

基于 BIM 的地图构建与导航研究

陈宇帆¹, 段中兴^{1,2}

(1. 西安建筑科技大学 信息与控制工程学院, 西安 710055;

2. 西部绿色建筑国家重点实验室, 西安 710055)

摘要: 为了提高移动机器人在作业过程中获得现场环境地图的效率, 提出了利用 BIM 技术建立导航地图的方式, 获取 IFC 信息映射到二维栅格, 从而快速构建地图; 对于室内移动机器人在移动过程中能更快更好的到达目标点的问题, 首先对传统 A* 算法做改进, 将原有的 8 邻域搜索扩展为 48 邻域搜索, 增加了搜索方向, 优化了搜索角度; 同时考虑了机器人的安全性, 对规划路径进行了改进, 使得规划的路径与障碍物保持了一定距离; 其次, 为了避开场地出现的动态障碍物, 采用将改进的 A* 算法与动态窗口法融合, 在保证全局路径最优的基础上, 实现避障效果; 通过实验仿真, 表明了改进的 A* 算法比传统 A* 的算法在运行时间上快了 2 倍以上, 路径转折点的角度差比原来减少了 28% 以上, 路径长度上更短且不再紧贴障碍物; 而融合算法比改进的 A* 算法在路径平滑性上有所提高, 能及时避开随机障碍物, 更加适用于环境变化的室内场景。

关键词: BIM 技术; 地图构建; 改进 A* 算法; 动态窗口法; 融合算法

Research on Map Construction and Navigation Based on BIM

CHEN Yufan¹, DUAN Zhongxing^{1,2}

(1. School of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China)

2. State Key Laboratory of Green Building in West China, Xi'an 710055, China)

Abstract: In order to improve the efficiency of the mobile robot in obtaining the scene environment map during the operation, a navigation map is established by using building information modeling (BIM) technology, and the industry foundation classes (IFC) information is obtained and mapped to a two-dimensional grid, so that the map can be quickly constructed. In order to make the indoor mobile robot reach the target point faster and better, Firstly, the algorithm of traditional A* is improved, and the original 8-neighborhood search is expanded to the 48-neighborhood search, which increases the search direction and optimizes the search angle. At the same time, considering the safety of robot, the planned path is improved, so that the planned path keeps a certain distance from the obstacles. Secondly, due to the emergence of dynamic obstacles in the site, the algorithm of improved A* is combined with the Dynamic Window Approach, which can not only ensure a better global path, but also achieve obstacle avoidance effect. Through the experimental simulation, it can be seen that the algorithm of improved A* improves the running speed by more than 2 times, reduces the turning angle difference by more than 28%, and reduces the path distance, and no longer clings to obstacles. Compared with the algorithm of improved A*, the fusion algorithm has better path smoothness, can avoid random obstacles in time, and is more suitable for the indoor environment of changing site.

Keywords: BIM technology; map construction; improve A* algorithm; dynamic window approach; fusion algorithm

0 引言

随着城市化发展的进程, 出现了越来越多的高楼大厦与大型公共建筑, 实现智能与自动化成为当今社会进一步发展的新契机。随着各类机器人的出现, 使得人们的生产生活方式发生了变革, 解放了很多人力劳动, 同时也存在着一些棘手的问题。

由于建筑室内环境的复杂与多变, 移动机器人需要在技术性能上要求更高。当前地图构建方式大多采用 SLAM (simultaneous localization and mapping) 技术^[1], 让机器人

通过传感器在未知的环境中得到识别信息, 并逐步获取相关地图。当环境发生较大变化时, SLAM 构建的地图将无法使用。并且面对面积较大的场景时, 使用 SLAM 技术也会使得构建地图的过程变得更加复杂和繁琐。由于建筑信息模型 BIM (building information modeling) 技术的快速推进, 因此在工程建设中越来越广泛使用 BIM 模型, 获取建筑的信息也更加的准确及时^[2]。使用 BIM 模型能够快速创建室内地图, 因此将 BIM 模型与移动机器人室内导航相结合, 能够提高机器人工作的效率。对于路径规划, 目前比

收稿日期: 2021-12-08; 修回日期: 2022-01-11。

基金项目: 国家自然科学基金(51678470)。

作者简介: 陈宇帆(1997-), 女, 陕西韩城人, 硕士研究生, 主要从事机器人的路径规划及其智能算法方向的研究。

段中兴(1969-), 男, 陕西西安人, 博士, 教授, 主要从事智能系统与智能信息处理、智能检测与机器视觉、建筑环境控制与节能优化、嵌入式技术与智能系统等方向的研究。

引用格式: 陈宇帆, 段中兴. 基于 BIM 的地图构建与导航研究[J]. 计算机测量与控制, 2022, 30(6): 144-149.

较常用的有人工势场法^[3]、A*算法^[4]、迪杰斯特拉算法(Dijkstra)^[5]、遗传算法^[6]、蚁群算法^[7]等。动态窗口法(DWA)是目前局部路径避障的主要算法,应用于实时碰撞避免策略^[8]。文献[9]对A*算法进行了改进,使得规划的路径更加平滑,但是对于有随机障碍物出现的情况不适用;文献[10]将Dijkstra算法与DWA算法相融合,在全局路径中规划出最短路径,并在局部路径规划中实现避障效果,但是没有考虑动态障碍物与机器人之间的距离,容易使得机器人在狭窄空间内无法通过;文献[11]中将传统A*算法的8邻域搜索改为5邻域搜索,使用DWA算法进行局部避障,使得路径距离变短且平滑度提高,但是没有考虑到其5个邻域方向同时存在障碍的情况,容易使得全局路径无法规划。

针对上述问题,本文在地图建立过程中采用BIM模型,提取建筑物的空间信息,从而使地图构建效率得到提高,让移动机器人能够快速获取和熟知室内环境。在室内路径规划中,采用全局路径与局部路径相结合,对A*算法进行改进,扩大其搜索邻域,考虑机器人通过的安全性,当有动态障碍时,采用DWA算法进行避障,保障运动的连续性。

1 BIM信息提取与地图构建

建筑信息模型BIM(building information modeling)是一个生产建筑数据和使用建筑数据的过程^[12],基于模型的智能化流程,包含与物理特性和功能特性相关的数据,BIM技术能更好的还原从概念到施工的建筑模型,使得建筑建造和运营建筑基础设施变得更加高效、更加精细化和透明化。

使用Autodesk Revit软件进行BIM建模,可以对现实三维事物进行准确的模拟,并且能够呈现出准确的信息。

1.1 IFC简述

IFC(industry foundation class)是针对BIM数据交换的唯一的开放式标准格式。文件IFC是一种交换格式,用于不同软件系统之间交换建筑设计和建造中使用的建筑数据模型,从而达到数据共享。IFC标准分为4个结构层次,分别为资源层、核心层、交互层和领域层^[13],4个层次共同来完成信息资源的稳定。IFC中运用EXPRESS语言描述建筑产品数据^[14]。EXPRESS是一种标准化数据建模语言,其定义在STEP国际标准中^[15]。EXPRESS语言与其他编程语言不同,更看重对计算机大规模数据的可读性,从而实现了对数据信息的描述。

1.2 IFC数据前期处理

通过BIM建模,导出IFC文件,并对IFC文件进行解析,提取出所需要的室内几何信息^[16-17]。IFC文件的基础结构包括项目的基本内容、基本空间结构以及几何表示。项目的基本内容主要由IfcProject提供,项目的基本空间结构主要由IfcBuilding提供,几何表示主要由IfcBuildingElement提供。在IFC文件中,实体的属性具有继承关系,属性分为:直接属性、导出属性和反属性。其中直接属性很直观的表达了构件信息,导出属性是通过不同构件实体的

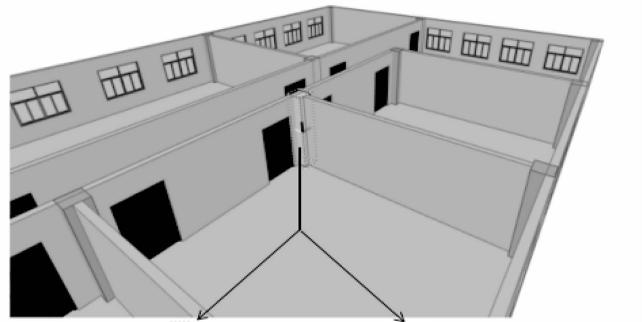
属性来构成表达,而反属性是构件可以通过关联实体从而获得相关信息。IFC文件的数据部分包含了构建模型的坐标位置、属性值、长宽高、空间隶属关系等相关信息。IFC数据段中的部分坐标信息如下:

```
# 138=IFCCARTESIANPOINT((0. ,0. ,10800. ));
# 140=IFCAXIS2PLACEMENT3D(# 138, $, $);
# 141=IFCLOCALPLACEMENT(# 32, # 140)
```

其中# 141是对象位置信息(ObjectPlacement),IFCLOCALPLACEMENT(# 32, # 140)描述了该构件的相对位置几何坐标,这里几何坐标引用了# 32父坐标体系。

140=IFCAXIS2PLACEMENT3D(# 138,,);指定了局部坐标系的原点位置,符号代替语句中次要信息。# 138=IFCCARTESIANPOINT是位置信息,表示(0., 0., 10800.)位置处。

使用IFC解析工具对IFC文件进行解析,对所需的室内模型构件的几何信息进行提取。如图1为其中一个柱IfcColumn的属性和地点信息。



属性	地点	分类	关系	属性	地点	分类	关系
名称			值	名称			值
- Element Specific				- Location			
Guid			OpVjRzNnj8rhm_RYcN\$y_N	Project			项目编号
IfcEntity			IfcColumn	Storey			四层
Name			钢管混凝土柱-矩形: 600*600; 217472	Top Elevation			3 400
ObjectType			钢管混凝土柱-矩形: 600*600	Bottom Elevation			0
Tag			217472	Global Top elevation			14 200
- Pset_ColumnCommon				Global Bottom Elevation			10 800
IsExternal			无	- Geometry			
LoadBearing			正确	Has Om Geometry			正确
Reference			600*600	Children Have Geometry			无
Slope			0	Global X			9 082.337744
- Pset_QuantityTakeOff				Global Y			-1 837.497743
Reference			600*600	Global Z			10 800
- Pset_ReinforcementBarPitchOfColumn				Bounding Box Length			600
Reference			600*600	Bounding Box Widty			600
				Bounding Box Height			3 400
				- Membership			
				Layer			S-COLS-___-0TLN

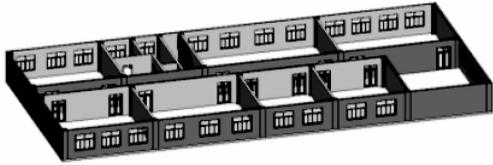
图1 IfcColumn的属性和地点信息

从属性中可以看出构件的实体名、类型、标签以及轮廓等。从地点中可以看到构件的位置坐标、长宽高以及所处的楼层信息等。图中这个柱子长度为600 mm,宽度为600 mm,高度为3 400 mm。位置坐标为(9 082.337 744, -1 837.497 743, 10 800)。

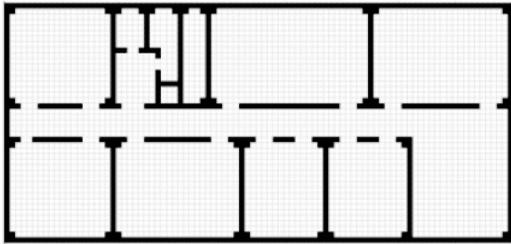
1.3 构建地图

获取建筑环境的边界顶点坐标,栅格平面的长为 $L = X_{\max} - X_{\min}$,宽为 $W = Y_{\max} - Y_{\min}$ 。则栅格平面行数为 $i = \lceil L/S \rceil$,列数为 $j = \lceil W/S \rceil$,S为每个栅格所代表的实际长宽。将几何信息映射到栅格平面时,使用构件在二维平面

图的起终点坐标进行映射，对应网格属性为 1。（属性为 1 代表不可通行，为 0 代表可通行）。当构件映射不满一个栅格时，则补齐这个栅格作为不可行区域。地图构建过程如图 2 所示。



(a) 建筑BIM模型



(b) 映射栅格地图

图 2 地图构建过程

2 A* 算法的改进

2.1 传统 A* 算法

A* (A-Star) 算法是一种启发式搜索，利用启发函数对搜索过程进行指导，在静态环境下的路径规划，A* 算法不但能获得较短路径，并且其搜索效率比较高，广泛应用于室内机器人路径搜索、游戏动画路径搜索。A* 算法的估价函数可以表示为：

$$f(n) = g(n) + h(n) \quad (1)$$

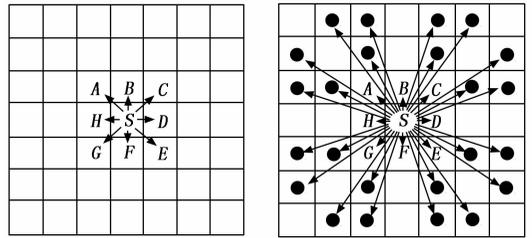
$f(n)$ 是从初始状态经由状态 n 到目标状态的代价估计， $g(n)$ 是状态空间中从初始到状态 n 的实际代价， $h(n)$ 是从状态 n 到目标状态的最佳路径的估计代价。

2.2 48 邻域扩展

传统 A* 算法为 8 邻域搜索，也就是在该点周围的 8 个相邻栅格方向搜索，使得机器人的转向方向也被限制于 $\pi/4$ 的整数倍，这样规划出来的路径转折角度较大。如果转折角度过大，不利于机器人在室内狭窄空间移动，也会使得移动时间变长，降低了工作效率。因此，将原先的搜索范围扩展至 48 邻域搜索如图 3 所示，对 8 个方向上每 2 个方向（如 D、E 之间）之间再增加 3 个方向搜索，由原来的 8 个方向变为 32 个方向，这样会使机器人在方向上有更多选择，减少转向次数，更加流畅的进行作业。图 4 为使用 48 邻域搜索的路径规划结果图。S 为起点位置，D 为终点位置，直线所连接的方块为规划的路径，周围区域为 48 邻域搜索空间。

2.3 搜索路径改进

传统 A* 算法在规划路径时没有考虑到机器人本身的轮廓大小，规划出来的路径容易紧贴墙壁或者障碍物。由于室内环境会有很多门或者相邻柱体，且随机障碍物较多，



(a) 传统的8邻域搜索

(b) 改进后的48邻域搜索

图 3 A* 算法的邻域搜索空间

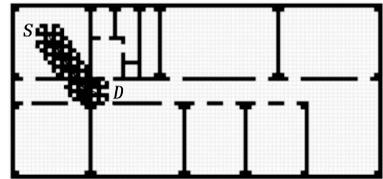


图 4 48 邻域搜索结果图

若按照传统 A* 算法规划出来的路径移动，会使得机器人与障碍物相撞，更有可能损伤机器人机身。因此，除了对传统 A* 算法在搜索方向上进行扩展，还需要考虑机器人安全性，将规划出来的路径进行改进。

机器人在移动过程中，将规划出来的路径与障碍物的距离记为 s (m)，机器人机身半径记 d (m)，规定安全距离为 0.2 m，当 $s \geq d + 0.2$ ，则表示规划的该路径可通过，反之，不可通过。

3 进 A* 与动态窗口法的融合算法

3.1 动态窗口法

动态窗口法 DWA (dynamic window approach) 作为局部路径规划避障的主要算法^[18]，能够接受全局路径规划生成的路径以及里程计的信息、地图的信息，输出为底盘运动的速度信息。算法流程为：根据机器人当前的速度，计算出当前可采样的速度范围，然后对采样的每一组速度都依据运动模型，模拟一段时间内的路径。把所有采样速度生成的路径都模拟出来后，接着根据评价函数对每一条路径都打分，然后选取得分最高的一条路径，那么这条路径就对应了一组采样速度，就把这一组速度作为机器人下一时刻的速度，那机器人就会以新的速度继续移动，这时候算法就会进入新的循环，以新的速度计算新的采样速度范围，然后模拟路径，再次根据得分选取路径和速度，把选取到的速度用于下一个时刻，以这种流程不断循环下去。所以这些模拟路径是不断的在变化，因为在不断的进行采样、模拟、评价再采样，一直刷新模拟的路径，那机器人就会不断的调整自身的速度和方向，往目标点移动，从而实现导航的功能。

3.1.1 机器人运动学模型

设机器人在全向运动过程中，在 X 方向的速度为 V_x ，在 Y 方向的速度为 V_y ，角速度为 ω 。若将机器人在相邻时刻两点之间的轨迹看成直线，那么在 Δt 的时间里，运动模型可表示为：

由 X 方向速度产生的机器人的坐标变化为:

$$\Delta x_x = v_x \Delta t \cos \theta_t \quad (2)$$

$$\Delta y_x = v_x \Delta t \sin \theta_t \quad (3)$$

由 Y 方向速度产生的机器人的坐标变化为:

$$\Delta x_y = -v_y \Delta t \sin \theta_t \quad (4)$$

$$\Delta y_y = v_y \Delta t \cos \theta_t \quad (5)$$

机器人下个状态的位置:

$$x = x + v_x \Delta t \cos \theta_t - v_y \Delta t \sin \theta_t \quad (6)$$

$$y = y + v_x \Delta t \sin \theta_t + v_y \Delta t \cos \theta_t \quad (7)$$

$$\theta_t = \theta_t + \omega \Delta t \quad (8)$$

3.1.2 速度采样

依据机器人的运动模型, 通过采样多组速度, 模拟出一段时间内的路径。

1) 机器人最大速度与最小速度限制。

$$v_m = \{(v, \omega) \mid v \in [v_{\min}, v_{\max}] \wedge \omega \in [\omega_{\min}, \omega_{\max}]\} \quad (9)$$

式中, V_m 为机器人速度; v 为线速度; ω 为角速度。

2) 机器人加速度限制。电机的力矩是有限的, 电机性能也会有所影响, 使得机器人移动时的加速度不能在短时间内无限大。所以加速度是限制了采样速度范围的一个因素。因此, 加速度限制范围为:

$$V_d = \{(v, \omega) \mid v \in [v_c - v_b \Delta t, v_c + v_a \Delta t] \wedge \omega \in [w_c - w_b \Delta t, w_c + w_a \Delta t]\} \quad (10)$$

式中, V_d 为机器人受加速度限制后的速度; V_c 为当前线速度; V_a 为最大线加速度; V_b 为最大线减速度; W_c 为当前角速度; W_a 为最大角加速度; W_b 为最大角减速度。

3) 预留安全距离。当有障碍物时, 机器人不是立马能够停下来, 因此需要预留安全距离范围为:

$$V_a = \{(v, \omega) \mid v \leq (2dist(v, \omega)v_b)^{1/2}, \omega \leq (2dist(v, \omega)w_b)^{1/2}\} \quad (11)$$

式中, $dist(v, \omega)$ 为机器人与障碍物之间的距离。

3.1.3 评价函数

在利用采样速度模拟了很多路径以后, 对速度的选择需要使用评价函数来评估轨迹的好坏^[19-20]。评价函数用以下表示:

$$G(v, \omega) = \sigma(\alpha \cdot heading(v, \omega) + \beta \cdot dist(v, \omega) + \gamma \cdot velocity(v, \omega)) \quad (12)$$

式中, $heading(v, \omega)$ 为方位角评价函数, 指的是机器人与目标之间的角度差, 角度差越小, 得分越高; $dist(v, \omega)$ 为机器人与最近障碍物之间的距离, 距离越远, 得分越高; $velocity(v, \omega)$ 为轨迹对应的速度大小, 速度越大, 得分越高。

为了避免这三项不同类别的得分基数相差太大, 影响结果。比如方位角的角度差得分在 0~180 之间, 而障碍物距离得分可能都在 5 以内, 这样就容易掩盖了障碍物距离的影响。因此, 需要对这 3 个方面的得分进行归一化处理, 把每一项的得分占比算出来再相加。归一化如下:

$$normal_head(i) = \frac{head(i)}{\sum_{i=1}^n head(i)} \quad (13)$$

$$normal_dist(i) = \frac{dist(i)}{\sum_{i=1}^n dist(i)} \quad (14)$$

$$normal_velocity(i) = \frac{velocity(i)}{\sum_{i=1}^n velocity(i)} \quad (15)$$

在归一化处理, 对每一部分赋权重, 不同的权重值会影响机器人选取的路径。将这 3 方面综合起来的物理意义就是使得机器人避开障碍物, 朝着目标点以较快的速度移动。

3.2 融合算法

将全局路径规划与局部路径规划相结合, 先由全局路径规划器规划出一条大致的路径, 接着局部路径规划器又会将它划分为许多小区段, 再完成整个局部路径规划。相当于是把一个总目标分成很多个小目标然后依次达成。这样做的好处是在全局路径规划的时候, 对地图保存过的障碍物进行避障, 而在局部路径规划的时候, 会对新增的障碍物信息也进行避障。并且它支持对动态的障碍物进行避障。本文算法的流程如图 5 所示。

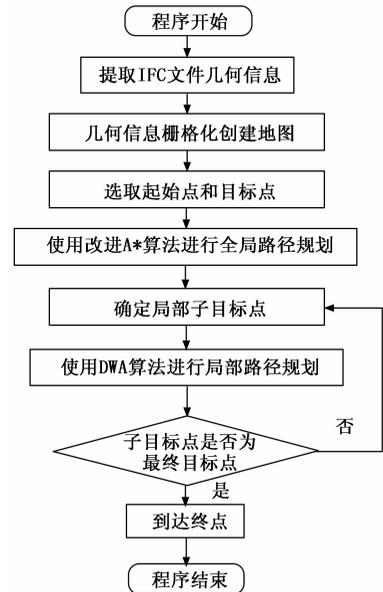


图 5 算法流程图

4 实验与仿真

在获取到的楼层地图上进行短距离和长距离两组路径规划, 每一组实验中都采用传统 A* 算法、改进 A* 算法、改进 A* 与 DWA 的融合算法 (无障碍物)、改进 A* 与 DWA 的融合算法 (有障碍物) 来进行路径规划。第一组实验的起点坐标为 (10, 28), 终点坐标为 (27, 14), 动态障碍物设置为 2 个, 坐标为 (14, 27)、(23, 20), 结果如图 6 和图 7 所示; 第二组实验的起点坐标为 (15, 16), 终点坐标为 (89, 27), 动态障碍物设置为 3 个, 坐标为 (32, 22)、(83, 22)、(84, 22), 结果如图 8 和图 9 所示。同时, 记录了每组实验中各算法路径规划的运行时间和路径长度, 对传统 A* 算法与改进 A* 算法的累计转折角度进行了计算, 结果如表 1~4 所示。

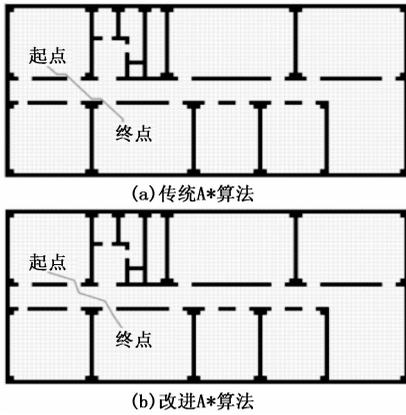


图 6 第一组路径传统与改进算法轨迹图

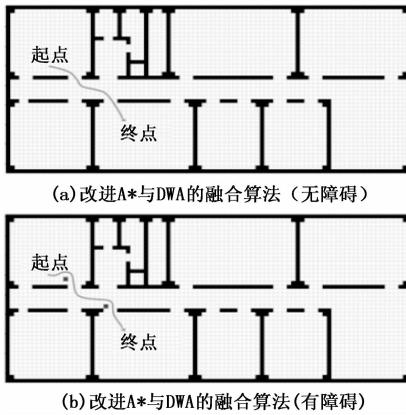


图 7 第一组路径融合算法轨迹图

表 1 第一组路径传统与改进算法的性能比较

算法	运行时间/s	路径长度/m	累计转折角度/(°)
传统 A* 算法	2.36	14.030 5	225
改进 A* 算法	1.14	13.092 5	162

表 2 第二组路径传统与改进算法的性能比较

算法	运行时间/s	路径长度/m	累计转折角度/(°)
传统 A* 算法	3.32	49.491 0	225
改进 A* 算法	1.56	48.941 8	128

表 3 第一组路径融合算法的性能比较

算法	时间/s	路径长度/m
第一组路径融合算法(无障碍)	30.99	13.053 6
第一组路径融合算法(有障碍)	31.89	14.402 6

表 4 第二组路径融合算法的性能比较

算法	时间/s	路径长度/m
第二组路径融合算法(无障碍)	87.35	48.241 8
第二组路径融合算法(有障碍)	89.13	49.872 5

由图 6 和图 8 可以看出：传统的 A* 算法在规划路径时，路径与墙或门的距离较近，安全性比较低。而使用改进的 A* 算法考虑了机器人安全性的问题，增加了路径与

障碍物的安全距离，使得机器人不再靠着墙壁移动，确保了机器人能够顺利通过。由表 1 表 2 可知，改进的 A* 算法比传统 A* 算法在运行时间上快了 2 倍以上，由于节点转折角度差的减小，改进后的两组路径长度也都相对的优化。其次，第一组仿真实验传统算法累计转折角度为 225°，改进算法为 162°，第二组仿真实验原始算法累计转折角度为 225°，改进算法为 128°，改进的 A* 算法优化了搜索角度，比传统 A* 算法的累计转折角度减少了 28% 以上。综合来看，改进后的算法在路径规划在性能上都有进一步的改善，且搜索效率有所提升。

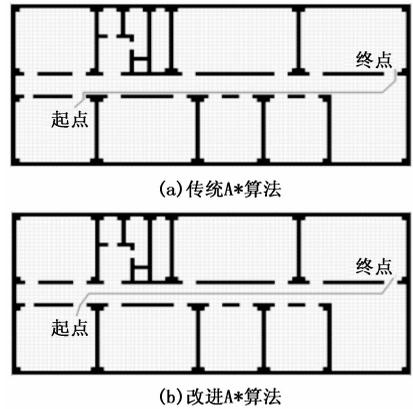


图 8 第二组路径传统与改进算法轨迹图

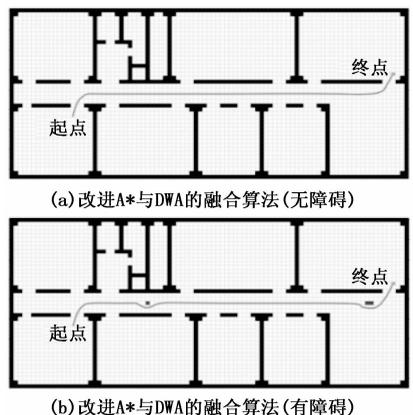


图 9 第二组路径融合算法轨迹图

采用改进 A* 与 DWA 的融合算法规划的路径如图 7、9 所示，分别为无障碍物和有障碍物两种情况。动态窗口法的参数设置如下：机器人运动学模型间隔时间为 0.1 s，最大速度为 1 m/s，最大角加速度为 20°/s，速度分辨率为 0.01 m/s，角加速度分辨率为 1°/s，加速度为 0.2 m/s²，角加速度为 50°/s²。评价函数参数为： $\alpha=0.2$ ， $\beta=0.2$ ， $\gamma=0.1$ ，预测时间周期为 3.0 s。改进 A* 与 DWA 的融合算法的配合过程就是改进的 A* 算法出一条大致的路径，然后 DWA 算法会把改进的 A* 算法规划的路径分割成很多段，大目标分割成很多小目标，然后用 DWA 算法依次导航到每一个小目标点。当没有障碍物时，融合算法比改进的 A* 算法的路径更加连续，平滑度更高；当有动态障碍物出现后，融合算法能够及时实现避障，由表 3 所知，无障碍

物时, 使用融合算法第一组路径规划整个算法用时 30.99 s, 路径长度为 13.053 6 m; 当出现两个障碍物时, 整个算法用时 31.89 s, 路径长度变为 14.402 6 m。由表 4 所知, 无障碍物时, 使用融合算法第二组路径规划整个算法用时 87.35 s, 路径长度为 48.241 8 m; 当出现 3 个障碍物时, 整个算法用时 89.13 s, 路径长度变为 49.872 5 m。因此随着障碍物的出现, 运行时间和路径长度也理应有所增加。综合来说, 融合算法规划的路径平滑度更好, 并且能及时避开出现的随机障碍物, 使得机器人在室内场所的移动更加安全。

图 10 为选取了第一组短路径下传统 A* 算法和融合算法的控制参数对比, 机器人在传统 A* 算法下的线速度和角速度变化范围较大, 变化不连续, 这样会不利于机器人的移动, 长时间会损伤机器人机身。而在融合算法下的线速度变化波动较小, 比较平稳, 并且角速度的曲率变化比较平滑连续, 更符合机器人的动力学控制, 也会使得机器人整个行驶过程变得流畅。

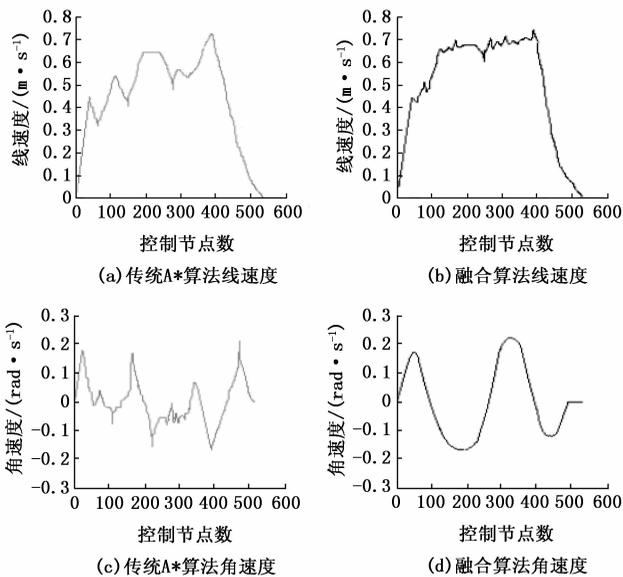


图 10 第一组路径控制参数对比

5 结束语

通过 BIM 技术提取建筑空间信息, 节省了机器人获取室内环境的时间, 同时也提高了地图构建的效率。针对移动机器人路径规划问题, 对传统 A* 算法进行了改进, 通过仿真结果得知累计转折角度减少了 28% 以上, 运行时间也快了 2 倍。同时, 设置了路径与障碍物的安全距离, 保证了机器人的安全性。此外, 为了解决移动机器人行驶过程中遇到动态障碍物的问题, 将改进 A* 的算法和动态窗口法相结合, 并对融合算法也进行了实验仿真和性能对比, 验证了本文的融合算法路径连续性更高, 达到了避障能力, 更具有环境适用性。

参考文献:

[1] 赵洋, 刘国良, 田国会, 等. 基于深度学习的视觉 SLAM 综述 [J]. 机器人, 2017, 39 (6): 1-9.

- [2] 王婷, 池文婷. BIM 技术在 4D 施工进度模拟的应用探讨 [J]. 图学学报, 2015, 36 (2): 306-311.
- [3] 祝敬, 杨马英. 基于改进人工势场法的机械臂避障路径规划 [J]. 计算机测量与控制, 2018 (10): 205-210.
- [4] FU B, CHEN L, ZHOU Y T, et al. An improved A* algorithm for the industrial robot path planning with high success rate and short length [J]. Robotics & Autonomous Systems, 2018, 106: 26-37.
- [5] LIM J, LEE S J, TEWOLDE G, et al. Indoor localization and navigation for a mobile robot equipped with rotating ultrasonic sensors using a smartphone as the robot's brain [J]. International Journal of Handheld Computing Research, 2016, 7 (1): 1-11.
- [6] PATLE B K, PARHI D R K, JAGADEESH A, et al. Matrix-binary codes based genetic algorithm for path planning of mobile robot [J]. Computers and Electrical Engineering, 2018, 67: 708-728.
- [7] LUO Q, WANG H B, ZHENG Y, HE J C. Research on path planning of mobile robot based on improved ant colony algorithm [J]. Neural Computing and Applications, 2020, 32 (6): 1555-1566.
- [8] 何壮壮, 丁德锐. 基于 D-star 和 DWA 的改进机器人导航方法 [J]. 电子测量技术, 2019, 42 (12): 122-128.
- [9] 王红卫, 马勇, 等. 基于平滑 A* 算法的移动机器人路径规划 [J]. 同济大学学报 (自然科学版), 2010, 38 (11): 1647-1650.
- [10] 徐波, 陈欢, 田定胜, 等. 基于 Dijkstra 的智能巡检机器人避障算法 [J]. 供用电, 2020, 37 (12): 74-80.
- [11] 曹毅, 周轶, 张亚宾. 基于优化 A* 和 DWA 算法的移动机器人避障路径规划 [J]. 机床与液压, 2020, 4 (24): 246-252.
- [12] 郑华海, 刘匀, 李元齐. BIM 技术研究与应用现状 [J]. 结构工程师, 2015, 31 (4): 233-241.
- [13] 李春霞. 基于 BIM 与 IFC 的 N 维模型研究 [D]. 武汉: 华中科技大学, 2009.
- [14] 刘强, 张建平, 胡振中. 基于键值缓存的 IFC 模型 Web 应用技术 [J]. 清华大学学报 (自然科学版), 2016 (4): 348-353, 359.
- [15] 周忠, 许速. 基于 IFC 和 City GML 映射转换的 BIM 与 3DGIS 集成研究 [J]. 河南科技, 2016 (21): 18-21.
- [16] 李佩瑶, 汤圣君, 刘铭崑, 等. 面向导航的 IFC 建筑模型室内空间信息提取方法 [J]. 地理信息世界, 2015, 22 (6): 78-84.
- [17] 陈远, 逯瑶. 基于 IFC 标准的 BIM 模型空间结构组成与程序解析 [J]. 计算机应用与软件, 2018 (4): 162-167, 194.
- [18] 王洪斌, 尹鹏衡, 郑维, 等. 基于改进的 A* 算法与动态窗口法的移动机器人路径规划 [J]. 机器人, 2020 (3): 346-353.
- [19] 劳彩莲, 李鹏, 冯宇. 基于 A* 与 DWA 算法融合的温室机器人路径规划 [J]. 农业机械学报, 2021, 52 (1): 14-22.
- [20] 刘末, 朱宏辉. 隧道安全预警机器人自主导航方法研究 [J]. 计算机测量与控制, 2021, 29 (8): 172-177.