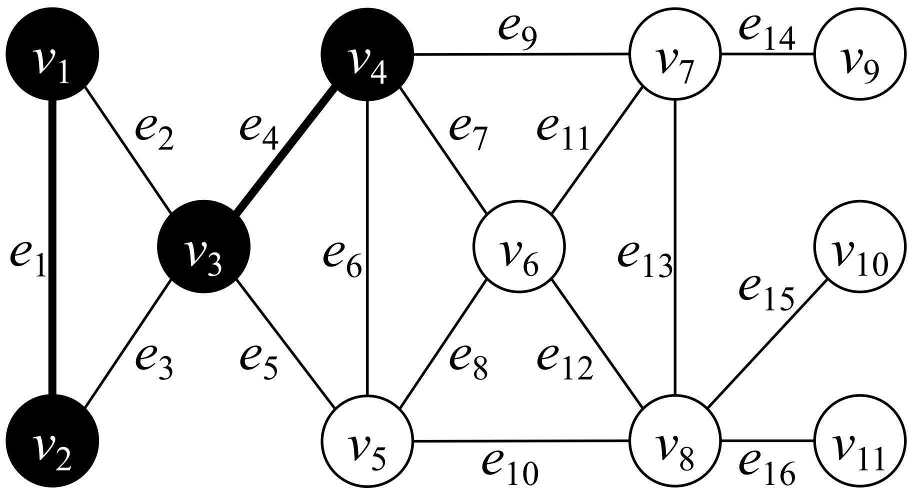
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I′ ← I′ ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I′);
```



# 边的独立、覆盖和支配

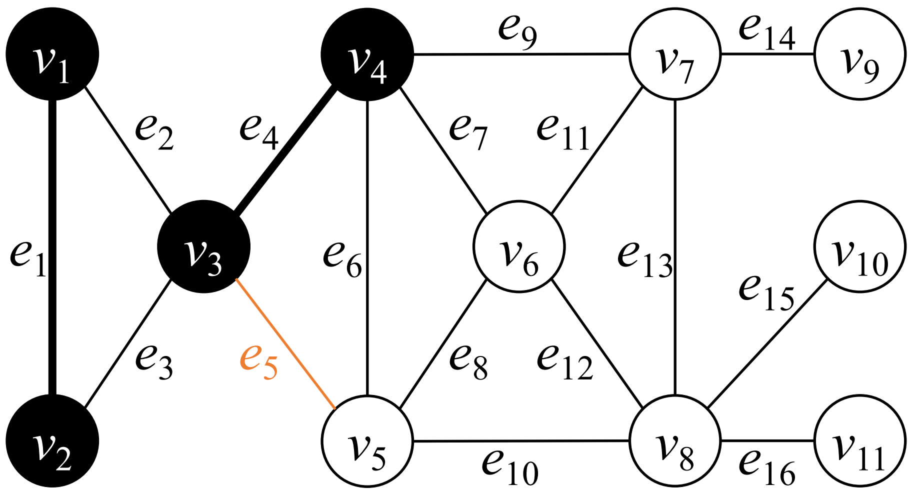
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
    **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
        $I' \leftarrow I' \cup \{(u, v)\}$;
        $u$.saturated $\leftarrow$ true;
        $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$； $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**

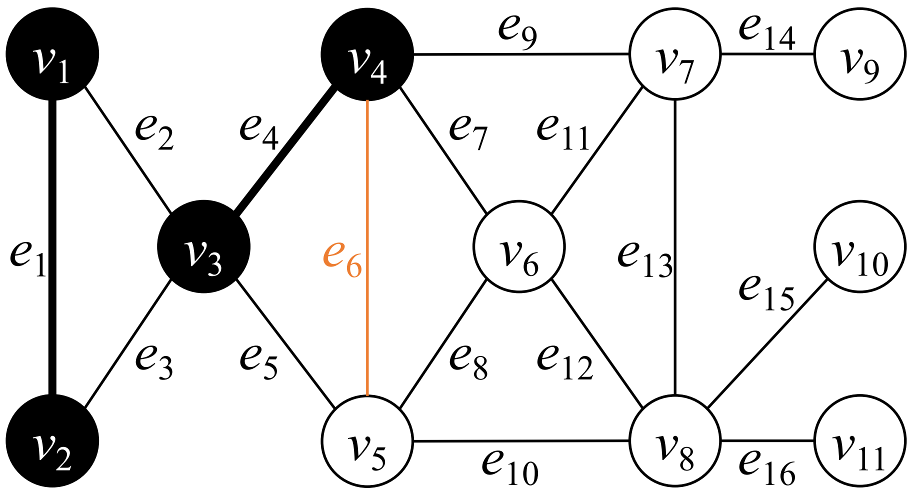
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**

    $I' \leftarrow I' \cup \{(u, v)\}$;

    $u$.saturated ← true;

    $v$.saturated ← true;

输出 $(I')$;



# 边的独立、覆盖和支配

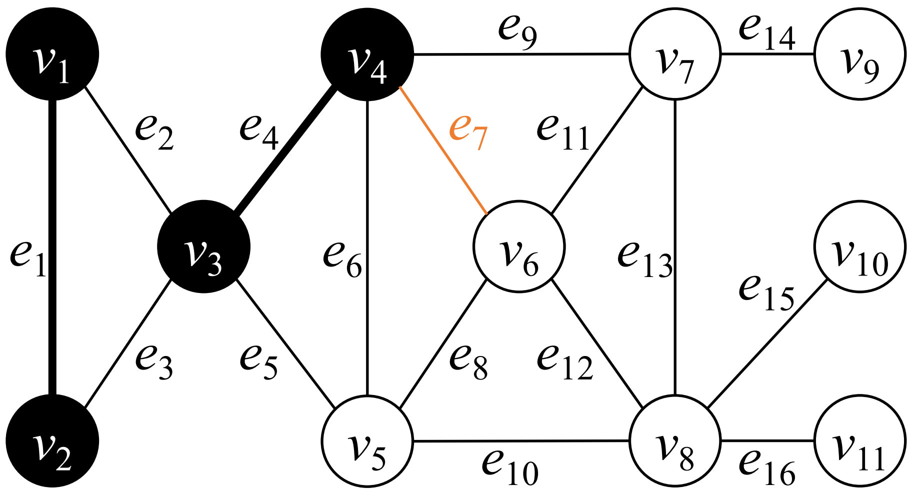
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G=\langle V,E\rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I' ← I' ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I');
```



# 边的独立、覆盖和支配

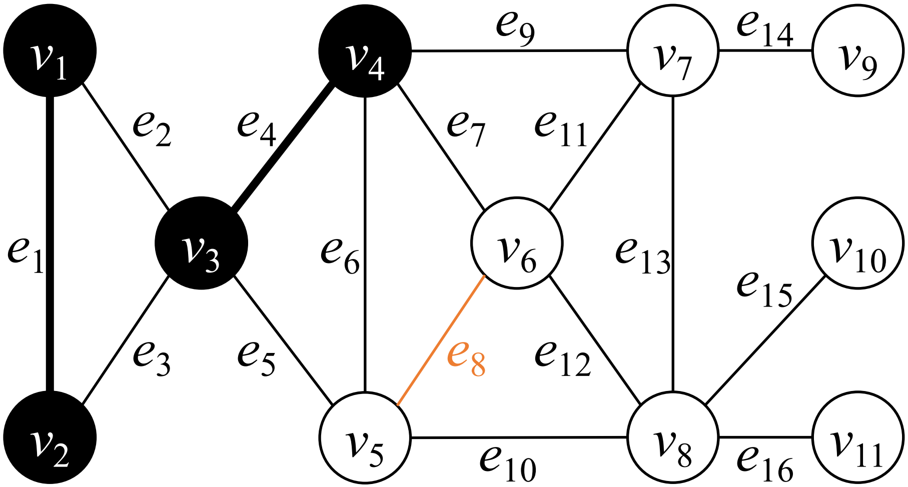
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E\rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I' ← I' ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I');
```



# 边的独立、覆盖和支配

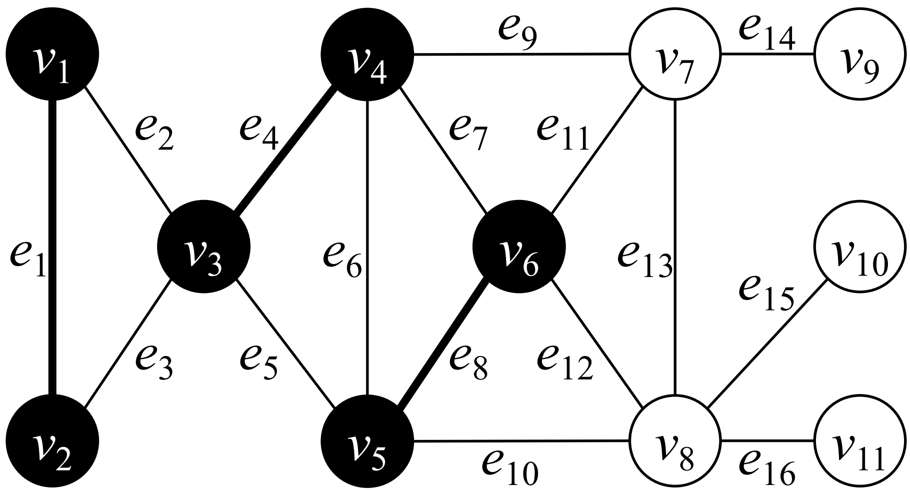
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
    $I' \leftarrow I' \cup \{(u, v)\}$;
    $u$.saturated $\leftarrow$ true;
    $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

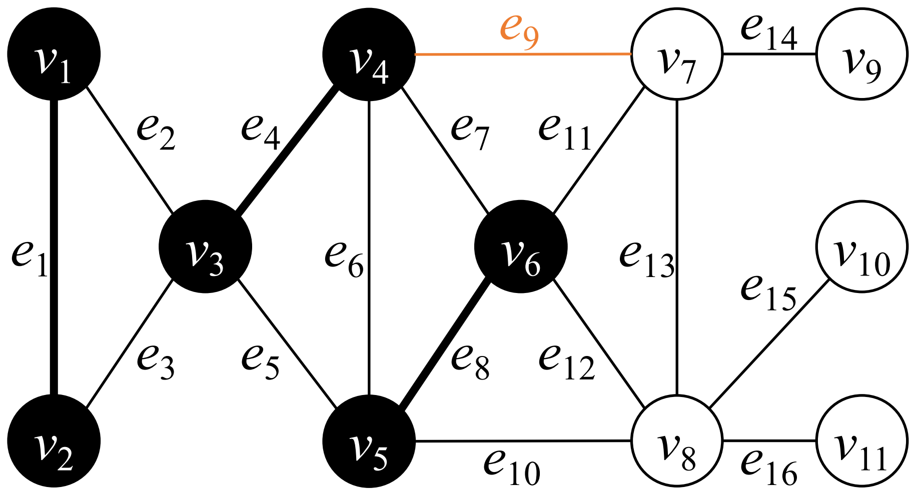
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
    $I' \leftarrow I' \cup \{(u, v)\}$;
    $u$.saturated $\leftarrow$ true;
    $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

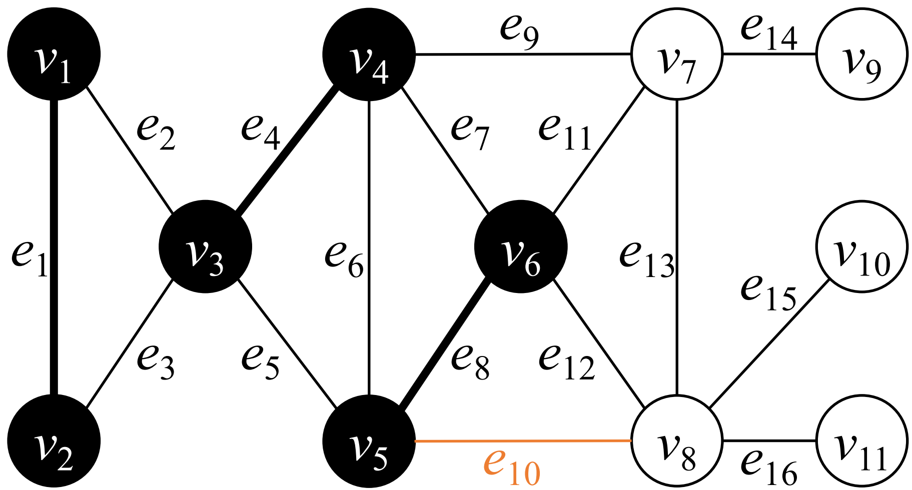
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I' ← I' ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I');
```



# 边的独立、覆盖和支配

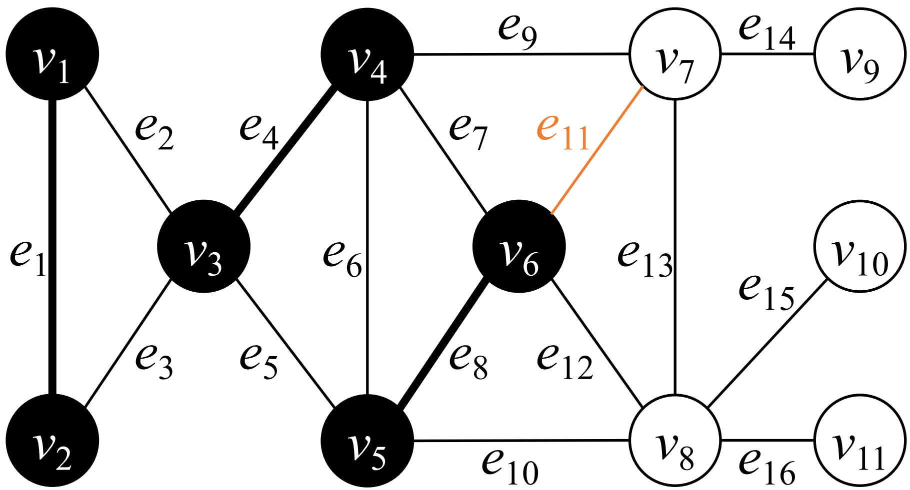
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I′ ← I′ ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I′);
```



# 边的独立、覆盖和支配

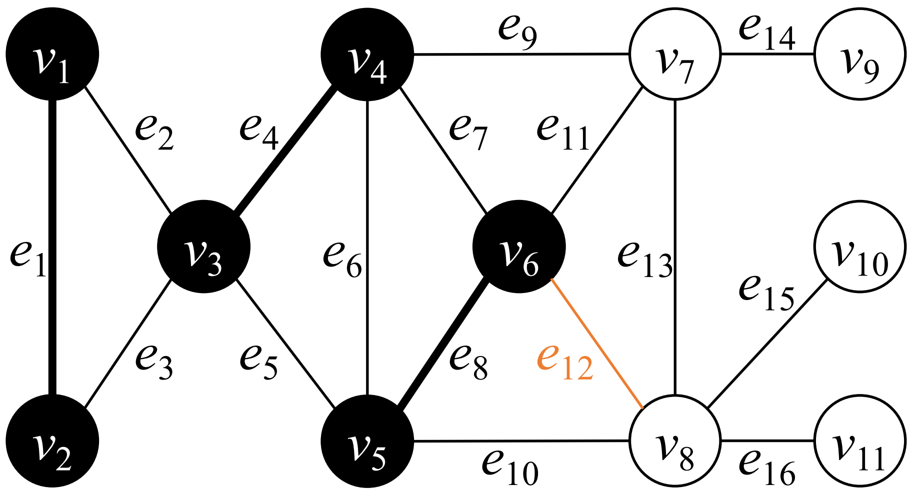
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
    **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
        $I' \leftarrow I' \cup \{(u, v)\}$;
        $u$.saturated $\leftarrow$ true;
        $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

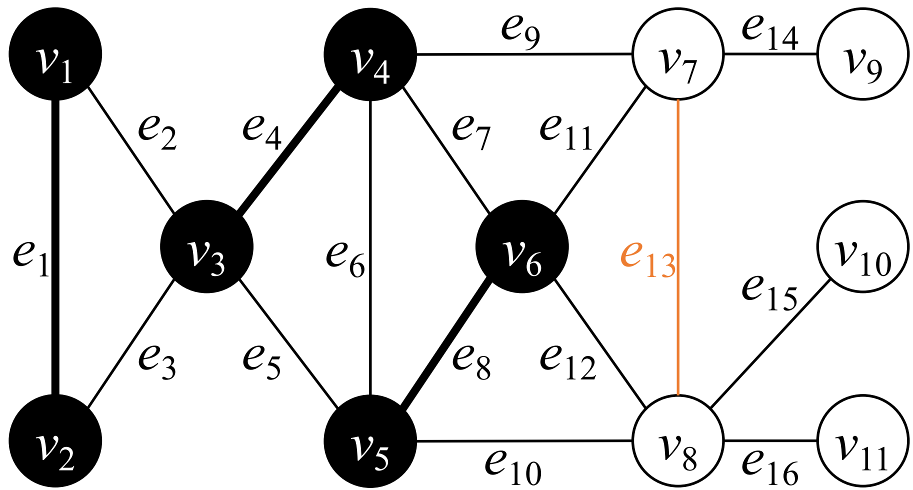
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I' ← I' ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I');
```



# 边的独立、覆盖和支配

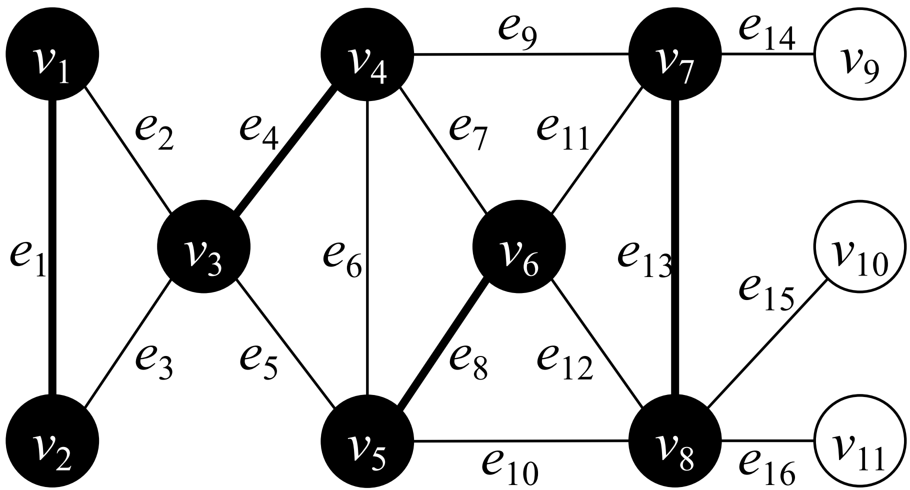
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
　**if** $u.saturated = false$ 且 $v.saturated = false$ **then**
　　$I' \leftarrow I' \cup \{(u, v)\}$;
　　$u$.saturated $\leftarrow$ true;
　　$v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

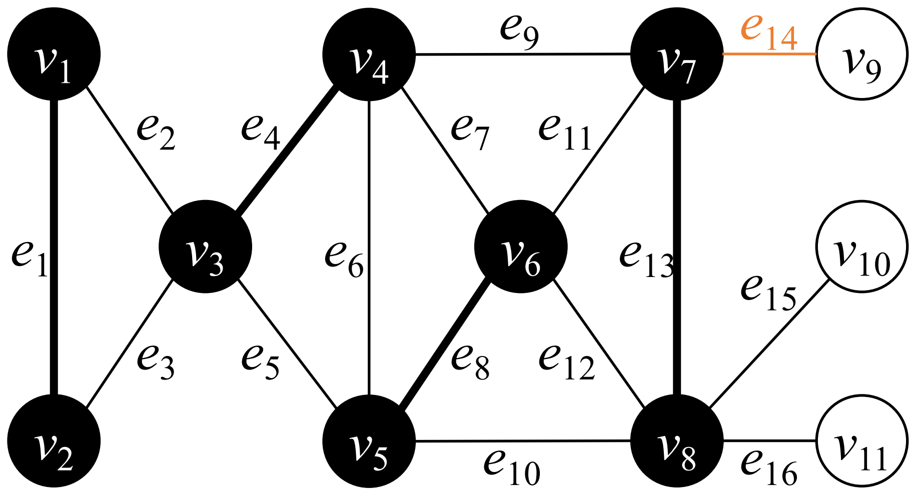
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
    $I' \leftarrow I' \cup \{(u, v)\}$;
    $u$.saturated $\leftarrow$ true;
    $v$.saturated $\leftarrow$ true;
输出（$I'$）;



# 边的独立、覆盖和支配

- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**

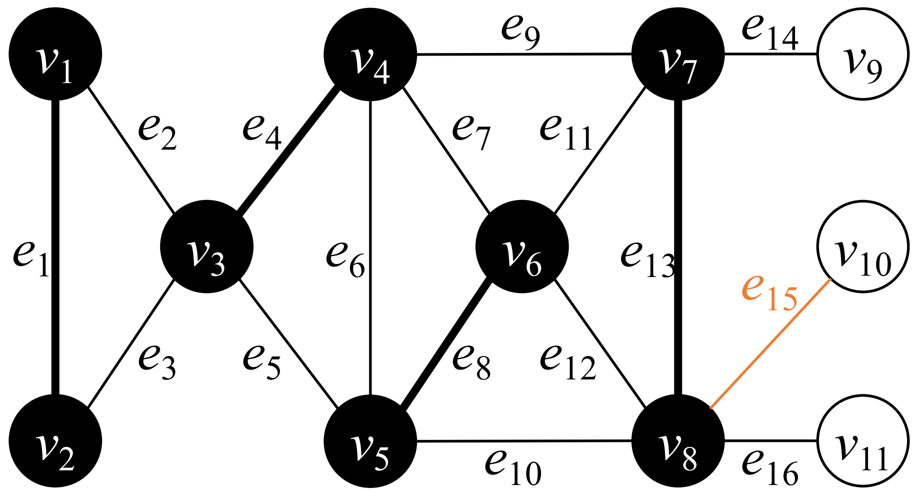
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**

    $I' \leftarrow I' \cup \{(u, v)\}$;

    $u$.saturated $\leftarrow$ true;

    $v$.saturated $\leftarrow$ true;

输出 $(I')$;



# 边的独立、覆盖和支配

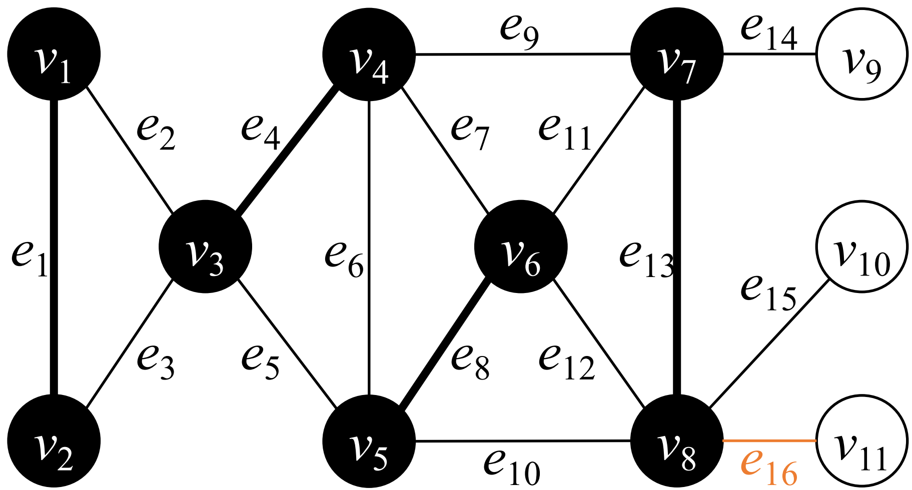
- 计算极大边独立集
  - 每轮循环尝试向$I'$中加入一条边$(u, v)$
    - 若端点$u$和$v$都未被$I'$饱和

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**
    $I' \leftarrow I' \cup \{(u, v)\}$;
    $u$.saturated $\leftarrow$ true;
    $v$.saturated $\leftarrow$ true;
输出 $(I')$;



# 边的独立、覆盖和支配

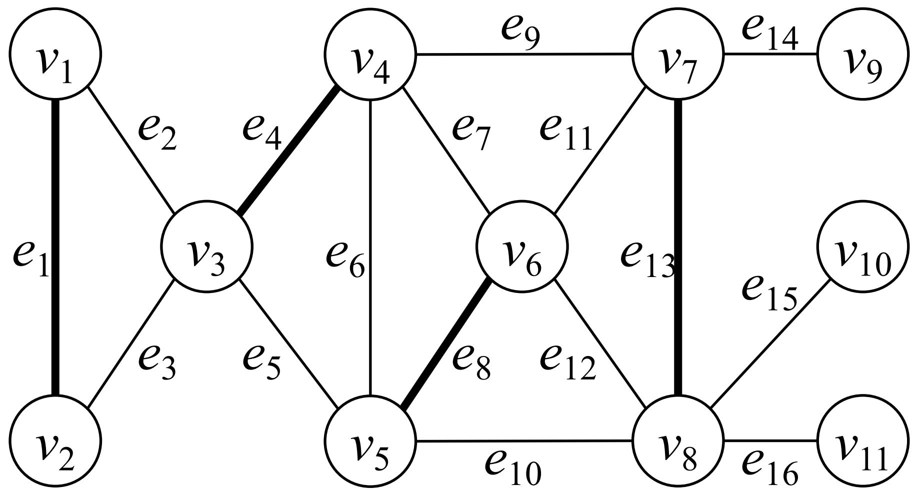
- 计算极大边独立集
  - 输出$I'$

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I′ ← I′ ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I′);
```



# 边的独立、覆盖和支配

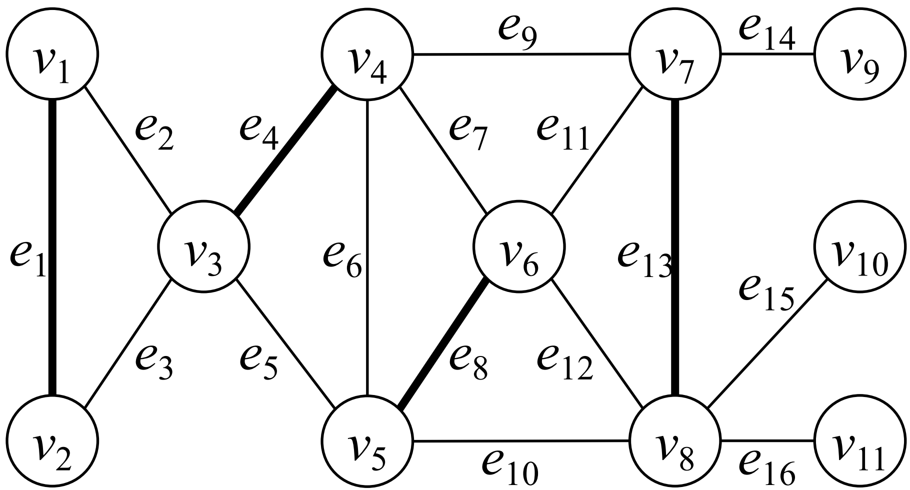
- 计算极大边独立集
  - 输出$I'$
    - $I'$为什么是边独立集？

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$; $V$ 中所有顶点的 saturated 初值为 false

```
foreach (u, v) ∈ E do
    if u.saturated = false 且 v.saturated = false then
        I′ ← I′ ∪ {(u, v)};
        u.saturated ← true;
        v.saturated ← true;
输出 (I′);
```



# 边的独立、覆盖和支配

- 计算极大边独立集
  - 输出$I'$
    - $I'$为什么是边独立集
    - $I'$为什么是极大边独立集？

**算法 14:** 计算极大边独立集的算法

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**

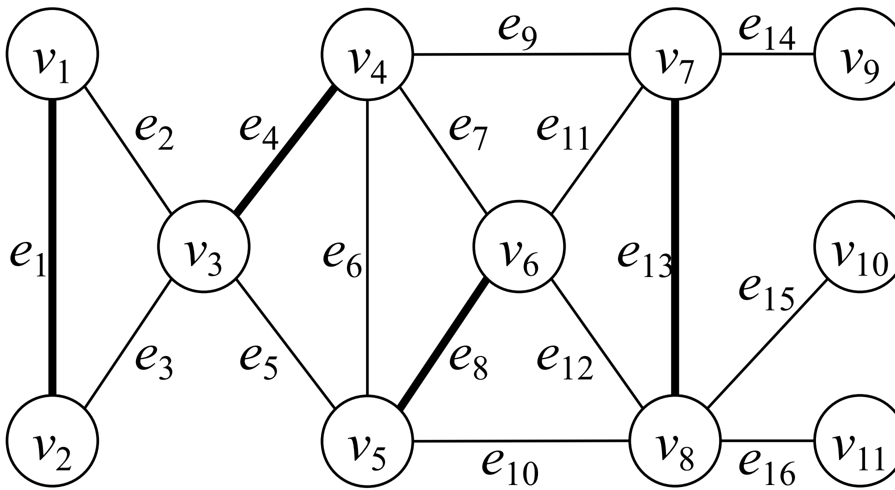
  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**

    $I' \leftarrow I' \cup \{(u, v)\}$;

    $u$.saturated $\leftarrow$ true;

    $v$.saturated $\leftarrow$ true;

输出 ($I'$);



# 边的独立、覆盖和支配

- 时间复杂度：O($n + m$)

---

**算法 14:** 计算极大边独立集的算法

---

**输入:** 图 $G = \langle V, E \rangle$

**初值:** $I'$ 初值为 $\emptyset$；$V$ 中所有顶点的 saturated 初值为 false

**foreach** $(u, v) \in E$ **do**

  **if** $u.saturated = false$ 且 $v.saturated = false$ **then**

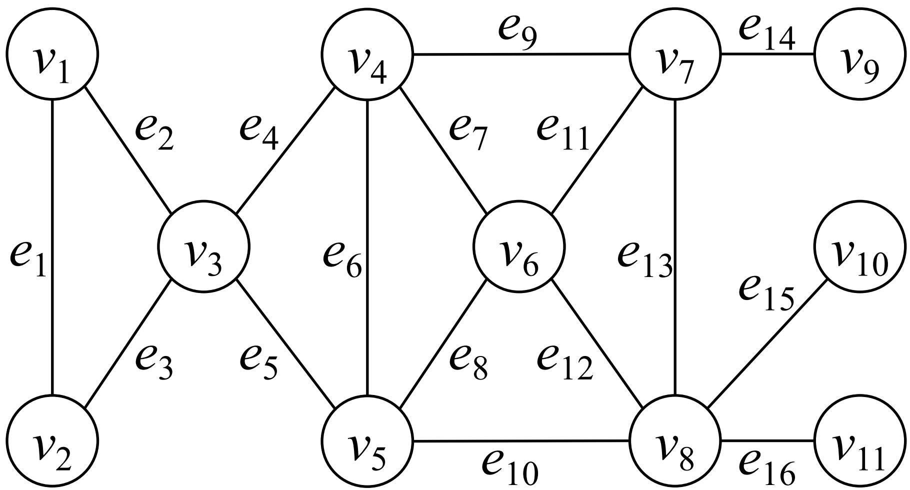
    $I' \leftarrow I' \cup \{(u, v)\}$;

    $u$.saturated $\leftarrow$ true;

    $v$.saturated $\leftarrow$ true;

输出 （$I'$）;

---



# 边的独立、覆盖和支配

- 计算最小边支配集的不可近似性
  - 极大边独立集包含的边的数量不超过最小边支配集包含的边的数量的2倍。

# 边的独立、覆盖和支配

- 计算最小边支配集的不可近似性
  - 极大边独立集包含的边的数量不超过最小边支配集包含的边的数量的2倍。
  - 在某种特定猜想成立的前提下，对于任意常数$\varepsilon > 0$，最小边支配集的计算具有$(2-\varepsilon)$-不可近似性，即不存在近似比为小于2的常数的多项式时间算法。
    - 这意味着什么？

# 边的独立、覆盖和支配

- 计算最小边支配集的不可近似性
  - 极大边独立集包含的边的数量不超过最小边支配集包含的边的数量的2倍。
  - 在某种特定猜想成立的前提下，对于任意常数$\varepsilon > 0$，最小边支配集的计算具有$(2-\varepsilon)$-不可近似性，即不存在近似比为小于2的常数的多项式时间算法。
    - 以极大边独立集作为最小边支配集的近似，在理论上似乎已经足够精确了。

# 书面作业

- 练习6.1、6.2
  （本周和下周的作业一起交）