

文章编号:1001-9081(2010)02-0490-05

基于供应商选择的大规模网络供应能力研究

蒋霖云^{1,2}, 陈虎²

(1. 西南交通大学 物流学院, 成都 610031; 2. 攀枝花学院 经济与管理学院, 四川 攀枝花 617000)

(jjyjsx@126.com)

摘要:在产品供不应求的市场环境下,为了最大限度地满足市场需求,核心企业需要对供应链中的供应商组合进行选择,实现整个供应链网络的最大供应能力。为了有效地解决组合方案规模庞大、计算困难的问题,采用了嵌套的混合遗传算法对此问题进行优化,提出了一种计算大规模网络最大流的新算法。算例结果显示,该方法能够正确选择供应商组合,并求解网络的最大供应能力。

关键词:大规模供应链网络;供应能力;供应商选择;混合遗传算法

中图分类号: TP391; TP301.6 **文献标志码:** A

Supply capability of large-scale supply network chain based on vendor-selection

JIANG Ji-yun^{1,2}, CHEN Hu²

(1. College of Logistics, Southwest Jiaotong University, Chengdu Sichuan 610031, China;

2. School of Economics and Management, Panzhihua University, Panzhihua Sichuan 617000, China)

Abstract: Under the circumstance that the supply falls behind the demand, in order to satisfy market demand as much as possible, the core enterprises need vendor-selection in the supply chain to achieve maximum supplying capacity for the entire supply chain network. In order to solve the large-scale portfolio alternatives and the difficult calculation effectively, a nested hybrid genetic algorithm was used to optimize this issue. Meanwhile a new algorithm was presented for solving large-scale network maximum flow. The simulation results show that this method draws the right vendor-selection, and figures out the maximum supplying capacity for the network.

Key words: large-scale network supply chain; supply capability; vendor-selection; hybrid genetic algorithm

0 引言

随着经济全球化、信息技术与通信技术的进步、全球采购的实施、虚拟企业的快速建立与分解、更多的非核心业务外包等,供应链的形式变得越来越复杂并迅速向大规模甚至是超大规模网络化的形式演变。国际上很多著名的跨国公司合作伙伴为数众多,遍布全球,其供应链中的企业达到数万个。作为大规模供应链网络中的核心企业,由于为市场提供产品的能力不仅受到自身的影响,还要受到合作伙伴的供应能力的影响,因此其市场最大供应能力取决于整个供应链网络的供应能力。同时,为了预防可能发生的风险,核心企业往往需要了解供应链中的薄弱环节,以便预先采取预防措施。这些问题,实质上可以转化为关于供应能力的供应链网络最大流问题。目前,从复杂网络的角度研究供应链的论文并不很多,其中文献[1]提出了一个局部世界网络模型用来描述供应链;美国学者 Nagurney 等人提出的超级网络模型可以很好地描述供应链中的物流活动^[2],通过求解网络系统的均衡点研究系统的行为;以 Helbing 为首的一批物理学家在采用复杂网络理论研究供应链网络^[3-5]。本文在供应商可选择的情况下研究了大规模供应链网络的最大供应能力问题。

1 问题描述

考虑构建一个供应链网络如图1所示,节点之间的配套关系中确定的,供应链网络节点上有多家供应商可供选择,由

于每家供应商在设备、人员、管理水平以及客户在其供应关系中的地位等方面是不同的,因此不同的供应商向相同的客户提供的产品供应能力是不同的。如果市场处于成长期,产品供不应求,核心企业为了满足市场的需要,必须尽可能地提高对市场的供应能力,这就需要在现有情况下通过选择合适的供应商来达到这个目的。

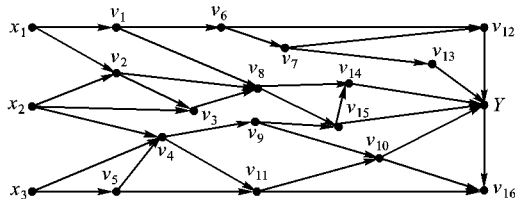


图1 基于供应商选择的大规模供应链网络

这样,基于最大供应能力的供应商选择问题实际上构成了一个双重优化问题。首先,选择最优的供应商组合,使其组合后的最大流达到最优,这是供应商的优化问题;其次,针对特定的供应商组合,找到该网络的最大流量(即最大供应能力),这是供应能力的优化问题。对于前一个问题,由于节点众多,组合方案随着变量(节点)的增加而呈指数增长,用常规的方法求解很困难。譬如,假设有15个供应链网络节点,每个节点可选择的供应商有3个,那么可供选择的供应商方案的组合数量应有 3^{15} 种,亦即,为了求得理想的供应商组合方案,需要求解14 348 907个供应链网络的最大流。这样,优化选择需要很长的运行时间,这个时间甚至是人们无法忍受

收稿日期:2009-08-25;修回日期:2009-09-25。

作者简介:蒋霖云(1973-),男,四川营山人,讲师,博士研究生,主要研究方向:优化算法、供应链物流管理; 陈虎(1963-),男,四川简阳人,博士研究生,主要研究方向:供应链物流管理、物流仿真。

的。因此,为了解决可供选择方案,本文采用遗传算法来进行优化。至于后一个问题,属于大规模网络最大流问题,采用常规方法求解效率将非常低下。因此,本文提出了一种嵌套的混合遗传算法解决整个问题的优化。

2 模型假设

为了简化讨论,在不影响问题性质的前提下,本模型作以下假设:

1) 核心企业处于强势地位,具有选择供应链网络节点企业的权利,其供应商没有选择前向供应商的权利。

2) 核心企业选择供应商的依据是唯一的,即对供应链最大供应能力的贡献大小,而质量、成本等因素不在考虑范围内。比如处于产品供不应求的核心企业就存在这种情况。

3) 供应链网络结构固定不变,即节点之间的供应关系是固定的,不随节点上供应商的不同而不同。

4) 对于供应链相同节点上的可选择企业而言,其产品是同质的,交付时间是相同的,不同的只是供应能力的大小。

5) 供应链的每一个网络节点(除核心企业),能够保证按时供应配套的零部件,因此同一节点不必选择两家供应商。

3 优化模型

基于供应商选择的大规模网络的最大供应能力优化问题,实际上分两步进行决策:首先完成供应商的选择,即完成供应链网络构成的决策;然后,需要在给定供应链网络的基础上,实现供应量的合理分配,从而实现整个供应链网络的最大供应能力。

对于供应链网络 $G = (X, M, Y, A)$, $V = X + M + Y$ 表示网络的顶点集。其中: X 为网络 G 的出发点集(供应链源点); Y 为网络的收点集(供应链核心企业); M 为网络的中间点集,代表供应链网络中的节点企业; A 为网络 G 的弧集,表示节点企业为客户提供产品的供应能力。

在网络中, $X + M$ 为供应链的供应商,每一个顶点都有若干可供选择的企业,假设某发点共有 n 个企业可供选择,表示发点集中顶点 x_i 的某个可选企业用 $x[k]_i$ 表示($k = 1, 2, \dots, n$)。

供应链网络中,若弧 $(v_i, v_j) \in A$,则表示节点企业 v_i 为 v_j 提供配套产品,若弧 $(v_i, v_j) \notin A$,则企业 v_i 为 v_j 没有合作关系。

因此,上述问题就是求关于供应量分配的一个可行流 $\{f_{ij}\}$,使其流量达到最大:

$$\max v(f) = \left| \sum f_{ij} - \sum f_{ji} \right|; v_i \in Y, v_j \in V \quad (1)$$

$$\text{s. t. } c_{ij} \in (v[k]_i, v[k]_j); k = 1, 2, \dots, n \quad (2)$$

$$0 \leq f_{ij} \leq c_{ij}; (v_i, v_j) \in A \quad (3)$$

$$\sum f_{xj} - \sum f_{jx} = v(f); v_x \in X, v_j \in V \quad (4)$$

$$\sum f_{ij} - \sum f_{ji} = 0; v_i, v_j \in M \quad (5)$$

$$\sum f_{ij} - \sum f_{ji} = -v(f); v_i \in Y, v_j \in V \quad (6)$$

上述模型中,式(1)表示优化目标函数,使所有收点的净流入量即整个网络的输出最大,从而实现供应链网络的最大供应能力。式(2)为节点企业约束,即同一节点相应弧的容量必须是同一可选择企业的供应能力。式(3)为网络容量限制条件,即对于每一弧上的流量 f_{ij} ,它小于选定企业的相应弧 $(v[k]_i, v[k]_j)$ 的容量。式(4)~(6)为流量平衡条件,式(4)表示所有发点的净流出量的和等于可行流;式(5)表示

所有收点的净流入量等于可行流的相反数;式(6)表示任一中间节点的流入量等于流出量。

4 优化算法

基于供应商选择的大规模网络供应能力优化问题,由于其组合方案数量巨大,不宜使用传统方法求解。而遗传算法是基于“优胜劣汰”的全局优化智能算法,非常适合于解决高维决策问题^[6]。因此,本文采用遗传算法来解决优化问题,同时,考虑到大规模网络的特点,在计算遗传算法的适应度函数中再次嵌入遗传算法,从而形成了求解上述模型的混合遗传算法。实现步骤如下^[7-8]:

1) 生成初始种群。

采用二进制编码方式,每条染色体可表示为图2所示的代码串,每条染色体对应一种供应商选择方案。供应链网络每一个节点的企业选择用若干个基因位表示,基因位长度取决于相应节点可选择企业的个数,譬如节点 x_i 有4家可选择企业,则选择方案可用2个基因位表示,基因值“00”表示企业A被选中,“01”表示企业B被选中,“10”表示企业C被选中,“11”表示企业D被选中。然后,随机产生一定数量的染色体作为遗传进化的初始种群。

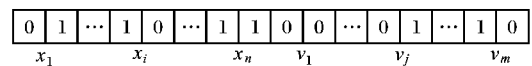


图2 编码方式

2) 适值计算。

因为每条染色体对应一种固定的供应链网络,所以其目标函数即为相应网络的最大供应能力。鉴于所研究的网络是大规模网络,用常规的最大流算法计算繁杂、效率低,因此采用基于遗传算法的最大流最小截集的计算方法来求解网络的最大供应能力,具体方法见第5章。

3) 适值排序。

根据适值采用等差为2的线性排序方法计算出各条染色体的适应度。

4) 选择。

利用轮盘赌算法和代沟技术形成下一代种群中的部分染色体。为了保证算法的收敛性,同时采用最佳保留策略,把当前代中适应度最好的个体保留到下一代群体,使其不被交叉变异算法破坏掉。

5) 交叉。

以交叉概率随机在新一代种群中选择两条染色体进行交叉,并采用两点法随机确定交叉位置。

6) 变异。

采用传统的基本位变异,即对个体编码串中以变异概率 p_m 随机指定某一位基因座上的基因值作变异运算(若原有基因值为0,则变异操作将该基因值变为1;反之,若原有基因值为1,则变异操作将该基因值变为0)。

7) 重插入。

根据代沟设置比例采用基于适应度选择的重插入。

8) 循环或终止遗传操作。

循环迭代步骤2)~7)以便寻求相对于整个系统的最优决策。如果种群中的最优适值在规定的进化代数内没有变动或迭代到了最大允许进化代数,则停止迭代,此时种群中的最优适值就是实际问题的最优值,最优适值对应的染色体就是实际问题的供应商选择方案。

5 容量确定情况下供应链网络最大流算法

5.1 算法的基本思想

对于供应链网络 $G = (X, M, Y, A)$, 以顶点集 $\{X, W\} = \{x_1, x_2, \dots, x_p; v_1, v_2, \dots, v_m\}$ 为行, 以顶点集 $\{W, Y\} = \{v_1, v_2, \dots, v_m; y_1, y_2, \dots, y_q\}$ 为列, 作供应链网络 G 的容量矩阵 $C(G)^{[9]}$:

容量矩阵中, c_{ij} 表示弧 (v_i, v_j) 的供应能力 ($c_{ij} \neq 0$), 若 v_i 和 v_j 之间没有供应关系, 则 $c_{ij} = 0$ 。

$C(G) =$

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1m} & c_{1,m+1} & c_{1,m+2} & \cdots & c_{1,m+q} \\ c_{21} & c_{22} & \cdots & c_{2m} & c_{2,m+1} & c_{2,m+2} & \cdots & c_{2,m+q} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ c_{p1} & c_{p2} & \cdots & c_{pm} & c_{p,m+1} & c_{p,m+2} & \cdots & c_{p,m+q} \\ c_{p+1,1} & c_{p+1,2} & \cdots & c_{p+1,m} & c_{p+1,m+1} & c_{p+1,m+2} & \cdots & c_{p+1,m+q} \\ c_{p+2,1} & c_{p+2,2} & \cdots & c_{p+2,m} & c_{p+2,m+1} & c_{p+2,m+2} & \cdots & c_{p+2,m+q} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ c_{p+m,1} & c_{p+m,2} & \cdots & c_{p+m,m} & c_{p+m,m+1} & c_{p+m,m+2} & \cdots & c_{p+m,m+q} \end{bmatrix} \begin{matrix} x_1 \\ x_2 \\ \vdots \\ x_p \\ v_1 \\ v_2 \\ \vdots \\ v_m \end{matrix}$$

按照网络截集的定义, 若顶点集 V 被剖分为两个非空集合 V_s, V_t , 使得 $X \subseteq V_s, Y \subseteq V_t$, 那么弧集 (V_s, V_t) 称为是 (分离 V_s 和 V_t) 的一个截集。

建立截集矩阵 K_t , 其建立过程是: 在容量矩阵 $C(G)$ 中以截集起点集 V_s 所对应的行为截集矩阵 K_t 的行, 在 $C(G)$ 中以截集终点集 V_t 所对应的列为截集矩阵 K_t 的列。假设 V_s 等于网络发点集 X 及网络中任意 t 个中间点 ($t = 0, 1, 2, \dots, m$), 则有截集矩阵:

$$K_t = (c_{ij})_{(p+t) \times (q+m-t)}$$

截集矩阵 K_t 的容量为:

$$C_t = \sum_{j=1}^{q+m-t} \sum_{i=1}^{p+t} c_{ij}$$

根据最大流量最小截量定理, 网络 G 中, 从发点集 X 到收点集 Y 的最大流的流量等于分离 X, Y 的最小截集的容量, 因此可得到最大供应能力 f^* :

$$f^* = \min C_t = \min \left(\sum_{j=1}^{q+m-t} \sum_{i=1}^{p+t} c_{ij} \right)$$

根据以上论述, 通过“最大流最小截”定理将供应链网络最大供应能力问题转化为顶点集的组优化问题, 即随着顶点组合的改变而截集容量不断改变的组优化问题。随着网络规模的扩大, 顶点优化组合的搜索空间也急剧扩大, 用枚举法或其他算法很难精确求出其最优解。这时候, 可以使用遗传算法求解顶点的优化组合, 使其分割得到的截集容量最小。在此问题的遗传算法中, 以中间顶点为染色体的基因位, 采用二进制编码, 以截集容量为适应度评价函数, 从而实现大规模供应链网络的最大供应能力求解。

本问题是一个典型的组合寻优问题, 用常规的遗传算法就可解决问题。中间点集中的每一个顶点都可以成为或不成为截集起点集中的顶点, 规避了传统网络优化问题约束条件的限制, 因此染色体不存在失效的问题, 可直接编码计算。关键的地方是适应度评价函数, 基本思想中的截集矩阵容量并不能直接作为适应度评价函数, 因为截集矩阵是不断变化的, 因此需要进一步转化, 将适应度函数建立在容量矩阵上。

5.2 编码

采用二进制编码方式, 每个染色体可表示为图3所示的

代码串, 表示中间点的选择方案, 染色体中的基因为 $v_1, v_2, \dots, v_i, \dots, v_m$ 表示该备选顶点的状态, 1 表示被选中, 与网络的发点集 X 一起成为截集起点集 V_s 中的顶点, 0 表示未被选中, 而与网络的收点集 Y 一起成为截集终点集 V_t 中的顶点。

1	0	...	0	...	1
v_1	v_2		v_i		v_m

图3 编码方式

5.3 适应度函数

根据前面的论述, 目标评价函数也就是截集矩阵 K_t 的容量, 即为:

$$f = C_t = \sum_{j=1}^{q+m-t} \sum_{i=1}^{p+t} c_{ij}$$

但这里的 c_{ij} 是截集矩阵的元素, 每取一次截集矩阵, c_{ij} 随之发生变化, 为便于计算机编码计算, 需要对上式进行变形, 用染色体编码值和容量矩阵 C 的元素值来表示。变换过程如下:

$$\begin{cases} f = C_t = \sum_{j=1}^{q+m-t} \sum_{i=1}^{p+t} c_{ij} = \sum_{i=1}^{p+t} c_{i,1} + \sum_{i=1}^{p+t} c_{i,2} + \cdots + \sum_{i=1}^{p+t} c_{i,m-t} + \sum_{i=1}^{p+t} c_{i,m-t+1} + \cdots + \sum_{i=1}^{p+t} c_{i,m-t+q} \\ \sum_{i=1}^{p+t} c_{ij} = 1 \times \sum_{i=1}^{p+t} C_{ij} + 0 \times \sum_{i=1}^{p+t} C_{ij}; \\ j = 1, 2, \dots, m-t+q \end{cases} \quad (7)$$

式(7)中, 前半部分 $\sum_{i=1}^{p+t} C_{ij}$ 表示截集起点集所对应的容量

矩阵取值, 后半部分 $\sum_{i=1}^{p+t} C_{ij}$ 表示 W 为中间点集中属于截集终点集的点所对应的容量矩阵取值, 用 $v_1, v_2, \dots, v_i, \dots, v_m$ 表示中间顶点的染色体基因值 1 或 0; 用 $v_{m+1}, v_{m+2}, \dots, v_{m+p}$ 表示发点集 X 的染色体基因值, 其值恒为 1, 用向量 $VS_{(p+m) \times 1} = (v_1, v_2, \dots, v_{p+m})^T$ 表示, 则式(7)变为:

$$\sum_{i=1}^{p+t} c_{ij} = \sum_{i=1}^{p+m} v_i C_{ij}; j = 1, 2, \dots, m-t+q \quad (8)$$

$v_1', v_2', \dots, v_j', \dots, v_m'$ 表示中间顶点的染色体基因值 1 或 0; $v_{m+1}', v_{m+2}', \dots, v_{m+q}'$ 表示收点集 Y 的染色体基因值, 其值恒为 0, 用向量 $VT_{1 \times (m+q)} = (v_1', v_2', \dots, v_{m+q}')^T$ 表示。则有:

$$\begin{aligned} f &= \sum_{j=1}^{q+m-t} \sum_{i=1}^{p+t} c_{ij} = \sum_{i=1}^{p+t} c_{i,1} + \sum_{i=1}^{p+t} c_{i,2} + \cdots + \sum_{i=1}^{p+t} c_{i,m-t} + \sum_{i=1}^{p+t} c_{i,m-t+1} + \cdots + \sum_{i=1}^{p+t} c_{i,m-t+q} = \\ &= (1 - v_1') \sum_{i=1}^{p+t} c_{i,1} + (1 - v_2') \sum_{i=1}^{p+t} c_{i,2} + \cdots + (1 - v_m') \sum_{i=1}^{p+t} c_{i,m-t} + (1 - v_{m-t+1}') \times \sum_{i=1}^{p+t} c_{i,m-t+1} + \cdots + \\ &= (1 - v_m') \sum_{i=1}^{p+t} c_{i,m} + (1 - v_{m+1}') \sum_{i=1}^{p+t} c_{i,m+1} + \cdots + (1 - v_{m+2}') \sum_{i=1}^{p+t} c_{i,m+2} + \cdots + \\ &= (1 - v_{m+q}') \times \sum_{i=1}^{p+t} c_{i,m+q} \end{aligned} \quad (9)$$

由于截集矩阵中没取到 v_k 列 (即 W 为中间点集中属于截集起点集的点所对应的列, 共有 t 列) 所对应的 $(1 - v_k)$ 为零, 故可将 $\sum_{i=1}^{p+t} c_{ij} = \sum_{i=1}^{p+m} v_i C_{ij} (j = 1, 2, \dots, m-t+q)$ 变为 $\sum_{i=1}^{p+t} c_{ij} = \sum_{i=1}^{p+m} v_i C_{ij} (j = 1, 2, \dots, m+q)$ 代入式(9), 得到:

$$f = \sum_{j=1}^{m+q} (1 - v_j') \sum_{i=1}^{p+m} v_i C_{ij} = \sum_{j=1}^{m+q} \sum_{i=1}^{p+m} (1 - v_j') v_i C_{ij}$$

为方便计算,将上式进一步化简,定义矩阵 $S_{(p+m) \times (m+q)}$ 矩阵 $T_{(p+m) \times (m+q)}$ 为:

$$S_{n,i} = S_{n,j} = VS_{n,1};$$

$$n = 1, 2, \dots, p + m; i, j = 1, 2, \dots, m + q$$

$$T_{i,n} = T_{j,n} = VT_{1,n};$$

$$n = 1, 2, \dots, m + q; i, j = 1, 2, \dots, p + m$$

定义矩阵相乘符号 \otimes 为: $D = A \otimes B$, 若 $D_{ij} = A_{ij} \times B_{ij}$ 。

令 $W = S \otimes (I - T) \otimes C(G)$, 其中 I 为单位矩阵, 则目标函数可表示为:

$$f = \sum_i \sum_j W_{ij}$$

由于最大流为最小截集的容量, 故目标函数的值越小, 适应度就越高, 则令适应度函数为:

$$F = C - f; C \text{ 为一适当的常数}$$

5.4 遗传操作

容量确定情况下供应链网络最大供应能力遗传算法的其他步骤, 如交叉、变异等, 和第4章的操作一致。

6 算例

设某以制造商为核心企业的大规模供应链网络如图1所示, x_1, x_2, x_3 是供应链的原料供应商, 也是供应链网络的发点集; Y 是供应链核心企业(制造商), 即供应链网络的收点; v_1, v_2, \dots, v_{16} 为网络的中间节点, 共分三层供应商结构: 第一层供应商为 $v_{12}, v_{13}, \dots, v_{16}$, 第二层供应商为 v_6, v_7, \dots, v_{11} , 第三层供应商为 v_1, v_2, \dots, v_5 。后一层的企业为前一层的企业配套, 但同时也可能为同层的企业配套, 如第三层的企业 v_2 就为同层的企业 v_3 提供配套服务。弧上的数字 c_{ij} 为弧的容量, 代表后向供应商 v_i 为前向企业 v_j 提供零部件的最大供应能力。设每个节点可选择的企业各有4家, 其向客户的供应能力(即弧容量)分布见表1。现需知道网络的最大供应能力, 并发现整个供应链网络的瓶颈环节, 以便有针对性的采取措施。

表1中, 对应不同的节点, $ABCD$ 代表的供应商是不同的, 如 x_1 节点的 A 供应商与 x_2 节点的 A 供应商是不同的企业。

从图1中可以看出, 节点 $v_3, v_{12}, v_{13}, v_{14}, v_{16}$ 各自只向一家下游企业服务, 因此只需选择供应能力最大的供应商即可。即使这样, 需要作出选择的供应商节点仍有14个, 可供选择的供应商组合方案也达到了 $4^{14} = 268435456$ 个, 显然用常规的优化方法无法解决计算问题。同时, 供应链网络属于大规模网络, 通用面临计算复杂的问题。在这种情况下, 我们采用混合遗传算法来进行优化计算。根据所给信息, 本文采用英国Sheffield大学推出的遗传算法工具箱编写基于Matlab程序, 经调试, 设置计算参数为: 种群染色体20, 最大进化代数200, 交叉概率0.9, 变异概率0.1, 代沟设置比例10%。利用该程序连续进行了5次运算, 各次运算结果见表2。

第一次运算的跟踪情形如图4所示, 由图4可以看出, 随着进化代数的增加, 种群的最优适值在逐渐减少(当进化到第31代时适值收敛)。各次运行结果的最优适值都为52, 即供应链网络的最大供应能力为52。最优染色体为:

0010101100000000100010111100

即使整个网络供应能力最大的各节点供应商选择方案为, $x_1:A; x_2:C; x_3:C; v_1:D; v_2:A; v_3:D; v_4:A; v_5:A; v_6:A; v_7:C; v_8:A; v_9:C; v_{10}:D; v_{11}:D; v_{12}:D; v_{13}:D; v_{14}:A; v_{15}:A; v_{16}:A$ 。

表1 节点可选择企业供应能力表

节点	弧	A	B	C	D
x_1	x_1, v_1	16	10	7	4
	x_1, v_2	4	5	6	7
x_2	x_2, v_2	2	5	7	9
	x_2, v_3	8	7	5	4
	x_2, v_4	4	6	9	13
x_3	x_3, v_4	13	10	7	5
	x_3, v_5	10	13	16	20
v_1	v_1, v_6	8	11	15	18
	v_1, v_8	5	4	3	2
v_2	v_2, v_3	2	3	4	5
	v_2, v_8	8	6	4	2
v_3	v_3, v_8	7	6	5	8
v_4	v_4, v_9	10	7	6	5
	v_4, v_{11}	4	7	11	15
v_5	v_5, v_4	4	7	10	13
	v_5, v_{11}	15	11	8	5
v_6	v_6, v_7	9	7	5	3
	v_6, v_{12}	3	4	6	8
v_7	v_7, v_{12}	3	5	7	9
	v_7, v_{13}	7	6	5	4
v_8	v_8, v_{14}	10	8	6	4
	v_8, v_{15}	3	5	6	7
v_9	v_9, v_{10}	2	4	6	8
	v_9, v_{15}	6	5	4	3
v_{10}	v_{10}, v_{16}	18	14	9	8
	v_{10}, Y	3	6	9	13
v_{11}	v_{11}, v_{10}	5	7	10	13
	v_{11}, v_{16}	11	8	6	4
v_{12}	v_{12}, Y	12	11	10	13
v_{13}	v_{13}, Y	5	6	7	8
v_{14}	v_{14}, Y	15	14	13	12
v_{15}	v_{15}, v_{14}	5	6	7	8
	v_{15}, Y	11	8	5	4
v_{16}	v_{16}, Y	16	15	14	13

表2 程序运算结果

运算次第	最优适值	最优值收敛代数	运行时间/s
1	52	9	2339.9
2	52	88	2341.6
3	51	43	2318.6
4	52	34	2298.2
5	52	121	2324.7

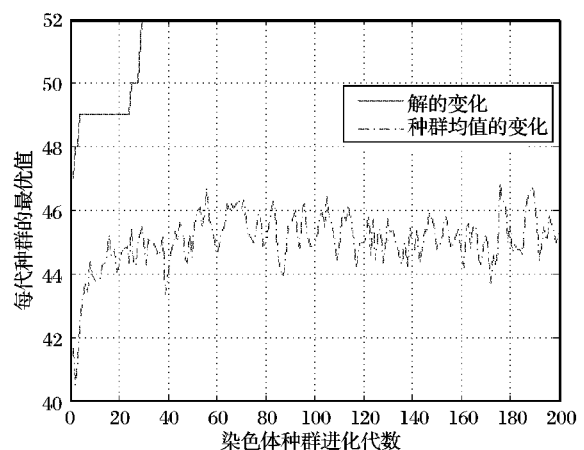


图4 运行结果跟踪

根据已选定的供应商组合,计算供应链网络的最小截集,其遗传算法计算参数设置为:种群染色体20,最大进化代数300,交叉概率0.9,变异概率0.1,代沟设置比例10%。利用该程序连续进行了5次运算,各次运算结果见表3。

表3 最大能力程序运算结果

运算次第	最优适值	最优值收敛代数	运行时间/s
1	52	71	1.2603
2	52	28	1.0601
3	52	24	1.0957
4	52	45	1.1186
5	52	75	1.1216

第一次运算的跟踪情形如图5所示,由图5可以看出,随着进化代数的增加,种群的最优适值在逐渐减少(当进化到第31代时适值收敛)。各次运行结果的最优适值都为52,即供应链网络的最大供应能力为52。最优染色体为:

1111110100100000

因此,网络的瓶颈弧段为 (v_6, v_7) , (v_6, v_{12}) , (v_8, v_{14}) , (v_8, v_{15}) , (v_4, v_9) , (v_{11}, v_{10}) , (v_{11}, v_{16}) 。因此,就供应能力而言,整个供应链网络的瓶颈节点企业是 v_6, v_8, v_4, v_{11} ,尤其是 v_{11} ,其截集上的容量之和最大(17),如果该企业中断,将导致最大供应能力损失最大。

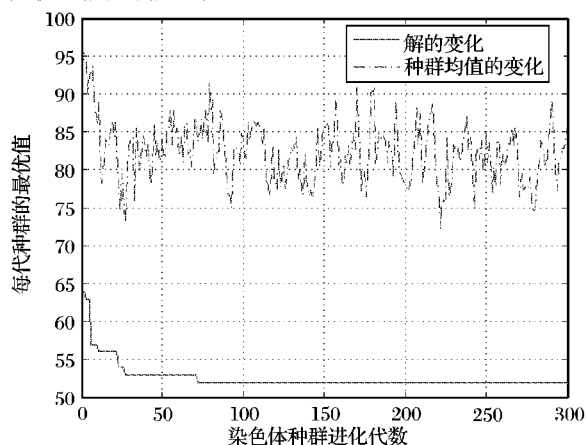


图5 供应商确定情况下最大供应能力程序运行结果跟踪

7 结语

从实例分析可见,本文采用混合遗传算法可以有效求解基于最大供应能力的供应商选择问题,它能够解决优化方案随

着变量的增加而呈指数增长所存在的计算困难问题。同时,通过求解网络最大流的最小截集,还可以发现制约整个供应链网络最大供应能力的瓶颈环节,以便有针对性地采取措施进行改进。

在使用遗传算法求解供应商最优组合的过程中,由于适应度函数使用的遗传算法并不能保证最优解百分之百收敛到最优值,而适应度函数一旦不是收敛到真正最大流,其目标函数值(并不是真正的最小截集)比实际值大,从而导致其所代表的组合被错误的以大概率机会复制并遗传到下一代,最终使得收敛结果不正确。为了解决这个问题,本文采用同一供应商组合多次求适应度值,然后取最小值的办法来解决,譬如一次求适应度值得错误率为1%,那么同一组合计算3次取最小值得到的最大流是错误的概率为 $1\% \times 1\% \times 1\% = 0.0001\%$,随着计算次数的增加,其精度更高。因此,此方法基本可以保证结果的正确性,但随着计算次数的增加,所需要花费的时间也呈倍数增长。如何解决计算精度和计算时间之间的矛盾还需进一步研究。

参考文献:

- [1] QI XUAN, LI YAN-JUN, WUA TIE-JUN. A local-world network model based on inter-node correlation degree[J]. *Physica A: Statistical Mechanics and its Applications*, 2007, 378(2): 561-572.
- [2] NAGURNEY A, TOYASAKI F. Supply chain supernetworks and environmental criteria[J]. *Transportation Research D*, 2003, 8(1): 185-213.
- [3] HELBING D. Information and material flows in complex networks[J]. *Physica A: Statistical Mechanics and its Applications*, 2006, 363(1): xi-xvi.
- [4] KUHNERT C, HELBING D. Scaling laws in urban supply networks[J]. *Physica A: Statistical Mechanics and its Applications*, 2006, 363(1): 89-95.
- [5] LAUMANN S M, LEFEBER E. Robust optimal control of material flows in demand-driven supply networks[J]. *Physica A: Statistical Mechanics and its Applications*, 2006, 363(1): 24-31.
- [6] 戴晓晖,李敏强,寇纪淞. 遗传算法理论研究综述[J]. *控制与决策*, 2002, 15(3): 263-273.
- [7] 丰建荣,刘正和,刘志河,等. MINP问题全局优化算法的研究[J]. *系统仿真学报*, 2005, 17(8): 1859-1863.
- [8] 雷英杰,张善文,李续武,等. MATLAB遗传算法工具箱及应用[M]. 西安:西安电子科技大学出版社, 2005: 62-106.
- [9] 邹豪思,王远志. 网络最大流的矩阵算法[J]. *内蒙古大学学报*, 2001, 32(4): 466-469.

(上接第471页)

参考文献:

- [1] PISINGER D. An exact algorithm for large multiple knapsack problems[J]. *European Journal of Operational Research*, 1999, 114(3): 528-541.
- [2] BALEV S, YANEV N, FREVILLE A, et al. A dynamic programming based reduction procedure for the multidimensional 0-1 knapsack problem[J]. *European Journal of Operational Research*, 2008, 186(1): 63-76.
- [3] FLESZAR K, HINDI K S. Fast, effective heuristics for the 0-1 multi-dimensional knapsack problem[J]. *Computers & Operations Research*, 2009, 36(5): 1602-1607.
- [4] 虞安波,杨家本. 多背包问题的遗传算法求解[J]. *计算技术与自动化*, 2002, 21(20): 59-63.
- [5] 宋海生,傅仁毅,徐瑞松,等. 求解多背包问题的混合遗传算法[J]. *计算机工程与应用*, 2009, 45(20): 45-48.
- [6] KONG MIN, TIAN PENG, KAO YU-CHENG. A new ant colony optimization algorithm for the multi-dimensional knapsack problem[J]. *Computers & Operations Research*, 2008, 35(8): 2672-2683.
- [7] 李晓磊,邵之江,钱积新. 一种基于动物自治体的寻优模式: 鱼群算法[J]. *系统工程理论与实践*, 2002, 22(11): 32-38.
- [8] 黄光球,朱华平,周静. 用鱼群算法求解石油运输系统多级站定位优化问题[J]. *系统工程理论与实践*, 2008, 28(3): 94-102.
- [9] 王锡淮,郑晓鸣,肖健梅. 求解约束优化问题的人工鱼群算法[J]. *计算机工程与应用*, 2007, 43(3): 41-43.
- [10] 李晓磊,路飞,田国会,等. 组合优化问题的人工鱼群算法应用[J]. *山东大学学报: 工学版*, 2004, 34(5): 64-67.