

基于协同自适应禁忌的 多时窗 VRP 算法实现*

朱玲玲¹, 杨爱琴¹, 吴宽仁²

(1. 南通大学 计算机科学与技术学院, 江苏 南通 226019; 2. 南京大学 计算机科学与技术系, 南京 211102)

摘要: 针对多时窗 VRP, 提出了一种对其进行求解的协同禁忌优化算法。首先定义了多时窗 VRP 的数学模型, 通过 C-W 算法求初始解, 采用四种算子进行邻域搜索, 定义目标函数衡量候选解好坏; 设计了自适应地修改禁忌长度的方法和多个子禁忌算法进行协同寻优的算法; 将协同寻优获得的最好最优解作为主禁忌算法初始解进行全局寻优。仿真实验证明, 该方法能有效地解决多时间窗的 VRP, 且与其他方法相比, 以较少的迭代次数获得了全局最优解为 386.38, 具有较大的优越性。

关键词: 车辆路径; 禁忌算法; 多时间窗; 协同

中图分类号: TP18

文献标志码: A

文章编号: 1001-3695(2012)12-4542-04

doi:10.3969/j.issn.1001-3695.2012.12.036

Implementation for VRP with multi-window based on collaborative adaptive tabu algorithm

ZHU Ling-ling¹, YANG Ai-qin¹, WU Kuan-ren²

(1. College of Computer Science & Technology, Nantong University, Nantong Jiangsu 226019, China; 2. Dept. of Computer Science & Technology, University Nanjing, Nanjing 211102, China)

Abstract: Aiming at solving multi-window VRP, this paper proposed a method based on collaborative tabu optimization algorithm. Firstly, it defined the mathematical model for multi-window VRP. Then it used the C-W algorithm to get the initial solution, and used four operators to search neighbor and defined the goal function to evaluate the quality of the solution. Then designed a tabu length modifying method and collaborative sub tabu algorithm adaptively. Used the best solution of collaborative sub tabu algorithm as the initial solution of the main tabu search algorithm. The simulation experiment shows the method can solve the multi-window VRP effectively. Compared with the other methods, the method reduces the iteration numbers and gets the best solution 386.38 with the big superiority.

Key words: vehicle routing; tabu search; multi-time window; collaborate

0 引言

车辆路径问题 (vehicle routing problem, VRP) 由 Dantzing 和 Ramser 于 1959 年首次提出, 其车辆路径方案数随着客户的增加呈指数级增长, 已被证明是一个 NP 难题^[1]。

VRP 可以描述为对于一系列客户配送需求, 设计从配送中心出发的若干车辆有序地经过客户, 满足客户货物配送需求, 最后回到配送中心, 使得每个客户仅被某辆车服务一次, 同时满足一定的约束。这些约束通常包含客户时间窗限制、车辆容量限制、路程最短和费用最少等^[2]。

目前国内外对 VRP 的研究往往集中在单时间窗 VRP (VRP with single time window, VRPTW), 单时间窗 VRP 限定车辆必须在用户指定的时间窗内访问, 而用于解决单时间窗 VRP 的常用方法有精确方法和近似方法两种。精确方法主要

包含分支界定算法和拉格朗日分解法等^[3]。由于其无法满足大规模问题的需要, 所以采用模拟退火算法^[4]、遗传算法^[5]、蚁群算法^[6]、禁忌算法^[7]等近似算法来解决 VRP, 并且已经成为当前的研究热点。

禁忌算法由 Glover 在 1986 年首次提出, 是一种通过记忆功能实现对局部领域的扩展、进而实现全局寻优的算法, 较传统优化算法具有全局收敛能力强的优点。

文献[8]设计了一种基于改进禁忌搜索算法解决有时间窗的车辆路径问题, 并采用动态变化的禁忌长度, 提高了算法的整体寻优能力; 文献[9]提出了采用并行节约法求初始解, 并采用禁忌搜索的混合算法对动态调度问题进行求解; 文献[10]采用将当前解分解成几个独立的路线集合, 并用禁忌搜索法分别进行求解, 采用重启和扰动策略, 以增加算法的全局寻优能力。

上述工作具有重要意义, 但仅解决了单时间窗 VRP, 事实

收稿日期: 2012-05-02; **修回日期:** 2012-06-07 **基金项目:** 国家自然科学基金资助项目 (60211227); 南通大学自然科学基金资助项目 (03040850, 03040846)

作者简介: 朱玲玲 (1976-), 女, 江苏启东人, 讲师, 硕士, 主要研究方向为中文知识抽取、机器学习 (zll2003@ntu.edu.cn); 杨爱琴 (1976-), 女, 江苏如东人, 讲师, 硕士, 主要研究方向为中文知识抽取、机器学习; 吴宽仁 (1974-), 男, 江苏启东人, 讲师, 博士研究生, 主要研究方向为中文知识抽取、机器学习。

上,客户通常有多个时间窗。所以,为了解决多时间窗 VRP,本文在对车辆路径问题进行深入分析的基础上,提出了多时窗 VRP 数据模型,通过 C-W 算法获取初始解,并采用协同的禁忌算法进行寻优,获得全局最优解作为车辆路径方案。

1 多时窗 VRP 数学模型

多时间窗 VRP 可以表示为:有 M 辆车为 N 个需要被派送的客户点 $(0, 1, \dots, N)$ 服务,其中 0 表示配送中心,第 i 个客户点所需要运送的货物需求量为 q_i ,服务时间为 s_i ,多时间窗为 $[a_i^\tau, b_i^\tau]$ ($\tau = 1, 2, \dots, \alpha$), α 为客户时间窗的个数,其中, a_i^τ 为客户 i 的第 τ 个时间窗的最早服务时间, b_i^τ 为客户 i 的第 τ 个时间窗的最晚服务时间,车辆到达客户的时间 t_i 必须满足时间窗的约束。当客户 i 与 j 之间的距离为 d_{ij} 时,其运行成本为 c_{ij} ,运行时间为 tw_{ij} 。 E_i 表示车辆到达客户点 i 的时间 t_i 比 a_i 早的等待成本; L_i 表示车辆到达客户点 i 的时间 t_i 比 b_i 晚的惩罚成本。

定义相关二进制变量如下:

$$x_{ijk} = \begin{cases} 1 & \text{车辆 } k \text{ 从客户 } i \text{ 到 } j \\ 0 & \text{其他情况} \end{cases} \quad (1)$$

$$y_{ik} = \begin{cases} 1 & \text{车辆 } k \text{ 访问客户 } i \\ 0 & \text{其他情况} \end{cases} \quad (2)$$

$$\phi_{i\tau} = \begin{cases} 1 & \text{选中客户 } i \text{ 的 } \tau \text{ 个时间窗} \\ 0 & \text{其他情况} \end{cases} \quad (3)$$

则多时间窗 VRP 的数学模型可以表示为

$$\begin{cases} Z = \min \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^M c_{ij} x_{ijk} + P \\ P = E_i \sum_{i=1}^N \phi_{i\tau} \max(a_i^\tau - t_i, 0) + L_i \sum_{i=1}^N \phi_{i\tau} \max(t_i - b_i^\tau, 0) \end{cases} \quad (4)$$

$$\text{s. t.} \quad \sum_{i=1}^N q_i y_{ik} \leq Q \quad k = 1, 2, \dots, M \quad (5)$$

$$\sum_{k=1}^M y_{ik} = 1 \quad i = 1, 2, \dots, N \quad (6)$$

$$\begin{cases} \sum_{i=1}^N x_{ijk} = y_{jk} \\ \sum_{j=1}^N x_{ijk} = y_{ik} \end{cases} \quad j = 1, 2, \dots, N \quad k = 1, 2, \dots, M \quad (7)$$

$$\phi_{i\tau} a_i^\tau \leq s_i \leq \phi_{i\tau} b_i^\tau \quad (8)$$

$$\sum_{\tau=1}^{\alpha} \phi_{i\tau} = 1 \quad i = 1, 2, \dots, N \quad (9)$$

其中:式(4)为目标函数, $\min \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^M c_{ij} x_{ijk}$ 为车辆的最小运输成本, P 为车辆到达时间 t_i 不在时间窗范围的惩罚成本;式(5)为车辆容量约束;式(6)和(7)为访问唯一性约束;式(8)为客户时间窗约束;式(9)为多时间窗的唯一约束,即车辆在访问客户 i 时,必须选择其中一个时间窗进行访问。

2 协同自适应禁忌算法

为了提高禁忌算法的求解质量,采用子禁忌算法进行协同寻优的方式。首先通过 C-W 算法构造初始解,并将其作为子禁忌算法的初始解进行寻优;在寻优过程中自适应地调整禁忌长度,选取所有子禁忌算法的最优解作为主禁忌算法的初始解开始寻优,直至达到最大迭代次数或最优解达到最大未变化次数,最后比较主禁忌和子禁忌算法的最优解情况,选取两者中的较优解作为全局最优解,其流程如图 1 所示。

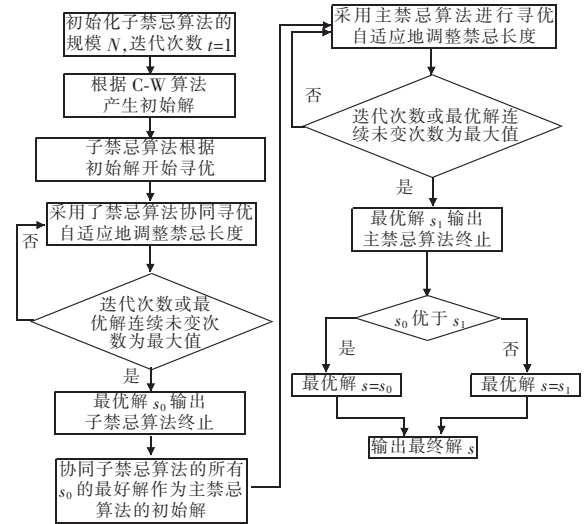


图 1 协同自适应禁忌算法流程

2.1 目标函数

在禁忌算法中,目标函数越大则解越优,所以对式(4)进行改进,得到的目标函数如式(10)所示。

$$\text{goal} = \frac{1}{Z} = \frac{1}{\min \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^M c_{ij} x_{ijk} + P} \quad (10)$$

2.2 初始解构造

初始解的好坏关系到禁忌算法的全局寻优能力。采用 C-W 算法进行初始路径求解,同样采用 c_{ij} 表示从客户点 i 到 j 的运输费用。具体步骤如下:

a) 首先将所有客户与配送中心进行连接,计算运行成本,然后对任意两可连接客户进行连接,计算成本节约值,如式(11)所示。

$$\cos t_{ij} = c_{i0} + c_{0j} - c_{ij} \quad (11)$$

b) 按式(11)所计算出的节约值进行排序,对 $\cos t_{ij}$ 进行从大到小排序,并将其存入集合 M 中。

c) 若 M 为空,则算法终止,否则转入步骤 d)。

d) 取出 M 中的第一个元素 $\cos t_{ij}$,对客户 i 和 j 进行判断,若符合下列条件之一则转入步骤 e); 否则转入步骤 g):

(a) 客户点 i 和 j 不在已有路径上;

(b) 客户点 i 和 j 在已有路径上,但不与配送中心直接相连;

(c) 客户点 i 和 j 在已有的路径上,但一个是起点另一个是终点。

e) 判断加入路径 $i-j$ 后的总货运量是否满足车辆容量 Q 约束,若不满足进入步骤 g); 否则进入步骤 f)。

f) 根据式(12)计算连接客户点 i 和 j 后,计算车辆到达 j 点的时间比原路线 j 上的任务的开始时间推迟量或提前量 EL_j :

$$EL_j = t_i + s_i + tw_{ij} - t_j \quad (12)$$

计算原路径 j 后的任务不需要等待的最大提前量:

$$E_j = \min_{r \geq j} \{ t_r - a_r \} \quad (13)$$

$$L_j = \min_{r \geq j} \{ b_r - t_r \} \quad (14)$$

如果 EL_j 满足下面三个条件之一,则连接客户点 i 和 j , 否则进入步骤 g):

- (a) 当 $EL_j = 0$;
- (b) 当 $EL_j < 0$ 且 $-EL_j \leq E_j$;
- (c) 当 $EL_j > 0$ 且 $EL_j < L_j$;
- g) $W = W - s_{ij}$, 转入步骤 c) 继续进行迭代。

2.3 邻域构造

采用四种邻域算子进行邻域搜索, 这四种算子分别是逆转、1-0 交换、2-opt 交换和 3-opt 交换。

a) 逆转算子。此算子属于线路内部变换, 从子路径内选择某部分进行逆转, 如将子路径 R_0 中 357 逆转为 753。

$$R_0: 0-4-3-5-7-8-0 \rightarrow R_0: 0-4-7-5-3-8-0$$

b) 1-0 交换。将路径 R_0 内子路径中某一客户插入到另一条子路径中去, 如将路径 R_0 中第一条子路径中的 3 插入到第二条子路径中 7 和 8 之间。

$$R_0: 0-4-3-0-5-7-8-0 \rightarrow R_0: 0-4-0-5-7-3-8-0$$

c) 2-opt 交换。选取路径 R_0 中两条子路径中的某两点, 将其路径尾部进行交换, 如将下面路径 R_0 中的 4 和 5 进行 2-opt 交换, 得到的结果如下所示。

$$R_0: 0-4-3-0-5-7-8-0 \rightarrow R_0: 0-4-7-8-0-5-3-0$$

d) 3-opt 交换。此算子属于线路内部变换, 即从路径 R_0 中某子路径中四个连续的点 $(i, i+1, j, j+1)$ 变换为 $(i+1, i, j+1, j)$, 当选取 i 为 4、 j 为 7 时, 路径 R_0 经过 3-opt 交换后如下所示。

$$R_0: 0-4-3-7-8-0-5-0 \rightarrow R_0: 0-3-4-8-7-0-5-0$$

2.4 禁忌表和禁忌长度

采用传统禁忌算法中的禁忌表, 将每次搜索的局部最优解加入到禁忌表中, 采用数组作为存储结构。

为了阻止算法在搜索过程中出现早熟收敛, 对禁忌长度进行自适应的调整。在初始搜索时, 扩大搜索空间, 以寻求解的多样性; 当搜索到一定阶段快达到最优时, 为了提高收敛性, 缩小搜索区域, 通过对禁忌长度进行自适应的调整来增强算法的全局寻优能力, 禁忌长度自适应调整的公式如下所示。

$$\text{len} = \text{len}_{\min} + t \frac{(\text{len}_{\max} - \text{len}_{\min})}{t_{\max}} \quad (15)$$

2.5 带时间窗的解结构

由于多时窗 VRP 中每个客户都具有一个或多个时间窗, 而车辆仅选择每个客户的某个时间窗进行访问, 采用数组 SV 来存储带有时间窗的解, 每个客户均使用节点表示, 如 (N, i) 表示客户 N 的第 i 个时间窗, 则解结构如下所示。

$$(0,0)-(2,1)-(4,2)-(5,1)-(0,0)-(7,3)-(0,0)$$

在上述解中, 包含了两条子路径即 $(0,0)-(2,1)-(4,2)-(5,1)-(0,0)$ 和 $(0,0)-(7,3)-(0,0)$ 。其中 $(0,0)$ 表示配送中心, $(7,3)$ 表示客户 7 的第 3 个时间窗。

2.6 终止规则和结果输出

当迭代次数达到最大值或连续寻优得到不变解的次数超过某个最大值时, 禁忌算法停止寻优, 并将当前解作为最优解进行输出。

2.7 协同禁忌算法

算法 1 协同禁忌算法

输入: 根据 C-W 算法获得初始解 x_{initial} , 设置最优解未变化

次数最大值 $r_{\max} = 0$ 、最大迭代次数 $t_{\max}$ 、禁忌表的最大长度 $\text{len}_{\max}$ 、最小长度 $\text{len}_{\min}$ 和长度 len , 协同禁忌的规模为 W , 候选解规模为 Y 。

输出: 最优解 s_{best} 。

a) 初始化当前解 $x_{\text{cur}} = x_{\text{initial}}$, 最优解 $s_{\text{best}} = x_{\text{initial}}$, 初始化禁忌表为 $TV = \text{null}$, 迭代次数 $t = 1$ 。

b) 将 x_{cur} 作为 $W-1$ 个子禁忌算法的初始解。

c) 利用文中的四种邻域算子产生 Y 个候选解, 根据式 (10) 所示目标函数计算当前最优解 x_{best} , 并判断其是否处于禁忌状态:

如果处于禁忌状态, 并满足赦免原则, 即 $\text{goal}(x_{\text{best}}) > \text{goal}(s_{\text{best}})$, 则忽略其禁忌特性, 将当前解及最优解的值均设置为当前最优解, 并将最优解未变化次数置为 0;

反之, 选择当前未被禁忌的最优解 x_{best} , 并设置当前解的值为当前最优解 x_{best} , 将最优解未变化次数加 1;

将当前解 x_{cur} 加入禁忌表, 替换禁忌表中禁忌长度为 0 的元素, 将其他禁忌元素的禁忌长度减 1。

d) $t = t + 1$, 根据式 (15) 自适应地调整禁忌长度 len , 并循环执行步骤 b)c), 直到子禁忌算法迭代次数达到最大值 $t = t_{\max}$ 或最优解连续未变化次数 $r = r_{\max}$, 则转入步骤 e)。

e) 将 $W-1$ 个子禁忌算法得到的最好最优解作为主禁忌算法的初始解。

f) $t^* = t^* + 1$, 并循环执行步骤 c)d), 若主禁忌算法当前迭代次数为达到最大值 $t^* = t_{\max}$ 或最优解连续未变化次数 $r^* = r_{\max}$, 则主禁忌算法结束。

g) 将主禁忌算法的最优解与前 $W-1$ 个子禁忌算法的最好最优解作比较, 输出其中的较优解。

3 仿真实验

为了验证算法的性能, 构造了 12 个客户的多时间窗算例。客户点分布在边长为 100 km 的正方形区域内, 配送中心位于正方形中心, 坐标点为 $(0,0)$, 车辆的载重量为 3 T, 客户点的详细信息如表 1 所示。

表 1 客户点详细信息

编号	坐标	配送量/T	时间窗 (a_i, b_i)
0	(0,0)	0.000	$[0, \infty]$
1	16, 10	1.324	$[18, 27]$
2	-17, 26	1.423	$[20, 40][100, 110]$
3	35, -35	1.212	$[10, 22][41, 62]$
4	-41, -23	1.611	$[61, 83][98, 120]$
5	19, 35	1.012	$[70, 80][88, 110]$
6	16, 43	1.903	$[59, 70][101, 121]$
7	34, -41	1.911	$[20, 50]$
8	-15, 0	1.901	$[30, 45][50, 70]$
9	42, 43	2.011	$[43, 56]$
10	38, -25	2.732	$[57, 68][77, 98]$
11	16, 23	1.834	$[20, 40]$
12	25, -31	1.903	$[14, 32][71, 92]$

首先采用 C-W 算法求初始解, 然后采用本文的协同自适应禁忌算法进行寻优, 经过 300 次迭代后, 可以得到算例所需要的车辆数为 4, 配送路径总距离为 386.38 km, 对应的四条优化路径为

路径 1: (0,0)-(8,2)-(2,1)-(0,0)

路径 2: (0,0)-(4,1)-(0,0)

路径 3: (0,0)-(1,1)-(11,1)-(6,1)-(5,2)-(0,0)

路径 4: (0,0)-(12,1)-(7,1)-(3,2)-(10,2)-(0,0)

实验结果如图 2 所示。

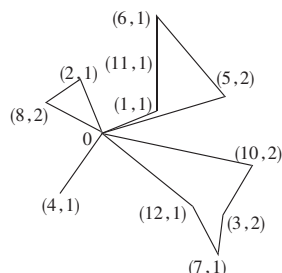


图 2 客户点连接路径

将本文方法与文献[5,6]以及经典禁忌算法进行比较,得到的路径长度随着迭代次数增加的变化如图 3 所示。

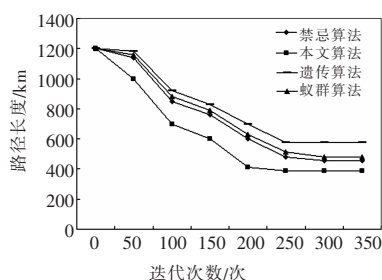


图 3 路径长度随迭代次数变化曲线

从图 3 可以看出,本文的协同自适应算法在迭代 200 次时趋于收敛,得到的路径长度为 386.38;遗传算法在迭代 250 次时趋于收敛,但其路径长度为 576.89,很显然算法陷入了局部最优;蚁群算法和禁忌算法在迭代到 300 次时趋于收敛,得到的路径长度分别为 481.57 和 453.62,显然本文方法最优。这是由于禁忌算法采用记忆机制从而具有较强的局部搜索能力,同时本文采用 C-W 方法求解初始值,并采用协同禁忌算法进行寻优,所以使得全局寻优能力更强,结果更符合真实值。

4 结束语

多时间窗的 VRP 是一个 NP 难题,本文在对传统禁忌算法改进的基础上提出了一种协同自适应算法用于解决多时间窗 VRP。采用 C-W 算法求初始解,提出了采用四种邻域算子进行邻域搜索,在此基础上采用子禁忌算法和主禁忌算法进行协同寻优,并自适应地修改禁忌长度以满足算法需要。通过实例仿真证明,该算法提高了全局搜索能力,改善了解的质量,同时具有计算效率高、计算结果稳定的优点。

参考文献:

- [1] LANTUEJOL C, MAISONNEUVE F. Geodesic methods in quantitative image analysis [J]. *Pattern Recognition*, 1984, 17 (2): 177-187.
- [2] BLANTON J L, WAINWRIGHT R L. Multiple vehicle routing with time and capacity constraints using genetic algorithms [C]//Proc of the 5th International Conference on Genetic Algorithms. 1993:452-459.
- [3] 宋伟刚,王聪,赵玲. 大规模客户车辆调度问题启发式算法研究 [J]. *系统仿真学报*, 2008, 20 (14): 3696-3701.
- [4] CHEN J F, WU T H. Vehicle routing problem with simultaneous deliveries and pickups [J]. *Journal of the Operational Research Society*, 2006, 57 (5): 579-587.
- [5] 黄明,林广智,梁旭,等. 改进的遗传算法在车辆路径问题中的应用 [J]. *大连交通大学学报*, 2010, 31 (1): 95-99.
- [6] 段征宇,杨东援,王上. 时间依赖型车辆路径问题的一种改进蚁群算法 [J]. *控制理论与应用*, 2010, 27 (11): 1557-1563.
- [7] 姜山佐,史忠科. 有限车辆调度问题的模型和改进遗传算法 [J]. *计算机应用研究*, 2006, 23 (4): 61-62.
- [8] 葛金辉. 有时间窗的车辆路径问题及改进禁忌搜索算法 [J]. *吉林大学学报*, 2011, 49 (1): 105-111.
- [9] 王训斌,陆慧娟,张火明. 物流动态车辆调度问题的混合禁忌搜索算法 [J]. *计算机工程与应用*, 2010, 46 (8): 228-231.
- [10] 李建,鲁植雄,高谋荣. 大规模同时集散货物路线问题的新禁忌搜索算法设计 [J]. *西南交通大学学报*, 2009, 44 (5): 787-793.

(上接第 4541 页)网络结构、提高分类器泛化能力和识别效率。

参考文献:

- [1] BAIK J W, KANG H W, KANG C W, et al. The optimal control limit of a G-EWMAG control chart [J]. *The International Journal of Advanced Manufacturing Technology*, 2011, 56 (1): 161-175.
- [2] RANAEE V, EBRAHIMZADEH A. Control chart pattern recognition using neural networks and efficient features; a comparative study [J]. *Pattern Analysis & Applications*, 2011, 10 (1): 1-12.
- [3] GAURI S K, CHAKRABORTY S. Recognition of control chart patterns using improved selection of features [J]. *Computers & Industrial Engineering*, 2009, 56 (4): 1577-1588.
- [4] PSARAKIS S. The use of neural networks in statistical process control charts [J]. *Quality and Reliability Engineering International*, 2011, 27 (5): 641-650.
- [5] GUH R S, HSIEH Y C. A neural network based model for abnormal pattern recognition of control charts [J]. *Computers & Industrial Engineering*, 1999, 36 (1): 97-108.
- [6] DAS P, BANERJEE I. An hybrid detection system of control chart patterns using cascaded SVM and neural network-based detector [J]. *Neural Computing and Applications*, 2011, 20 (2): 287-296.
- [7] 杨元世,吴德会,苏海涛. 基于支持向量机技术的智能工序诊断研究 [J]. *微电子学与计算机*, 2006, 23 (5): 42-45.
- [8] 吴常坤,赵丽萍. 基于小波分析和支持向量机的控制图模式识别 [J]. *中国机械工程*, 2010, 21 (13): 1572-1576.
- [9] EBRAHIMZADEH A, RANAEE V. Control chart pattern recognition using an optimized neural network and efficient features [J]. *ISA Transactions*, 2010, 49 (3): 387-393.
- [10] 杨元世,吴德会,苏海涛. 基于 PCA 和 SVM 的控制图失控模式智能识别方法 [J]. *系统仿真学报*, 2006, 18 (5): 1314-1318.
- [11] WANG C H, KUO W. Identification of control chart patterns using wavelet filtering and robust fuzzy clustering [J]. *Journal of Intelligent Manufacturing*, 2007, 18 (3): 343-350.
- [12] HOUCK C R, JOINES J A, KAY M G. A genetic algorithm for function optimization; a matlab implementation NCSU-IE TR 95-09 [R]. 1995.
- [13] HARVEY I. The microbial genetic algorithm [J]. *Advances in Artificial Life. Darwin Meets von Neumann*, 2011, 5778: 126-133.
- [14] SCHÖLKOPF B, SMOLA A J. Learning with kernels [M]. Cambridge: MIT Press, 2002.
- [15] VAPNIK V. 统计学习理论的本质 [M]. 张学工,译. 北京:清华大学出版社, 2000.