

基于改进 A* 算法的分拣搬运机器人路径规划

蔡旻, 刘士凡¹, 陶重萍¹, 李春光², 孙云飞¹, 班建民¹

(1. 苏州科技大学 电子与信息工程学院 建筑智慧节能江苏省重点实验室, 江苏 苏州 215009;

2. 常州工学院 计算机信息工程学院, 江苏 常州 213002)

摘要: 为了解决分拣搬运机器人在路径规划过程中, 遇到目标点众多的情况时存在路径寻优效率低、容易出错等问题; 针对 A* 算法存在多个最小值时, 无法实现路径最优化的问题进行研究, 提出一种将蚁群算法与 A* 算法相结合的改进 A* 算法; 首先使用 A* 算法筛选出一条最优化的路线来分布信息素, 从而简化 A* 算法在路径规划上的运算; 其次以筛选出的路线为基础, 针对不同情况结合蚁群算法设计了 3 种通用方案, 以此为基础进行具体的路径规划, 从而解决 A* 算法本身存在的容易带入大量重复数据的问题; 通过仿真与实际实验验证了本文提出的改进的 A* 算法能够满足自动分拣搬运的需求, 值得推广与使用。

关键词: 分拣搬运机器人; 蚁群算法; 路径规划; A* 算法

Path Planning of Sorting and Handling Robot Based on Improved A* Algorithm

Cai Min¹, Liu Shifan¹, Tao Chongben¹, Li Chunguang², Sun Yunfei¹, Ban Jianming¹

(1. College of Electronics and Information Engineering, Suzhou University of Science and Technology,

Suzhou 215009, China; 2. School of Computer and Information Engineering,

Changzhou Institute of Technology, Changzhou 213002, China)

Abstract: In order to solve the sorting handling optimal robot path planning in the process, often encounter target many situations exist path planning problems such as low efficiency, error prone. Multiple minimum value of A* algorithm cannot be achieved under the condition of path optimization problems are studied, and an improved A* algorithm which combine ant colony algorithm and A* algorithm is proposed. Firstly, the A* algorithm is used to select the optimal route, distribution information, so as to simplify the operation of A* algorithm. Secondly, based on the selected route, according to different situation combined with ant colony algorithm to design three kinds of general scheme based on a specific path planning, and then solve the A* algorithm itself easily into a large number of duplicate data problems. The proposed improved A* algorithm of this paper is verified by simulation and practical experiment can meet the demand of automatic sorting handling, which is worth popularizing and using.

Keywords: sorting and handling robot; ant colony algorithm; path planning; A* algorithm

0 引言

随着我国工业生产的不断进步与发展, 分拣搬运机器人越来越多的参与到了工厂的货物分拣和搬运等应用中^[1]。它的作用不局限于处理货物的搬运问题, 更能够减轻人力工作的压力, 提高整体的工作效率, 从而降低不必要的成本。张岩岩等人针对搬运机器人的路径寻优问题, 通过将人工免疫算法中的相关参数进行迭代优化后, 将其与蚁群算法相结合, 提出了一种改进的蚁群路径规划算法^[2]。然而, 该算法虽然能够较为准确地找到最优路径, 但是该算法中的参数迭代优化计算量较大, 影响到了路径规划算法的实时性。吕太之等人提出在路径规划过程中加入同步构造可视图环节, 有效地降低了路径规划的耗时^[3]。然而该算法能够扩展的节点数量比较有限, 无法用于范围较大的障碍物环境。孙炜等人在路径规划时首先采用栅

格方法建立环境模型, 接着将 A* 算法用于初步的路径规划, 并按较小的分割步长进行分割, 剔除中间路径点的方法实现最优路径规划^[4]。然而, 该算法中的分割步长需要人工确定, 无法实现自主的路径规划。

针对上述问题, 本文提出了一种将蚁群算法与 A* 算法相结合的改进了 A* 算法用于解决自主最优路径规划问题, 并自主开发设计了一种以改进 A* 算法为核心的分拣搬运机器人用于算法验证。该机器人使用 STM32 单片机为核心控制器, 能够为电机运动、机械臂的夹取提供灵活的控制。机器人使用的颜色传感器, 具备在一定的误差范围内能够分辨不同类型障碍物的功能。机器人采用 S 型加减速控制法来控制运行速度, 使用 PID 控制法对电机进行调控, 使得整体机器的运行更加稳定^[5]。

1 分拣搬运机器人的硬件设计

1.1 分拣搬运机器人的整体结构

机器人以 STM32 芯片的电路板为平台搭建。驱动模块控制电机的行动和机械臂的抓取。机械臂上部安装了铁针, 用来配合机械臂夹取。机器臂的底部装有颜色传感器, 用来辨识不同的物料。机器人共有 3 个电机, 左右轮电机作为主动轮运行, 中间电机则作为万向轮对方向进行调整。

收稿日期: 2017-09-01; 修回日期: 2017-09-27。

基金项目: 国家自然科学基金(61401297); 江苏省自然科学基金(BK20140283); 江苏省高等学校自然科学研究项目;(17KJB120011); 苏州市科技项目(SYG201708)。

作者简介: 蔡旻(1978-), 男, 江苏张家港人, 硕士, 讲师, 主要从事环境建图与机器人自主导航方向的研究。

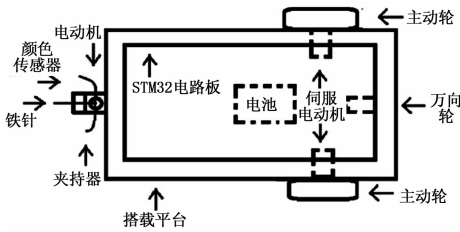


图1 搬运机器人的结构图

1.2 颜色传感器模块设计

本文设计的机器人使用了 TCS230 颜色传感器, CMOS 电路上集成了硅光电二极管和电流频率转化器, 且单一芯片上有红、蓝、绿 3 种滤光器, 在没有 A/D 转换电路的情况下, 能够与微处理器直接连接。传感器上共 64 个光电二极管, 分别为红色、蓝色、绿色、全透。当选定一种色彩的滤波器后, 便只能通过这类的颜色, 其余颜色不能够通过, 从而得到此颜色的光强, 并能够进行区分。

在颜色识别开始前, 需要对颜色识别模块进行白平衡。白平衡调整检测系统, 使其对三原色的敏感度相同, 从而确保测量的准确性。当颜色的计算值在一定误差范围内, 可以得出对应颜色的检测结果, 从而能够正确判断不同的障碍物。

本文中的颜色传感器采用笛卡尔距离算法来辨别不同颜色。公式如下:

$$d = \sqrt{(R_u - R_r)^2 + (G_u - G_r)^2 + (B_u - B_r)^2} \quad (1)$$

R_r 、 G_r 、 B_r 为不同颜色的传感器参考值, R_u 、 G_u 、 B_u 为未知色彩的传感器实际测量值。

2 A* 算法原理

A* 算法是一种直接的启发式搜索算法, 常用于在静态路网中求解最短路线^[6]。启发式搜索就是在一定范围空间内, 每一个地位都进行搜索, 得出最好的位置, 再以此持续进行搜索, 直到到达目标点。A* 算法的路线规划由估价函数决定, 每走一段距离就需要进行估算。在不断前进的路线上, 舍去死节点, 而保留最优的节点, 并且不断比较新节点与旧节点的优劣。得出价值最低的节点, 向此节点前进, 并进行下一步的计算, 最终逐步到达目标点。公式如下:

$$f(n) = g(n) + h(n) \quad (2)$$

$f(n)$ 为全局环境中最优理论消耗值, $g(n)$ 为实际操作中额外的消耗值, $h(n)$ 为理论上最优化路线的消耗值, 整体估价函数 $h(n)$ 占主导地位。因此简化 $h(n)$ 的具体算法, 便可在理论上减小 $f(n)$ 的量。为了匹配本文的模拟环境, 采用欧几里得距离算法作为启发函数, 能够在复杂环境里计算出最优解。欧几里得距离算法即两点之间的距离, 能够简化二维模型里的距离问题, 从而化简整体的运算。公式如下:

$$h(n_i) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (3)$$

其中: x_i 、 y_i 设定为起始位置, x_j 、 y_j 设定为目标位置^[5-6]。

3 改进的 A* 算法

3.1 蚁群算法

蚁群算法是一种启发式全局优化算法。此算法来源于意大利学者 Dorigo、Maniezzo 等人对蚁群搬运食物路线的研究^[7]。蚂蚁在搬运食物的路径上, 会留下信息素。蚁群对信息素具有

很强的感知力, 根据之前蚁队留下的信息素, 后来的蚁队能够花费更少的时间到达食物点。蚁群算法简单而实用, 但相较于 A* 算法, 多了盲目性与不稳定性。

3.2 与蚁群算法相结合改进 A* 算法

A* 算法在路径规划中有极高的价值, 但算法本身存在一定的问题, 在路径规划中存在多个最小解的情况下, 无法做到选择最优化。蚁群算法是一种广度优先搜索, 在进行初次搜索时, 采用信息素均匀分布, 各条路线上的信息素均定义为一个常数 C 。因此在搜索的初期, 收敛速度会较慢, 算法的搜索时间也较长。A* 算法的启发函数能够明确从起点到目标点搜索的方向, 本文中利用简化的 A* 算法, 对地图初始位置信息素进行设置, 从而加快算法初期收敛速度, 避免了大量无意义的计算。

本文介绍算法的主要改进在于使用 A* 算法筛选出一条合适的路线来分布信息素, 从而简化 A* 算法在之后路线规划上的计算。从起点出发, 选取多个适合的无障碍节点, 从中选取估价函数最低的节点。再以此节点为中心, 选取下一个估价函数最低的节点, 从而能够顺利筛选出一条从起点到目标点估价函数最低的路线。而简化了新节点与旧节点的比较的过程。

将 $f(n)$ 作为评价函数得出的最优化的路线 R 。将此路线的初始信息素设置为:

$$X(R) = kc \quad (4)$$

c 为其他路线信息素的分布, k 为大于 1 的常数。

3.3 改进后 A* 算法的路径规划

设定两个决定数 a 、 b 。 ab 属于 $(0, 1)$ 。设 p 为决定概率。 O 为最终决定节点。

情况一: 蚁群算法受信息素与启发信息的影响, 所以在 $a < p$ 、 $b < p$ 条件下, 选择信息素与启发信息乘积最大的节点。公式如下:

$$O = \max \{ [X(m, n)^a] * [Y(m, n)]^b \} \quad (5)$$

情况二: 在决定参数 $a < p$ 、 $b > p$ 条件下, 可以采用轮盘赌策略选择节点。公式如下:

$$O = \frac{[X(m, n)^a] * [Y(m, n)]^b}{\sum_{jk(r) \in o} [X(m, n)^a] * [Y(m, n)]^b} \quad (6)$$

情况三: 当决定参数 $a > p$ 条件下, 在栅格中随机选择非障碍物的栅格作为前进路线。公式如下:

$$O = \text{random}, jk(r) \in n \quad (7)$$

m 对应信息素因数, n 对应能见度因数, X 对应信息数, Y 对应能见度或者启发因数, $jk(r)$ 对应下一步到达的栅格位置。

采用改进后的 A* 算法, 结合了蚁群算法, 指引出一条高于平均信息素分布的一条路线, 解决了 A* 算法在存在多个最小值时的路径优化问题, 并且让整个算法系统更加灵活, 能够应对各种复杂的环境^[8]。

4 场地模拟与路径规划

4.1 场地模拟

在实际的搬运过程中, 物料与生产车间总有一定的距离, 且沿途都会有相应的阻碍物。在仿真实验中设置红圈为障碍物, start 为起始点, target 为目标点。机器人需在方格内避开红圈, 并找出一条最短的行径, 从而实现物料搬运。

4.2 路径规划

分拣搬运机器人将从起点 start 出发, 避过途中设置的不

同的障碍，将物料送入目标位置 target，从而完成搬运任务。在相同大小 10 × 10 的栅格模型中进行比较，如图 2 所示。从图 3、图 4 和图 5 这三组对应不同方法的实验中可以看出，在图 2 所示的障碍物环境条件相同的情况下，机器人都能够成功规避相应的障碍物，最终准确把物料搬运到目的地。

对比 3 种算法，也能很容易发现在前期的运动轨迹中 3 种算法并没有太多区别。在运动后期，图 3 中的 A* 算法在路径规划上能做到避开障碍物，将货物搬运到指定目标点，但在搬运的最后阶段，沿直角行走，并不能做到路线的最优化。图 4 中的人工势场法能够在运动的最后阶段简化路线，但在路线最后的规划上受最后障碍物分布的影响，对最后路线的选择有一定的干扰，因此不能做到抉择最优化。而图 5 中本文提出的改进 A* 算法在规避障碍的同时也做到了路线的最优化。

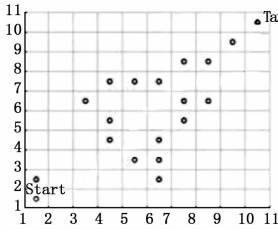


图 2 障碍物分布图

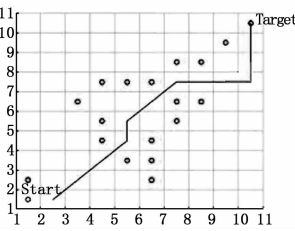


图 3 A* 算法仿真

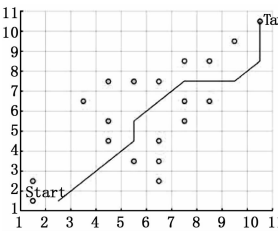


图 4 人工势场法仿真

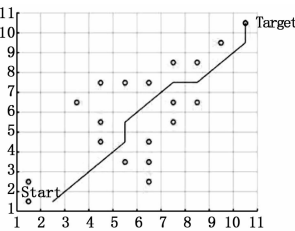


图 5 改进后 A* 算法仿真

最终实验结果的区别源于这 3 种算法本身的差异。人工势场法在路径规划后期受到障碍物影响较大，容易出现无法实现路径最优化的情况^[9-10]。A* 算法由于带入大量重复数据和复杂的估价函数，未能找到最佳路径。而本文提出的改进 A* 算法利用在路径上分布信息素的方法，简化了节点抉择最优解的计算量，因此最后的障碍物安放位置不会影响到最优路线的规划。

4.3 实际路径规划实验

将本文设计的分拣搬运机器人用于实际实验，其具体结构如图 6 所示。由夹持器、电机、STM32 开发板、颜色传感器和蓄电池组成。其中铁针配合着半圆形塑料臂组成夹持器夹取物料。机器人整体结构简单，运动流畅，能够在实际实验环境中完成分拣搬运任务。

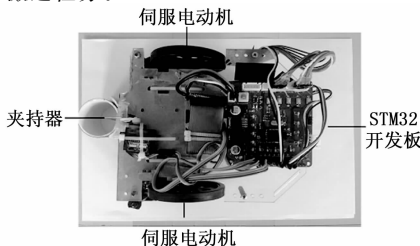


图 6 本文设计的分拣搬运机器人的实物图

在相同的实验环境下，分别采用了 A* 算法、人工势场法算法、改进后 A* 算法，分别进行了实际的实验，从搬运时间、出错率、能耗 3 个方面进行比较。在多次实验后得到以下数据。

表 1 3 种算法在时间、出错率、能耗率对比表

	搬运时间/%	出错率/%	能耗率/%
A* 算法	76	5	13
人工势场法	72	12	11
改进 A* 算法	57	2	8

从表 1 中可以看出，A* 算法与人工势场法在时间方面相差不多，出错率方面 A* 算法较低，而能耗较大。人工势场法出错率较高，但能耗略低。改进后的 A* 算法时间大大缩短，而且在出错率和能耗方面都优于 A* 算法和人工势场法，本文提出的改进 A* 算法比 A* 算法、人工势场法更适合分拣搬运机器人使用。

在实际实验中，机器人从图 7 (a) 起点位置出发，将货物夹紧后，开始搬运任务。如图 7 (b) 与图 7 (c) 所示，机器人顺利规避了沿途设置的障碍。选择本文提出的改进 A* 算法 3 种情况中的一种，自主的规划最优行进路径。最终，图 7 (d) 中机器人搬运货物并沿着最优路线准备到达目的地，从而完成了搬运任务。机器人在实验过程中没有与障碍物发生碰撞，且最优路径选择准确率高、速度快。从而验证了本文提出的改进 A* 算法在路径规划时具有较强的准确性与实用性。

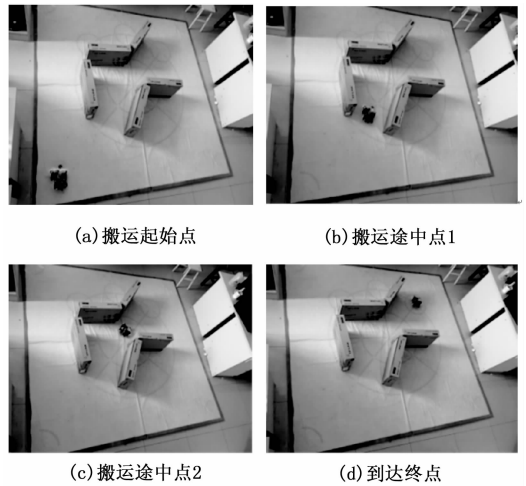


图 7 搬运机器人自主路径规划实验

5 结论

本文设计的基于 STM32 单片机控制的分拣搬运机器人，在 2016 年中国机器人大赛暨 RobotCup 公开赛获得了分拣搬运组一等奖的成绩。本文利用该平台围绕改进后的 A* 算法进行模拟和实际实验，验证了改进后的 A* 算法能够很好地提高机器人规避障碍能力。然而由于本文采用的机器臂较小，一次搬运物料有限，当需要搬运的物料较多时，效率便会有所降低。因此，本文设计的分拣搬运机器人仍需进一步改进，不断提升其实用性。

(下转第 170 页)