算　法　图　论

云南大学数学系

李 建 平

2015年9月

第十章 NP-完备性

10.1 最优化问题的三种提法

定义10.1 一个最优化问题的实例（例子）就是一对元素（F，c），其中F是一个集合的定义域，c是费用函数，即$c:F\rightarrow R^{+}$，问题是要寻找一个$f\in F$，使得对一切$y\in F$，均有

$$c(f)\leqslant c(y)\ （或c(f)\geqslant c(y)）$$

这样的f称为给定实例的整体最优解。

定义10.2 一个最优化问题就是它的一些实例构成的集合。

衡量一个算法的效果，广泛采用的标准是该算法在得到最终答案前所用的时间（或初等运算步骤之和）。

一般地，算法解决优化问题得到最终答案前所用的时间依赖于实例的输入规模。

对于所有规模为n的输入，算法对这些输入进行的行为可能不同，我们把其中**最坏的行为**定义为该算法关于输入规模为n的计算复杂性（简称**复杂性**）。从而，算法复杂性是输入规模的函数。

计算机科学家们认为，解决一个计算问题的算法，仅当其计算复杂性随输入规模n的增加而成（关于n的）多项式地增加时，该算法才是实际有效的算法，并称该算法是**多项式时间算法**。特别，有些算法的复杂性虽然不是多项式的，但是它有一个多项式作为上界，这样的算法也称多项式时间算法。

同时，计算机科学家们认为，解决一个计算问题的算法如果不是多项式时间算法，则称之为指数时间算法。

下面优化问题都存在多项式时间算法：

① 最小支撑树问题、最短路问题；

② 网络流中的最大流、最小费用流；

③ 图的最大匹配与最优匹配；

④ 线性规划。

但是，单纯形算法不是多项式时间算法；求解线性规划的椭球算法、内点算法是多项式时间算法。

预定用算法A_F和A_c来隐含地给出F和c。已知组合对象f和参数集合S，算法A_F决定f是否是由给定参数规定的可行解集合F中的元素；已知可行解f和另一参数集合Q，算法A_c给出c(f)之值。

例10.1 最大团问题的组合最优化问题：

已知图G=(V,E)，求最大子集$C \subseteq V$，使得对于所有不同的u,v∈C，都有uv∈E。

此时，参数集合S就是图G，算法A_F就是确定f是否构成G的一个团，算法A_c就是计算f的基数，此时Q是空集。

组合最优化问题有下面三种计算形式：

定义10.2（最优化形式）已知算法A_F和A_c的参数集合S和Q，求最优可行解。

定义10.3（计值形式）已知算法A_F和A_c的参数集合S和Q，求最优解的费用。

定义10.4（判定形式）已知组合优化问题的实例（即算法A_F和A_c参数集合S和Q的表达式）及整数L，是否存在一个可行解$f \in F$，使得$c(f) \leq L$?（或者$c(f) \geq L$?）

定理10.1 上述三种形式的最优化问题是等价的。

10.2 P类和NP类

例10.2 （停机问题）已知一个算法A及其输入，算法A总会停止吗？

例10.3 （哈密尔顿问题）已知一个图G，G是否存在一个圈C，使得C经过每个顶点恰好一次？

例10.4 （图的连通性问题）已知一个图G，G是连通的吗？

例10.5 （最大匹配问题）已知一个图G和整数k，G中存在边数不小于k的匹配吗？

定义10.5 对于组合最优化问题（判断性问题）B的任何一个输入I，如果存在多项式时间算法A能够求得输入I的最优解（回答为是或者否），称该问题是多项式时间内是可解的，并简称该问题为P类问题。

用P来表示多项式时间算法所能够解决的判断性问题类（即全体）。

如果x是问题的答案为是的实例，则存在对x的简短证明，使得在多项式时间内检验这个证明的真实性。

定义10.6 称最优化问题（判断问题）B是NP类的，如果存在一个多项式p(n)和算法A（证明检验算法），使得下面的论断成立：

符号串x是问题B的一个答案为“是”的实例，当且仅当存在一个符号串c(x)（“证明”），使得$|c(x)| \le p(|x|)$，并具有性质：

若给算法A提供输入x$c(x)，则至多经过$p(|x|)$步之后，算法A能够给出答案为“是”。

例10.6（最大团问题）已知一个图G=（V，E）及整数k，问图G有k个点的团吗？

例10.7（货郎担问题）已知一个nXn的对称矩阵D=（d_{ij}）及整数L，问是否存在一个循环排列（环游）τ，使得 $\sum_{i=1}^{n} d_{i\tau(i)} \le L$ ？

例10.8（适定性问题）给定包含布尔变量的$x_1,x_2,\ldots,x_n$的m个句子，布尔表达式$C_1 \cdot C_2 \cdots C_m$是可适定的吗？

例10.9（整数线性规划）已知一个mXn的整数矩阵A=（d_{ij}）及整数向量b，问是否存在一个整数向量x≥0，使得Ax=b？

10.3 多项式时间的归结

定义10.7 设B_1和B_2都是判断性问题，称B_1在多项式时间内归结到B_2，当且仅当B_1有（多项式时间）算法A_1，并且A_1是多次地以单位费用把B_2的(假象)算法A_2用作子程序的算法。把算法A_1称为问题B_1到问题B_2的多项式时间归结。

定理10.2设B_1和B_2都是判断性问题，如果B_1多项式地归结到B_2，并且B_2有多项式时间算法，也有多项式时间算法。

定义10.8 设B_1和B_2都是判断性问题，称B_1多项式地变换到B_2，如果对于任意的符号串x，在（|x|的）多项式时间内能够构造出符号串y，使得x为问题B_1的“是”的实例，当且仅当y为B_2的“是”的实例。

定义10.9 判定性问题B∈NP称为是NP-完备的，如果所有其他的NP问题都能够在多项式时间内归结到问题B。

10.4 典型的NP-完备性问题

为证明一个问题B是NP-完备性问题，必须证明：

① 该问题B是NP类的；

② 所有其他的NP类问题都能够在多项式时间内变换到该问题B。

事实上，②的证明通常是证明某个已知的NP-完备性问题可以多项式时间内变换到要证明的问题即可（图示）。

定理10.3（Cook定理）适定性问题是NP-完备的。（ Cook直接给出证明）

定理10.4 3-适定性问题是NP-完备的。
（把适定性问题变换到3适定性问题）

定理10.5 整数线性规划问题是NP-完备的。（把适定性问题变换到整数线性规划问题）

定理10.6 团问题是NP-完备的。
（把3-适定性问题变换到团问题）

定理10.7 哈密尔顿圈问题是NP-完备的

（把3-适定性问题变换到哈密尔顿圈问题）

定理10.8 货郎担问题是NP-完备的。

（哈密尔顿圈问题是货郎担问题的特例）

定理10.9 0-1背包问题、整数背包问题是NP-完备的。

定理10.10 划分问题、3-划分问题是NP-完备的。

定理10.11最小强连通支撑子图问题是NP-完备的（由有向哈密尔顿圈问题变换而来）。

10.5 近似算法

定义10.10 设B是以正整数函数c为费用的一个最优化（最小或最大）问题，A是一个多项式时间算法，对于优化问题B的任何输入实例I，算法A会产生一个可行解f_I，如果存在某个整数k，使得下面式子对任何实例I成立

$$\frac{|c(f_I)|}{OPT_I} \le k \qquad \text{（或} \quad \frac{|c(f_I)|}{OPT_I} \ge k \text{）}$$

称算法A是优化问题B的k-近似算法，这里OPT_I是实例I理论上的最优值。

例10.10 (点覆盖问题)已知一个图G=(V，E)，求最小的点覆盖C，即对任意一条边uv∈E，使得u∈C或者v∈C，其目标是使得|C|达到最小。

解：每次选取一个顶点，使得该顶点覆盖尽可能多的边，算法如下：

算法　1

输入：一个图G=（V，E）

输出：G的一个点覆盖C

Begin

C:=Φ

While E≠Φ do

1. 在V中选取一个最大度点，当最大度点不止一个时，可任意选取一个最大度点；

2. 从图G中去掉该点及相关联的边，把该顶点加入C中。

End

算法 2

输入：一个图G=（V，E）

输出：G的一个点覆盖C

Begin

C:=Φ

While E≠Φ do

1. 在E中选取一条边e=uv；

2. 从图G中去掉顶点u,v及相关联的边，把该两个顶点u,v加入C中。

End

算法 3 （Gavril）

输入：一个图G=（V，E）

输出：G的一个点覆盖C

Begin

在图G中寻找一个极大（或最大）匹配M，

把M中每条边的两个顶点u,v加入C中。

End

算法Gavril是 2 -近似算法，证明如下：

显然，输出集合 C 是图 G 的一个点覆盖，从而有|C|=2|M|；对于极大匹配中的每条边，至少该边上有一个顶点必须属于最优的点覆盖OPT，从而|M| ≤|OPT|。于是

$$|C|=2|M| \le 2\ |OPT|,$$

说明算法Gavril是 2 -近似算法。

例10.11（货郎担问题）已知一个赋权图G=(V，E；w)，寻找图G的一个闭回路(环游)，使得上边的权重之和达到最小。

这是一般形式下的货郎担问题，该问题没有任何n^k-近似算法，这里k是任意正数。

定义10.10　设有一个n×n的对称矩阵D=（d_{ij}），称矩阵D满足三角不等式（Δ-不等式），如果对于所有1≤ i，j，k≤n，都有

$$d_{ij} + d_{jk} \geq d_{ik}$$

定义10.11 Δ-货郎担问题就是满足三角不等式的货郎担问题。（此时的图是完全图）

定理10.11 Δ-货郎担问题是NP-完备的。

证明：把哈密尔顿圈问题的实例变换到Δ-货郎担问题的实例，变换形式为：

给定密尔顿圈问题的实例 I ： $G=(V, E)$，构造Δ-货郎担问题的实例 J ： $H=(V, E_H, w)$，若 $uv\in E$，构造 $uv\in E_H$，取 $w(u,v)=1$，若 $uv\notin E$，构造 $uv\in E_H$，取 $w(u,v)=2$ 。

可知实例 J 满足三角不等式，并且实例 I 有H圈iff实例 J 的最优值等于n。

算法　树算法

输入：一个完全图G=(V,E)及距离矩阵D＝(d_{ij})

输出：G的一个环游

Begin

1. 根据距离矩阵(d_{ij})，求出图G的最小支撑树T＝（V，E_T）；
2. 把T中的每条边使用两次，得到一个Euler图H；
3. 对图H利用Fleury算法找到Euler回路，利用三角不等式，把此Euler回路变成一条简单回路，并输出最后的回路。

End

算法　Christofids

输入：一个完全图G=(V,E)及距离矩阵 $D=(d_{ij})$

输出：G的一个环游

Begin

1. 根据距离矩阵(d_{ij})，求出图G的最小支撑树 $T=(V, E_T)$；
2. 找出T中的所有2k个奇数次顶点，并求出所有奇数次顶点之间的最短路；
3. 利用奇数次顶点及最短路长度构造2k个顶点的完全图H，求出图H的最小完美匹配M，|M|=k;
4. 把M中的每条边对应于G中的最短路，并把这k条最短路迭加到图G上，得到一个Euler图G*;
3. 对图G*利用Fleury算法找到Euler回路，利用三角不等式，把此Euler回路变成一条简单回路，并输出最后的回路。

End

定理10.11　树算法是Δ-货郎担问题的2-近似算法。

定理10.12　Christofids算法是Δ-货郎担问题的3/2-近似算法。

10.6 分支定界算法与动态规划算法