

文章编号: 0253-9888(2001)03-0289-05

# 基于变长编码求解一维下料问题的演化算法

李元香, 张进波, 徐静雯, 王 琳

(武汉大学 软件工程国家重点实验室, 湖北 武汉 430072)

**摘 要:** 针对一维下料问题的特点, 将线性规划方法与演化算法相结合, 提出了一种基于变长编码求解一维下料问题的演化算法. 该算法设计了一种新颖的遗传算子, 实现简单, 求解快速. 实验表明, 运用该法求解下料问题, 材料利用率高, 平均达到 97.5% 以上, 具有很好的实用价值.

**关 键 词:** 下料问题; 整数规划; 线性规划; 演化算法

**中图分类号:** TP 301.6 **文献标识码:** A

## 0 引 言

制造、建筑等行业对角钢、钢管等一维型材消耗数量巨大, 最大限度地节约材料, 提高材料利用率, 是工厂实际生产中的一项根本原则. 目前, 实际应用中大量采用的是人工下料法, 不仅效率低下, 而且浪费非常大, 材料利用率一般在 80% 左右. 也有一些应用行业采用计算机辅助下料, 但是所用的算法和人工下料法一样, 本质上都是贪婪法, 利用率并没有得到明显改善, 仅仅是提高了工作效率. 从计算复杂性理论来看, 一维下料问题属于 NP 难问题. 国内外关于这方面的研究十分活跃, 并且提出了不少近似算法, 如 Dyckhoff H 提出的线性规划方法<sup>[1]</sup>以及 Sarker B R 提出的动态规划方法<sup>[2]</sup>等. 但这些算法过于复杂, 也未能有效地解决巨大数量切割方式的优选问题. 本文将线性规划方法与演化计算<sup>[3,4]</sup>相结合, 提出了一个求解一维下料问题的实用演化算法(简称 VEA). 大量的计算机实验表明, 此算法实现简单, 求解快速, 材料利用率高, 平均达到 97.5% 以上, 具有很好的实用价值. 此外, 在 VEA 中还给出了一种新颖的杂交变异算子, 对于求解组合优化问题具有一定的启发性和通用性.

本文第 2 部分是问题的数学描述及其简化的模型, 第 3 部分给出算法描述和算法的实现方法, 第 4

部分给出若干计算机实验结果.

## 1 下料问题的数学模型

设共有  $m$  种规格的原材料, 长度分别为  $H_i (i = 1, 2, \dots, m)$ , 每种原材料的根数为  $p_i$ . 要利用它们加工成  $n$  种型号的坯料, 它们的长度分别为  $L_k (k = 1, 2, \dots, n)$ , 每种型号坯料的需求量分别为  $q_i (i = 1, 2, \dots, n)$ .

针对  $n$  种坯料, 设长度为  $H_i$  的原材料共有  $h_i$  种截取方法. 假设这些截取方法是已知的, 分别为  $(a_{j1}^i, a_{j2}^i, \dots, a_{jn}^i) (j = 1, 2, \dots, h_i)$ , 其中  $a_{jk}^i (k = 1, 2, \dots, n)$  表示长度为  $H_i$  的原材料的第  $j$  种截取方法中截取长度为  $L_k$  的坯料的根数. 比如, 对于长度为 10 m 的原材料, 要将其截成长度为 2 m, 3 m 和 5 m 的坯料. 其中的一种截取方法是 (1, 1, 1), 表示长度为 2 m, 3 m 和 5 m 的坯料各截出一根; 另一种截取方法是 (5, 0, 0), 表示截出长度为 2 m 的坯料 5 根, 而没有截出长度为 3 m 和 5 m 的坯料. 当然, 还有很多其他的截取方法, 这里就不一一列举了.

显然, 对  $i = 1, 2, \dots, m; j = 1, 2, \dots, h_i$ , 满足:

$$\sum_{k=1}^n a_{jk}^i L_k = H_i \quad (1)$$

设  $x_j^i$  表示长度为  $H_i$  的原材料采取第  $j$  种截取方法的根数, 当  $x_j^i = 0$  时表示不采取此截取方法. 为

收稿日期: 2001-02-02

基金项目: 国家自然科学基金资助项目 (6970303011)

作者简介: 李元香 (1962-), 男, 教授, 主要从事演化计算和并行计算方面的工作. E-mail: yxli@whu.edu.cn

© 1995-2005 Tsinghua Tongfang Optical Disc Co., Ltd. All rights reserved.

使用料最省, 则必须使原材料用料总长度  $\sum_{i=1}^m H_i x_j^i$  达到最小同时还必须满足下料要求和约束条件(1). 由此, 得到如下的整数规划问题:

$$\min Z = \sum_{i=1}^m H_i x_j^i \quad (2)$$

$$x_j^i \leq p_i, (i = 1, 2, \dots, n) \quad (3)$$

$$\sum_{j=1}^m a_{jk} x_j^i \leq q_k, (k = 1, 2, \dots, n), \quad (4)$$

$$x_j^i \geq 0, x_j^i \text{ 取整数} \quad (5)$$

其中,  $a_{jk}$  还需满足约束条件(1).

问题(2) ~ (5) 是一个整数规划问题, 它的解  $x_j^i$  就是采取每种截取方法的原材料根数. 下料问题的一种下料方案由原材料  $H_i$  的  $h_i$  种截取方法  $(a_{j1}^i, a_{j2}^i, \dots, a_{jn}^i)$  及采取这种截取方法的原材料根数  $x_j^i$  表示, 其中  $i = 1, 2, \dots, m; j = 1, 2, \dots, h_i$ .

## 2 下料问题的演化算法设计

下面是算法的主要流程, 其中 random 产生 [0, 1) 的随机数, eval( $S$ ) 表示个体  $S$  的适应值, 也就是  $S$  花费的原材料的总长度,  $p$  表示选择变异的概率.

初始化种群  $P$ ;

while (不满足终止条件) do

{

$\forall S_i \in P$  do

{

$S = S_i$ ;

if random >  $p$

随机从种群中另选两个个体  $S_1, S_2$ ;

else

随机生成一个新的个体  $S_1, S_2$ ;

将个体  $S, S_1$  和  $S_2$  合并得到  $T$ ;

由  $T$  生成新的个体  $S$ ;

}

if (eval( $S$ ) < eval( $S_i$ ))

$S_i = S$ ;

}

}

### 2.1 编码方案

由问题(1) ~ (5) 就可以得到问题一种下料方案, 即  $(a_{j1}^i, a_{j2}^i, \dots, a_{jn}^i)$  和  $x_j^i (i = 1, \dots, m; j = 1, \dots, h_i)$ . 由于通过  $(a_{j1}^i, a_{j2}^i, \dots, a_{jn}^i)$  就可以得到相应的

$x_j^i$ , 所以在此算法中, 直接用  $(a_{j1}^i, a_{j2}^i, \dots, a_{jn}^i)$  来对个体进行编码

实际应用中, 由于原材料的规格很多, 或坯料的型号很多时, 所有的截取方法  $(a_{j1}^i, a_{j2}^i, \dots, a_{jn}^i)$  不可能完全给出, 即使是一小部分也很困难. 因此, 在 VEA 中, 对原材料  $H_i$  的  $h_i$  种截取方法, 只是从中选取  $c_i$  种进行编码 (在后面的叙述中将会发现这样选取是合理的):

$$S = \{ [(a_{11}^1, \dots, a_{1n}^1), \dots, (a_{c_1 1}^1, \dots, a_{c_1 n}^1)], \dots, [(a_{11}^m, \dots, a_{1n}^m), \dots, (a_{c_m 1}^m, \dots, a_{c_m n}^m)] \}$$

其中, 如果  $c_i = 0$ , 那么在编码  $S$  中  $[(a_{11}^i, \dots, a_{1n}^i), \dots, (a_{c_i 1}^i, \dots, a_{c_i n}^i)]$  这个片段就不存在

这样(2) ~ (5) 式也相应地变为:

$$\min Z = \sum_{i=1}^m H_i x_j^i \quad (6)$$

$$x_j^i \leq p_i, (i = 1, 2, \dots, n) \quad (7)$$

$$\sum_{j=1}^{c_i} a_{jk} x_j^i \leq q_k, (k = 1, 2, \dots, n), \quad (8)$$

$$x_j^i \geq 0, x_j^i \text{ 取整数} \quad (9)$$

这里  $a_{jk}^i$  也要满足约束条件(1).

$S$  与相应的解向量  $[(x_1^1, \dots, x_{c_1}^1), \dots, (x_1^m, \dots, x_{c_m}^m)]$  就给出问题的一种下料方案. 对于确定的  $i$ ,  $(a_{j1}^i, \dots, a_{jn}^i)$  给出第  $i$  种材料的被选取的第  $j$  种截取方法, 而片段  $[(a_{11}^i, \dots, a_{1n}^i), \dots, (a_{c_i 1}^i, \dots, a_{c_i n}^i)]$  给出第  $i$  种材料被选取的  $c_i$  种截取方法

这样, 所有可能的编码  $S$  与由  $S$  导出的问题(6) ~ (9) 的解就构成了实际问题的求解空间

### 2.2 适应值函数

在 VEA 中, 适应值函数 eval( $S$ ) 取为(6) 式, 而适应值计算的核心是运用整数规划方法求解问题(6) ~ (9). 由于整数规划问题的计算量比较大, 在 VEA 中采用了一种近似的方法, 即: 先采用单纯形法求解线性规划问题(6) ~ (8), 然后对求得  $x_j^i$  向上取整得到新的  $x_j^i (i = 1, 2, \dots, m; j = 1, 2, \dots, c_i)$ , 然后将新的  $x_j^i$  带到(2) 式中得到的  $Z$  值即为 eval( $S$ ). 由于新的  $x_j^i$  是由原来的  $x_j^i$  向上取整得到的, 这样就

可能存在某个或某几个  $i$ , 使得  $x_j^i > p_i$ . 在 VEA 中, 当碰到这种情况时, 处理如下:

1) 保存  $p_i$  的值;

2)  $\forall i$ , if  $x_j^i > p_i$ ,  $p_i = p_i - 1$ ;

3) 重新求解线性规划问题(6)~(8), 然后对求得的  $x_j^i$  向上取整得到新的  $x_j^i$ ;

4) 如果仍存在  $i$  使得  $\sum_{j=1}^{c_i} x_j^i > p_i$ , 转 2;

5) 恢复原来的  $p_i$  值;

6) 结束

事实上, 由于在约束条件(3)中包含  $m+n$  个线性不等式, 因此用单纯形法求得的  $x_j^i (i=1, \dots, m; j=1, \dots, c_i)$  中最多有  $m+n$  个不为零, 这样对  $x_j^i$  向上取整后得到的  $x_j^i$  中也是最多有  $m+n$  个不为零, 而每个不为零的分量对应着一个被选取的截取方法, 也就是说每种下料方案中最多含有  $m+n$  种截取方法. 所以在编码的时候, 并不需要把所有的截取方法都一一列举出来, 而是给出其中的一部分就可以. 这也是我们从原材料  $H_i$  的  $h_i$  种截取方法中选取  $c_i$  种进行编码的原因.

### 2.3 群体初始化和演化算子

初始化: 在  $[0, \min(6, n)]$  上随机取  $m$  个整数赋给  $c_i (i=1, \dots, m)$ , 对于每个  $i$ , 再随机取  $c_i$  组, 每组  $n$  个小于等于  $[H_i/L_k]$  的非负整数赋给  $(a_{1k}^i, \dots, a_{c_i k}^i) (i=1, \dots, m; j=1, \dots, c_i; k=1, \dots, n)$ . 这样就得到一个个体:

$$S = \{[(a_{11}^1, \dots, a_{1n}^1), \dots, (a_{c_1 1}^1, \dots, a_{c_1 n}^1)], \dots, [(a_{11}^m, \dots, a_{1n}^m), \dots, (a_{c_m 1}^m, \dots, a_{c_m n}^m)]\}$$

对于  $S$  中的  $a_{jk}^i$ , 很可能出现  $\sum_{k=1}^n a_{jk}^i L_k > H_i$  或者  $\sum_{k=1}^n a_{jk}^i L_k$  比  $H_i$  小得比较多的情形. 显然是不允许的; 若  $\sum_{k=1}^n a_{jk}^i L_k$  比  $H_i$  小比较多, 还可以多截出几段坯料来, 以便达到最优值. 对于这两种情形, 必须作一定的修正. 下面是笔者设计的修正算子:

1)  $\forall i (i=1, \dots, m)$ , 若  $\sum_{k=1}^n a_{jk}^i L_k > H_i$ , 则随机选取一些大于 0 的  $a_{jk}^i$ , 置  $a_{jk}^i = a_{jk}^i - 1$ , 直至  $\sum_{k=1}^n a_{jk}^i L_k \leq H_i$  为止

2)  $\forall i (i=1, \dots, m)$ , 若  $\sum_{k=1}^n a_{jk}^i L_k < H_i$ , 则尝试将某个  $a_{jk}^i$  置为  $a_{jk}^i + 1$ , 直到所有的  $a_{jk}^i$  都不能再增大为止

杂交: 随机从种群的其它个体中选择两个个体  $S_1, S_2$  与当前个体  $S_0$  合并起来. 比如:

$$S_0 = \{[(1, 1, 1), (0, 0, 2)], [(1, 1, 2), (5, 0, 1)]\}$$

$$S_1 = \{[(2, 2, 0)], [(3, 2, 0), (1, 1, 1)], [(0, 3, 1)]\}$$

$$S_2 = \{[(1, 1, 1)], [(1, 3, 0)], [(1, 1, 2), (5, 0, 1), (0, 3, 1)]\}$$

合并以后得到:

$$T = \{[(1, 1, 1), (0, 0, 2), (2, 2, 0), (1, 1, 1)], [(3, 2, 0), (1, 1, 1), (1, 3, 0)], [(1, 1, 2), (5, 0, 1), (0, 3, 1)]\}$$

去掉每个片段中相同的部分为

$$T = \{[(1, 1, 1), (0, 0, 2), (2, 2, 0)], [(3, 2, 0), (1, 1, 1), (1, 3, 0)], [(1, 1, 2), (5, 0, 1), (0, 3, 1)]\}$$

求解  $T$  导引的整数规划问题(2)~(4), 假设除  $(0, 0, 2), (2, 2, 0), (3, 2, 0), (1, 1, 2), (5, 0, 1)$  外, 其他的截取方案对应的  $x_j^i$  均为 0, 则杂交以后得到的新的个体为:

$$S = \{[(0, 0, 2), (2, 2, 0)], [(3, 2, 0)], [(1, 1, 2), (5, 0, 1)]\}$$

可以证明  $S$  的适应值不会差于  $S_0, S_1$  和  $S_2$  中的任何一个的适应值

变异: 与初始化的方法类似, 随机产生两个个体  $S_1, S_2$ , 将  $S_1, S_2$  与当前个体  $S_0$  合并起来得到  $T$ . 由  $T$  得到新个体  $S$  的方法与杂交得到新个体的方法完全相同

同样可以证明  $S$  的适应值不会差于  $S_0, S_1$  和  $S_2$  中的任何一个的适应值

分析杂交变异操作可以发现, 个体的长度并不是固定不变的, 这也正是称 VEA 为变长编码的由来

值得说明的是, 如果参与杂交或变异的个体过多, 那么求解线性规划问题(2)~(3)的时间就会急剧增加. 但是如果参与杂交或变异的个体过少, 就会破坏种群的多样性. 另外, 若参数  $p$  的值过大, 则会降低收敛速度; 否则, 又会破坏种群的多样性. 建议如果对解的质量要求不是特别高, 将  $p$  设置为 0.10 左右比较合适.

### 3 计算实例

运用上面设计的算法 VEA, 进行了大量的计算机实验, 所得下料方案的材料利用率均在 97% 以上, 有的甚至达到 99% 以上. 本节选取四个计算实例加以介绍.

**实例 1** 由文献[5] 给出. 原材料长度  $H_1 \sim H_5$  分别为: 320, 340, 360, 380, 400 m, 每种原材料的根数分别为 30, 40, 50, 40, 30 所需坯料长度  $L_1 \sim L_4$  分别为: 35, 52, 71, 97 m, 需求量分别为  $q_1 = 100, q_2 = 80, q_3 = 50, q_4 = 100$

表 1 是用本文设计的算法 VEA 得到的下料方案, 利用率为 98.632%. 表 2 是文献[5] 给出的下料方案, 利用率为 95.479%.

表 1 运用 VEA 求解实例 1 得到的下料方案

所选的原材料		每根原材料 截取的坯料根数			
长度	根数	$L_1$	$L_2$	$L_3$	$L_4$
$H_2$	25	4	0	0	2
$H_2$	15	0	0	2	2
$H_4$	20	0	4	1	1
每种坯料的总根数		100	80	50	100

表 2 文献[5] 给出的下料方案

所选的原材料		每根原材料 截取的坯料根数			
长度	根数	$L_1$	$L_2$	$L_3$	$L_4$
$H_2$	29	1	0	0	3
$H_3$	10	0	0	5	0
$H_3$	15	0	5	0	1
$H_4$	8	9	1	0	0
每种坯料的根数		101	83	50	102

由表 1 可以得到如下的下料方案:

长度为  $H_2$  的原材料选取了两种截取方法, 第一种: 每根原材料截取的坯料  $L_1, L_2, L_3, L_4$  根数分别为 4, 0, 0, 2, 采取这种截取方法的原材料根数为 25; 第二种: 每根原材料截取的坯料  $L_1, L_2, L_3, L_4$  根数分别为 0, 0, 2, 2, 采取这种截取方法的原材料根数为 15. 长度为  $H_4$  的原材料选取了一种截取方法: 每根原材料截取的坯料  $L_1, L_2, L_3, L_4$  根数分别为 0, 4, 1, 1, 采取这种截取方法的原材料根数为 20. 截出的坯料  $L_1, L_2, L_3, L_4$  的根数分别为 100, 80, 50, 100.

**实例 2** 某工厂提供的一个实际问题, 原材料长度  $H_1 \sim H_5$  分别为: 12, 11, 10, 9, 8 m, 每种原材料的根数分别为 2 500, 500, 7 000, 1 000, 1 000 所需坯料长度  $L_1 \sim L_6$  分别为: 6 1, 5 1, 4 3, 3 6, 2 8, 5 7 m, 需求量分别为  $q_1 = 7 500, q_2 = 2 500, q_3 = 2 500, q_4 = 4 500, q_5 = 2 000, q_6 = 3 000$

下料方案见表 3, 利用率为 97.21%.

实例 3 和实例 4 是对实例 2 的扩展

**实例 3** 原材料长度  $H_1 \sim H_5$  分别为: 60, 55,

50, 45, 40 m, 每种原材料的根数分别为 1 200, 500, 1 000, 500, 500 所需坯料长度  $L_1 \sim L_6$  分别为: 6 1, 5 1, 4 3, 3 6, 2 8, 5 7 m, 需求量分别为  $q_1 = 7 500, q_2 = 2 500, q_3 = 2 500, q_4 = 4 500, q_5 = 2 000, q_6 = 3 000$

表 3 运用 VEA 求解实例 2 得到的下料方案

所选的原材料		每根原材料 截取的坯料根数					
长度	根数	$L_1$	$L_2$	$L_3$	$L_4$	$L_5$	$L_6$
$H_1$	250	0	0	0	1	3	0
$H_1$	2 000	1	1	0	0	0	0
$H_2$	500	0	1	0	0	0	1
$H_3$	4 250	1	0	0	1	0	0
每种坯料的根数		7 500	2 500	2 500	4 500	2 000	3 000

下料方案见表 4, 利用率为 99.58%.

表 4 运用 VEA 求解实例 3 得到的下料方案

所选的原材料		每根原材料 截取的坯料根数					
长度	根数	$L_1$	$L_2$	$L_3$	$L_4$	$L_5$	$L_6$
$H_1$	1005	7	0	0	4	1	0
$H_1$	41	3	1	3	7	1	0
$H_1$	22	0	4	0	0	14	0
$H_3$	433	0	0	3	0	1	6
$H_3$	35	0	0	8	0	5	0
$H_4$	403	0	6	2	0	0	1
每种坯料的根数		7 508	2 506	2 508	4 507	2 012	3 001

**实例 4** 原材料长度  $H_1 \sim H_5$  分别为: 120, 110, 100, 90, 80, 每种原材料的根数分别为 400, 50, 50, 1 000, 500. 所需坯料长度  $L_1 \sim L_6$  分别为: 6 1, 5 1, 4 3, 3 6, 2 8, 5 7 m, 需求量分别为  $q_1 = 7 500, q_2 = 2 500, q_3 = 2 500, q_4 = 4 500, q_5 = 2 000, q_6 = 3 000$

下料方案见表 5, 利用率为 99.65%.

表 5 运用 VEA 求解实例 4 得到的下料方案

所选的原材料		每根原材料 截取的坯料根数					
长度	根数	$L_1$	$L_2$	$L_3$	$L_4$	$L_5$	$L_6$
$H_1$	38	0	0	0	13	26	0
$H_1$	209	9	12	0	1	0	0
$H_1$	153	1	0	15	5	3	4
$H_2$	10	1	0	2	0	34	0
$H_3$	48	0	0	4	0	5	12
$H_4$	607	9	0	0	5	0	3
每种坯料的根数		7 507	2 508	2 507	4 503	2 027	3 009

## 4 结 论

通过实验, 可以得出如下几点的结论:

- 1) 算法简单, 易于实现.
- 2) 求解快速, 对材料的利用率很高, 均可达到 97% 以上, 而相关文献[5~ 7]中给出的其他方法最好只能达到 95% 左右; 若只凭人工经验下料, 最多只能达到 80% 左右.
- 3) 提出了一种新颖的遗传算子, 新的个体都是通过线性规划问题求解以后得到的.

## 参考文献:

- [1] Dyckhoff H. A New Linear Programming Approach to the Cutting Stock Problem [J]. *Opns Res*, 1981, **29** (6): 1094-1104
- [2] Sarker B R. An optimum solution for one dimensional slitting problem s: A Dynamic Programming Approach [J]. *J op l Res Soc*, 1988, **39** (8): 749-755
- [3] Mitsuo Gen, Cheng Run-wei. *Genetic Algorithms and Engineering Design* [M]. New York: John Wiley&Sons Inc, 1997.
- [4] PAN Zheng-jun, Kang L i-san, CHENG Yu-ping. *Evolution Algorithms* [M]. Beijing: Qinghua University Press, 1998 (Ch).
- [5] PAN Xiao-yu, LI Hai-yan. Automatic Setting Up Optimum Calculating of One-dimensional Blanking Model [J]. *Journal of Anshan Institute of I & S Technolog*, 1998, **21** (3): 35-38 (Ch).
- [6] FAN Xiao-ying, WEI Wei. Application of Linear Plan for Cutting Out of Raw and Processed Materials [J]. *Journal of Shengyang Architectural and Civil Engineering Institute*, 1998, **14** (2): 154-157 (Ch).
- [7] LU Fei, DENG Zhong-qing, CAI Zheng-jun. The Problems in a Kind of Mathematical Models for Stock Cutting and their Solution [A]. Proc of 11th ICRP [C], New York: 1991. 643-646

# An Evolutionary Algorithm of One Dimensional Stock Cutting Problem Based on Variable Length Coding

LI Yuan-xiang, ZHANG Jin-bo, XU Jing-wen, WANG Lin

(State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, Hubei, China)

**Abstract:** One dimensional stock cutting problem is a typical combinatorial optimization problem, on which many recent research work are concerning. In this paper, an algorithm for the problem coupling the linear programming and the evolutionary algorithm is proposed, which is based on variable length coding and a designed novel genetic operator. Experiments show that our algorithm is easy of implementation and fast to produce satisfactory results. The usage rate of material is up to 97.5% on average and it is convenient to put into practice.

**Key words:** linear programming; integer programming; evolutionary algorithm; stock cutting problem