

基于改进 A* 与 DWA 算法融合的温室机器人路径规划

劳彩莲^{1,2} 李 鹏¹ 冯 宇¹

(1. 中国农业大学现代精细农业系统集成研究教育部重点实验室, 北京 100083;
2. 中国农业大学农业农村部农业信息获取技术重点实验室, 北京 100083)

摘要: 根据温室环境下移动机器人作业的实时路径规划要求,提出一种基于改进 A* 算法与动态窗口法相结合的温室机器人路径规划算法。针对传统 A* 算法搜索算法拐点过多的问题,对关键点选取策略进行改进,融合动态窗口法,构建全局最优路径评价函数,采用超声传感器进行局部避障,实现实时最优的路径规划。仿真实验结果证明,与传统 A*、Dijkstra、RRT 算法相比,基于改进 A* 算法的路径更为平滑和高效。真实环境下实验表明,移动机器人能够实现自主导航,跟踪误差保持在 0.22 m 以内、定位误差不大于 0.28 m,能够满足实际需求。

关键词: 温室机器人; 路径规划; 改进 A* 算法; 动态窗口法; 算法融合

中图分类号: TP242.6 **文献标识码:** A **文章编号:** 1000-1298(2021)01-0014-09 **OSID:** 

Path Planning of Greenhouse Robot Based on Fusion of Improved A* Algorithm and Dynamic Window Approach

LAO Cailian^{1,2} LI Peng¹ FENG Yu¹

(1. Key Laboratory of Modern Precision Agriculture System Integration Research, Ministry of Education, China Agricultural University, Beijing 100083, China
2. Key Laboratory of Agricultural Information Acquisition Technology, Ministry of Agriculture and Rural Affairs, China Agricultural University, Beijing 100083, China)

Abstract: Path planning is the premise of greenhouse robot operation, and an optimal continuous barrier free path is planned with great significance. An algorithm based on the combination of improved A* algorithm and dynamic window approach was proposed to solve real-time path planning of mobile robot in greenhouse. The core was based on the search algorithm of traditional A* algorithm. Aiming at the problem of too many inflexion points, the key point selection strategy was improved. The dynamic window method was integrated to construct a global optimal path evaluation function. Local obstacle avoidance was achieved through ultrasonic sensors to achieve real-time optimal path planning. The simulation results showed that compared with the traditional A*, Dijkstra, RRT algorithms, the improved A* algorithm had a smoother path and higher efficiency, which was conducive to the motion control of the robot in the greenhouse, which showed the effectiveness of the algorithm. Considering the size of the robot, the grid map of the real environment was expanded to ensure the safety of the path. The experimental results showed that the fusion algorithm can satisfy the smoothness of path and effectively avoid obstacles. The mobile robot can achieve autonomous navigation, and the tracking error was kept within 0.22 m, and the positioning error was no more than 0.28 m, which met the actual needs. The research result had an important reference value for the application of greenhouse mobile robot navigation.

Key words: greenhouse robot; path planning; improved A* algorithm; dynamic window approach; fusion algorithm

0 引言

随着我国农业的发展,温室技术变得高度智能

化和高效化。在植株高度密集、高温和高湿度甚至有毒的温室环境下,不利于长时间进行人工作业,再加上劳动力不足和成本增加的影响,温室机器人应

用越来越多^[1]。利用移动机器人完成温室自主化作业符合精细化农业的需求,路径规划是移动机器人在温室作业自主导航的重要前提^[2-4]。根据机器人的作业范围,可以将路径规划分为基于静态全局环境的全局路径规划和基于局部障碍的局部路径规划^[5]。常见的全局路径规划算法包括以Dijkstra^[6-7]、A*^[8-9]为代表的搜索规划算法,以RRT^[10]为代表的采样规划算法和以遗传算法^[11]为代表的启发式规划算法。依据实时传感器数据的局部路径规划算法有人工势场法^[12]和动态窗口法^[13]。

Dijkstra算法采用遍历搜索方式,规划节点数较多,节点网络非常庞大,算法效率低下^[14]。在Dijkstra算法的基础上,A*算法引入目标点到当前点的估计代价,根据估计代价决定路径搜索方向,提高了算法效率^[15]。A*算法在已知环境下能快速实现移动机器人无碰、最短全局路径规划,主要通过节点状态检测和简单的估值功能规划出一条最佳的安全道路。但是,该算法忽略了机器人的运动约束,不能保证曲率连续,导致运动参量在拐点处发生跳变。文献[16]动态计算出机器人的旋转方向及角度,简化了路径,但增加了运行时间,实时性不高。文献[17]使用跳点搜索减小了算法搜索面积,加快了速度,但是规划出来的路径拐点过多。文献[18]采用直线-圆弧策略对路径进行平滑处理,消除了锯齿效应。文献[19]使用微分方法减少了拐点数,但数学计算过多,比较耗时。文献[20]优化了A*算法的启发函数,改进了关键点的选取策略,减少了冗余的路径点。文献[21]提出对邻域进行扩展,将8邻域扩展到24邻域,使机器人可以小角度行进,从而轨迹更加平滑。但这些改进方法没有考虑未知的障碍物,不具备动态避障能力。

动态窗口法^[13,22-23](Dynamic window approach, DWA)通过结合传感器的数据进行在线实时局部路径规划,具有良好的避障能力,适用于动态环境中的机器人路径规划,但并不能满足全局路径的最优。将局部路径规划的算法结合到全局路径规划之中,在实际作业过程中,既考虑到环境中的障碍物,又能保证路径规划的完备性、实时性,满足机器人的运动约束。

本文提出基于改进A*算法与动态窗口法结合的温室机器人路径规划算法,利用选取策略对路径进行优化。通过结合动态窗口算法与超声波信息进行局部避障。在一维弦机器人操作系统上对改进A*算法与Dijkstra算法、A*算法、RRT算法进行仿真对比,验证改进A*算法的有效性。考虑机器人

的实际尺寸,在真实环境下对栅格地图进行膨胀化处理,并结合超声波进行局部避障,完成真实环境下的机器人自主导航任务,并对算法的可行性进行验证。

1 基于改进A*算法的路径规划

1.1 传统A*算法

A*算法采用启发式搜索与广度优先算法结合,是静态环境中用于求解最优路径最有效的直接搜索算法。A*算法通过一个代价函数 $F(n)$,选择搜索方向,从起点开始向周围扩展,通过启发函数 $H(n)$ 计算得到周围每个节点的代价值,选择最小代价值作为下一个扩展点,重复这个过程,直到到达终点,生成从起点到终点的路径。在搜索过程中,由于路径上的每一个节点都是具有最小代价节点,得到的路径代价是最小的。A*算法的代价函数为

$$F(n) = G(n) + H(n) \quad (1)$$

式中 $G(n)$ ——从起点到当前节点的实际路径代价

$F(n)$ ——当前节点所在路径从起点到终点预估的路径代价总和

A*算法在执行路径规划时主要用2个表实现节点的扩展和最优点的选择,即通过Openlist表和Closelist表来记录节点。其中Openlist表作用是保存搜索过程中遇到的扩展节点,同时将这些节点按代价进行排序,取出 $F(n)$ 值最小的节点,作为当前节点;然后将当前节点的所有邻节点按邻节点规则加入到Openlist表中。

1.2 A*算法优化策略

传统A*算法根据载入的地图,按照8邻域搜索得到一系列的点,规划出的路径转折次数太多,包含了一部分冗余点。如果转折点过多,移动机器人每次只能移动很短的距离,会造成机器人卡顿现象。在实际的机器人移动过程中,单次指令下,机器人的运动距离越长,机器人移动流畅度越好,产生的误差也会越少。这些问题造成在机器人的运动中平滑性欠缺,因此本文在传统A*算法基础上进行关键点的路径优化策略。该策略能够在栅格地图上求解出更优的路径,移动机器人路径执行具有了更高效率。

传统A*算法规划的路径如图1a所示,其中起始点为绿色方格,终点为红色方格,灰色方格为规划路径,从图中可以看出,路径转折点过多。

本文先从A*算法规划出的轨迹获得一系列的点,然后选取当前节点运动方向相同的点,最终优化成一条线段。通过相邻节点的斜率对机器人

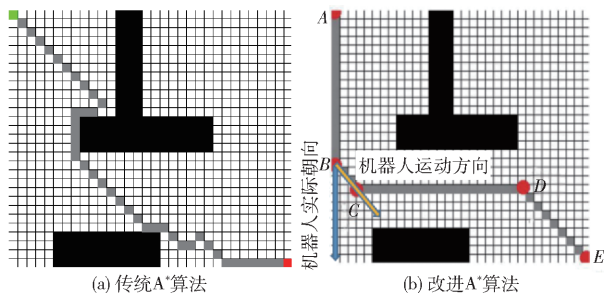


图1 传统A*算法与改进A*算法路径规划对比图

Fig. 1 Comparison between traditional A* algorithm and improved A* algorithm in path planning

的运动方向进行判断。从路径规划的第2个节点开始,如果当前节点与前一个节点的运动方向一致,则认为前一个节点为冗余的,删除前一个节点,依次遍历所有路径的节点,删除所有的冗余点。机器人只需执行包含起点、拐点的一系列路径点,减少向机器人下达运动命令的次数,保证了机器人执行路径平滑性。

经过关键点选取策略处理后的路径如图1b所示,其中点A到点B有18格子,机器人的运动方向都是朝下,将A与B之间的点优化成2个点,即点A和点B,删除AB之间的冗余点,更新路径,使得机器人运动较为流畅。使用相同的优化策略,形成BC、CD、DE段路径,最终将路径优化成没有冗余点的最优路径。经过改进A*算法的路径平滑后,图1a中的多余转折点被消除,得到如图1b所示的路径。

2 动态窗口算法

在结构化温室环境中,作物按照一定的规则进行放置,其相对位置已知,但存在一些易变因素,很难获得完整的环境信息。为避免损害植物、保证机器人安全性,本文在全局路径规划的基础上,采用动态窗口法进行局部路径规划。动态窗口法根据机器人超声传感器检测到局部环境问题,进行实时的路径规划,具有良好避障能力。载入全局地图后,通过改进A*算法进行全局路径规划,下达给机器人运动命令开始运动。由于机器人本身的运动学约束和环境因素影响,造成机器人在运动过程中产生误差。动态窗口法主要是在速度空间内采样线速度和角速度,并根据机器人的运动学计算预测其下一时间间隔的轨迹。对其获得的待评价轨迹进行评分,从而获得更加安全、平滑的最优局部路径。

2.1 机器人运动模型

动态窗口法将移动机器人的位置控制转换为速度控制^[24]。在利用速度模式对机器人运动轨迹进行预测时,首先需要对机器人的运动模型进行分析。移动机器人采用的是两轮差速模型, $v(t)$ 和 $w(t)$ 分

别代表机器人在世界坐标系下的平移速度与角速度,反映了机器人的运动轨迹。在机器人的编码器采样周期 Δt 内,位移较小,机器人作匀速直线运动,则机器人运动模型为

$$\begin{cases} x(t) = x(t-1) + v(t) \Delta t \cos(\theta(t-1)) \\ y(t) = y(t-1) + v(t) \Delta t \sin(\theta(t-1)) \\ \theta(t) = \theta(t-1) + w(t) \Delta t \end{cases} \quad (2)$$

式中 $x(t)$ 、 $y(t)$ 、 $\theta(t)$ —— t 时刻机器人在世界坐标系下的位姿

2.2 速度采样

动态窗口法将避障问题描述为速度空间中带约束的优化问题,其中约束主要包括差速机器人的非完整约束、环境障碍物的约束以及机器人结构的动力学约束。DWA算法的速度矢量空间示意图如图2所示,横坐标为机器人角速度 w ,纵坐标为机器人线速度 v ,其中 $v_{\max}$ 、 $v_{\min}$ 为机器人最大、最小线速度, $w_{\max}$ 、 $w_{\min}$ 为机器人最大、最小角速度;整个区域为 V_s ,所有白色区域 V_a 为机器人安全区域, V_d 为考虑电机扭矩在控制周期内限制的机器人可达速度范围, V_r 为上述3个集合的交集最终确定的动态窗口。

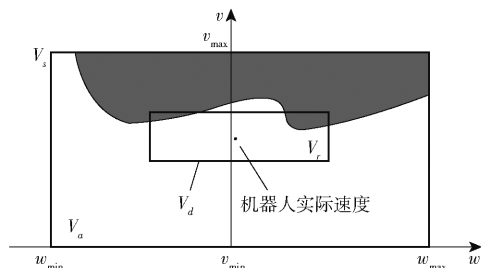


图2 速度矢量空间示意图

Fig. 2 Speed vector space schematic

根据机器人的速度限制,定义 V_s 为机器人线速度与角速度的集合,即动态窗口算法搜索求解的最大范围,满足

$$V_s = \{ (v, w) \mid v_{\min} \leq v \leq v_{\max}, w_{\min} \leq w \leq w_{\max} \} \quad (3)$$

整个机器人的运动轨迹,可以细分为若干个直线或圆弧运动,为保证机器人安全区域,在最大减速度条件下,当前速度应能在撞击障碍物之前减速为0,则定义机器人碰撞可行区域的线速度与角速度集合 V_a 满足

$$V_a = \{ (v, w) \mid v \leq \sqrt{2 \text{dist}(v, w) \dot{v}_b}, w \leq \sqrt{2 \text{dist}(v, w) \dot{w}_b} \} \quad (4)$$

式中 $\dot{v}_b$ ——机器人最大线减速度

$\dot{w}_b$ ——机器人最大角减速度

$\text{dist}(v, w)$ ——轨迹上与障碍物最近的距离

由于机器人电机扭矩的限制,采样周期 Δt 内存

在机器人最大、最小可到达的速度 v 和角速度 w 范围, 需要进一步缩小动态窗口。在给定当前线速度 v_c 和角速度 w_c 条件下, 下一时刻动态窗口 V_d 满足

$$V_d = \{ (v, w) \mid v_c - \dot{v}_b \Delta t \leq v \leq v_c + \dot{v}_b \Delta t, \\ w_c - \dot{w}_b \Delta t \leq w \leq w_c + \dot{w}_b \Delta t \} \quad (5)$$

式中 $\dot{v}_a$ ——机器人最大线加速度

$\dot{w}_a$ ——机器人最大角加速度

最终的速度范围为上述 3 个速度的合集, 定义动态窗口 V_r 满足

$$V_r = V_s \cap V_a \cap V_d \quad (6)$$

在速度矢量空间 V_r 中, 根据线速度、角速度采样点数, 将连续的速度矢量空间 V_r 离散化, 得到离散的采样点 (v, w) 。对于每一个采样点, 根据机器人运动学模型预测下一时刻机器人的多个运动轨迹生成, 如图 3 所示。

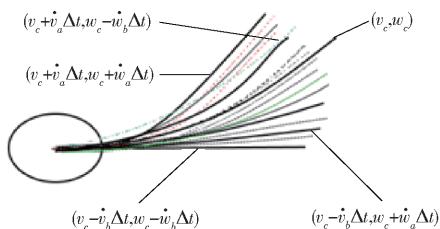


图3 机器人多个轨迹生成图

Fig. 3 Robot multiple trajectory generation map

2.3 评价函数

通过设计评价函数, 在速度空间内有选取最优轨迹。在局部规划中, 保证最优轨迹靠近 A* 算法的全局路径, 完成避障任务, 朝向目标快速运动。定义评价函数为

$$G(v, w) = k(\alpha \text{Heading}(v, w) + \beta \text{Goal}(v, w) + \\ \gamma \text{Path}(v, w) + \sigma \text{Occ}(v, w)) \quad (7)$$

式中 k ——平滑函数

$\alpha, \beta, \gamma, \sigma$ ——各子函数的加权系数

$\text{Path}(v, w)$ ——路径评价子函数, 计算轨迹末端点到全局路径的距离, 距离越短说明轨迹越靠近全局路径

$\text{Heading}(v, w)$ ——方位角不断地朝向终点位置函数

$\text{Goal}(v, w)$ ——评价机器人局部路径末端点到终点的距离函数, 其目的是不断缩短与终点的距离

$\text{Occ}(v, w)$ ——评价机器人轨迹到障碍物距离函数

当 $G(v, w)$ 值最小时, 获得最佳路径。本文中通过改进 A* 算法路径规划获得全局路径, 如图 4 中黑色实线所示, 虚线为局部规划路径, 之间的距离计

算公式为

$$\text{Path}(v, w) = \min \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (8)$$

式中 (x_1, y_1) ——机器人根据运动学模型估计的局部路径末端点坐标

(x_2, y_2) ——全局路径规划的节点坐标

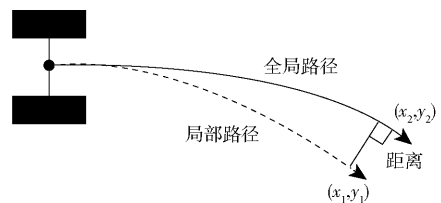


图4 全局路径评价函数示意图

Fig. 4 Schematic of global path evaluation function

在移动过程中, $\text{Head}(v, w)$ 函数用于使机器人的朝向不断趋向终点方向, θ 越小, 说明与终点的方位角越小, 其示意图如图 5 所示, 计算公式为

$$\text{Head}(v, w) = 180^\circ - \theta \quad (9)$$

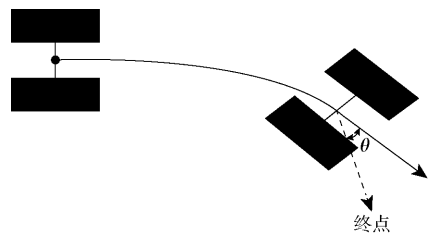


图5 方位角评价函数示意图

Fig. 5 Schematic of azimuth evaluation function

$\text{Goal}(v, w)$ 函数用于评价机器人局部路径末端与终点的距离, 通过函数不断缩短与终点的距离, 其示意图如图 6 所示。

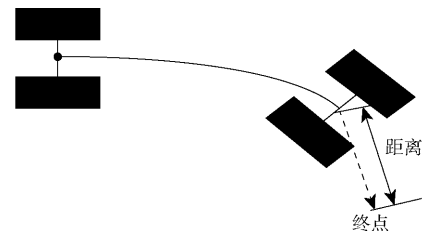


图6 局部路径评价函数示意图

Fig. 6 Schematic of local path evaluation function

$\text{Occ}(v, w)$ 函数用于评价机器人轨迹到障碍物的距离, 体现了机器人的避障能力, 如果机器人的轨迹到障碍物的距离大于机器人半径, 则没有发生碰撞的危险; 反之, 就说明碰撞风险大, 舍弃这条轨迹。其示意图如图 7 所示, R 为机器人底盘半径, H 为机器人轨迹到障碍物的距离。

3 融合算法

移动机器人超声波检测温室局部环境信息, 结合动态窗口法进行在线实时局部路径规划, 检测窗口滚动前进, 具有避障能力, 获取局部的最优路径。

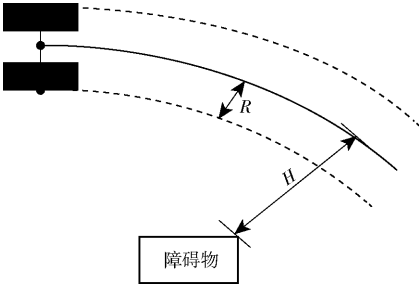


图 7 障碍物评价函数示意图

Fig. 7 Schematic of obstacle evaluation function

通过全局路径规划与局部路径规划结合,将改进A*算法进行全局路径规划与动态窗口算法融合,以保证动态规划路径的全局最优性。

在已知地图上进行全局路径规划,根据当前速度、角速度、线速度计算并发布轨迹,通过评价标准下发各个轨迹,抛弃不合理的轨迹和遇到障碍的轨迹,运用4个评价函数对某一路径进行评价,将各个评价函数累加,将成本最低的轨迹设定为最佳轨迹。如果当前最佳轨迹是可通过的,依照速度模式根据最佳轨迹的速度移动;如果遇到障碍,根据相关规则进行避障,融合算法流程如图8所示。

4 仿真实验与分析

4.1 改进A*算法仿真

Moro 机器人是中国一维弦公司研发的移动机器人,其配套 EwayOS 机器人操作系统包含了 SDK 接口、运动算法等模块,能够对 Moro 机器人路径规划算法进行离线仿真,验证其算法的有效性。本文

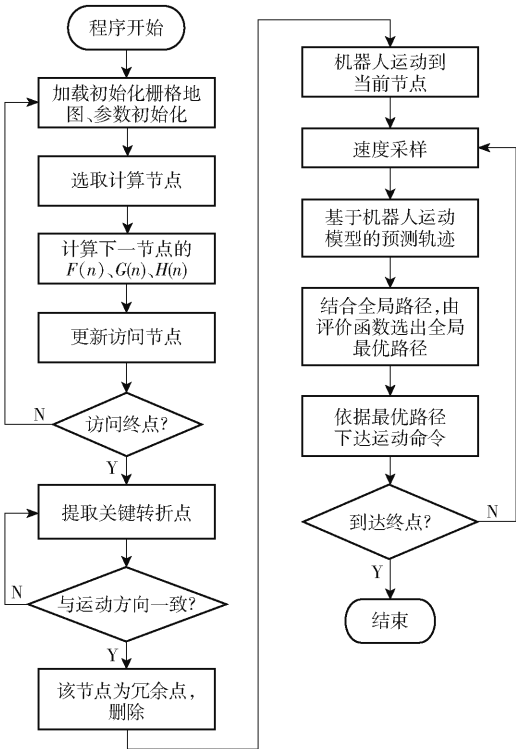


图 8 融合算法流程图

Fig. 8 Flow chart of fusion algorithm

构建5个尺寸一致的栅格地图场景1~5,其水平方向上32个方格,竖直方向上29个方格,分辨率为10 cm × 10 cm,其中黑色方块代表环境中的障碍物信息,白色区域为可行区域。本文将改进A*算法与传统A*、Dijkstra、RRT算法进行对比仿真实验,设定相同的起点坐标为(0,0),终点坐标为(31,28),其仿真实验结果如图9~13所示。

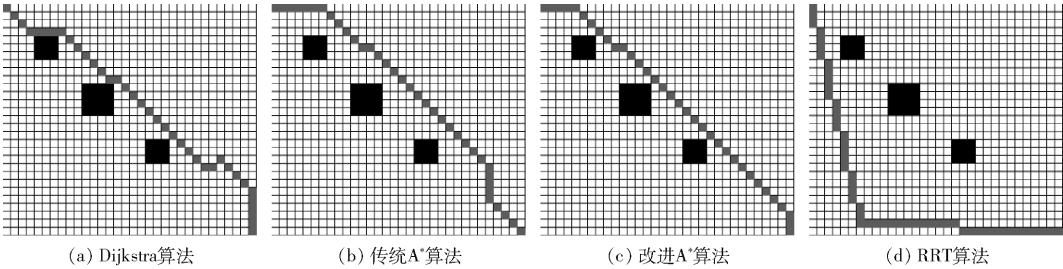


图 9 场景1算法仿真结果

Fig. 9 Algorithm simulation results of scenario 1

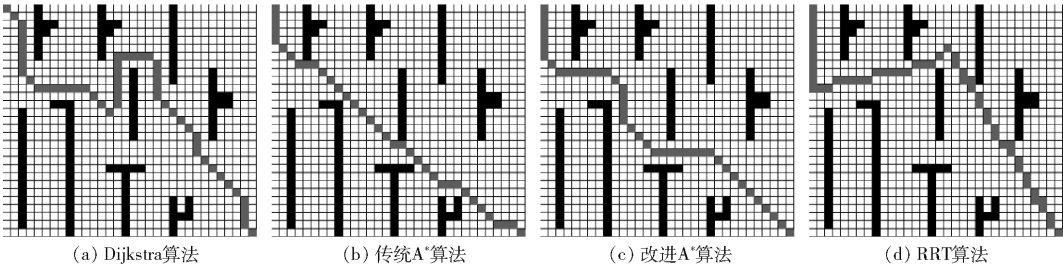


图 10 场景2算法仿真结果

Fig. 10 Algorithm simulation results of scenario 2

5个栅格地图场景障碍物的覆盖率逐渐增大,复杂程度也逐渐增大,其中包含了规则和不规则

障碍物,具有一定代表性。针对每个栅格地图场景,本文使用4种路径规划算法运行10次,初始设

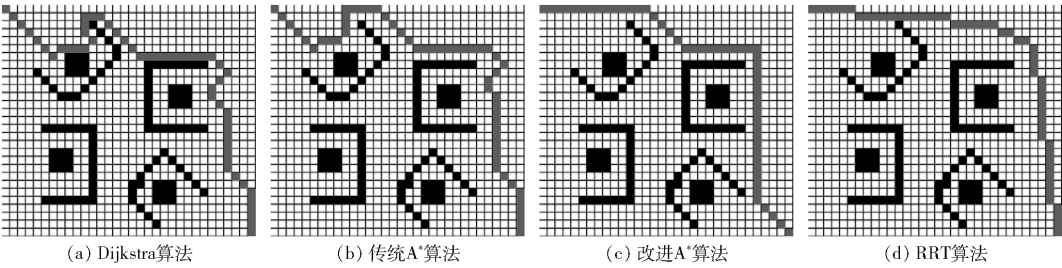


图 11 场景 3 算法仿真结果
Fig. 11 Algorithm simulation results of scenario 3

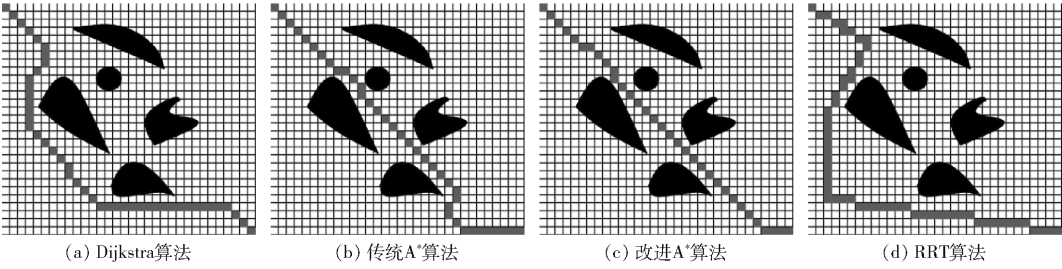


图 12 场景 4 算法仿真结果
Fig. 12 Algorithm simulation results of scenario 4

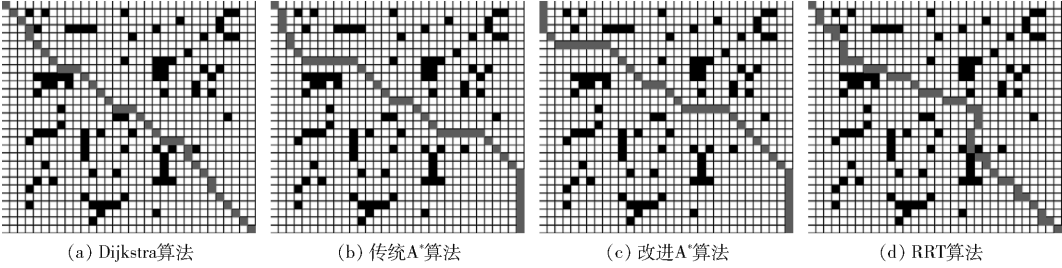


图 13 场景 5 算法仿真结果
Fig. 13 Algorithm simulation results of scenario 5

置参数一致,得到 10 组路径规划算法的路径长度、计算时间以及拐点数,取其平均值,结果如表 1 所示。

从表 1 来看,在 5 个不同栅格地图场景中,4 种路径规划算法都能最终规划出路径,但其路径长度、计算时间、拐点数存在一定差异。RRT 算法的计算时间最短,但 RRT 算法是由随机树节点组成,导致生成的路线转折点过多,并非理想的平滑曲线,这对于机器人运动来说不利。当机器人运动是从一个节点走到另一个节点,首先需要在原地完成转向后,转向下一个节点再继续行走,拐点过多或路径过长,会造成机器人时间和能量的消耗。而改进 A* 算法、传统 A* 算法、Dijkstra 算法获得的全局路径较优。Dijkstra 算法是一种遍历算法,相对于改进 A* 算法更加耗时,尤其面对不规则障碍时,改进 A* 算法规划的路径是最优的。改进 A* 算法相对于传统 A* 算法在计算时间、路径长度和拐点数都有较大的改善。从实验数据可以看出,改进 A* 算法有较好的实时性,路径更短,路径平滑性更好,便于机器人执行路径轨迹达到目标点。

表 1 不同路径规划算法的路径长度、计算时间与拐点数对比
Tab.1 Comparison of length, calculation time and number of inflection points of different path planning algorithms

场景	路径规划算法	路径长度/	计算时间/	拐点数/
		cm	s	个
场景 1	Dijstrak 算法	464	0.053 672	8
	传统 A* 算法	464	0.050 349	5
	改进 A* 算法	440	0.045 953	4
	RRT 算法	578	0.005 681	15
场景 2	Dijstrak 算法	542	0.051 606	14
	传统 A* 算法	576	0.049 326	9
	改进 A* 算法	482	0.040 191	7
	RRT 算法	546	0.006 156	30
场景 3	Dijstrak 算法	560	0.047 315	16
	传统 A* 算法	514	0.045 126	14
	改进 A* 算法	506	0.033 735	4
	RRT 算法	518	0.004 236	16
场景 4	Dijstrak 算法	582	0.045 475	8
	传统 A* 算法	460	0.040 817	9
	改进 A* 算法	444	0.038 253	5
	RRT 算法	616	0.005 965	28
场景 5	Dijstrak 算法	482	0.035 263	12
	传统 A* 算法	480	0.034 425	11
	改进 A* 算法	450	0.025 261	8
	RRT 算法	504	0.008 086	30

4.2 环境模型处理

在算法仿真实验中机器人理想化成一点,忽略了机器人本体尺寸,在真实环境中还要考虑机器人尺寸,避免机器人边缘和障碍物发生碰撞,执行路径失败。用栅格图构建机器人的工作环境,移动机器人在栅格间运动时,将固定障碍和机器人运动过程中超声波检测到的障碍设置为不可行区域。在此基础上对地图的障碍物进行膨胀化处理,保证机器人与障碍物间的安全区域,提高路径的可行性。如图 14 所示,膨胀后的路径,设有安全区域,更利于移动机器人安全执行规划后的路径。

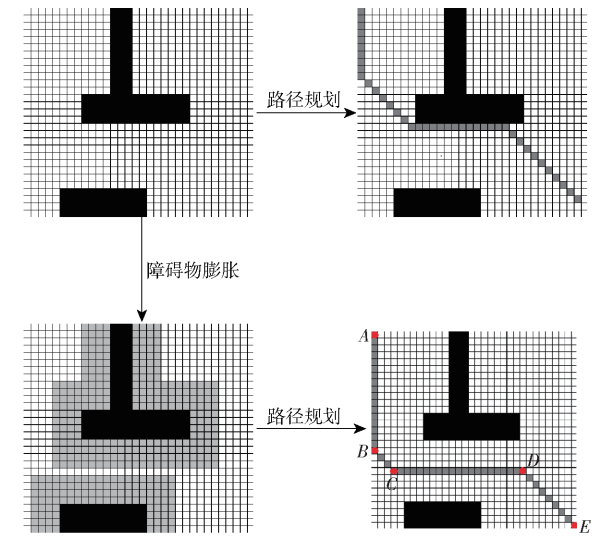


图 14 障碍物膨胀后的路径规划
Fig. 14 Path planning after obstacles swell

在已知温室环境地图基础上,移动机器人通过超声波获取未知的动态障碍信息。本文使用的 Moro 机器人底盘前方的 4 个超声波雷达,检测范围为 0 ~ 250 cm。移动机器人在地图中的位置已知,超声波雷达将检测到与动态障碍物的距离转换到世界坐标系下,增加局部障碍物,将其加入全局地图,机器人再根据融合算法进行局部路径规划。在实际环境中,如图 15a 所示,Moro 机器人遇到动态障碍物,超声检测到障碍物如图 15b 所示,白色表示障碍,检测到障碍物与机器人坐标原点的距离为 80 cm,实际测量值为 82 cm,测量结果误差较小,可以满足实际的精度要求。如图 16a 所示,机器人对局部障碍进行膨胀处理,生成局部地图,重新规划路径,避开障碍物;图 16b 中障碍物厚度增加,绿色的线即为重新规划出的路径。

通过构建如图 17a 所示环境,验证机器人运动的定位精度,起点为绿色,终点为红色。EwayOS 机器人操作系统实时记录融合算法的路径,即图 17a 中的灰色曲线与实际运动轨迹和图 17a 中的黄色曲线。Moro 机器人按照规划路径成功抵达终点,路线

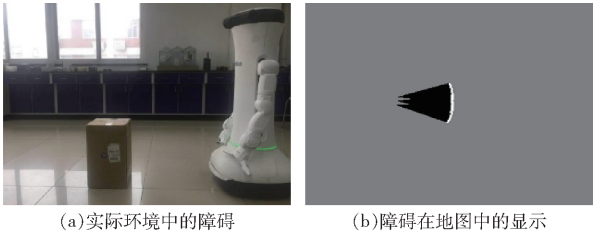


图 15 超声检测障碍物
Fig. 15 Ultrasonic detection of obstacles

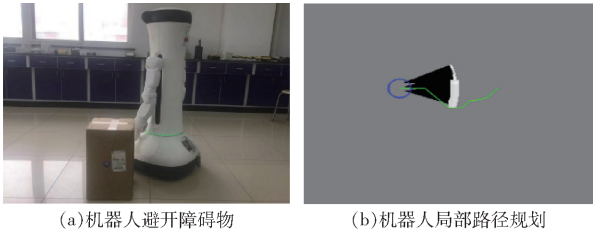


图 16 局部超声避障
Fig. 16 Local ultrasound obstacle avoidance

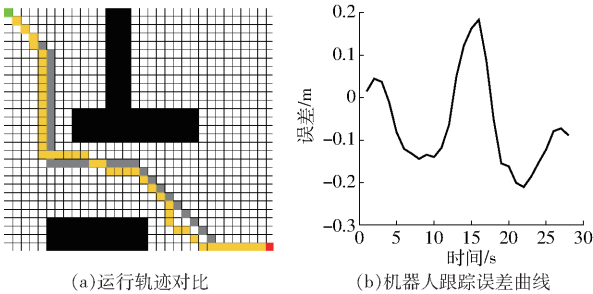


图 17 实际运行轨迹对比及机器人跟踪误差曲线
Fig. 17 Comparison of actual running trajectory and robot tracking error curve

中存在不重合的地方,分析实时的跟踪数据,如图 17b 所示,机器人实际跟踪误差始终小于 0.22 m,基本满足实际需求。

5 真实环境下的实时路径规划

本文的融合算法适用于小范围结构化环境,其环境具有一定的特殊性,选择的 3 个与实验相似的结构化环境 1 ~ 3,对算法可行性进行验证。利用 Moro 机器人分别在楼层走廊、模拟温室环境和真实的温室环境进行实验。3 个实验环境地面光滑程度相似,其栅格地图中黑色为机器人可以到达的无障碍空间,白色为障碍物信息,融合算法的实验步骤描述为:①加载环境栅格地图,设定机器人作业起点与目标点。②改进 A* 算法全局路径规划。通过机器人定位方法,获取机器人实时位置信息,然后通过改进 A* 算法规划机器人当前位置至目标点的全局路径。③障碍物信息处理。机器人运动过程中通过超声波雷达实时检测环境障碍信息,并将传感器信息转换为机器人地图上的障碍信息,得到膨胀后的地图。④动态窗口法局部路径规划。根据机器人的当

前位置信息,结合实时超声波信息,在全局路径的基础上进行动态窗口法局部路径规划,获得机器人最优运动轨迹,机器人按照最优轨迹对应的线速度与角速度执行运动。⑤根据机器人当前位置到目标点的距离判断是否到达。如果未到达终点,则返回步骤③继续运动;如果到达了目标点,机器人导航任务结束。

机器人的实验参数设置如下:最大线速度为 0.6 m/s,最大角速度为 0.3 rad/s,线速度采样个数为 10,角速度采样个数为 20,采样周期为 0.2 s,子函数的加权系数 $\alpha=0.1$ 、 $\beta=10.0$ 、 $\gamma=0.1$ 、 $\sigma=0.2$ 。

环境 1 为中国农业大学信息与电气工程学院楼层走廊,地面为光滑地板。图 18a 所示走廊实际长为 10.5 m,宽为 6.5 m,图 18b 所示栅格地图分辨率为 10 cm × 10 cm,起点为 (1.0 m, 1.0 m),终点设为 (9.1 m, 4.2 m),实际机器人路线如图 18b 中红线所示,终点为 (9.3 m, 4.0 m),误差为 0.22 m。

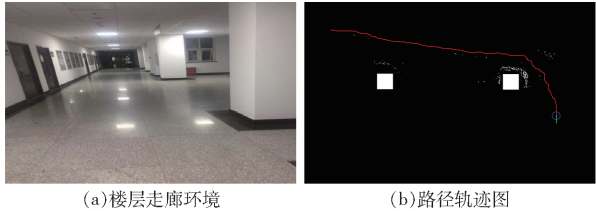


图 18 大楼走廊环境及路径轨迹图
Fig. 18 Building corridor environment and path trajectory map

环境 2 为模拟温室盆栽环境,如图 19a 所示,设置 2 排 10 个盆栽花盆,地面为光滑地板。场景实际长为 6 m,宽为 4 m,图 19b 所示栅格地图分辨率为 10 cm × 10 cm,起点为 (0.5 m, 0.5 m),终点设为 (5.5 m, 3.5 m),实际机器人路线如图 19b 中红线所示,终点为 (5.5 m, 3.4 m),误差为 0.1 m。

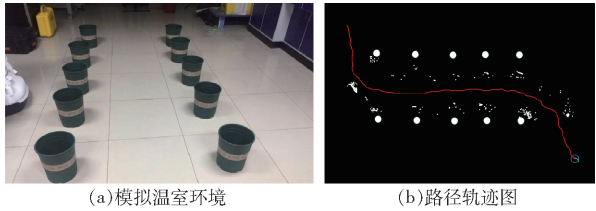


图 19 模拟温室环境及路径轨迹图
Fig. 19 Simulation of greenhouse environment and path trajectory map

环境 3 为实际温室环境,如图 20a 所示,为复合薄膜型结构化温室,地面为光滑的地板,有 4 排植物

生长金属框架。场景的实际长为 8 m,宽为 6 m,图 20b 所示栅格地图分辨率为 10 cm × 10 cm,起点为 (0.5 m, 0.5 m),终点设为 (7.0 m, 2.5 m),实际机器人路线如图 20b 中红线所示,终点为 (7.1 m, 2.3 m),误差为 0.28 m。

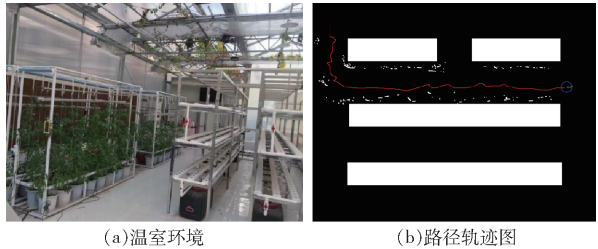


图 20 温室环境及路径轨迹图
Fig. 20 Greenhouse environment and path trajectory map

从实验结果来看,本文的融合算法能够实现避障,绕过障碍物前进到目标点,且规划的路径能够保持全局最优,机器人基本按照规划路径抵达目标。随着场景面积变大和复杂程度增高,机器人到达的实际目标点与设定的目标点之间的误差逐渐增大,主要是由于机器人转弯时与地面打滑以及超声波出现噪声点造成的,其误差不大于 0.28 m,满足实际需求。

6 结论

(1)根据环境信息已知的结构化温室环境进行机器人路径规划,从路径的平滑性、实时性和局部避障出发,提出一种基于改进 A* 算法与动态窗口法相结合的融合算法,在全局最优路径的基础上,进行实时避障,保证了路径的平滑性和实时性。

(2)仿真结果表明,相较于传统 A* 算法、Dijkstra 算法和 RRT 算法,改进 A* 算法在兼顾实时性基础上采取的关键点策略,使路径更为平滑,符合实际机器人运动需求。

(3)通过对环境模型处理,保证机器人运动的安全性。首先,对栅格地图的障碍进行膨胀化处理,结合超声波传感器对动态障碍进行检测,并将膨胀化处理添加到全局地图中。在仿真环境中对融合算法的有效性进行了验证,结果表明,其跟踪误差保持在 0.22 m 以内。

(4)在 3 个相似的结构化环境下进行了 Moro 机器人实验,结果表明,测量定位误差不大于 0.28 m,能够满足实际需求。验证了融合算法在温室机器人路径规划中的可行性。

参 考 文 献

[1] MOHD S, MOHAMAD S, ZAHARUDDIN M, et al. Multi-objective path planner for an agricultural mobile robot in a virtual greenhouse environment[J]. Computers and Electronics in Agriculture, 2019, 157: 488 - 499.
[2] 李秀智, 彭小彬, 方会敏, 等. 基于 RANSAC 算法的植保机器人导航路径检测[J/OL]. 农业机械学报, 2020, 51(9): 40 - 46.
LI Xiuzhi, PENG Xiaobin, FANG Huimin, et al. Navigation path detection of plant protection robot based on RANSAC algorithm

- [J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2020, 51(9): 40–46. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx?flag=1&file_no=20200905&journal_id=jcsam. DOI:10.6041/j.issn.1000-1298.2020.09.005. (in Chinese)
- [3] TELEWEEK P, CHANDRASEKARAN B. Path planning algorithms and their use in robotic navigation systems[C]//2019 3rd International Conference on Control Engineering and Artificial Intelligence, 2019.
- [4] 白国星, 刘丽, 孟宇, 等. 基于非线性模型预测控制的移动机器人实时路径跟踪[J/OL]. 农业机械学报, 2020, 51(9): 47–52, 60. BAI Guoxing, LIU Li, MENG Yu, et al. Real-time path tracking of mobile robot based on nonlinear model predictive control[J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2020, 51(9): 47–52, 60. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx?flag=1&file_no=20200906&journal_id=jcsam. DOI:10.6041/j.issn.1000-1298.2020.09.006. (in Chinese)
- [5] 曲道奎, 杜振军, 徐殿国, 等. 移动机器人路径规划方法研究[J]. 机器人, 2008, 30(2): 97–102. QU Daokui, DU Zhenjun, XU Dianguo, et al. Research on path planning for a mobile robot[J]. Robot, 2008, 30(2): 97–102. (in Chinese)
- [6] DIJKSTRA E. A note on two problems in connexion with graphs[J]. Numer. Math., 1959, 1(1): 269–271.
- [7] CARDONA M, MARIA J, OMAR D, et al. Determination of the optimal method of bimanual assembly operations with Dijkstra algorithm (or shortest path)[J]. Informacion Tecnologica, 2017, 28(4): 125–134.
- [8] SOLTANI A, TAWFIK H, GOULERMAS J, et al. Path planning in construction sites: performance evaluation of the Dijkstra, A*, and GA search algorithms[J]. Advanced Engineering Informatics, 2002, 16(4): 291–303.
- [9] 黄辰, 费继友, 刘洋, 等. 基于动态反馈 A* 蚁群算法的平滑路径规划方法[J/OL]. 农业机械学报, 2017, 48(4): 34–40. HUANG Chen, FEI Jiyu, LIU Yang, et al. Smooth path planning method based on dynamic feedback A* ant colony algorithm[J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2017, 48(4): 34–40. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx?flag=1&file_no=20170404&journal_id=jcsam. DOI:10.6041/j.issn.1000-1298.2017.04.004. (in Chinese)
- [10] DONGGYUNG K, YOUNSUNG C, SANGHO K, et al. Adaptive rapidly-exploring random tree for efficient path planning of high-degree-of-freedom articulated robots[J]. Journal of Mechanical Engineering Science, 2015, 229(18): 203–210.
- [11] MANIKAS T, ASHENAYI K, WAINWRIGHT R. Genetic algorithms for autonomous robot navigation[J]. IEEE Instrumentation and Measurement Magazine, 2007, 10(6): 26–31.
- [12] MATOUI F, BOUSSAID B, METOUI B, et al. Path planning of a group of robots with potential field approach: decentralized architecture[J]. IFAC Papers on Line, 2017, 50(1): 11473–11478.
- [13] FOX D, BURGARD W, THRUR S. The dynamic window approach to collision avoidance[J]. IEEE Robotics & Automation Magazine, 1997, 4(1): 23–33.
- [14] CHOUBEY N, GUPTA M. Analysis of working of Dijkstra and A* to obtain optimal path[J]. International Journal Computer Sciencena Management Research, 2013, 2: 1898–1904.
- [15] ADEL A, HACHEMI B, IMEN C, et al. Relaxed Dijkstra and A* with linear complexity for robot path planning problems in large-scale grid environments[J]. Methodologies and Application, 2016, 20: 4149–4171.
- [16] 王殿君. 基于改进 A* 算法的室内移动机器人路径规划[J]. 清华大学学报(自然科学版), 2012, 52(8): 1085–1089. WANG Dianjun. Indoor mobile-robot path planning based on an improved A* algorithm[J]. Journal of Tsinghua University (Science and Technology), 2012, 52(8): 1085–1089. (in Chinese)
- [17] 赵晓, 王铮, 黄程侃, 等. 基于改进 A* 算法的移动机器人路径规划[J]. 机器人, 2018, 40(6): 903–910. ZHAO Xiao, WANG Zheng, HUANG Chengkan, et al. Mobile robot path planning based on an improved A* algorithm[J]. Robot, 2018, 40(6): 903–910. (in Chinese)
- [18] 张文, 刘勇, 张超凡, 等. 基于方向 A* 算法的温室机器人实时路径规划[J/OL]. 农业机械学报, 2017, 48(7): 22–28. ZHANG Wen, LIU Yong, ZHANG Chaofan, et al. Realtime path planning of greenhouse robot based on directional A* algorithm[J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2017, 48(7): 22–28. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx?flag=1&file_no=20170703&journal_id=jcsam. DOI:10.6041/j.issn.1000-1298.2017.07.003. (in Chinese)
- [19] TROCATO K, DORST L. Differential A*[J]. IEEE Transactions on Knowledge and Data Engineering, 2002, 14(6): 1218–1229.
- [20] 程传奇, 郝向阳, 李建胜, 等. 融合改进 A* 算法和动态窗口法的全局动态路径规划[J]. 西安交通大学学报, 2017, 51(11): 137–143. CHENG Chuanqi, HAO Xiangyang, LI Jiansheng, et al. Global dynamic path planning based on fusion of improved A* algorithm and dynamic window approach[J]. Journal of Xi'an Jiaotong University, 2017, 51(11): 137–143. (in Chinese)
- [21] 王小红, 叶涛. 基于改进 A* 算法机器人路径规划研究[J]. 计算机测量与控制, 2018, 26(7): 282–286. WANG Xiaohong, YE Tao. Research on robot path planning based on improved A* algorithm[J]. Computer Measurement & Control, 2018, 26(7): 282–286. (in Chinese)
- [22] SEDER M, PTROVIC I. Dynamic window based approach to mobile robot motion control in the presence of moving obstacles[C]//IEEE International Conference on Robotics and Automation, 2007.
- [23] DOOPALAM T. Hybrid motion planning method for autonomous robots using Kinect based sensor fusion and virtual plane approach in dynamic environments[J]. Journal of Sensors, 2015(5): 1–13.
- [24] 王永雄, 田永永, 李璇, 等. 穿越稠密障碍物的自适应动态窗口法[J]. 控制与决策, 2019, 34(5): 927–936. WANG Yongxiong, TIAN Yongyong, LI Xuan, et al. Self-adaptive dynamic window approach in dense obstacles[J]. Control and Decision, 2019, 34(5): 927–936. (in Chinese)