对拟阵的初步研究

浙江省杭州第二中学 刘雨辰

摘要

拟阵中文又称矩阵胚，英文名 matroid。1935 年美国数学家 Whitney 首先提出了拟阵的概念。拟阵是组合优化与图论的重要内容，在近几十年得到了空前的发展，成为了一门博大精深的学科。

本文对拟阵进行了初步探讨，第一部分引入了拟阵的概念，第二部分提出了拟阵的最优化问题，并论证了其贪心算法的正确性，这两部分都将同步讲解 2 个实例，力求做到严谨而生动。第三部分讨论了一个拟阵最优化问题的实例。这三部分是本文的重点所在。第四部分给出一些拟阵的实例，重点是线性拟阵。拓展部分对一个有趣问题进行了简单讨论，有相当难度。附录 I 介绍了罗素悖论，附录 II 主要讨论第三部分中的问题如何用并查集实现。

拟阵理论非常难，需要有良好的数学功底才能深入研究。希望本文对拟阵的初步探讨能够使读者对拟阵理论有所认识。

关键字

拟阵 贪心算法 Shannon 开关游戏

序言

拟阵中文又称矩阵胚，英文名 matroid。1935 年美国数学家 Whitney 首先提出了拟阵的概念。拟阵是他在研究线性无关时发现的。拟阵是组合优化与图论的重要内容，在近几十年得到了空前的发展，成为了一门博大精深的学科。

本文第一部分讨论了拟阵的的基本概念，第二部分提出了拟阵的最优化问题，并论证了其贪心算法的正确性，这两部分都将同步讲解 2 个实例，分别是众所周知的部分背包问题（即物品是不用整个拿，可以分割）与最小生成树问题，希望理论结合实例能够帮助读者更好地理解。第三部分讨论了一个拟阵最优化问题的实例。前三部分是本文的重点所在，难度适中，只要读者能够仔细读，慢慢读，反复体会，必能发现其奥妙所在。本文的第四部分和拓展部分比较难，其中有一些结论与证明，我自己也尚未搞明白，但因为这些结论都很有趣，我还是把它们写入了本文，这两部分主要为了拓宽读者视野，对拟阵论的博大精深有个更全面的认识。

理论和实践是不分家的，但历史上却有很多理论与应用不协调的例子：有的是为了实际用途，而发现了一些新方法，但理论根据却不很清楚，过了很久才被完善，并在完善后得到了空前的发展，比如著名的微积分；有的是先有了理论研究，但一直只作为一种智力游戏，并无实际用途，但却在某样事物发明以后，大显神威，于是成为了应用的理论根据并得到空前的发展，比如著名的数论、组合数学。我们发现这些例子仅仅是时间上的一种暂时分离，而并不是说理论是没有用的，或实践不需要理论，只是两者被发现的早晚而已，而当两者都被发现以后，相辅相成，却能走得更快更好更远。如果我们认为拟阵没什么实际用途，应该不是因为应用还没被发现，只是没被我们发现，因为我们的知识还不够，我们不知道，仅此而已。拟阵是好的、美的，所以我要把它介绍给大家。

拟阵论很难，需要非常良好的数学基础才能学好，所以一般作为研究生的课程。我在研究过程中深感自己水平有限，诸如线性代数，拓扑学等数学基础知识都不懂，无奈只得边研究边学习。当然这样临时性的学习必然只能获得皮毛，但我还是在这种研究性学习中获得了无限乐趣。“吾生也有涯，其知也无涯”。知识是无穷尽的，我们要做的就是永远保持对知识的渴望，对真理的追求。这样也就称的上一个爱智慧的人了。

正文

第一部分：子集系统与拟阵的概念

定义：

子集系统是一个二元组 $M=(S,L)$，它须满足以下条件：

1、S 是一个有限集。

2、L 是由 S 的一些子集组成的有限非空集

3、遗传性：对任意 $B\in L$，任意 $A\subseteq B$,有 $A\in L$（可知 ϕ 必须是 L 的元素）。

拟阵是一个子集系统，它须满足：

4、交换性：对任意 $A\in L, B\in L, |A|<|B|$,存在一个 $x\in B-A$,使 $A\cup\{x\}\in L$。

对该定义做些解释是有益处的。我们把 S 看成一个班级所有的同学，L 看成若干张名单的集合。一张名单记录了若干个同学，表示他们能够组成一个团队。遗传性说明，从一张名单中选出若干个同学组成的子名单仍存在于我们的名单集合，这是一种包容性。交换性说明，名单 A 上的人数如果少于名单 B，则必然可以从 B 中选出一个人，该人不属于 A，将该人加到 A 里形成的新名单仍然存在于我们的名单集合中，要注意，我们加到 A 中的人必须本来不是 A 的，并且必须是 B 的人，而不能是 2 张名单以外的人，外人不能干预。当然你也可以将 S 想象成别的事物组成的集合，而 L 则是将这些事物的组合看成基本元素的集

合。由于是集合，万物皆可做元素（这句话严格地说是有问题的，比如著名的罗素悖论,详见附录I，任何数学体系都有其局限性与不完备性,因此需要公理化。在此只想说明拟阵是个抽象概念，而不是一个具体问题）。

遗传性和交换性是拟阵最根本的2条性质，拟阵上的其他性质都是基于这2个性质发展出来的。古人云："君子务本，本立而道生。"这2条性质就是拟阵的本。拟阵是一种组合结构。研究组合结构的意义正如研究代数结构，一旦我们发现某个问题具有拟阵结构（符合4个基本条件，前2条比较显然，因为构造时就已完成，所以后面2个本的成立是关键），则我们已研究出的有关拟阵的性质都可以直接应用于该问题，而不需重新证明。

我们来同步地看我们的两个实例：

对于背包问题，我们首先进行一步转化：将体积为V，单位价值为W的物品变成V个体积为1，价值为W的物品。这样物品就全是单位体积的了。我们可以定义这样一个$M=(S,L)$：

1、 S是所有物品的集合

2、 $L=\{x:x\subseteq S,|x|\le MAX\}$

这个M显然满足子集系统的前两条件。根据L的定义，对任意$x\in L$，任意$y\subseteq x$，满足$|y|\le|x|\le MAX$，有$y\in L$,因此M满足遗传性,是一个子集系统。对任意$A\in L,B\in L$，$|A|<|B|$，随意选取一个$x\in B-A$,令$C=A\cup x$,显然有$|C|=|A|+1\le|B|$，所以M满足交换性。因此M也是一个拟阵。我们称M为背包问题的拟阵。

最小生成树问题是在无向图上进行的。因此考虑对于无向图$G=(V,E)$，我们可以定义这样一个$M=(S,L)$：

1、 S是边集E

2、 $L=\{x:x\subseteq E$且x组成的图无环$\}$

这个M显然满足子集系统的前两条件。根据L的定义，对任意$x\in L$，任意$y\subseteq x$，假设y形成环，则x形成环，矛盾，所以y不形成环，所以$y\in L$,因此M满足遗传性。因此M是一个子集系统。接下来证明M满足交换性：考虑任意$A\in L,B\in L,|A|<|B|$，我们将A组成的森林命名为G_A,B组成的森林命名为G_B。G_A有$|V|-|A|$个连通分量，G_B有$|V|-|B|$个连通分量。（这是根据一个基本事实：n个点k条边的森林有$n-k$个连通分量）。$|A|<|B|$，所以$|V|-|B|<|V|-|A|$，所以G_B中存在一个连通分量T，T中的点在G_A中不连通（如果对G_B中的每个连通分量的点在G_A中也连通，则$|V|-|B|\ge|V|-|A|$）。那么T中必然存在一条边x连接G_A中不同连通分量的边，显然$x\notin A$且$x\in B$，即$x\in B-A$，且$A\cup\{x\}$无环，即$A\cup\{x\}\in L$。所以M满足交换性。因此M是一个拟阵。我们称M为无向图$G=(V,E)$的拟阵。

为了方便以后的讨论，我们以一些命名来结束本部分：

对于$U\subseteq S$，如果$U\in L$，那么称U为独立集。

对于独立集A,若存在$x\in S$,满足$x\notin A$且$A\cup x\in L$，则称A为可扩展的。不可

扩展的独立集称为极大独立集。对于 $U \subseteq S$，我们定义：$r(u) = \max\{|x| : x \subseteq U, 且x是独立集\}$，我们称 r(U)为 U 的秩，r 称为拟阵的秩函数。

定理 1：拟阵的极大独立集大小相同

证明：假设 A 和 B 是拟阵的 2 个极大独立集，如果 $|A| \neq |B|$,不失一般性，我们可以设 $|A| < |B|$，那么根据交换性，A 是可扩展的，不是极大独立集，矛盾。所以命题成立。证毕。

第二部分：拟阵上的最优化问题

上一部分我们给出了拟阵的定义，并做了一些有益的解释，同步给出了 2 个实例。

这部分我们将提出拟阵上的最优化问题，并论证可以用贪心法解决。我们一旦发现了最优化问题的拟阵结构，那么就可以对该问题实施贪心算法，这是由拟阵的性质决定的。

当然仍有大量可以用贪心算法解决的问题无法被拟阵结构覆盖（比如 huffman 问题，最多不冲突区间问题等），但拟阵理论博大精深，这只是拟阵最简单最直接的一个应用，只是冰山一角。况且，爱美之心，人皆有之，细心体会你会发觉拟阵是种很优美的结构。

对于拟阵 $M = (S, L)$，我们对 S 的每个元素 x 赋予一个正整数权值 $w(x)$，S 的任意子集 U 的权值 $w(U) = \sum_{x \in U} w(x)$，即为其所有元素的权值和。对于 M，我们希望求出它的一个权值最大独立集。现在我们在名单上写下分数，分数为名单上所有人的分数和。我们的目标是找出分数最高的一张名单。

继续同步地来看我们的两个例子：

背包问题是希望求出一种方案使得带走的物品价值和最大。那么我们上面提到了我们的 S 是所有物品的集合（物品是单位体积的），我们将 S 的任意元素 x 的权值 $w(x)$ 定义为 x 的价值（价值总为正的），那么问题就转化为了求背包问题的拟阵的权值最大独立集。显然权值最大的独立集是极大独立集。

最小生成树问题乍看有一些困难，因为我们的目标是使权值最小，如何将一个权值最小问题转化为一个权值最大的问题呢？一般方法就是对权值取负，但对权值取负以后会出现负数，不满足我们拟阵权值的前提条件，怎么办？由于权值只是个相对概念，我们可以改变我们的绝对零点，使得所有权值变正。也就是说，将所有权值先取负，然后再统一加上一个足够大的值，使得权值都是正的，那么问题就等价转化为在新的权值下求一棵最大生成树了。即对于 S 的每个元素 x(x 是图的一条边)，使 $w(x) = -g(x) + delta$，其中 $g(x)$ 为 x 的边权，$delta$ =图中边权最大值 +1。显然 $w(x) > 0$。这样，求最小生成树的问题就等价为求图的拟阵的权值最大独立集了。

我们先给出求拟阵最大权值独立集的贪心算法：这个算法的输入是一个加权拟阵 $M = (S, L)$ 和一个正权函数 w，返回的是一个最大权值独立集 A。在我们的伪代码中，用 $S[M]$ 和 $L[M]$ 表示 M 的组成，用 w 表示权函数。算法的基本思想是按权值的非增序来依

次考虑每个元素 $x \in S$,如果 $A \cup \{x\}$ 是独立集，就立刻把 x 加入 A。

伪代码：

```
Greedy(M,w)
    A := 空集;
    根据 w 按非增长顺序对 S[M] 排序
    for 每个 x∈S[M] ,根据权 w(x) 的非增长顺序 do
        if ( A∪{x} ∈L[M] ) then A:=A∪{x} ;
    return A;
```

同步看我们的两个例子：按该算法做背包问题，相当于每次取价值最大的物品，如果背包还没满（说明 $A \cup \{x\} \in L[M]$ ），那么该物品就放入背包了。按该算法做最小生成树问题，相当于每次取权值最小的边，如果该边加入 A 不形成环（说明 $A \cup \{x\} \in L[M]$ ），则将该边加入。我们发觉两个算法只在判断是否为独立集这一步是不同的。

该算法的时间复杂度是很容易分析的。设 n 表示 $|S|$ 。排序阶段的时间复杂度为 $\Theta(n \log n)$ ，贪心阶段要进行 $\Theta(n)$ 次判断是否为独立集的操作，若判断是否为独立集的时间复杂度为 $\Theta(f(n))$ ，那么总的时间复杂度就为 $\Theta(n \log n + n \times f(n))$ 。我们发现该算法的时间复杂度的瓶颈在判断是否为独立集。

其实这个算法的关键也就在于判断是否为独立集这一步，因为该算法虽然是针对拟阵的算法，但可以归结为拟阵的实际问题是各不相同的，而拟阵体现了它们的一种共性，正是由于这种共性，它们才能够使用这个贪心算法，而它们的不同之处也就集中体现在判定独立集这一步了。

我们接下来讨论这个算法的正确性。

我们只需证明在算法的每一步 A 都是某个最优解的子集，那么当算法结束时 A 就为一个最优解（因为无法扩展了）。

初始时，A 为空集，显然是任何集合的子集，即是某个最优解 T 的子集。我们只需证明下面命题：

定理 2：对于拟阵 $M = (S, L)$ ，A 是 L 的一个元素，且 A 是该拟阵的某个权值最大独立集 T 的子集，x 满足 $A \cup \{x\} \in L$,且没有其他满足 $A \cup \{y\} \in L$ 的 y 使得 $w(y) > w(x)$ ，即 x 是能使 A 扩展的元素中的权值最大的。令 $A' = A \cup \{x\}$ 。那么存在一个包含 A' 的最优解

证明：我们用证明贪心的常用方法，反证。我们假设 A' 不是任何最优解的子集，我们来构造一个包含 A' 的独立集 T' ，并说明这个 T' 不会比 T 差，即 T' 也是最优解，以推出矛盾。我们这样得到 T' ：首先令 $T' = A'$ ，当|T'|<|T|时，根据交换性，必然可以找到一个属于 T 但不属于 T' 的元素 z,使得 $T' \cup \{z\} \in L$,于是就令 $T' = T \cup \{z\}$,直到 $|T'| = |T|$,显然 $T' \in L$ 。此时 $T' = (T - \{y\}) \cup \{x\}$,其中 y 为 $T - A$ 中未被加入 T' 的元素，所以

$w(y) \le w(x)$,所以 $w(T') = w(T) - w(y) + w(x) \ge w(T)$。如果 $w(T') > w(T)$,则 T 不是最优解，矛盾。如果 $w(T') = w(T)$,即也是最优解，且 A' 是 T' 的子集，矛盾。所以存在一个最优解使得 A' 是其子集。证毕。

我们发现以上证明中 T' 的构造是关键，而构造 T' 的过程中交换性起了关键作用。因此子集系统上是无法贪心的，只有给子集系统赋予交换性，才能派上用处，这就是拟阵。

可以将该证明和最小生成树算法的正确性证明做一个比较，可以发现最小生成树证明中也是采用了构造一个 T' 的方法，构造方法是：把边 x 加到 T 里，那么 T 中必然形成一个环，这个环上必然有一条边权值大于 x,去掉以后得到生成树 T'。这与我们拟阵的 T' 的构造方法是本质相同的。拟阵的 T' 的构造方法是最小生成树的构造方法的抽象形式，而最小生成树的构造方法是拟阵的构造方法的具体表现之一。

第三部分：一个任务调度问题

上一部分介绍了拟阵结构的最优化问题及其贪心算法，这一部分我们介绍一个实际应用。我们可以发现洞察问题中隐藏着的拟阵结构是关键。

一个有趣的问题：我们要在单个处理器上对若干个单位时间任务进行最优调度，其中每个任务都有一个截止时刻和超时的罚款。

单位时间任务（如要在计算机上运行的一个程序）恰需要一个单位的运行时间。给定一个单位时间任务的集合 S，S 有 n 个任务，我们将任务标号为 1..n。对 S 的一个调度即 S 的一个排列，它规定了各任务执行的顺序。该调度第一个任务开始于时刻 0，结束于时刻 1；第二个任务始于时刻 1，结束于时刻 2，以此类推。现在给出：

1、 n 个整数 $d_1, d_2, \ldots, d_n$ $(1 \le d_i \le n)$, d_i 表示第 i 个任务的截止时刻

2、 n 个正整数 $w_1, w_2, \ldots, w_n$, w_i 表示第 i 个任务的罚款

对于一个调度,如果任务 i 的结束时刻大于 d_i，则要交付 w_i 的罚款。我们希望求一个调度，使得罚款最少。

这题乍看之下似乎不能贪心，因为有罚款和截止时刻 2 个衡量标准，应该以哪个为标准贪心呢？似乎都不太可以。我们不妨先来分析一下，得出一些有益的结论。

对于任何一个调度，我们都可以将任务分成两类：完成的；未完成的。我们只需关注我们要完成哪些任务，能否做到，无法完成的任务可以全部放到最后做。于是我们就先考虑这么一个问题：对于给定集合 A（A 是 S 的子集），是否存在调度方案使 A 中的任务都被完成。这个问题可以贪心解决，将 A 按任务的截止时刻从小到大排序作为调度方案，即截止时刻越早的任务越先处理，如果按这个调度无法全部完成 A 的任务，则其他任意调度方案都无法完成。证明很容易，简单地说，只要有其他可完成的调度方案，就可以调整为

这种截至时刻从小到大的方案。定理 3 中给出了另一种证明。这种证明技巧很常用。

现在已经解决了这么一个问题：对于给定集合 A（A 是 S 的子集），是否存在调度方案使 A 中的任务都被完成。也就是说对于给定的任务集合 A，我们能够有效地判断这些任务能否全部完成。我们把能全部完成的任务集合 A 称为可行的。

为了方便后面的讨论，我们定义一个函数 $F(A,t)$,其中 A 是一个任务集合，且是可行的，t 是一个整数，$F(A,t)$ 的值是 A 中截止时刻不大于 t 的任务数。

为了更好地论述这个问题，我们看一个定理。

定理 3：以下 3 个命题是等价的：

1、 集合 A 是可行

2、 对所有 t=1,2,…,n,有 $F(A,t) \le t$

3、 按截止时刻从小到大排列的调度方案可以完成 A

证明：

我们先证明命题 1 推出命题 2：集合 A 是可行的。我们假设存在 t($1 \le t \le n$，t 是整数)，使得 $F(A,t) > t$,则至少要到时刻 $F(A,t)$ 才能完成所有 A 中截止时刻不超过 t 的任务，则在时刻 $F(A,t)$ 完成的那个任务已经超时，不可行，矛盾。

再证明命题 2 推出命题 3：在按时间从小到大排列的调度方案中，截止时刻为 t 的任务的完成时刻 $t' \le F(A,t) \le t$，所以 A 中所有任务都被完成。

命题 3 推出命题 1 是显然的。

我们现在对该问题定义 $M=(S,L)$：

1、 S 就是所有任务的集合，它显然是个有限集

2、 $L=\{x : x是S的子集且x是可行的\}$

3、 遗传性：对任意 $B \in L$，显然 B 的子集仍然是可行的，即 B 的任意子集 $A \in L$

4、 交换性：对任意独立集 A，B，$|A| < |B|$，设 k 为使 $F(B,k) \le F(A,k)$ 成立的最大的 k，即对 $k+1 \le j \le n$ 中的所有 j 有 $F(B,j) > F(A,j)$ 而 $F(B,k) \le F(A,k)$。因为 $F(B,n) = |B| > |A| = F(A,n)$，所以 k<n，又因为 $F(B,0) = 0 = F(A,0)$，所以 $k \ge 0$。因为 $F(B,k) \le F(A,k)$，所以 $|B| - F(B,k) > |A| - F(A,k) \ge 0$，即 B 中截止时刻超过 k 的任务数比 A 中截止时刻超过 k 的任务数多，所以可以找到一个截止时刻超过 k 的任务 x，x 属于 B 但不属于 A，并设该任务的截止时刻为 t。令 $A' = A \cup \{x\}$，因为 t>k，所以对任意 $n \ge s \ge t$，$F(A',s) = F(A,s) + 1 \le F(B,s) \le s$,对任意 $1 \le s < t, F(A',s) = F(A,s) \le s$，根据定理 3，$A'$ 是独立集。

所以 M 是拟阵，我们的问题也就是在 M 上求一个权值最大独立集，我们在第二部分已经讨论了贪心法解决该问题。最朴素的实现，时间复杂度为 $\Theta(n^2)$，因为判定算法可以通过计算 F(A,s)在 $\Theta(n)$ 时间里解决。聪明的读者一定发现时间复杂度还可以进一步提高。的确，我们可以实现这样一个算法：设所有 n 个时间空位开始时都是空的，其中空位 i 是终止于时间 i 的单位长度时间空位。我们按罚款的单调递减顺序来考虑各个任务。在考虑任务 j 时，如果有处于 j 的期限 d_j 之前的时间空是空的，则将任务 j 填入离 d_j 最近的这样的空位。可以用并查集实现该算法。由于算法的具体实现不是本文的重点，故把详细讨论放到

附录 II。

第四部分：拟阵实例

这一部分，我们将来看一些拟阵实例，其中线性拟阵比较重要，因为它就是第一个被发现的拟阵。除此之外的只须大致了解一下即可。匹配拟阵的证明比较复杂，叙述的也比较简洁，跳跃性比较大，可以略过不看，有兴趣的读者可以仔细研究一下。

线性拟阵:

给一个 $m \times n$ 的矩阵T(T的元素需在一个域中,我们不妨认为是实数)。可以定义 $M = (S, L)$，其中S是T的所有列组成的集合，S的子集U是独立集当且仅当U中的列是线性无关的。这是个拟阵，称为线性拟阵。证明要涉及向量空间的概念，作者也未能深入研究，只能大致理解，不过给出这个严格证明还是有必要的。

证明：拟阵的前2条是显然的。

遗传性：一个向量组如果线性无关，那么它们的子向量组也线性无关。

交换性：X和Y是S的2个独立子集，且 $|Y| < |X|$ 。设W是 $X \cup Y$ 形成的向量空间。我们dim W表示W的维度，dim W至少为 $|X|$ 。假设所有 $Y \cup \{x\}$ 是线性相关的，其中 $x \in X - Y$ 。那么W被Y所形成的向量空间包含，所以dim W至多为 $|Y|$ 。那么 $|X| \le \dim W \le |Y|$，矛盾。所以交换性成立。

Ural1041是一道在线性拟阵上的最优化问题,有兴趣的读者可以研究一下,其中线性拟阵的独立集判定要判断向量组是否线性无**关**,而判断线性无关就是判断一个齐次线性方程组是否有解，判断齐次线性方程组是否有解的方法是计算矩阵的秩,当然用行列式来计算矩阵的秩效率太低,所以要通过消元法来求矩阵的秩，这些都是线性代数的内容，故不展开讨论。

我们接下来探讨一下图的拟阵和线性拟阵的关系，我们会发现其实图拟阵就是一种线性拟阵。我们对无向图 $G = (V, E)$ 定义它的点边矩阵B，B是一个 $|V| \times |E|$ 的矩阵，矩阵的每一列只有两个元素不为0，一个为1，一个为 -1，这2个元素的行号就是这条边所连接的2个点的编号。可以发现，这个矩阵所形成的拟阵和这个图的拟阵是等价的，也就是说图中的一个边集不形成环，当且仅当这些边所对应的列线性无关。证明很容易，但细节比较烦琐，故略去。

组合拟阵：

$M = (S, I_k)$ 是个拟阵，其中 S 为任意有穷集合，I_k 为由 S 的所有大小至多为 k 的子集构成。我们发现背包问题的拟阵就是组合拟阵。

拟阵的补拟阵：

如果 $M=(S,L)$ 是一个拟阵，则 $M'=(S,L')$ 也是一个拟阵，此处 $L'=\{A':$ 存在M的极大独立集A,使 $A\subseteq S-A'\}$，我们称 M' 为 M 的补拟阵。

我们接下来证明该 M' 确实为拟阵：

遗传性：对任意 $B'\in L',A'\subseteq B'$,存在 M 的极大独立集 x,使得 $x\subseteq S-B'$,因为 $A'\subseteq B'$,所以 $x\subseteq S-B'\subseteq S-A'$,所以 $A'\in L$,遗传性得证。

交换性：对任意 $A',B'\in L',|A'|<|B'|$，那么存 M 的极大独立集 A，B，使得 $A\subseteq S-A',B\subseteq S-B'$。随意找一个 $x\in B'-A'$,令 $C'=A'\cup\{x\}$。因为 $x\in B'$,所以 $x\notin B$。如果 $A=B$,则 $A\subseteq S-C'$；如果 $A\neq B$ 且 $x\notin A$,则 $A\subseteq S-C'$；如果 $A\neq B$ 且 $x\in A$，则令 $C=A-\{x\}\cup\{y\}$,其交换性成立中 $y\in B-A$,易知 C 是 M 的极大独立集,且 $C\subseteq S-C'$。综上所述 C' 是 M' 的独立集。所以交换性成立。

匹配拟阵：

对于无向图 $G=(V,E)$，定义 $M=(S,L)$，其中 S=V，S 的子集 A 是独立集当且仅当 A 集合中的点能被该图的一个匹配覆盖。

很显然，如果 G 有完美匹配，则 V 的任意子集可以被一个匹配覆盖，即完美匹配，这时 L 就是 S 的所有子集组成的集合，是个组合拟阵。这只是一种很特殊的情况

我们接下来证明这个 M 是一个拟阵，并且我们称这个 M 为无向图 G 的匹配拟阵。

拟阵的前 2 个条件是显然成立的。

遗传性：很显然，如果任意 $A\in S$ 是独立的，那么存在一个匹配 P 覆盖 A 中所有点，则匹配 P 也覆盖 A 的任意子集 B 中所有点，所以 B 是独立集，遗传性成立。

交换性：这是关键所在，也是最困难的。对任意独立集 A 和 B 且 $|A|<|B|$。假设 P_A 和 P_B 分别是覆盖 A 和 B 的匹配。如果 P_A 覆盖了一个属于 $B-A$ 的点，那么交换性显然成立。因此这种情况不成立。在 $P_A\Delta P_B$ 中，从任意 $B-A$ 中的点出发都有一条交替路径（交替的含义是 P_A 的边和 P_B 的边交替出现的，其实 $P_A\Delta P_B$ 中的路径都是交替路径）。这些交替路径可能在 $A-B$ 中结束，但不可能全部在 $A-B$ 中结束，因为 $|B-A|>|A-B|$。因此 $B-A$ 中至少存在一个点从它开始的交替路径在别的地方结束。但这条路径的终点不可能在 $B\cap A$，因为这些点在 $P_A\Delta P_B$ 中的度为 0 或 2。因此我们有一条从 $B-A$ 到一个不在 A 中的点的交替路径，现在 $P_A\Delta P$ 就是一个覆盖 A 和某个属于 $B-A$ 的点的匹配，因此满足交换性。该证明有点晦涩难懂，主要是为了说明在 A 中存在一条增广路径，该增广路径是以 $B-A$ 中的某点为起点的，并且这条增广路径上的未匹配边是 B 中的匹配边。

拓展部分：Shannon 开关游戏浅谈

Shannon 开关游戏是由信息论鼻祖 Shannon 提出的，而该游戏的优美解法是 A.Lehman 发现的，很可惜我没能找到他解决这个问题的精彩论文《A solution of the Shannon switching game》，而我找到的资料都杂乱而晦涩难懂。我尽力在找到的资料中提炼出有关这个问题的内容。

先提出这个游戏，并给出正方后手必胜的充分必要条件，最后给出一个要用到拟阵的证明，但证明中用到的一个定理的证明非常复杂，作者也未能研究得很明白，故略去。

Shannon 开关游戏由一个正方玩家和一个反方玩家在一个无向图 $G=(V,E)$（可有重复边）上进行。正方玩家每次操作可以选择一条没有画'+'的边画上'+'，反方玩家可以选择一条没有画'+'的边删除。反方先手，轮流操作，直到无法操作为止。游戏结束后，该图若还是连通的，则正方玩家获胜，否则反方玩家获胜。

正方玩家后手必胜的充要条件是：图中有 2 个边独立生成树，边独立生成树就是指两棵生成树没有公共边。

我们如果考虑图 G 的拟阵 $M=(S,L)$，S=E，S 的子集 U 是独立的当且仅当 U 不形成环。那么充要条件可以叙述为：存在 2 个极大独立集 A,B 满足 $A\cap B=\Phi$。

我们先给出一些有趣的事实。

对于 k 个拟阵 $M_1,M_2,\ldots,M_k$，$M_i=(S_i,L_i)$，我们定义它们的并为 $M=(S,L)$。$S=\bigcup_{i=1}^{k}S_i$，$L=\left\{\bigcup_{i=1}^{k}I_i \mid I_i\in L_i \text{对于} 1\le i\le k\right\}$ M 是拟阵，证明很复杂，作者能力有限，故略去。下面再不加证明给出一个公式：

$$r(U)=\min_{T\subseteq U}\left(|U-T|+\sum_{i=1}^{k}r_i(T\cap S_i)\right) \text{对任意} U\subseteq S$$

其中 r_i 为 M_i 的秩函数,r 为 M 的秩函数。

现在我们来考虑拟阵 M=(S,L)是否存在 k 个两两交集为空的独立集。令 M^k 表示 k 个拟阵 M 的并。我们用 r(U)表示 M 的秩函数,R(U)表示 M^k 的秩函数，那么有：$R(U)=\min_{T\subseteq U}(|U-T|+k\times r(T))$。因此，M 存在 k 个两两交集为空的独立集当且仅当：$k\times r(S)=\min_{T\subseteq U}(|U-T|+k\times r(T))$。等价地说就是对于所有 $T\subseteq S$：$|S-T|\ge k(r(S)-r(T))$，后面的证明将用到这个结论。

好，有了上面的结论以后，我们接下来就对正方玩家后手必胜的充要条件给出一个证明：

1、必要性：如果 S 不存在 2 个交集为空的独立集，那么根据上面的讨论，存在 $T\subseteq S$，使得 $|S-T|<2(r(S)-r(T))$。这时反方的策略是每次删除一个 $x\in S-T$。那么反方选

手至少可以从 $S-T$ 中删除 $\left\lfloor \frac{|S-T|}{2} \right\rfloor$ 个元素，正方选手最多只能拯救 $S-T$ 中的 $\left\lfloor \frac{|S-T|}{2} \right\rfloor$ 个元素。$\left\lfloor \frac{|S-T|}{2} \right\rfloor < r(S)-r(T)$，而在 T 中最多拯救 $r(T)$ 条有用（无环）的边（所谓有用就是指能减少连通块数量的边，即使拯救超过 $r(T)$ 边会形成环，一些边是不必要的）。所以我们能拯救的有用的边数 $e=\left\lfloor \frac{|S-T|}{2} \right\rfloor + r(T) < r(S)-r(T)+r(T)=r(S)=V-1$，所以最后至少还有 2 个连通块，即不连通。所以正方败。得证。

2、充分性：M 有 2 个交集为空的独立集 B_1, B_2，即存在 2 个边独立生成树 T_1, T_2。我们的基本思想是，在反方选手走完一步，紧跟着该正方选手走的时候，我们都要构造出一对新的生成树，它比原来的一对多一条公共边，且在这条公共边上画'+', 这样我们每步都保证我们的那对生成树的公共边全有'+'。开始时，T_1 和 T_2 没有公共边，我们记为：

$$T_1^0 = T_1 \text{ 和 } T_2^0 = T_2$$

在比赛的第一个回合，反方选手先走，并删除某条边 β，我们考虑两种情况。

情况 1：β 是树 T_1^0 或 T_2^0 的一条边，不妨设是 T_1^0 的边。因为 T_1^0 和 T_2^0 是生成树，所以存在 T_2^0 的一条边 α，使得通过向 T_1^0 中加入边 α 去掉边 β 后，所得到的图 T_1^1 仍是一棵生成树。我们给 P 的指令是：在 α 上画个+号。令 $T_2^1 = T_2^0$，则 T_1^1 和 T_2^1 恰好有一条共同的边，即带有+号的边 α

情况 2：β 既不是 T_1^0 的边，也不是 T_2^0 的边。

这时，我们给 P 的指令是：在 T_1^0 或 T_2^0 的任意一条边 α 上画+号，不妨设是 T_1^0 的边。因为 T_2^0 是一棵生成树，存在 T_2^0 的一条边 γ，使得通过向 T_2^0 中加入边 α 去掉边 γ 后，所得到的图 T_2^1 仍是一棵生成树，令 $T_1^1 = T_1^0$，则树 T_1^1 和 T_2^1 恰好有一条共同的边，即带有+号的边 α。

我们断定：在比赛第一个回合结束时，G 有两棵生成树 T_1^1 和 T_2^1，它们恰好只有一条共同的边，且被画上+号。

后面的策略非常类似于第一回合的策略，在比赛的第 k 个回合结束时，两棵生成树 T_1^k 和 T_2^k 恰好有 K 条共同的边，且这 K 条边都被画上+号。设图有 M 个点，在比赛的第 M-1 个回合结束时，生成树 T_1^{M-1}，T_2^{M-1} 恰好有 M-1 条共同的边，因为具有 M 个点的

树只有 M-1 条边，这就意味着 T_1^{M-1} 和 T_2^{M-1} 是同一棵树，所以带+号的边使图连通。

这就是必胜策略。

总结

拟阵是一种组合结构，遗传性和交换性是它本质特性，证明一种结构是拟阵的关键在于证明遗传性和交换性，遗传性往往很简单，而交换性往往很难，因此证明交换性是关键。

我们在进行证明时用的最多的是两种常见的证明技巧：构造，反证，两者各有所长，相辅相成。

贪心算法的正确是由拟阵的本性决定的。

拟阵最初就是以线性拟阵提出的，我们发现，图的拟阵本质上是线性拟阵。

Shannon 开关游戏很有趣，它还可以有很多变化，比如正方先手必胜条件，或者把问题改成使两个点 u,v 连通，而不是使正张图连通。有关拓展部分没有证明的一些内容与拟阵的交算法有关。

虽然本文介绍的只是拟阵理论冰山一角，不过相信这一角已足以使大家体会到：拟阵是美的。

参考文献

1、《Introduction to Algorithms》 Thomas H.cormen , Charles E.Leiserson , Ronald L.Rivest , Clifford Stein 著

2、 MIT Topics in Combinatorial Optimization

3、《组合数学》 [美]Richard A. Brualdi 著

4、《算法艺术与信息学竞赛》 刘汝佳 黄亮 著

5、《What is a matroid?》 JAMES OXLEY 著

6、《什么是数学》 [美] R柯朗 H罗宾 著

7、《拟阵论》 赖虹建 著

8、同济大学线性代数课件

9、Strategies for the Shannon switching game Richard Mansfield 著

10、《代数》 [美]Michael Artin 著

附录

附录 I 罗素悖论

虽然直觉主义者的那种不妥协的立场对大多数数学家来说是太极端了，但是当美妙的无限集理论中出现了一些逻辑上明显的悖论时，集论受到了严重的威胁。人们很快就发现毫无约束的滥用"集合"的概念必然引出矛盾。有一个由罗素（R.Russel）解释出的悖论可叙述如下。大多数集合不包含自身作为元素。例如，全体整数集 A 只包含数为元素；A 本身，不是一个整数，而是一个整数集，A 并不包含它自身为元素。这样的集我们可以称之为"普通的"。有许多集可能包含它自身为元素，例如集 S 定义如下："凡是可以用不超过三十个字来定义的集合是 S 的元素"可以看到，S 是包含了它自身为一元素的。这样的集我们可以称之为"非普通集"。但无论如何，多数集将是普通的。为了排除"非普通"集的反常状态，我们可以只着眼于所有普通集组成的集，称它为 C。集合 C 的每一个元素本身是一个集合，而且事实上是一个普通集。现在产生了一个问题：C 本身是普通集还是非普通集？它必须是这二者之一。如果 C 是普通集，由于 C 定义为包含所有普通集，它包含了它本身作为一个元素。这样的话，C 必须是非普通集，因为非普通集是那些包含了它本身为元素的集。这是一个矛盾。因此 C 必须是非普通集。但这时 C 包含了一个非普通集（即 C 本身）为其元素，这与 C 只包含普通集的定义相矛盾。因此无论哪一种情形，仅仅是 C 的存在，就已经使我们陷入矛盾。

摘自参考文献 6

附录 II 调度问题的高效算法

回顾这个算法：我们按罚款的单调递减顺序来考虑各个任务。在考虑任务 j 时，如果有处于 j 的期限 d_j 之前的时间空是空的，则将任务 j 填入离 d_j 最近的这样的空位，否则考虑下一个。

这个算法的正确性很明显。因为如果 1.. d_j 没空位，则说明把 j 放入形成的 A'，$F(A',d_j)>d_j$，所以考虑下一个。如果有空位，则说明放入形成的 A' 是独立的,所以要把 j 加入，放哪呢？贪心原则就是在截止时间前尽量靠后放，反证法很容易说明这个贪心的正确性（即不这样放不会更好）。

我们定义一种时刻之间的关系：

时刻 i 和时刻 j 有关系当且仅当 i 和 j 之间的时间空位全被填满。

容易证明这是一种等价关系，即满足自反性，对称性，传递性，所以我们可以用并查集来维护这种关系。每次把一个任务放到某个时间空位，就把这个时间空位的 2 个端点，即

2 个时刻所在的集合合并，并保证并查集中每个集合的代表都是该集合中时刻值最小的元素。于是对于任务 j,如果 d_j 所在集合的代表是 0，则说明 d_j 前已无空位，否则 d_j 前最靠后的空位就是 d_j 所在集合的代表时刻前的那个空位。下面我们给出核心代码：

```
    fillchar(f , sizeof(f) , 255);//初始化存储并查集的父亲指针数组
      ret := 0; //答案初始化
for i := 1 to n do
begin
      k := find(d[i]); //找代表
      if (k > 0) then Union(k - 1 , k)  //如果有空位，放，并将 2 端的集合合并
                 else ret := ret + w[i];//无法完成，罚款
end;
```

因为要使集合代表是最小的，所以 Union 要这样实现：

```
procedure Union(u , v : longint);
begin
u := find(u); v := find(v);//find 是找代表
if (u > v) then f[u] := v
else f[v] := u;
end;
```

源代码在：task\task.pas