

# 基于改进遗传算法的移动机器人路径规划

张毅, 代恩灿, 罗元

(重庆邮电大学 国家信息无障碍研发中心, 重庆 400065)

**摘要:** 针对传统遗传算法存在的搜索效率低、易于陷入局部最优解的问题, 提出了一种改进的遗传算法; 采用简单的一维编码替代复杂的二维编码, 节约了存储空间; 在遗传算子的设计中, 重新定义了交叉算子和变异算子, 避免了陷入局部最优; 最后将最短路径和免碰撞相结合作为适应度函数进行遗传优化; 在种群的各项参数均相同的情况下, 分别对改进遗传算法和传统遗传算法进行了 100 次实验; 其中, 改进遗传算法搜索到最优路径的次数为 95 次, 最短路径长度为 20.970 6, 平均搜索用时 217 ms; 传统遗传算法搜索到最优路径的次数为 62 次, 最短路径长度为 25.071 1, 平均搜索用时 345 ms; 实验结果表明, 相比于传统遗传算法, 改进遗传算法搜索效率更高且能获得更好的解。

**关键词:** 遗传算法; 移动机器人; 路径规划; 交叉算子; 变异算子

## Mobile Robot Path Planning Based on Improved Genetic Algorithm

Zhang Yi, Dai Encan, Luo Yuan

(National Engineering Research and Development Center for Information Accessibility, Chongqing University of Posts and Telecommunication, Chongqing 400065, China)

**Abstract:** In order to solve the problems of low search efficiency and easily falling into the local optimal solution in traditional genetic algorithm, an improved genetic algorithm is proposed in this paper. It adopts the simple one-dimensional code to replace the complex two-dimensional coding, which can save storage space. In the design of genetic operators, many operations such as crossover and mutation are redefined to avoid getting into the local optimum. Then the two fitness functions—collision-free path and the shortest distance—are fused into one for the following genetic optimization. In the case of the same population parameters, 100 trials are respectively developed with the method of improved genetic algorithm and traditional genetic algorithm. Among them, the improved genetic algorithm to search the optimal path gets to 95 times, and the shortest path is 20.970 6. Besides, the average searching time takes up 217 ms. While the number of traditional method to search for the optimal path reaches up to 62 times, the shortest path can be 25.071 1, and the average searching time needs 345 ms. So compared to the tests results referred above, the improved genetic algorithm is more efficient and can get a better solution than the traditional genetic algorithm.

**Keywords:** genetic algorithm; mobile robot; path planning; crossover operator; mutation operator

## 0 引言

导航技术是移动机器人研究领域的关键技术, 其中, 路径规划又是导航中最重要的任务之一。所谓路径规划, 就是指在一个含有障碍物的环境中, 按照某一性能指标 (可以是行走时间最短、路径最短或能量消耗最少等), 为移动机器人规划出一条从起始节点到目标节点的最优或者近似最优的无碰撞路径。移动机器人路径规划可分为两种类型: 一种是基于环境先验信息的全局路径规划; 另一种是基于传感器信息的局部路径规划, 后者环境信息是未知或者局部未知的, 即需要传感器来实时获取障碍物的位置、尺寸、形状等信息。全局路径规划是根据已有的环境地图搜索一条从起始节点到目标节点的最优或近似最优的免碰撞路径, 全局路径规划涉及的主要问题是环境信息的表示以及搜索策略的选择。全局路径规划通常应用在障碍物为静态障碍物、环境信息已知的环境中, 当障碍物不断运动

变化时, 全局路径规划不再奏效, 需要采用依赖传感器信息的局部路径规划方法。

国内外研究人员已经花了很长时间在移动机器人路径规划方面, 也产生了许多方法<sup>[1-2]</sup>。目前在路径规划领域应用较多的方法有: 滚动窗口规划方法、人工势场法、神经网络法、蚁群算法、Dijkstra 算法、A\* 算法等。每一种算法都有各自的思想, 并且都在不同的方面表现出了优势。但总的来看, 上述方法都存在着某一或者某些方面的不足。如算法计算量大、搜索效率低、缺乏灵活性、易于陷入局部最优解、全局路径规划的质量不高以及自适应能力差等方面。而遗传算法作为一种优化算法具有鲁棒、灵活、适应能力强等优点, 因此被广泛应用于许多优化问题中并取得了不错的效果。近十几年来, 遗传算法在解决移动机器人路径规划问题方面也得到了广泛的应用。随着研究的不断深入, 应用遗传算法进行移动机器人路径规划的不足逐渐被人们发现<sup>[3-6]</sup>, 如存在着种群规模大、搜索空间大、收敛速度慢、易于陷入局部极小点等问题。针对这些问题, 提出了一种改进的遗传算法, 编码方式上摒弃复杂的二维编码方式采用简单的一维编码, 减小了编码长度节约了存储空间, 重新定义了传统遗传算法中的交叉、变异等遗传算子, 加速了进化过程。并把免碰撞要求和最短路径要求融合成一个适应度函数, 选取适应度更高的个体, 提高了种群的适应能力。最后通过实验验证了改进遗传算法的可行性和有效性。

收稿日期: 2015-08-04; 修回日期: 2015-11-11。

基金项目: 国家自然科学基金资助项目(51075420)。

作者简介: 张毅(1966-), 男, 博士, 教授, 主要从事机器人导航技术、数据融合、信息无障碍技术方向的研究。

罗元(1972-), 女, 博士, 教授, 主要从事信号与信息处理、数字图像处理方向的研究。

## 1 遗传算法的基本原理

遗传算法是一种并行搜索算法<sup>[5-8]</sup>，它从一个随机生成的初始解开始，操作过程中按照一定的规则，如选择，复制，交叉，变异，不断迭代计算，以获得下一代种群中的个体。用适者生存优胜劣汰的原则来指导搜索过程，使适应度更强的个体不断保留下来，并逐渐演变成更多更好的近似解，最终收敛到全局满意解。

下面是遗传算法的基本步骤。

Step1: 由  $N$  个随机生成的个体组成初始种群。

Step2: 通过适应度函数计算出当前种群中每一个个体的适应度函数值

Step3: 判断算法是否满足终止条件，如果满足终止条件则转 Step8。

Step4: 按照个体适应度值的大小进行选择操作。

Step5: 根据交叉概率进行交叉操作。

Step6: 根据变异概率进行变异操作。

Step7: 如果由  $N$  个新个体组成的新一代群体已经产生，则转 Step2；否则，转 Step4。

Step8: 输出搜索结果，算法终止。

## 2 基于改进遗传算法的机器人路径规划

本部分的主要目的是提出一种用于解决移动机器人路径规划问题的改进遗传算法。主要改进点有以下几个方面：把复杂的二维编码方式简化为一维编码，适应度函数融合了最短路径要求和免碰撞要求；在选择操作过程中，将轮盘赌选择和精英选择进行了结合；优化了遗传操作中的交叉算子和变异算子；增加了新的遗传算子——插入算子和删除算子。有效防止了遗传算法在进行路径规划时易于陷入局部最优的情况。改进遗传算法主要包括以下三个方面。

### 2.1 环境建模及路径编码

本文运用栅格法来描述机器人的环境模型<sup>[3]</sup>，如图 1 所示，二维平面上的栅格区域  $E$  即为机器人的运行空间。在此环境中存在着一定数量的障碍物。将栅格区域  $E$  的左下角设为坐标原点，水平向右为  $X$  轴，竖直向上为  $Y$  轴，每一个栅格区间代表一个坐标轴上的单位长度，把栅格区域  $E$  划分成  $m \times n$  个大小相同的方形栅格。按照从左到右，从下到上的顺序，从栅格区域左下角第一个栅格开始给每一个栅格编号  $i$  (从零开始计)，则栅格编号  $i$  与坐标  $(x_i, y_i)$  互为映射关系，对应关系如公式 (1) 所示。

$$\begin{cases} x_i = i \bmod N \\ y_i = (\text{int})(i/N) \end{cases} \quad (1)$$

其中： $\bmod$  为取余运算， $\text{int}$  为取整运算， $N$  为每行的栅格数。

对机器人工作空间  $E$  和机器人本体作如下假设：

(1) 机器人的工作空间为二维结构化空间，机器人的高度不予考虑。

(2) 环境中的障碍物位置信息利用其占据的栅格号来表示，障碍物的形状、大小、位置已知且不发生变化。

(3) 规划过程中，将机器人作为质点处理。

(4) 规划过程中，机器人速度大小不发生变化。

机器人路径由多条线段组成，这些线段将移动机器人的起始节点、中间节点和目标节点相连。路径被编码成固定长度。图 1 对移动机器人的工作空间和运行路径进行了描述。

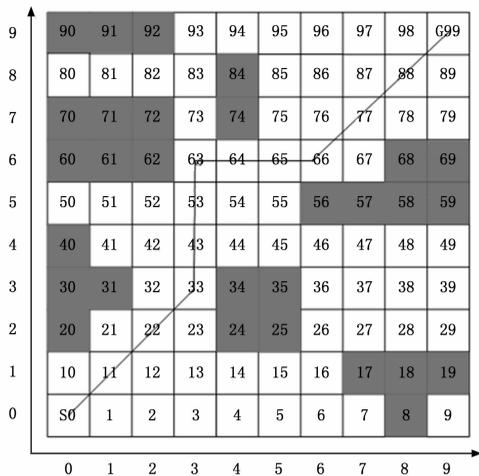


图 1 直角坐标法和序号法建立的栅格模型

路径编码策略的选择对适应度函数以及遗传算子（尤其是交叉算子和变异算子）的设计具有重要影响。路径编码策略主要有实数编码、二进制编码、自适应编码和树编码等。下述讨论中，将实数编码中的序号法与坐标法结合使用<sup>[4]</sup>，采用序号法表示机器人运动路径，因为这样的表示简单直观，可以节约存储空间，并且便于遗传算子的操作。在评估路径的适应度时，则将序号转换成坐标形式，因为坐标法更便于表示栅格的相对位置，更易于计算路径长度和检验路径可行性。

### 2.2 适应度函数

适应度函数可以有效的评价每条路径的优劣，通常适应度值的大小与路径的优劣成正比，它对遗传算法的收敛性和稳定性有着重要的影响。进行移动机器人路径规划的目的是使机器人以最短的时间顺利到达目标点<sup>[5]</sup>。因此，在进行路径规划时，适应度函数的设计应该主要满足两个条件<sup>[7-10]</sup>：机器人运行轨迹长度最短及在行进过程中可以有效的避开障碍物。与障碍物不发生碰撞是移动机器人路径规划的基本条件，是机器人在工作空间内安全行驶的必要条件。免碰撞要满足两个条件：第一，任意路径点不能落在任何一个障碍物区域内；第二，相邻两个路径点的连线与障碍物不相交。

(1) 任意路径点 (用  $p_i$  表示) 不能落在任何一个障碍物区域，用集合  $W$  表示障碍物的集合，则路径点不落在障碍物区域内的适应度函数可表示为

$$fit_{11} = \begin{cases} 1, p_i \notin W (i = 0, 1, \dots, n) \\ 0, p_i \in W (i = 0, 1, \dots, n) \end{cases} \quad (2)$$

式中， $p_i$  代表工作空间中路径上的点， $n$  为栅格数。公式 (2) 表明，路径点不在障碍物区域内时，适应度为 1，否则为 0。

(2) 设  $p_i p_{i+1}$  为机器人相邻两个路径点的连线，则其与各个障碍物区域没有交点的适应度函数为

$$fit_{12} = \begin{cases} 1, p_i p_{i+1} \cap W \neq \emptyset (i = 0, 1, \dots, n) \\ 0, p_i p_{i+1} \cap W = \emptyset (i = 0, 1, \dots, n) \end{cases} \quad (3)$$

由公式 (1)、(2) 可知，机器人在路径规划过程中，可有效避免碰撞的适应度函数为

$$fit_1 = fit_{11} \times fit_{12} \quad (4)$$

适应度函数的选取对于遗传算法来说至关重要，除了在运动过程中能够有效避开障碍物，适应度函数还要能够快速计算每条路径的长度。用公式 (5) 表示相邻两条路径点的长度

$$L_{(i)} = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (5)$$

公式 (5) 中,  $(x_i, y_i)$  表示机器人的当前位置坐标,  $(x_{i-1}, y_{i-1})$  则代表机器人前一位置的坐标。综上, 整条路径的长度可以用公式 (6) 来计算:

$$Length = \sum_{i=1}^n L_{(i)} \quad (6)$$

因此, 表示最短路径的适应度函数  $fit_2$  为:

$$fit_2 = \frac{1}{Length} \quad (7)$$

由公式 (4) 和公式 (7) 可知, 用于表示移动机器人路径规划的最优适应度函数是:

$$fit = C_{max} + fit_1 + \alpha \times fit_2 \quad (8)$$

公式 (8) 中,  $C_{max}$  的取值与地图的复杂程度以及起始点和目标点之间的直线距离有关,  $\alpha$  为惩罚系数, 把公式 (8) 作为移动机器人路径规划的适应度函数, 既可以有效避免碰撞又可以使得路径最短。

## 2.3 遗传算子

### 2.3.1 选择算子

执行选择操作的主要目的是使得群体中适应度大的个体被复制下来, 去除适应度小的个体, 并且在这一过程中群体的规模保持不变。然后, 将执行选择操作选出来的个体再进行交叉和变异操作从而产生新的后代。选择策略是确保机器人以最佳适应度存活下来的根本原因。本文采用轮盘赌和精英保留策略相结合的方法来执行选择操作<sup>[11-12]</sup>。

### 2.3.2 交叉算子

交叉操作是遗传算法中基因重新排列的基本机制, 在遗传算法中它被应用于选择操作之后。在进行遗传操作的时候, 交叉点的选择是随机的, 可以采用单点交叉也可以采用多点交叉的方式。在两个解决方案之间交换部分基因片段从而产生新的基因组合, 进而创造出新的解决方案。本文采用单点交叉的方式执行交叉操作<sup>[9]</sup>。分为两种情况。

#### (1) 共有节点处一点交叉:

共有节点处一点交叉是指两个待交叉的个体存在相同的节点 (不包括起始点和目标点), 并且两个个体在该节点之前 (或之后) 的基因不同。则将此节点作为交叉节点进行交叉。举一个例子来说明该交叉操作的过程。假设两个待交叉的父代个体为  $V_1$  和  $V_2$ , 其中, 父代个体  $V_1$  (0, 2, 52, 53, 93, 99); 父代个体  $V_2$  (0, 33, 53, 64, 66, 99)。

可以看出, 两个父代个体  $V_1$  和  $V_2$  中存在共有节点 53, 而且两个父代个体在该节点之前和之后的基因均不相同。因此, 可以采用共有节点一点交叉的方式进行交叉。交叉之后所得的子代个体为:

子代个体  $V_1'$  (0, 2, 52, 53, 64, 66, 99)

子代个体  $V_2'$  (0, 33, 53, 93, 99)

#### (2) 潜在节点处一点交叉:

所谓潜在节点是指在一个待交叉的父代个体中出现的节点, 在另一个待交叉的父代个体中没有出现, 但是另一父代个体中存在两个相邻节点的连线经过此节点。只需要将其插入到另一个父代个体中连线能够经过此节点的相邻节点之间, 便可采用共有节点一点交叉的方法进行交叉。

假设父代个体  $V_1$  和  $V_2$  分别为:

父代个体  $V_1$  (0, 2, 52, 53, 93, 99)

父代个体  $V_2$  (, 0, 33, 63, 66, 99)。

可以看出, 节点 53 在父代个体  $V_1$  中出现, 而没有在父代

个体  $V_2$  中出现, 但是父代个体  $V_2$  中存在两个相邻节点 (节点 33 和节点 63) 的连线经过节点 53, 因此, 节点 53 可以作为潜在节点进行交叉。首先, 将节点 53 插入到父代个体  $V_2$  中, 此时父代个体  $V_2$  变为 (0, 33, 53, 63, 66, 99)。然后, 按照共有节点处一点交叉的方法进行交叉, 得到的子代个体分别为:

子代个体  $V_1'$  (0, 2, 52, 53, 63, 66, 99);

子代个体  $V_2'$  (0, 33, 53, 93, 99)。

### 2.3.3 变异算子

基因突变是一个过程, 它是指应用一些随机的变化方式使得染色体上的位发生小的改变, 从而产生新的基因片段。这一操作对维持种群的多样性是非常必要的。本文对交叉后产生的子代个体基因按小概率  $p_m$  扰动进行变异, 通常取很小的值, 一般为 0.001~0.4, 这里取 0.01。

### 2.3.4 插入算子

引入插入算子的作用是使自由栅格将间断路径转化为连续路径。判断两个相邻序号的路径点  $p_k, p_{k+1}$  是否连续, 可用如下方法:

$$\Delta = \max\{\text{abs}(x_{k+1} - x_k), \text{abs}(y_{k+1} - y_k)\} \quad (9)$$

若  $\Delta = 1$ , 则  $p_k$  与  $p_{k+1}$  连续, 否则, 不连续。如果不连续则按中值法插入候补点。

$$\begin{cases} x'_k = \text{int}\left[\frac{1}{2}(x_k + x_{k+1})\right] \\ y'_k = \text{int}\left[\frac{1}{2}(y_k + y_{k+1})\right] \\ p'_k = x'_k + 10y'_k \end{cases} \quad (10)$$

若计算得到的  $p'_k$  为自由栅格, 则可直接插入。若  $p'_k$  是障碍栅格, 则选择一个与  $p'_k$  最近的自由栅格作为新的候补插入点; 如果找不到符合条件的候补插入点, 则将该个体舍弃。否则用新的候补插入点插入到  $p_k$  与  $p_{k+1}$  之间。

重复执行上述插入过程, 直至将间断路径转化为连续路径或因找不到新的候补插入点而舍弃该个体。

### 2.3.5 删除算子

增加删除算子的目的是删除同一路径中出现的相同序号以及相同序号之间可能存在的冗余序号, 以简化机器人的行走路径。

## 3 实验结果及分析

### 3.1 实验结果

本部分证明了改进的遗传算法在移动机器人路径规划问题上的实现。描述了实验所需要的软硬件环境。设定了遗传进化过程中种群的各项参数, 并对产生的实验结果进行了分析。本文的实验是在 1.7 GHz, 酷睿 i5 处理器的计算机平台上, 由 MATLAB2012b 64Bit 编制完成的。

在实验过程中, 种群的各项参数如下: 种群规模  $M=60$ , 交叉操作的概率  $p_c=0.6$ , 变异操作的概率为  $p_m=0.01$ , 最大进化代数  $T=100$ 。移动机器人的工作空间由  $15 \times 15$  的栅格地图构成。移动机器人的起始点和目标点分别用 S 和 G 表示。实验结果如图 2 和图 3 所示, 其中, 实线是由遗传算法计算出的最优路径。

从图 2 和图 3 能够看出, 改进的遗传算法相比于传统的遗传算法能够搜索到更短的最优路径。改进的遗传算法产生最优路径的适应度函数值和收敛代数分别为:  $F_1=10.074$ ,  $N_1=21$ ; 传统遗传算法产生最优路径的适应度值和收敛代数分别

为:  $F_2 = 10.068$ ,  $N_2 = 57$ 。

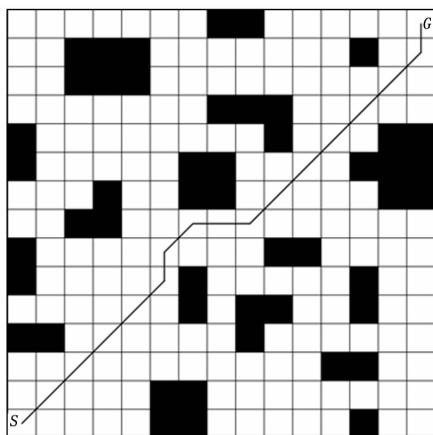


图 2 改进遗传算法实验结果

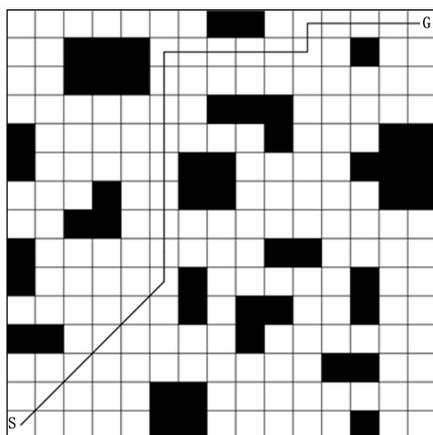


图 3 传统遗传算法实验结果

### 3.2 算法对比分析

在与传统遗传算法的对比过程中, 移动机器人的工作空间如图 2 或图 3 所示, 在实验的过程中, 两种算法的控制参数相同, 包括种群规模的大小、发生交叉和变异的概率、最大遗传代数等。图 4 为两种算法对比的实验结果, 从图中可以看到, 改进的遗传算法能够更快的收敛到全局最优解, 而且在收敛的过程中能够有效克服传统遗传算法易于陷入局部最优的问题。此外, 还对路径规划的时间、所获取的最短路径的长度以及产生最优路径的比例进行了比较, 结果如表 1 所示。从表中可以看到, 改进遗传算法比传统遗传算法所获得的路径更短, 而且运行时间少于后者。此外, 在获得最优路径比例方面, 改进遗传算法远远高于传统遗传算法。当移动机器人路径规划的环境变得复杂时, 更能显现改进遗传算法的优点。因此, 所提出的改进遗传算法在搜索效率及获取最优解方面都优于传统遗传算法。

表 1 改进遗传算法与传统遗传算法性能比较

	最短路 径长度	搜索时间/ ms	最优路径 比例/(%)
改进遗传算法	20.970 6	217	95
传统遗传算法	25.071 1	345	62

## 4 结束语

本文提出了一种改进的遗传算法, 并采用该算法对移动机器人路径规划问题进行了研究。在存在障碍物的栅格环境中,

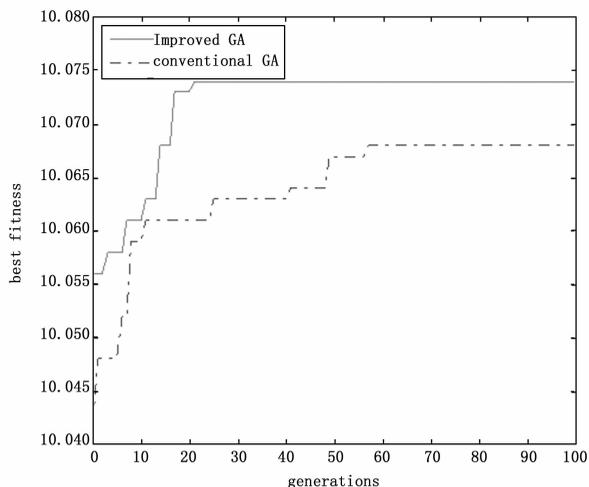


图 4 改进遗传算法与传统遗传算法最优适应度和收敛代数比较

使用改进的遗传算法进行路径规划, 并且获得了最优路径。正如实验部分所证明的那样, 改进的遗传算法相比与传统遗传算法在路径规划方面具有明显优势, 具体表现在搜索的效率更高、规划的最优路径更短。采用序号编码的方法, 缩短了染色体的长度, 简化了计算过程, 提高了搜索效率。特殊的交叉算子的使用则利于寻找到最短的可行路径。

### 参考文献:

- [1] 张捍东, 郑睿, 岑豫皖. 移动机器人路径规划技术的现状与展望 [J]. 系统仿真学报, 2005, 17 (2): 439-443.
- [2] 朱大奇, 颜明重. 移动机器人路径规划技术综述 [J]. 控制与决策, 2010, 25 (7): 962-967.
- [3] 周兰凤, 洪炳熔. 用基于知识的遗传算法实现移动机器人路径规划 [J]. 电子学报, 2006, 34 (5): 911-914.
- [4] 陈曦, 谭冠政, 江斌. 基于免疫遗传算法的移动机器人实时最优路径规划 [J]. 中南大学学报 (自然科学版), 2008, 39 (3): 577-583.
- [5] Kala Rahul. Multi-robot path planning using co-evolutionary genetic programming [J]. Expert Systems with Applications, 2012, 39 (3): 3817-3831.
- [6] 陈华华, 杜歆, 顾伟康. 基于遗传算法的静态环境全局路径规划 [J]. 浙江大学学报 (理学版), 2005, 32 (1): 49-53, 61.
- [7] Hsu-Chih Huang, Ching-Chih Tsai. Global path planning for autonomous robot navigation using hybrid metaheuristic GA-PSO algorithm [A]. SICE Annual Conference 2011 [C], 2011: 1338-1343.
- [8] 郝博, 秦丽娟, 姜明洋. 基于改进遗传算法的移动机器人路径规划方法研究 [J]. 计算机工程与科学, 2010, 32 (7): 104-107.
- [9] Hu Y R, Simon X. Yang, Li Z X. A knowledge based genetic algorithm for pathing in unstructured mobile robot environments [A]. Proceedings of the 2004IEEE International Conference on Robotics and Biomimetics [C], 2004: 767-772.
- [10] 唐国新, 陈雄, 袁杨. 基于改进遗传算法的机器人路径规划 [J]. 计算机工程与设计, 2007, 28 (18): 4446-4449.
- [11] 卢瑾. 基于遗传算法的移动机器人路径规划研究 [D]. 杭州: 浙江工业大学, 2005.
- [12] Masoud Samadi, Mohd Fauzi Othman. Global path planning for autonomous mobile robot using genetic algorithm [A]. 2013International Conference on Signal-Image Technology&Internet-Based Systems [C], 2013: 726-730.