解批量问题的模拟退火算法[1)]

韩丽敏 王 飞 冯允成
(北京航空航天大学 管理学院)

摘 要 多品种、有能力约束的批量问题是多品种批量生产中生产计划的关键性问题. 这一问题就是要确定: 在每一期生产多少产品以使得所用费用最少且满足已知的每一期的需求量. 本文设计了一种模拟退火算法的实现形式, 通过大量的算例分析表明, 该算法具有良好的寻优特性与运算效率.

关键词 批量; 启发式算法; 网络流; 多品种; 模拟退火算法; 生产计划
分类号 O 22

多品种、有能力约束的动态批量问题[1,2]是多品种批量生产中生产计划的关键性问题. 这一问题就是要确定不同产品在有限的生产时期内的生产量和生产时间, 并且满足每一生产期内产品的需求量, 同时使生产和存贮费用达到最小. 在每一期有生产能力的限制, 准备结束费用和线性成本费用是固定的, 存贮费用与存贮量和存贮时间成正比. 对不同的产品, 上面的费用可能是不同的.

这一问题是一个NP 完全问题. 在 70 年代和 80 年代, 人们提出了一些启发式算法来求解这一问题. Eisenhut 提出了一种向前运行过程(forward pass procedure) 的启发式方法. 对这一算法, Lambrecht 和 Vanderveken 结合一个回朔过程来除去不可行解. Dixon 和 Silver 设计了一种“固定”技术(fix-up) 来指明在每一期结束时生产的最小费用. Dogramaci, Panayiotopoulos 和 Adam 设计了一种可行的向前运行并指定费用的四步骤启发式方法. Thizy 和 Van Wassenhove 运用拉格朗日松弛法对这一问题进行了求解. 对于这一问题还有一些优化算法. 最著名的方法是将能力限制看成约束条件, 基于一般的对偶理论求得一个下界. 最初, Manne 求解这一问题线性规划的松弛; Dzielinski 和Gomory 用 Dantzig-Wolfe 分解算法求解这一问题; Martin 用变量重定义法来解这一问题.

对许多组合优化问题中的 NP 完全问题, 一般的启发式方法虽用很短的时间求得一个可行解, 但解的质量往往不能令人太满意, 而求取最优解的算法往往是运行时间过长. 因而尝试着去寻找更好的启发式算法, 使得在可接受的时间内找到比一般的启发式方法更好的解. 本文设计了一种模拟退火算法的实现形式.

1 问题的描述

单阶层、多品种、有能力约束的批量问题(single level, multi-item capacitated lotsizing problems) 混合整数规划形式可表示如下:

$$\min \sum_{i=1}^{N}\sum_{t=1}^{T}(S_i y_{it} + h_i I_{i,t} + p_{it}x_{it}) \tag{1}$$

满足

$$I_{i,t-1} + x_{it} - d_{it} = I_{i,t}$$
$$i = 1, \cdots, N;\ t = 1, \cdots, T \tag{2}$$

$$\sum_{i=1}^{N} b_i x_{it} \leqslant C_t \qquad t = 1, \cdots, T \tag{3}$$

$$x_{it} \leqslant \left(\sum_{\tau=t}^{T} d_{i\tau}\right) y_{it}$$
$$i = 1, \cdots, N;\ t = 1, \cdots, T \tag{4}$$

$$x_{it} I_{i,t} \geqslant 0 \qquad i = 1, \cdots, N;\ t = 1, \cdots, T \tag{5}$$

$$I_{i,0} \geqslant 0 \qquad i = 1, \cdots, N \tag{6}$$

$$y_{it} \in \{0, 1\} \qquad i = 1, \cdots, N;\ t = 1, \cdots, T \tag{7}$$

其中 N ——生产的产品数; T ——期数; S_i ——第 i 种产品的准备结束费用; h_i ——第 i 种产品的存贮费用; p_{it} ——第 i 种产品在第 t 期的线性生产费用; d_{it} ——第 i 种产品在第 t 期的需求量; b_i ——在第 t 期和第 $t+1$ 期存贮一个单位 i 产品的存贮费用; C_t ——第 t 期的生产能力; x_{it} ——第 i 种产品在第 t 期的生产量; $I_{i,t}$ ——第 i 种产品在第 t 期的存贮量; y_{it} ——二元的准备结束变量.

这一模型假设没有初始存贮(约束(6)), 而且在每一期生产的产品立即被用户所用, 不允许拖

收稿日期: 1998-04-16 第一作者 女 34 岁 博士生 100083 北京
1) 国家自然科学基金(79770008) 和航空科学基金(98J51094) 资助项目

欠. 模型中的约束含意如下: 目标函数(1) 表明要使总费用最小; 方程(2) 是平衡方程; 约束(3) 表示生产量不能超过生产能力; 约束(4) 表明某一期的生产量必须小于或等于后面几期的需求总和; 约束(5) 表示生产量和存贮量的非负性; (7) 表示是否有准备结束费用; 0 表示无, 1 表示有.

单一品种的有能力约束的批量问题的计算复杂性已被证明是 NP 完全问题. 故多品种批量问题也是NP 完全问题. 因此, 在可接受的时间内对大规模批量问题的求解是很困难的, 由此导致人们寻找启发式方法进行求解.

2　批量问题的网络流表示

在批量问题的模型中, 当对所有的 i, t, 固定 y_{it}为 0 或 1 时, 就可通过求解线性规划得到问题的一个解. 因此, 解空间中解的数量为2^{NT}. 特别是当固定 y_{it}时, 这是一特殊的线性规划 ——运输问题. 这一运输问题有 T 种资源和TN 种需求. 运输问题的表示如下:

$$P_y:\ z_y = \min \sum_{i\in N}\sum_{t\in T}\sum_{\tau\in T}\delta_{it\tau}\xi_{it\tau} \qquad (8)$$

$$\sum_{t\in T}\xi_{it\tau} \geqslant b_i d_{i\tau}$$

$$i = 1, \cdots, N;\ t = 1, \cdots, T \qquad (9)$$

$$\sum_{i\in N}\sum_{t\in T}\xi_{it\tau} \leqslant C_t \qquad t = 1, \cdots, T \qquad (10)$$

$$\xi_{it\tau} = 0 \qquad \text{如果 } y_{it} = 0 \text{ 或 } t \geqslant \tau$$

$$i = 1, \cdots, N;\ t, \tau = 1, \cdots, T \qquad (11)$$

$$\xi_{it\tau} \geqslant 0 \qquad i = 1, \cdots, N;\ t, \tau = 1, \cdots, T \qquad (12)$$

其中　对所有 $i\in N$, $t\in T$, $\tau\in T$,

$$\delta_{it\tau} = \begin{cases} \left(p_{it} + \sum_{q=t}^{\tau-1} h_{iq}\right) \backslash b_i & \text{如果 } t \leqslant \tau \\ \infty & \text{其它} \end{cases}$$

$\xi_{it\tau}$是在第T 期的生产量, 这一生产量要满足在 τ 期的第 i 产品的需求.

下面用一个例子来说明如何将批量问题转化成运输问题来求解. 表 1(3 个品种 4 个生产期) 列出了每一产品在每一生产期的需求量和每一期的生产能力. 表 2 中的数字表示为: 第 1 列为生产某一单位产品所需的生产能力(时间) ; 第 2 列为生产某一产品的准备结束费用; 第 3 列为某一产品在某一期的存贮费用, 本例中没有线性的生产费用. 假设表 1 中带有横线的位置表示在这一期生产已给定的产品, 也就是说在这一期, y_{it}= 1, 否则 y_{it}= 0, 那么所对应的运输问题可写为如表 3 中的标准形式. 值得注意的是弧(t, $i\tau$) 可能被禁止, 一方面由于解的选择的限制 y_{it}= 0, 以× 来表示; 另一方面由于不允许拖欠, 即 $\tau < t$, 以* 来表示. 同时运输问题亦可表为最小费用网络流的问题.

表 **1**　每一期中每一产品的需求量和生产能力

品种	t= 1	t= 2	t= 3	t= 4
i= 1	20	30	40	10
i= 2	20	10	10	10
i= 3	25	30	30	30
能力限制	450	400	450	300

表 **2**　各种参数值

品种	能力吸收 a_i	准备结束费用 s_i	存贮费用 h_i
i= 1	5	70	3
i= 2	4	90	4
i= 3	6	200	5

表 **3**　表示成运输问题的费用表

t	i= 1				i= 2				i= 3				虚拟需求	能力限制
	τ= 1	τ= 2	τ= 3	τ= 4	τ= 1	τ= 2	τ= 3	τ= 4	τ= 1	τ= 2	τ= 3	τ= 4		
1	0	0.6	1.2	1.8	0	1	2	3	0	5/6	5/3	5/2	0	450
2	*	0	0.6	1.2	*	×	×	×	*	0	5/6	5/3	0	400
3	*	*	0	0.6	*	*	0	1	*	*	0	5/6	0	450
4	*	*	*	×	*	*	*	×	*	*	*	0	0	300
能力需求	100	150	200	50	80	40	40	40	150	180	180	210	210	

对批量问题的求解时间长短取决于对这一运输问题的求解时间. 本文采用目前速度较快的最小费用网络流的松弛方法[3,4] 求解这一问题.

3　邻域结构的确定及解的产生机制

在局部搜索算法的迭代过程中, 瞬时解不断地在解空间中从一个解变换到另一个解. 因此, 邻域规模和解的产生机制对逐次迭代的局部搜索算法是十分重要的. 有能力约束的批量问题的一个解自然地定义为如下的集合: $\{y_{it}, i = 1, \cdots, N;\ t = 1, \cdots, T\}$. 如果在第 t 期生产第 i 种产品, 那么 y_{it}= 1, 否则 y_{it}= 0. 令 I 是对应于第 t 期生产第 i 种产品的(i, t) 指标集合(即 y_{it}= 1) , I' 是它的余集. 因此, 解的变换可定义为将(i, t) 指标从 I 变换到I' 或者从I' 变换到I. 所对应的邻域是通过解的变换可达的可行解. 但是, 这样的邻域太大了.

特别是对一些大规模的实际问题以至于使大规模搜索变得不可能. 为此, 对邻域的定义加以限制. 对邻域的选取采用了 2 种方法. 在每一次迭代中, 找出那些生产量小于需求量(即 $0< x_{it}< d_{it}$) 的那些(i, t) 指标作为其邻域结构, 变换 1 如下: 随机选取一个这样的(i, t) 指标, 如果 $y_{it}= 0$, 则置 $y_{it}= 1$; 如果 $y_{it}= 1$, 则置 $y_{it}= 0$. 变换 2 如下: 在每一次迭代中, 找出那些生产量小于需求量(即 $0< x_{it}< d_{it}$) 的那些(i, t) 指标, 随机选取一个这样的(i, t) 指标, 如果 $y_{it}= 0$, 则置 $y_{it}= 1$, 同时对($i, t- 1$) 或($i, t+ 1$) 也进行相同的变换. 这样做的原因是基于一个观察, 在一个比较好的计划安排中生产量小于需求量的期可能产生的较少. 如果在某一期有准备结束费用的话, 那么它的前一期可能没有准备结束费用.

下面用一个例子来说明如何进行解的变换. 表4 给出了 y_{it} 的值, 将 y_{it} 代入上面的运输问题中, 求解这一运输问题可得对应的批量问题的解, 即求得在每一期的生产量, 设所得解为表5 所示. 找出那些生产量小于需求量(即 $0< x_{it}< d_{it}$) 的那些(i, t) 指标, 此例中为(i, t) = (1, 4), (2, 2), (2, 4) (如 $x_{22}= 0< d_{22}= 10$), 则邻域有 3 个元素. 随机选取这一邻域中的一个(i, t) 指标, 例如为(2, 2), 对变换 1, 若 $y_{22}= 0$, 则令 $y_{22}= 1$, 反之则相反. 对变换 2, 若 $y_{22}= 0$, 则令 $y_{22}= 1$, 反之则相反. 同时进行如下的变换: 若它的后一期是最后一期, 则进行变换, 本例中置 $y_{23}= 0$(因为 $y_{23}= 1$) , 否则则相反. 若它的后一期是最后一期, 则对它前面的一期进行相同的变换.

表 **4** 一个解的表示

品种	$t= 1$	$t= 2$	$t= 3$	$t= 4$
$i= 1$	1	1	1	0
$i= 2$	1	0	1	0
$i= 3$	1	1	1	1

表 **5** 每一期的生产量

品种	$t= 1$	$t= 2$	$t= 3$	$t= 4$
$i= 1$	20	30	50	0
$i= 2$	30	0	20	0
$i= 3$	25	30	30	30

4 模拟退火算法

1982 年, Kirkpatrick 等将退火思想[5,6] 引入组合优化领域, 提出一种解大规模组合优化问题的近似算法——模拟退火算法. 它源于对固体退火过程的模拟, 采用 Metropolis 接受准则, 并用一种称为冷却进度表的参数控制算法进程, 使算法在多项式时间里给出一个近似最优解. 假定存在邻域结构和产生器, 再设 t_k 表示 Metropolis 算法第 k 次迭代时控制参数值, L_k 表示 Metropolis 算法第 k 次迭代时产生的变换个数, 则模拟退火算法的设计如下:

1) 确定初始解变量 n_0, 这一解变量是由启发式算法求得的, 并计算其目标函数值 f_0. 令 $n= n_0, f= f_0$, 设定模拟退火算法的初始温度 $T= T_0$.

2) 在当前解的邻域结构 1(或邻域结构 2) 中随机产生一个新解 n_1, 计算新的目标函数值 f_1 和目标函数的差值 $\Delta f= f_0- f_1$.

3) 若 $\Delta f< 0$, 接受当前解; 否则, 若 $\Delta f> 0$, 则当 $\exp(- \Delta f/T)$ 大于[0, 1) 间的随机数时, 接受当前解; 否则, 原解仍为当前解. 当接受解时, 将 f_0 与 f_1 进行比较, 如果 $f_1< f_0$, 则令 $n= n_1, f= f_1$.

4) 如果尚未达到模拟退火温度的热平衡条件, 转到 2), 否则继续下一步.

5) 确定模拟退火的衰减温度 $T'= \beta T'$, β 为衰减函数.

6) 如果不满足停止判据, 转到 2); 如果满足停止判据, 停止计算, 输出问题的优化解.

在实际计算中, 模拟退火算法中的参数选取如下: 初始温度 $T_0= 5$; 若品种数大于 100, 马尔可夫链长 $L_k= 100$; 若品种数小于 100, $L_k=$ 品种数 × 期数; 模拟退火的衰减系数可以取 $\beta= 0.9$, 停止准则选为在某一马尔可夫链内解无变化. 为了节省运行算法的搜索时间, 初始解用一启发式方法求得, 这一启发式方法为 Eisenhut 启发式方法的扩展形式, 由 Lambrecht 和 Vanderveken 在文献[7] 中给出. 这也是算法的初始温度可以选择得较小的原因.

5 实验结果

14 个数据被用来对算法的优劣进行测试(表6) .“启发式方法” 下的解是用启发式方法求得的解. 模拟退火算法分别对邻域结构 1 和结构 2 运行 3 次求得它们的平均解值和最好解值. 算法是在 166MHz, 80586PC 兼容机上运行的. 从以上实验结果可以得出如下结论: 使用邻域结构 2 比邻域结构 1 求得的解的质量更好一些, 而邻域结构 1 是多数文献所采用的. 运用文中的启发式方法可使求解大规模的批量问题在 PC 机上成为可能, 并获得比一般启发式方法更好的解.

表 6　几组数据的测试结果

名称	品种数量	期限数量	启发式方法	模拟退火算法(邻域结构 1)			模拟退火算法(邻域结构 2)		
			解值	最好解	平均解	时间/s	最好解	平均解	时间/s
TVW1	8	8	8 760.00	8 480.00	8 480.00	2	8 480.00	8 480.00	2
TVW2	8	8	8 780.00	8 250.88	8 360.00	3	8 230.00	8 370.33	2
TVW3	8	8	8 320.00	7 720.00	7 720.20	4	7 720.00	7 720.00	5
TVW4	8	8	8 470.00	7 830.00	7 840.00	4	7 800.00	7 833.33	4
TVW50	50	8	52 878.00	52 640.64	52 742.87	13	51 715.00	51 825.60	15
TVW100	100	8	99 894.00	99 666.70	99 688.90	48	98 803.34	98 900.06	120
TVW150	150	8	146 258.80	145 212.30	145 372.60	48	143 325.10	143 813.93	152
TVW200	200	8	197 619.31	197 390.00	197 435.90	45	195 390.00	195 451.06	122
DS115L	100	10	21 269.60	207 745.80	207 808.20	45	20 689.08	20 704.20	111
DS110L	100	10	21 289.99	207 550.40	207 984.30	130	20 743.95	20 781.33	73
DS105L	100	10	21 425.79	209 170.60	209 298.70	130	20 869.66	20 901.53	103
DS100M	100	10	21 893.86	212 004.60	212 060.20	132	21 127.27	21 164.09	130
DS199T	100	10	22 021.91	212 960.00	213 153.00	140	21 269.28	21 290.03	120
DS200M	200	10	45 655.09	43 936.40	43 990.96	250	43 934.16	44 033.85	243

注: 解值——批量问题的最小费用; 平均解——算法运行 3 次所得的平均解; 最好解——算法运行 3 次所得的最好解.

6　结论和进一步要做的工作

本文对用模拟退火算法求解单阶层、多品种、有能力约束的批量问题进行了一些初步探讨, 通过大量的实验表明, 应用这一算法能大大提高求解效率, 而且能够在很大程度上保证解的质量. 虽然算法本身在求解组合优化问题上显示出了较强的寻优能力, 但算法还有不完善的地方. 今后还将进一步研究.

参　考　文　献

1　Eppen G D, Martin R K. Solving multi-item capacitated lot-sizing problems using variable redefinition. Operations Research, 1987, 35(6): 832~848

2　Hindi K S. Solving the CLSP by a tabu search heuristic. J Opl Res Soc, 1996, 47(1): 151~161

3　Eisenhut P S. A dynamic lot sizing algorithm with capacity constraints. AIIE Transactions, 1975, 7(2): 170~176

4　Bertsekas D P, Tseng P. Relaxation methods for minimum cost ordinary and generalized network flow problems. Operations Research, 1988, 36(1): 93~114

5　Glove F, Laguna M. Modern heuristic techniques for combinatorial problems. Oxford: Blackwell Scientific Publishing, 1993

6　刘　勇, 康立山, 陈毓屏. 非数值并行算法——模拟退火算法. 北京: 科学出版社, 1995

7　Lambrecht M R, Vanderveken H. Heuristic procedures for the single operation, multi-item loading problem. AIIE Transactions, 1979, 11(4): 319~326

Simulated Annealing Heuristic for Lot-Sizing Problem

Han Limin　Wang Fei　Feng Yuncheng

(Beijing University of Aeronautics and Astronautics, School of Management)

Abstract　The multi-item capacitated lot-sizing problem consists of determining the magnitude and the timing of some operations of durable results for several items in a finite number of processing periods so as to satisfy a known demand in each period. An approximation algorithm is described, which is based on simulated annealing algorithm. The method incorporates a primal partitioning scheme —with a network flow subproblem —to obtain good feasible solutions.

Key words　lot size; heuristic approach; network flows; multi-item; simulated annealing; production planning