

# 用变长度染色体遗传算法优化加工路径的方法

郭华芳<sup>1</sup>, 刘海利<sup>2</sup>, 李海生<sup>3</sup>, 张严林<sup>3</sup>

GUO Hua-fang<sup>1</sup>, LIU Hai-li<sup>2</sup>, LI Hai-sheng<sup>3</sup>, ZHANG Yan-lin<sup>3</sup>

1.中国科学院 广州能源研究所, 广州 510640

2.广东工业大学 自动化学院, 广州 510090

3.广东省科学院自动化工程研制中心, 广州 510070

1.Guangzhou Institute of Energy Conversion, Chinese Academy of Sciences, Guangzhou 510640, China

2.College of Automation, Guangdong University of Technology, Guangzhou 510090, China

3.Science Academy Automatization Center of Guangdong Province, Guangzhou 510070, China

E-mail: harryliu007@163.com

**GUO Hua-fang, LIU Hai-li, LI Hai-sheng, et al. Approach for machining paths optimization based on variable-length genome genetic algorithm. Computer Engineering and Applications, 2009, 45(6): 207-209.**

**Abstract:** The problem of machining path optimization is a special Traveling Salesman Problem(TSP), to solve this problem, the machining paths are sorted into a series of machining elements such as point, line segment, curve and close curve, and an approach for it's optimization based on variable-length genome genetic algorithm is presented. In this algorithm, each machining element is represented by some points which are encoded into a combination of an index and its attribute, and some kinds of lines can be cut and spliced by the crossover and mutation operation, thus the machining paths can be further optimized. Simulation results show that this approach is effective for the problem of machining paths optimization, it can dramatically shorten machining paths.

**Key words:** machining path optimization; variable-length genome; Genetic Algorithm(GA); Traveling Salesman Problem(TSP); NC machining

**摘 要:** 加工路径优化问题属于一类特殊的旅行商问题(TSP), 针对此问题将加工路径细分为点、线段、曲线段及闭合曲线等加工要素, 并提出一种基于变长度染色体遗传算法的优化方法。该方法将每个点编码为一个二元组用以表示各种加工要素, 其交叉和变异操作能对一些线进行分割和合并, 使加工路径能得到更大程度的优化。仿真结果表明, 该方法具有良好的优化效果, 可以显著地缩短辅助运动路径的长度。

**关键词:** 加工路径优化; 变长度染色体; 遗传算法; 旅行商问题; 数控加工

**DOI:** 10.3778/j.issn.1002-8331.2009.06.059 **文章编号:** 1002-8331(2009)06-0207-03 **文献标识码:** A **中图分类号:** TP301.6

## 1 引言

对数控系统的加工路径进行优化可以减少机床辅助运动路径, 缩短加工时间, 特别对加工路径复杂、进行批量生产的加工系统, 如绘图仪、雕刻机、点胶机、PCB 钻孔机、微量射出标签机等, 将大大地提高生产效率, 降低生产成本。因而有很多这方面的研究, 如文献[1-3]等, 然而, 这些方法主要针对特定的工艺只完成对点、“开放式线”或“闭合式线”中的一种或几种进行优化, 不具有通用性, 特别是把各种线作为一个整体进行处理使得路径不能进一步优化。将加工路径细分为点、线段、曲线段及闭合曲线等加工要素, 并采用变长度染色体的遗传算法, 其特定的编码、交叉和变异方案可以对折线进行分割和合并而保存曲线段和闭合曲线, 从而能够得到更优的加工路径。

## 2 问题描述

数控系统的加工路径是由一系列的点和线组成的, 如图 1 所示。由于加工质量的要求, 在加工曲线时不能被打断(进、退刀), 而对于折线则可以在线的交点处将其分割为线段。例如, 在图 1 中封闭线条 7-8-9-10-11-7 可以分割为 7-8, 8-9, 7-11 和 9-10-11, 但曲线段 9-10-11 不能被分割, 而闭合曲线 12-13-14-15-12 则只能有一个进、退刀点。但是, 对折线的这种“分割”使刀具需要增加一个“抬刀”和“落刀”的动作, 需要将其折合成平面路径长度。这样, 加工要素就可以细分为点、线段、曲线段及闭合曲线。因而加工路径优化的问题可以描述为: 在众多的待加工的点、线段、曲线段及闭合曲线中, 如何寻找一条路径, 使得加工刀具(或画笔等)从加工原点出发完成加工后

**作者简介:** 郭华芳(1964-), 男, 博士生, 研究员, 主要研究方向为机电工程; 刘海利(1980-), 男, 硕士生, 研究方向为自动化装备控制技术; 李海生(1975-), 男, 博士, 讲师, 主要研究兴趣为工业过程的建模、仿真、控制和优化; 张严林(1974-), 男, 副研究员, 主要研究方向为企业信息化及工业自动化。

**收稿日期:** 2008-01-14 **修回日期:** 2008-04-18

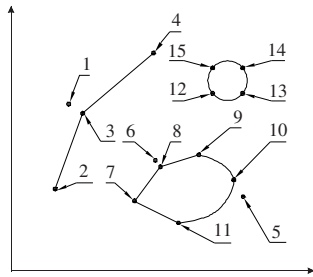


图1 待加工路径示例

又回到原点所走的折合平面路径最短?其中待加工的点和线不能被重复遍历(加工),但线的端点可以被重复访问(作为进、退刀点)。

由此可以看出,当只有点加工时此问题属于典型的 TSP 问题,而将线段、曲线和闭合曲线看做一个整体(城市)时则属于非对称的 TSP 问题<sup>[1]</sup>,将线上的点看成有轨迹约束的个体(城市),则路径优化问题可以归结为一类特殊的有轨迹约束的 TSP 问题。TSP 问题是一个典型的易于描述却难于处理的 NP 完备问题,用传统的线性及非线性优化算法难于求解。遗传算法是一种基于演化的概率性全局优化算法,十分适合于处理以 TSP 问题为代表的各种优化组合优化问题,因而得到了广泛的应用<sup>[4-5]</sup>。而基于变长度染色体的遗传算法对于一些问题的解决能得到比简单遗传算法更好的性能<sup>[6-7]</sup>,加工路径优化中折线的分割与拼接都会导致染色体长度的变化,因而非常适合采用变长度染色体的遗传算法。

### 3 算法的关键技术

采用遗传算法求解本问题关键在于路径的编码和各种遗传操作的设计,下面分别介绍。

#### 3.1 变长度染色体的编码方法

在现有的路径编码方式中路径表示法自然直观,且易于加入启发式信息,因而使用最多<sup>[1]</sup>。参照路径表示法也将每个点用一个唯一的序号进行标识,但为了区分点、折线、曲线段及闭合曲线,还要对每个点赋予一个属性值,这个属性值随着点所在线条的位置的不同而不同,而且两种线的交点也有相应的属性值。这样每个点由一个二元组组成:(*point\_index*, *point\_type*),其中 *point\_index* 是点的序号,同路径表示法相同,将待加工路径中的点按自然数进行编号;*point\_type* 是点的属性值,采用八位二进制编码(为书写方便下面用十六进制表达),其编码方法如下:

- (1) 0FH 表示独立的点;
- (2) 第 4、5、6 位为 1 分别表示是线段、曲线段或闭合曲线上的点;
- (3) 当第 7 位为 1 时表示该点为两种线的交点;
- (4) 0 位为 1 表示该点为某种线的起始点,1 位为 1 表示该点为某种线的终点,当两位都为 0 时表示该点为某种线的中间点;
- (5) 2、3 位与 0、1 位作用相同,当所在点为两种线的交点时才有意义,例如图 1 中 8-9-10-11 的路径中点 9 表示为(9, B6H)。

这种编码方法可以有效地表示加工要素,使遗传操作能顺利进行。几种加工要素在加工路径中的表示如下:

##### (1) 曲线段的基因表示

加工轨迹的优化问题实际上是使完成加工的辅助路径最

短,而待加工部分的长度不会变化。对于曲线段,影响辅助路径长度的是其在加工轨迹中的顺序和其两个端点的加工顺序。因而只需要将其两个端点编码即可,其轨迹长度也以两点间的距离代替。如图 1 中的 9-10-11 若为孤立的曲线段则在染色体中可表示为:{(9, 21H), (11, 22H)}。

##### (2) 闭合曲线的基因表示

闭合曲线一般是圆或椭圆等,由于闭合曲线只能有一个进、退刀点,所以影响辅助路径长度的是其在加工轨迹中的顺序和进退刀点在闭合曲线中的位置。因而需要在闭合曲线上取若干点进行优化计算,显然采用的点越密集其优化效果越好,但过多的点会使染色体过长,降低运算效率,而对整体的加工轨迹优化作用并不大。可以根据实际情况进行调整,以获得优化效果和优化效率的平衡<sup>[1]</sup>。这里根据曲线长度确定点的个数,如图 1 采用 4 个点,若 13 点为进、退刀点闭合曲线可以表示为:{(13, 41H), (14, 40H), (15, 40H), (12, 40H), (13, 42H)}。

##### (3) 点和折线的基因表示

点和折线的编码较简单,例如在图 1 中先加工点 1 再加工过 2-3-4 的折线可表示如下:

{(1, 0FH), (2, 11H), (3, 10H), (4, 12H)}

而先加工线段 2-3 再加工点 1 然后再加工线段 3-4 则可以表示为:

{(2, 11H), (3, 12H), (1, 0FH), (3, 11H), (4, 12H)}

在图 1 所示的情况下,2-3-1-3-4 的路径长度要小于 1-2-3-4 的长度,但这带来了染色体长度的增加,并使染色体出现重复个体(点),这也使得本算法不能直接使用简单遗传算法的一些遗传操作。

### 3.2 选择操作

选择操作直接以加工路径的总长度为适应度函数,并采用竞赛规模为 2 的锦标赛法。即在种群中随机地选择两个个体,再在其中选适应度较小的(路径最短)作为父个体,经过重复选择直至生成新一代种群的所有父个体。但是这种选择操作具有随机性,有可能没有选到当前种群中的最优个体或者在后面的操作中将其破坏,从而损失了优秀的基因,对算法的效率和收敛性有不利影响。采用精英保留策略可以较好地解决这个问题:用父代的最优个体不经交叉、变异直接取代子代中最差的个体。

### 3.3 交叉操作

由于染色体的编码是一个有约束的序列,而在交叉过程中要求子代不能有重复的加工要素,也不能缺少加工要素,对于此问题顺序交叉方法最为合适,为了便于对变长度染色体和有约束的编码序列进行处理,这里采用单点的顺序交叉。

在交叉过程中有时需要对线条进行分割和合并,线条的分割是指在线条的交点处将线分为两段,例如在图 1 中 8-9-11 可以分割成 8-9 和 9-11 两段,而合并则是其逆过程。这种分割和合并往往能带来加工路径的优化,但是分割操作使刀具需要增加一个抬刀和落刀的动作,这会花费一定的时间,可以将其折算为平面的路径长度加入该染色体的适应度,这样在选择操作时就可以衡量出这种分割能否带来路径的优化,若不能则该种分割可能会被淘汰,这也是没有直接将折线分为小线段的原因。同样,合并操作会使刀具动作减少,也要在其适应度上减去一定数值。交叉过程如下:

##### (1) 截断

截断是对两个父染色体随机地选择交叉点, 在交叉点处将染色体截断生成待拼接的两个基因段。若交叉点在线条的交点上则将线条分割成两个部分, 若交叉点为线条的尾部或闭合曲线的中部则需要向前搜索至该染色体的头部作为交叉点。

以图 1 的路径为例, 为了描述方便, 将折线表示为 2~3~4, 将闭合曲线表示为 12~13~14~15~12, 曲线表示为 9\_11。设交叉操作的两个父个体如下:

$p_1$ : {2~3|8~9 5 7~11\_9 6 12~13~14~15~12 4~3 1 8~7}

$p_2$ : {7~8|1 3~4 15~14~13~12~15 9\_11~7 2~3 6 8~9 5}

设交叉点选在第 3 位, 则  $p_1$  经截断后在交叉点左右生成两串代码:

左侧: 2~3

右侧整串: 8~9 5 7~11\_9 6 12~13~14~15~12 4~3 1 8~7 2~3

右侧整串是指从交叉点右侧开始到  $p_1$  的尾部再从  $p_1$  起始点到交叉点的基因段, 显然, 它包含了  $p_1$  的所有加工要素。

同样,  $p_2$  截断后的代码串为:

左侧: 7~8

右侧整串: 1 3~4 15~14~13~12~15 9\_11~7 2~3 6 8~9 5 7~8

(2) 拼接

这一步主要完成去除重复加工要素并对有重点的线进行合并的操作。

对  $p_1$  和  $p_2$  进行交叉, 首先将  $p_1$  的“右侧整串”中的  $p_2$  “左侧”元素去除, 并将此子串附加在  $p_2$  “左侧”元素的后面则生成了子个体  $o_1$ :

$o_1$ : {7~8 8~9 5 7~11\_9 6 12~13~14~15~12 4~3 1 2~3}

这里出现了重复的线段端点, 需要进行合并, 即将 7~8 和 8~9 合并为 7~8~9 并从其适应度中减去一定数值。

同理可得子个体  $o_2$ :

$o_2$ : {2~3 1 3~4 15~14~13~12~15 9\_11~7 6 8~9 5 7~8}

至此交叉操作完成。由此可见, 这种分割和合并操作能将 2~3~4 的折线分开并产生“2~3 1 3~4”这种优良的基因片段。

### 3.4 变异操作

变异操作的作用是维持群体的多样性和改善算法的局部搜索能力。这里采用两种方式: 一种是按一定的变异概率交换各种线的头尾顺序, 如对折线段 2~3~4 经变异后为 4~3~2, 对于闭合曲线可随机选其中的一个点做新的进、退刀点, 如 12~13~14~15~12 变为 14~15~12~13~14; 另外, 还采用一种“近邻进化”的策略: 以一定的变异概率按最近邻算法构造一段基因放在染色体中, 这个过程也涉及到线条的分割和合并。引入“近邻进化”操作的主要目的是利用邻域的启发式搜索方法来改善

遗传算法的局部搜索能力。在简单遗传算法操作中, 交叉操作在可行解空间中动作范围较宽, 步伐较大。因此, 在遗传算法框架中加入适当的、基于邻域的局部搜索机制, 构成一种全局搜索和局部搜索相结合的优化算法<sup>[9]</sup>, 可以提高寻优效果, 加快收敛速度。

## 4 算法仿真与分析

在 Matlab 上对算法进行了编程仿真, 以图 2(a) 所示的加工路径为例, 其中实线为待加工路径, 该路径由 17 个点、13 条折线段、一条曲线段和一个椭圆共 61 个点组成。为了分析算法的性能, 分别对几种优化算法进行了比较。首先, 将各种线作为整体, 用最近邻算法和遗传算法进行优化, 这里的遗传算法未采用变长度染色体的编码方法, 但用“启发式交叉”<sup>[10]</sup>进行了改进: 交叉时在父代个体  $p_1$  中随机选择一个交叉位置“a”, 而相应的父代个体  $p_2$  的交叉位置“b”选在离“a”上的点最近的点所在的位置。为表述方便下面简称其为“定长遗传算法”; 然后采用变长度染色体的编码方法, 并用“启发式交叉”对遗传算法进行改进, 下面简称为“启发式交叉”; 最后用前面所述的变长度染色体的遗传算法, 由于采用了“近邻进化”策略, 为区别起见下面简称“近邻进化”。

表 1 为几种算法优化结果的比较, 表中路径长度及运行时间为 20 次仿真的平均值, 而对于“启发式交叉”和“近邻进化”, 其路径长度包括了由于将折线分割成线段而增加的折算路径长度。

表 1 几种算法仿真结果的比较

算法	优化后路径长度/m	运行时间/s	进化代数/种群大小	备注
随机路径	6.51	0.01	...	线条未分割
最近邻法	3.99	0.02	...	线条未分割
定长遗传算法	3.56	29.54	150/50	定长染色体启发式交叉
启发式交叉	3.32	44.28	200/50	变长度染色体
近邻进化	2.77	24.25	100/50	变长度染色体

图 2(a) 和图 2(b) 分别是经表中“定长遗传算法”和“近邻进化”算法优化后的加工路径, 其中虚线为辅助加工路径, 图 2(c) 为“近邻进化”算法优化过程中平均值与最优值的变化趋势。

由表 1 和图 2 中的比较可以看出:

(1) 采用基于变长度染色体的遗传算法优化后的路径长度比采用“定长遗传算法”减少 0.79 m, 相对减少 22%, 而相对“随机路径”则减少了 57%, 这是由于前者可以对折线段进行分割和合并, 从而能最大程度地优化加工路径;

(2) “启发式交叉”和“定长遗传算法”采用同样的交叉操作, 但前者将折线进行了分割从而增加了问题的规模, 因而需

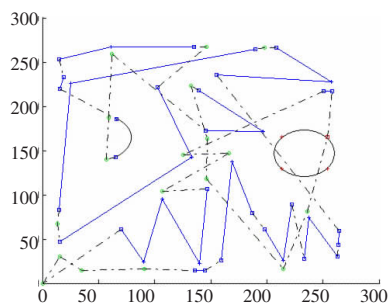


图 2(a) 用“定长遗传算法”优化后的加工路径

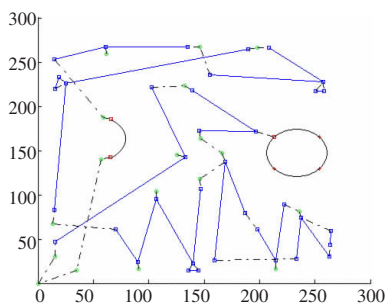


图 2(b) 用“近邻进化”算法优化后的加工路径

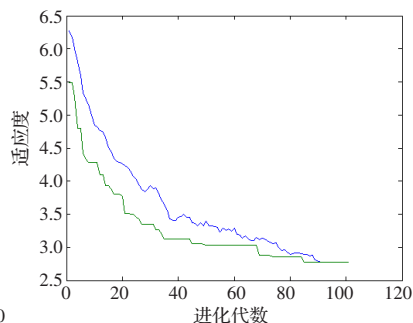


图 2(c) “近邻进化”算法的优化过程