

第九章 拟阵与组合最优化

9.2 拟阵与组合最优化问题

拟阵有一个最为基本的优化性质：极大独立集一定是最大独立集合——表示为拟阵的 Greedy 算法。

记号：

设 $M = M(E, \mathcal{I})$ 是拟阵

(1) R^E 表示定义在 E 上的全体实函数（若 $|E| = m$ ， R^E 可以看作 R^m ）。

(2) 每个映射 $c \in R^E$ 称为拟阵的一个权函数。

(3) 对于子集 $X \subseteq E$ ，记 $c(X) = \sum_{e \in X} c(e)$ ，约定 $c(\emptyset) = 0$ 。

(4) 对于每个子集 $J \subseteq E$ ，令 x_J 为 J 的特征函数，即

$$x_J(E) = \begin{cases} 1, e \in J \\ 0, e \notin J \end{cases}$$

显然 $c(J) = c^T x_J$ 。

(5) 若 $x = (x_1, x_2, \dots, x_n)^T, y = (y_1, y_2, \dots, y_n)^T$ ，且 $x_i \leq y_i (i = 1, 2, \dots, n)$ ，则记

为 $x \leq y$ 。

一、Greedy 算法和最大权独立集问题

定义 9.6 对于给定权函数 c 的拟阵 M ，拟阵的最大权独立集问题是：求一个确定的子集 $J \subseteq E(M)$ ，使得 $J \in \mathcal{I}(M)$ 并且 $c(J) = \max\{c(J) : J \in \mathcal{I}(M)\}$ 。

拟阵最大权独立集的性质启发了下面的 Greedy 算法（Greedy Algorithm）。

拟阵最大权独立集的 Greedy 算法：

给定 E 上的拟阵 M 和 E 上的实值函数 c ，以及一个查询时间为 $O(|E|)$ 的权独立集查询子程序，Greedy 算法步骤如下：

(i) 对于 $E(M)$ 中的元素重新标号，使得 $E(M) = \{e_1, e_2, \dots, e_m\}$ 满足

$$c(e_1) \geq c(e_2) \geq \dots \geq c(e_m)$$

(ii) 置 $J := \emptyset$

(iii) 对 $i = 1$ 到 m ，若 $J \cup \{e_i\} \in \mathcal{I}(M)$ ，则置 $J := J \cup \{e_i\}$

定理 9.16 拟阵最大权独立集的 Greedy 算法得到拟阵最大权独立集问题的最

有解。若对任意的 $x, y \in E(M)$ ，当 $x \neq y$ 时总有 $c(x) \neq c(y)$ ，则算法得到问题的唯一最优解。
证明：反证法。

例 9.10: MWF (最大森林问题)

$G = (V, E), \forall e \in E, w(e) \geq 0$ 。

问题：求 G 的权重最大的一个森林 (G 的无圈图)。

子集系统： $M = (E, \mathcal{F})$ 是一个拟阵

其中 \mathcal{F} : E 中所有的森林——无圈图的集合

组合优化问题：求 \mathcal{F} 中权重最大者

Greedy 算法能够得到问题的最优解

例 9.11: 设 E 是一个有限集合， $\forall e \in E$ ，定义 $w(e) \geq 0$ Π 是 E 的一个划分，即 $\Pi = \{E_1, E_2, \dots, E_p\}$ 为 E 的不相交子集的集合，并且覆盖 E 。定义子集系统 $M_\Pi = (E, \mathcal{I})$

其中 E 的一个子集 I 是独立的 ($I \in \mathcal{I}$) $\Leftrightarrow I$ 中任何两个元素不在 Π 的同一个集合里，即 $|I \cap E_j| \leq 1 (j = 1, 2, \dots, p)$ 。

子集系统 $M_\Pi = (E, \mathcal{I})$ 是一个拟阵——称为划分拟阵

组合优化问题：求 \mathcal{I} 中权重最大者

Greedy 算法能够得到问题的最优解

例 9.12: $G = (V, A)$ ， $\forall a \in A$ ，定义 $w(a) \geq 0$ 。

问题：找出 A 的一个最大权的子集 B ，使得 B 任何两条弧都没有公共终点。

子集系统： (A, \mathcal{B}) 是一个拟阵

其中 $B \subseteq A$ ， $B \in \mathcal{B} \Leftrightarrow B$ 中任何两条弧都没有公共终点

组合优化问题：求 \mathcal{B} 中权重最大者

Greedy 算法能够得到问题的最优解

注: $B \subseteq A$ ， $B \in \mathcal{B} \Leftrightarrow B$ 中任何两条弧都没有公共终点，等价于把弧集合 A 进行划分，使得指向同一个节点 v_j 的弧之集合定义为 E_j 。因此，这个子集系统

是一个划分拟阵 M_Π ，并称其为有向图 $G = (V, A)$ 的入弧划分矩阵。当然也可以等定义有向图 $G = (V, A)$ 的出弧划分矩阵。

例 9.13: $G = (V, E), \forall e \in E, w(e) \geq 0$

子集系统: $M = (E, \mathcal{I})$

其中 $\mathcal{I}: I \subseteq E, I \in \mathcal{I} \Leftrightarrow I$ 是一些不交路之并集

组合优化问题: 求 \mathcal{I} 中权重最大者

关于子集系统 (E, \mathcal{I}) 的上述组合优化问题，是一个变了形的 TSP 问题，TSP 是一个环游，而此处为路，与 TSP 问题等价
它不是一个拟阵
Greedy 算法不能得到问题的最优解

二、两个拟阵的交

二部图的匹配问题

$B = (V, U, E), \forall e \in E, w(e) \geq 0$

问题: 求 B 的最大匹配

$M = (E, \mathcal{M})$

$\mathcal{M}: E$ 中所有的匹配 ($E = \{e_1, e_2, \dots, e_n\}$ 的一些子集组成的集合)

组合优化问题: 求 \mathcal{I} 中权重最大者

算法: 不能用 Greedy 算法

所以, $M = (E, \mathcal{M})$ 不是一个拟阵。但是, \mathcal{M} 有比较好的性质, 它是两个拟阵的交。即:

\mathcal{M} 是两个拟阵 $M = (E, \mathcal{F})$ 和 $N = (E, \mathcal{K})$ 的公共独立子集的集合, 即 $\mathcal{M} = \mathcal{F} \cap \mathcal{K}$, 其中 M 和 N 都是划分拟阵。

M 是 $B_1 = (V, E)$ 的划分拟阵 $M_{\Pi_V} = (E, \mathcal{F}_V)$, 其中 $E_{v_j} \in \Pi_V \Leftrightarrow E_{v_j}$ 是 V 中第 v_j 个节点所关联的所有边的集合;

N 是 $B_2 = (U, E)$ 的划分拟阵 $M_{\Pi_U} = (E, \mathcal{F}_U)$, 其中 $E_{u_i} \in \Pi_U \Leftrightarrow E_{u_i}$ 是 U 中第 u_i 个节点所关联的所有边的集合;

所以：

二部图的匹配问题的子集系统 $M = (E, \mathcal{M})$ 满足：

$I \subseteq E$ 是一个匹配 ($I \in \mathcal{M}$) $\Leftrightarrow I$ 在拟阵 $M_{\Pi_V} = (E, \mathcal{F}_V)$ 中是独立集 ($I \in \mathcal{F}_V$, 即 I 中任何两条边在 V 中没有公共端点) 且 I 在拟阵 $M_{\Pi_U} = (E, \mathcal{F}_U)$ 中是独立集 ($I \in \mathcal{F}_U$, 即 I 中任何两条边在 U 中没有公共端点)。

所以： $\mathcal{M} = \mathcal{F}_V \cap \mathcal{F}_U$

所以，二部图的匹配问题可以变为求两个拟阵的最大公共独立子集问题。

一般的，我们假设 $M_1 = (E, \mathcal{F}_1)$ 和 $M_2 = (E, \mathcal{F}_2)$ 是定义在同一个集合上的两个拟阵，且 $r_1 = r_{M_1}$ ， $r_2 = r_{M_2}$ 分别表示 M_1 和 M_2 的秩函数。

定义 9.7 给出定义在同一个集合上的两个拟阵 $M_1 = (E, \mathcal{F}_1)$ 和 $M_2 = (E, \mathcal{F}_2)$ ，最大公共独立集问题是找出一个 E 的元素最多的子集 X ，满足 $X \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)$ 。

下面只讨论不带权拟阵的最大公共独立集问题。

定理 9.16 设 $J \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)$ ，则

(i) 对任意的 $X \subseteq E$ ， $|J| \leq r_1(X) + r_2(E - X)$ 。

(ii) 若有 $X \subseteq E$ ，使得 $|J| = r_1(X) + r_2(E - X)$ ，则 J 必是一个最大公共独立集。

证明：

(i) 由于 $J = (J \cap X) \cup (J \cap (E - X))$ ，根据秩函数的定义，

$|J| = |J \cap X| + |J \cap (E - X)| \leq r_1(X) + r_2(E - X)$ 。

(ii) 设 J' 是一个最大公共独立集，则有 $|J| \leq |J'| \leq r_1(X) + r_2(E - X) = |J|$ ，从而 $|J| = |J'|$ ，即 J 也是一个最大公共独立集。

问题：寻找最大公共独立集是否等价于寻找满足定理 9.16 中(ii)的子集 J 和 X ？下面的著名定理回答了这个问题。

定理 9.17 (Edmonds) 对 E 上的两个拟阵 $M_1 = (E, \mathcal{F}_1)$ 和 $M_2 = (E, \mathcal{F}_2)$ ，定义 $k_1 = \max\{|J| : J \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)\}$ 和 $k_2 = \min\{r_1(X) + r_2(E - X) : X \subseteq E\}$ ，则恒

有 $k_1 = k_2$ 。

上述定理满足定理 9.16 中(ii)的子集 J 和 X 的存在性，但不能从上述定理的证明中把 J 构造出来。因此，需要讨论拟阵交算法(Matroid Intersection Algorithm: MIA)。

二部图的匹配问题是两个拟阵交的特殊情况，求两个拟阵 $M = (E, \mathcal{F})$ 和 $N = (E, \mathcal{K})$ 的最大公共独立子集的算法是基于二部图的匹配的增广路算法。

利用增广路逐步增大匹配等价于逐步增大在两个拟阵 $M = (E, \mathcal{F})$ 和 $N = (E, \mathcal{K})$ 中均独立的集合。

所以，先复习一下二部图的匹配的增广路算法。

因此，对于公共独立集 $J \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)$ （即对应于二部图的一个匹配），引进辅助有向图 $G(J)$ 。

定义 9.8 对子集 $J \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)$ ，定义辅助有向图 $G(J)$ 如下：图 G 的节点集合为 $E \cup \{s, t\}$ ，其中 s 和 t 是两个在 E 之外的元素； $G(J)$ 的边集则由下列有向边组成

- (i) 对每个 $e \in E - J$ ，若 $J \cup e \in \mathcal{F}(M_1)$ ，则 (e, t) 为 $G(J)$ 中的有向边；
- (ii) 对每个 $e \in E - J$ ，若 $J \cup e \in \mathcal{F}(M_2)$ ，则 (s, e) 为 $G(J)$ 中的有向边；
- (iii) 对每个 $e \in E - J$ 和 $f \in J$ ，若 $(J \cup e) - f \in \mathcal{F}(M_1)$ ，则 (e, f) 为 $G(J)$ 中的有向边；
- (iv) 对每个 $e \in E - J$ 和 $f \in J$ ，若 $(J \cup e) - f \in \mathcal{F}(M_2)$ ，则 (f, e) 为 $G(J)$ 中的有向边；

定理 9.17 给定 $J \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)$ ，令 $G = G(J)$ ，则

- (i) 若 G 中有一条 $s-t$ 有向路 $P = \{s, e_1, f_1, \dots, e_k, f_k, e_{k+1}, t\}$ ，则

$$J' = (J \cup \{e_1, e_2, \dots, e_{k+1}\}) - \{f_1, f_2, \dots, f_k\} \in \mathcal{F}(M_1) \cap \mathcal{F}(M_2)。$$

- (ii) 若有子集 $X \subseteq E$ ，使得在 G ，没有以节点 u 为尾，以节点 v 为头的有向

边 (u, v) ，满足 $u \in X \cup s$ 和 $v \in (E - X) \cup t$ （即 G 中不存在任何 $s-t$ 有

向路), 则 $|J| = r_1(X) + r_2(E - X)$ 。

算法 (拟阵交算法: MIA)

第 1 步: 置 $J = \emptyset$;

第 2 步: 构造辅助有向图 $G = G(J)$;

第 3 步: 在辅助有向图 $G = G(J)$ 搜索 $s-t$ 有向路:

$$P = \{s, e_1, f_1, \dots, e_k, f_k, e_{k+1}, t\}$$

若不存在 $s-t$ 有向路, 则 J 为最大公共独立集, 算法停止; 否则转第 4 步;

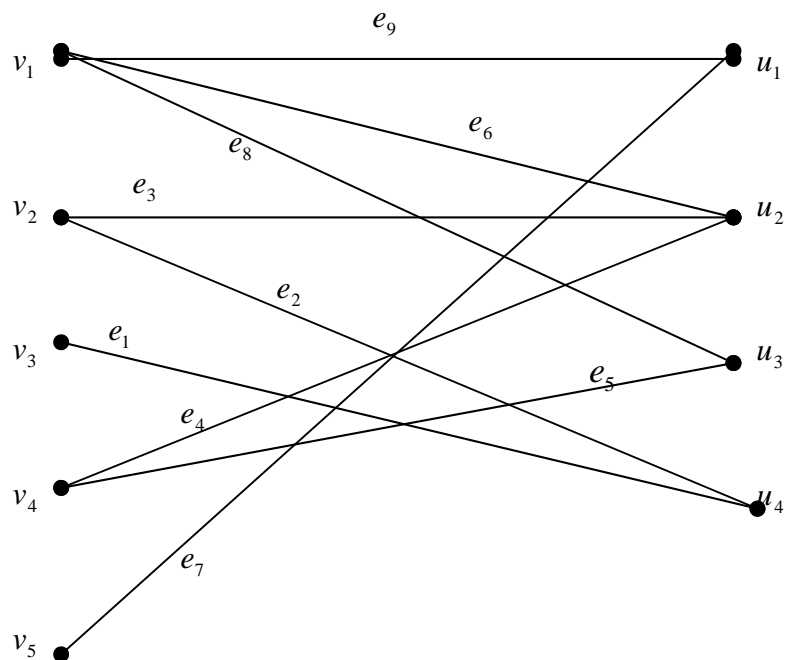
第 4 步: 置 $J := (J \cup \{e_1, e_2, \dots, e_{k+1}\}) - \{f_1, f_2, \dots, f_k\}$, 转第 2 步。

定理 9.18 上述拟阵交算法是正确的, 并且是多项式算法。

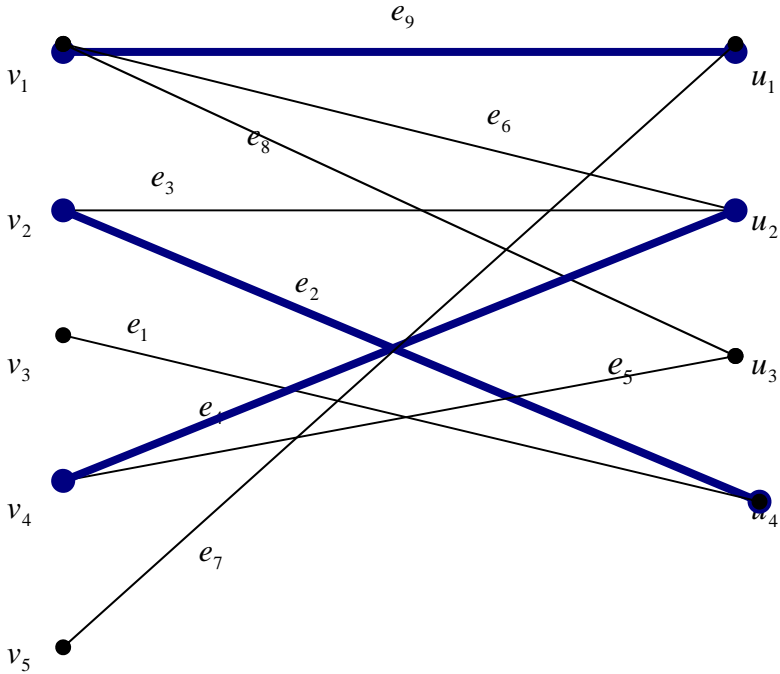
证明: 正确性由定理 9.17 直接可得。

利用二部图的增广路算法设计算法:

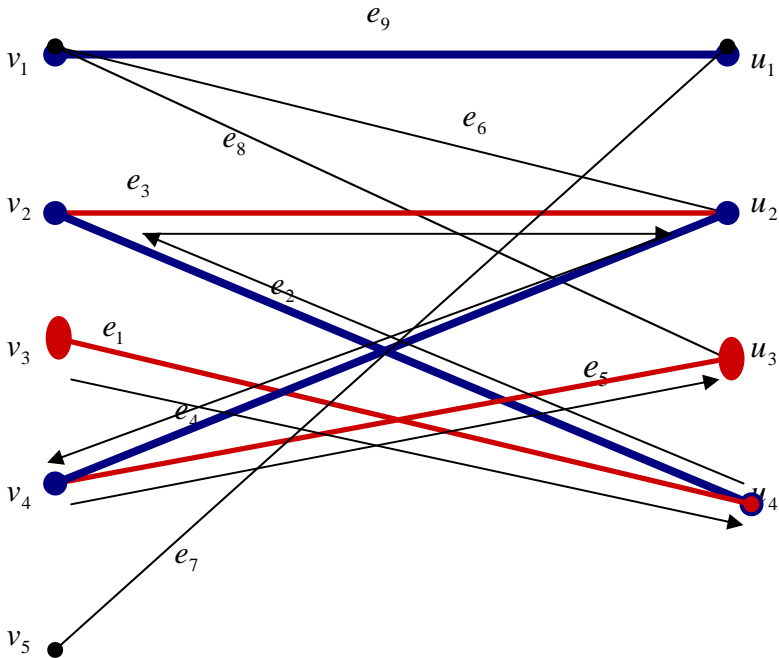
例 9.14: 求下列二部图的最大匹配



(e_9, e_2, e_4) 是一个极大匹配。



从未盖点 e_1 开始， $P = [e_1, e_2, e_3, e_4, e_5]$ 是关于匹配 (e_9, e_2, e_4) 的增广路。



$$I \Rightarrow I + e_1 - e_2 + e_3 - e_4 + e_5 \Rightarrow I \oplus P$$

因为：从 V 中的未盖点 e_1 开始，所以： $I + e_1$ 是 M 的独立集（从 v_3 出发只

有一条边)

但是: $I+e_1$ 在 N 中不是独立集 (从 u_4 出发有两条边 e_1 和 e_2 , 否则 e_1 就是一条增广路),

所以: $I+e_1$ 在 N 中有一个圈 $\{e_1, e_2\}$ (有公共端点 u_4)

所以: 去掉 e_2 , 从而 $I+e_1-e_2$ 在 M 中和 N 中都是独立的, 但它和 I 的基数相等, 再加上 e_3 ,,

我们希望得到一个元素 e_k , 使得 $I+e_1-e_2+e_3-e_4+\cdots+e_k$ 在 M 中和 N 中都是独立的, 且比 $|I|$ 多一个。

二、拟阵交问题的某些推广

1. 两个带权拟阵的交问题

更一般地, 还可以考虑两个定义在同一个集合上的两个带权拟阵 $M_1 = (E, \mathcal{I}_1)$ 和 $M_2 = (E, \mathcal{I}_2)$, 权函数分别为 c_1 和 c_2 。带权拟阵最大公共独立集问题是找出一个 E 的子集 $X \in \mathcal{I}(M_1) \cap \mathcal{I}(M_2)$, 使得 $c_1(X) + c_2(X)$ 最大。

类似于赋权二部图的匹配, 可以设计多项式算法。

2. 三个拟阵的交问题

三个拟阵的最大公共独立集问题——NP 完备问题。