第16卷 第4期　郑州工学院学报　Vol.16 No.4.
1995年 10 月　Journal of Zhengzhou Institute of Technology　Oct. 1995

可行性分配问题的一种解法

赵玉莹

(郑州工学院数力系)

摘　要: 本文给出了可行性分配问题的一种算法，并且研究了算法的界限。

关键词: 偶图，对集，极大流

中图分类号: O221

可行性分配问题是常遇到的一类问题，已有一些不同的解法。本文利用 karzanov 方法对可行性分配问题进行研究，给出了可行性分配问题的一种解法，并对算法进行分析，得出算法的极限是 O（N^2)。

1　可行性分配问题

可行性分配问题的提法: m 个人员分配去干几项工作，每人只能适应其中一项或几项工作，寻找一个分配方案，使尽可能多的人分配到适合的工作岗位上。（假定 $m \leqslant n$）。

构造一个偶图 $G=(S,T;R)$, 其中 $S=\{s_1,s_2,\cdots s_m\}$ 表示人员的集合；$T=\{t_1,t_2,\cdots t_n\}$ 表示工作的集合；若人员 S_i 适合做工作 t_j，则定义一条弧 $(s_i,\ t_j)$, 所有弧的全体记为 R。一个分配方案就相当于选择一组无公共端点的弧——对集，于是问题就转化为求偶图的最大对集。

例如，有三人分配去作 4 件工作，每人适应工作如图 1 的偶图 G 中直线弧所示。偶图 G 中粗线表示最大对集，即是一所求分配方案。

在偶图 G 中添上一个源 p, 且从 p 到 s 的诸顶点连以容量 b_{pi} 为 1 的弧；再添上一个汇 q，且从 T 各顶点到 q 连以容量 b_{jq} 为 1 的弧。原 R 中弧容量 b_{ij} 为$+\infty$, 这样得到一个网络图。如图 2 所示。

此网络图上每一整值流（各弧上流量为 0 或 1)，对应于偶图的一个对集（由流量为 1 的弧组成)，反之亦然。因此，求偶图最大对集，等价于求此网络的最大流。此网络的结构比较特殊。从源 p 流入 s 中各顶点的流量 x_{pi} 及 T 中各顶点流入汇 q 的流量 x_{jq} 均是 1 或 0，R 中的弧流不能饱和。

收稿日期: 1994-09-24

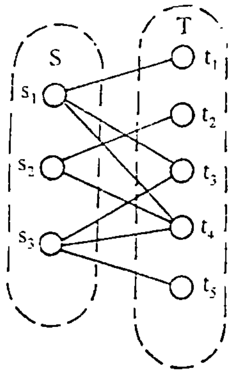

图 1

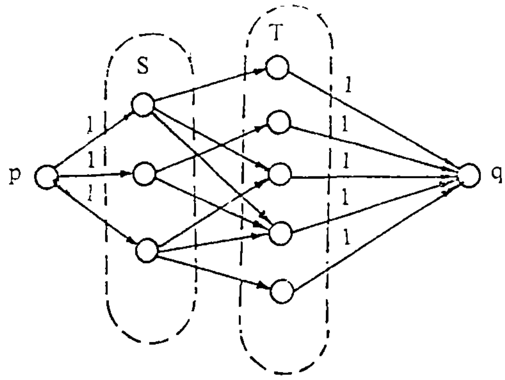

图 2

首先给出一些定义:

定义 1: 对任一个流 $\{x_{ij}\}$, 由 b_{ij}' 组成的网络称为剩余网络。记为 $N(f)$ 。其中

当 $x_{ij} < b_{ij}'$ 时, $b_{ij}' = b_{ij} - x_{ij}$

当 $x_{ij} > 0$ 时, 反向弧 $b_{ji}' = b_{ji} + x_{ij}$

b_{ij} 为弧 e_{ij} 的容量, x_{ij} 为弧 e_{ij} 上流量。

定义2: 剩余网络的层次划分:

①$P \in$ 第0层;

②$N(f)$ 中弧 (u,v), $u \in$ 第 i 层, $V \bar{\in}$ 第 i 层 $(j \leqslant i)$, 则 V 属于第 $i+1$ 层。

定义 3: 如果流 x_{ij} 不允许变化, 则说弧 e_{ij} 是关闭的。否则, 是开放的。

定义 4: 如果从顶点 V_j 到汇点 q 的每一条路至少包含一条饱和的弧, 则说顶点 V_j 是阻塞的。

显然, 一个顶点一旦是阻塞的, 就不可能再有流通过这个顶点。

2 算法

算法的特点是在一个分层网络上, 从源 P(第 0 层) 推出予备流来开始一个阶段。从第 0 层到第一层, 第一层到第二层, …推进予备流直到汇 q, 则转入第二步。第二步是在包含不平衡点的最高层第 i 层平衡顶点, 第 i 层所有顶点被平衡后, 将引起第 i-1 层顶点的不平衡, 其他层顶点不变化, 从第 i-1 返回第一步。一个阶段由推进予备流和平衡予备两步组成。

在平衡步骤中要减少进入顶点 V_j 的流时按"后进先出"方式减去 x_{ij}, 即首先减少最后一个进入 V_j 的流。

2.1 算法步骤:

第 0 步, 在一个网络流上考虑分层剩余网络。

第 1 步, 从源 P 推出流到第一层顶点, 放置流量 $x_{pj} = b_{pj} = 1$。一般地, 从低层的

不平衡点推进予备流到下一层，且相继到汇点 q。对顶点 V_j，弧 e_{jk} 按下标 K 增加顺序排列，对最小的 K，放置弧流 $x_{jk}=1$，其余 $x_{jk}=0$。如果顶点 V_j 的所有流出弧或是关闭的或是饱和的，则推进子程序停止。

第 2 步，平衡予备流，从含有不平衡点 V_j 的最高层开始对每一个这样的顶点 V_j 按“后进先出”规则减少进人顶点的流，直到顶点被平衡。对新平衡的顶点 V_j，所有进人弧注上 * 号表示关闭。如果这一层中所有未平衡的顶点均被平衡，则返回第一步。

重复这两步使流在剩余子网络中变成极大流。由此得出偶图的最大对集，即得出所求的一个分配方案。

2.2 算例:

用偶图表示一个 $m=n=6$ 的可行性分配问题。试求一个可行性分配方案。

为简化，集合 S 中顶点记为 1——6，集合 T 中顶点记为 7－ 12。

首先，补上源 P，汇 q，从 P 到 S 及从 T 到 q 连以容量为 1 的弧，R 中弧容量为 $+\infty$，并做分层网络。

① 第 1 步，从源 P 推出予备流，进人第一层诸顶点．进一步推进流到第二层顶点: $x_{18}=x_{27}=x_{38}=x_{48}=x_{57}=x_{68}=1$，第三层顶点 $x_{7q}=x_{8q}=1$。然后转到第 2 步。如图 3。

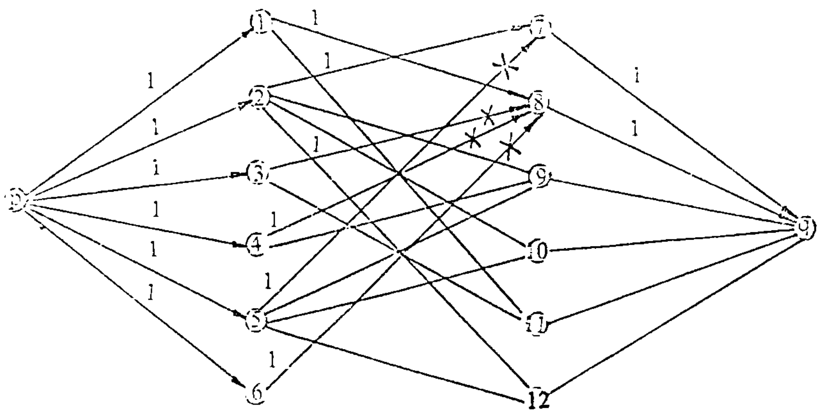

图3

第 2 步，从第 2 层平衡预备流，减少流量 $x_{57}=x_{38}=x_{48}=x_{68}=1$，（在图 4 中划 x 的弧流）使所有进人顶点 V_7，V_8 的弧被关闭。返回第 1 步。

②第1步，从第1层推进流 $x_{311}=x_{49}=x_{59}=1$ 及 $x_{9q}=x_{11q}=1$，转到第2步。

第 2 步，平衡顶点 V_9，减少流量 $x_{59}=1$，使 V_9 的全部进人弧关闭。

③第1步，再推进流 $x_{510}=1$，$x_{10q}=1$。此时第2层顶点均平衡。

第 2 步，从第 1 层顶点开始平衡步骤。顶点 V_1 至 V_5 均平衡，平衡顶点 V_6，减少流量 $x_{p6}=1$，V_6 的进人弧被关闭。

此时，从源 P 再也推不出流，且所有顶点是平衡的，所以此时网络流是极大流。R

中流量为1的弧构成一组无公共顶点的弧——对集，它对应一个分配方案。如图4所示。

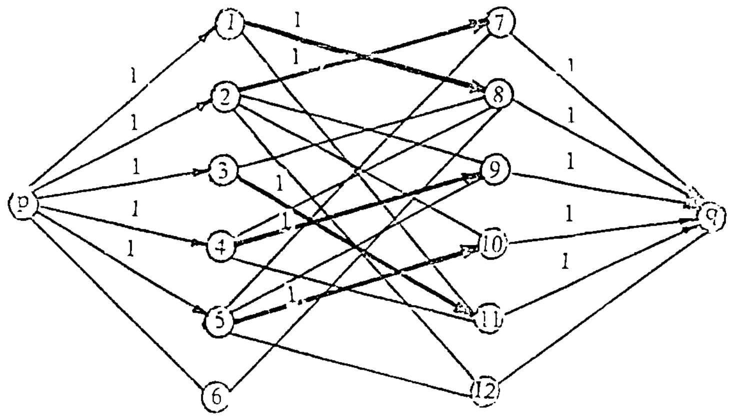

图4

分配方案不唯一，从一层推进流进入下一层时，顶点顺序不同时，则对应不同的分配方案。

3　算法分析

定理：经过一个阶段后得到的流是一个极大流，且算法的极限是$O(N^2)$。($N=m+n$)。

首先证明二个引理。

引理1　在每一时刻，所有不平衡和被平衡过的顶点均阻塞。

证明：对平衡次数K进行数学归纳法证明。

$K=1$时，第一次推进中出现的不平衡点，由于流出弧饱和，所以它是阻塞的。最高层被平衡过的顶点V_j，由于只减少流入弧流，所以仍保持阻塞。平衡V_j后其余不平衡点V_i，若V_i到汇q的路中包含V_j，则由V_j的阻塞，V_i也必阻塞。若路中不包含V_j，即V_j是推进中出现的不平衡点，已证是阻塞的。

假设进行了$K-1$次平衡之后，所有不平衡点和被平衡过的点均是阻塞的。

对第K次推进时出现的不平衡点V_i，从V_i到q的路中必有饱和弧或关闭弧。（否则与推进流的定义矛盾）。若路中有饱和弧，则V_i阻塞。若路中有关闭弧，则路中必包含被平衡过的顶点V_j，由归纳假设V_j阻塞，所以V_i也阻塞。

对第K次平衡步骤中被平衡的点V_l，由于只减少其流入弧流量，所以V_l仍阻塞。而平衡V_l时不会改变其他顶点的阻塞性。（证明类似$K=1$的证明）。

综上所述，一个顶点一旦在某一步推进中变成不平衡的，则在以后整个过程中都是阻塞的。

引理2　每个顶点至多被平衡一次。

证明　一个顶点V_i被平衡后全部流入弧被关闭，则其流出弧流不能增加。只须证V_i的流出弧流不会减少，也就是证任意比V_i层高的所有弧流不会减少。如若不然，设e_{jk}是所有高层弧中最先减少弧流的一个弧，显然V_j，V_k位于V_i的高层。在V_i被平

衡时，V_k 应是平衡的。而 x_{jk} 的减少意味着在此之前 V_k 又变成不平衡点了，引起 V_k 不平衡的可能情况有二:

① 由于流的推进，使 V_k 的进人弧流增加，这是不可能的。因为由算法可知，V_k 被平衡之后，所有进人弧已被关闭。

② 由于比 V_k 高层的顶点的平衡使 V_k 的流出弧流减少．这与 e_{jk} 是最先减少流的弧的假设矛盾。

综上讨论，被平衡过的顶点不会变成不平衡的。

现在来证明定理。

证明: 源 P 在阶段开始是阻塞的，因为 $x_{pj}=b_{pj}=1$。由引理 1 和引理 2，每一个不平衡点至多被平衡一次且在以后过程中始终保持阻塞和平衡。若干次推进、平衡步骤后，所有顶点均平衡，此时流是网络流且源保持阻塞性。因此得到的流是极大流。

在平衡步骤，如果一个弧流是减少的，则这个弧变成关闭的，所以流减少的全部次数可用弧的数量 M 来限定为 $0(M)$。

在推进步骤，当弧流增加时，弧或者饱和或者不饱和。饱和现象对每一条弧至多发生一次，这是因为流的任何减少将关闭这条弧。所以饱和现象的极限也是 $0(M)$。

如果推进中一个弧流是增加但没达到饱和，在最初予备流推进中这情况至多可能发生 $N-1$ 次。（因为 N 个顶点最多 $N-1$ 层，由算法每层只能有一条 $x_{jk}<b_{jk}$ 的不饱和弧。）而在下一次流的推进中，流不能到达平衡步骤中被平衡过，弧已关闭的顶点，所以增加流但没饱和的情况至多发生 $N-2$ 次。继续下去，得到增流但没饱和的弧流的次数是

$$(N-1)+(N-2)+\cdots+2+1=\frac{1}{2}N(N-1)=0(N^2)$$

所以一个阶段求极大流的算法极限是

$$2\ )(M)+0\ (N^2)=0\ (N^2)。$$

算法至多有 $n-1$ 个阶段，所以以算法的全部工作极限是 $0\ (n^3)$。

参 考 文 献

1 T.C.Hu　Combinatorial Algorithms

2 Robin J.Wilson　Introduction to Graph Theory.

3 林贻勋. 线性规划与网络流

An Algorithm for the Assignment Problem

Zhao Yuying

Zhengzhou Institute of Yhechnology)

Abstract: In this paper, we give an algorithm for the assignment problem and consider the bound of the algorithm.

Keywords: bipartite graph, matching, maximal metwork flow.