

基于 D-star 和 DWA 的改进机器人导航方法*

何壮壮 丁德锐

(上海理工大学 光电信息与计算机工程学院 上海 200093)

摘要: 移动机器人导航是机器人领域的一项基础研究,其重要意义在于自主移动是机器人能够完成其它复杂任务的前提。在 ROS 机器人操作环境下提出了一种改进的基于 D-star 算法和动态窗口法融合的改进导航算法。该算法由两部分组成:1)基于 D-star 的全局导航算法,可以为机器人规划出一条最优的全局路径;2)基于动态窗口法的局部路径规划算法,可以规避动态或机器人行进过程中突然出现的障碍物。有效的改善了传统导航算法在已经规划好全局路线的情况下无法动态避障的能力,提高了机器人导航性能的稳定性。最后,基于设计的两轮机器人平台,通过仿真和真实环境联合论证,验证了所开发机器人导航方法的有效性。

关键词: 移动机器人;导航;D-star 算法;动态窗口法;动态避障

中图分类号: TN201 **文献标识码:** A **国家标准学科分类代码:** 520.20

Improved robot navigation method based on D-star and DWA

He Zhuangzhuang Ding Derui

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Mobile robot navigation is a basic research in the field of robotics. Its important significance is that autonomous movement in the environment is the premise that the robot can complete other complex tasks. In this paper, an improved new navigation algorithm based on D-star algorithm and dynamic window algorithm fusion is proposed in the robot operating system (ROS). The algorithm consists of two parts. One is based on D-star's global navigation algorithm, which can plan an optimal global path for the robot. The second is based on dynamic window algorithm's local path planning algorithm, which can avoid dynamic or a sudden obstacle in the robotic travel. It effectively improves the ability of traditional navigation algorithms to avoid obstacles dynamically when the global route has been planned, and improves the stability of robot navigation performance. Finally, based on the design of the two-wheeled robot platform, the effectiveness of the developed robot navigation method is verified by the joint demonstration of simulation and real environment.

Keywords: mobile robot; navigation; D-star algorithm; dynamic window algorithm; dynamic obstacle avoidance

0 引言

机器人导航的目的就是让机器人具备从当前位置移动到环境中某一目标位置的能力,并且在这个过程中能够保证机器人自身和周围环境的安全性^[1]。导航部分分为两部分:全局路径规划部分用于在环境中找到最佳或接近最佳的路线,局部路径规划描述了如何让机器人遵循全局路径,同时避免移动障碍物。安全避障意味着当机器人移动到目标时,机器人将通过其自身的传感器避开移动或静态障碍物。局部路径规划和避障将控制机器人的运动,它改变了

机器人的方向和速度。在环境未知或环境信息不完整的情况下,路径规划是移动机器人研究的重要课题。

1969 年诞生于美国的世界上第一台智能机器人“Shakey”^[2]就可以在不同的房间之间移动,甚至可以定位和攀爬到平台高处。早期在工业领域中应用广泛的自动引导运输车(automated guided vehicle, AGV)根据其导引方式不同可分为电磁轨道和射频标记等多种方式^[3]。在服务机器人领域,就有小米公司生产的智能扫地机器人,集成了旋转激光和同步定位与建图(simultaneous localization and mapping, SLAM)算法,能够自动创建家庭环境地图并根据

收稿日期:2018-12

* 基金项目:国家自然科学基金(61573246)、上海市自然科学基金(18ZR1427000)、上海市科委部分地方院校能力建设项目(15550502500)资助

• 122 •

地图规划出清扫路线。

路径规划的结果将直接确定机器人是否能够准确到达目的地。在全局导航中,应该有已知的环境条件,现在已经开发了许多用于全局导航的方法,如Voronoi图^[4]、人工势场法^[5-6]、Dijkstra算法^[7]、网格法^[8]和细胞分解方法^[9],以及模糊逻辑^[10]、神经网络^[11]等成功应用于各种研究人员解决本地导航问题。在局部导航中,机器人可以使用配备的传感器自动决定或控制其运动和方向,例如超声波测距传感器,红外线距离传感器和视觉(相机)传感器等。

在本文中,提出融合了两种导航方法的改进的改进导航方法。该方法基于D-star算法和动态窗口法(dynamic window algorithm,DWA)。D-star是全局路径规划的核心算法,DWA是局部路径规划算法的关键算法。

1 路径导航算法

1.1 全局路径规划算法

在栅格地图中,Dijkstra^[12]、广度优先搜索(breadth first search,BFS)^[13]和D-star算法是常用的典型全局路径优化算法。该算法用于查找一个节点与所有其他节点之间的最短路径。Dijkstra算法的主要特征是它从起始点延伸到外层,直到到达终点。Dijkstra使用贪婪策略,它将首先从相邻点中选择最近的点,然后它将点延伸到最近的点并再次找到最近的点,直到找到最终目标。Dijkstra的优点是该算法总能找到节点之间的最短路径。但是Dijkstra的时间复杂度非常高。它必须访问起点周围的所有节点,所以Dijkstra的效率非常低。

BFS(广度优先搜索)使用启发式方法以指导搜索方向^[14],BFS的操作与Dijkstra相同,但BFS会评估节点到目标的成本。它不是选择最接近初始节点的节点,而是通过选择最接近目标的节点来对打开的列表进行排序。BFS无法保证将选择最短路径。但是,它比Dijkstra算法快得多,因为它使用启发式函数快速找到目标节点的路径。

D-star现在是寻找最短路径的常用方法。D-star结合了Dijkstra和BFS算法,D-star将启发式函数放入Dijkstra算法当中去,算法的效率增加了很多。每个节点的代价函数是:

$$C_{cost} = f(n) = g(n) + h(n) \quad (1)$$

式中: $g(n)$ 表示节点和起始点之间的代价值; $h(n)$ 表示节点和目标点之间的代价值,决定能否找到最优解; $d(n)$ 则代表着节点和目标点之间的实际距离。我们有如下三种情况,当 $h(n) < d(n)$ 时,代价估算小于实际距离。在这种情况下,算法将搜索更多的点,更大的范围。所以效率会很低,但是这种算法可以得到最优解;当 $h(n) = d(n)$ 时,代价估算与实际距离相等。在这种情况下,算法将严格地以最短路径规则搜索空间,效率最高;当 $h(n) > d(n)$ 时,在这种情况下,搜索范围和搜索点将很低,效率会很高。

D-star算法结合动态A*算法^[15],算法首先使用

Dijkstra找到从开始到目标的最短路径,然后机器人通过全局路径移动。如果下一个节点无效,则D-star算法将更新当前姿态的成本,然后它将生成A*路径,当新A*路径中的节点已经在原始路径中时停止,其算法框架如图1所示:

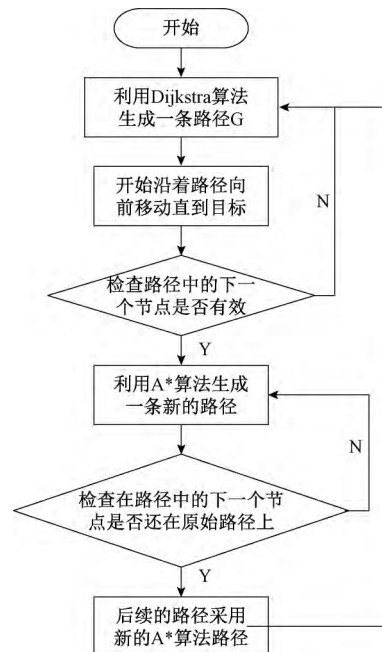


图1 D-star算法框架

1.2 局部路径规划算法

局部路径规划与全局路径规划不同。例如,如果您想将车开到不知名的地方。您首先从手机或计算机获取全局导航路径。然后在途中你必须考虑如何开车,如何避免其他车。第1步是全球路径规划的过程,第2步就像局部路径规划一样。局部路径规划也是一个问题的路径。局部路径规划是关于机器人的运动学问题,且涉及到障碍物识别算法。物体识别通常需要获取机器人的相对位置和障碍物的移动方向。通过障碍物识别,机器人可以制动,减速或改变方向。由于机器人的形状不同,不同的机器人必须选择面向相同环境的不同策略^[16]。最近常用的局部路径规划方法之一是动态窗口法。

动态窗口方法由Fox提出^[17],该算法主要根据机器人运动学模型对速度空间 (v, ω) 中的不同速度进行采样,该算法将根据速度在一段时间内模拟机器人的轨迹,然后它将通过评分规则对这些模拟轨迹进行评分。最后,机器人将以与下一个圆圈得分最高的轨迹相对应的速度移动。算法的突出优点是动态特征,它根据机器人的加速度将机器人速度设置在一定范围内。

速度采样的关键是可以根据速度模型预测轨迹模型。但是有无限的速度组,速度的采样必须设定在一定范围内。下面根据机器人和环境的限制描述限制速度的条件。

1)速度限制,机器人最大和最小速度的限制。该等式可以设定速度 v 和旋转速度 ω 的范围:

$$V_m = \{v \in [v_{\min}, v_{\max}], \omega \in [\omega_{\min}, \omega_{\max}]\} \quad (2)$$

2)加速度限制,机器人的最大和最小加速度根据机器人电机的性能来限制。因此,当机器人移动时,速度处于动态窗口内的某个时间段内。动态窗口中的速度是机器人在模拟时间内实际可达到的速度。式中: v_c 和 ω_c 为当前速度。

$$V_d = \{(v, \omega), v \in [v_c - v_b \cdot \Delta t, v_c + v_a \cdot \Delta t], \omega \in [\omega_c - \omega_b \cdot \Delta t, \omega_c + \omega_a \cdot \Delta t]\} \quad (3)$$

3)安全距离限制,速度也具有安全距离限制。在最大减速条件下,机器人必须在撞到障碍物之前停下来。

$$V_{safe} = (v, \omega) \mid v \leq \sqrt{2 \cdot dist(v, \omega) \cdot v_b} \wedge \omega \leq \sqrt{2 \cdot dist(v, \omega) \cdot \omega_b} \quad (4)$$

$dist(v, \omega)$ 为轨迹 (v, ω) 上机器人到障碍物的最短距离。在模拟轨迹之前不预测到障碍物的距离,在模拟速度的轨迹之后可以计算距离。如果机器人在与障碍物碰撞之前可以停止,则该组速度是可接受的,否则这个速度将被舍弃。

4)得分规则,动态窗口方法的另一个关键点是评分规则。这些规则用于对每个轨迹进行评分。得分最高的轨迹将是下一循环中机器人的最佳轨迹。对每个轨迹进行评分

的规则是:

$$G_{(v, \omega)} = \sigma(\alpha \cdot heading(v, \omega) + \beta \cdot dist(v, \omega) + \gamma \cdot velocity(v, \omega)) \quad (5)$$

5)角度评估,该评估规则用于对机器人朝向目标的目标与机器人处于模拟轨迹的末端时的机器人方向之间的角度差进行评分。角度差越小,分数越大。

6)速度评估, $velocity(v, \omega)$ 将根据速度曲率的变化对轨迹进行评分,得分方程为:

$$K = \frac{|k_t - k_{last}|}{\sum_{i=0}^n |k_i - k_{last}|} \quad (6)$$

式中: n 表示轨迹是个数; k_t 表示 t 时刻速度的曲率; k_{last} 表示最后时刻的最佳曲率。

2 ROS 中 move_base 框架分析与改进

2.1 move_base 中导航模块分析

在机器人操作系统(robot operating system, ROS)中 move_base 模块的整个框架中,导航模块是最重要的核心部分,导航模块主要处理过程如图 2 所示。现在,move_base 包使用插件模式来获取全局路径并获得机器人速度。在这个 move_base 框架中,机器人有 3 个主要的机器人状态。它包括规划状态(PLANNING)、控制状态(CONTROLLING)和清除状态(CLEANING)。

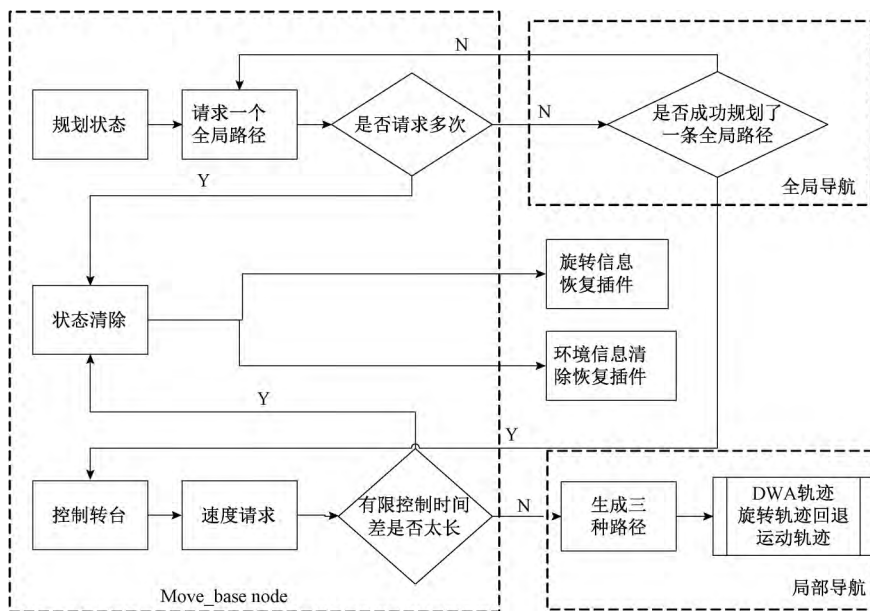


图 2 导航模块主要处理过程

move_base 框架中导航部分的主要过程是:

1)机器人首先等待目标消息。如果机器人收到目标消息,机器人将切换到 PLANNING 状态。然后,move_base 节点将尝试从全局路径规划插件请求全局路径计划,此时在 ROS 中的插件使用 A* 算法。

2)如果节点已获得规划好的路径,则机器人状态将更

改为 CONTROLLING 状态,否则模式将再次尝试请求全局路径。如果节点已连续尝试多次未成功,则机器人状态将更改为 CLEARING 状态。

3)机器人状态更改为 CONTROLLING 后,节点将从局部路径规划插件请求速度命令。稍后如果局部插件在一段时间内无法返回有效速度,则机器人状态将更改为

CLEARING。

在CLEARING状态下,机器人将清除障碍物信息并旋转以生成新地图。如果所有恢复行为都失败,程序将中止。局部路径规划部分主要使用DWA轨迹。

下面在move_base中对DWA轨迹的分析:ROS中的DWA轨迹与原始DWA不同。此DWA有3个评分规则。它包括路径评分,目标评分和方向角评分。前2个是局部目标和路径之间的距离,第3个是模拟角度和目标角度之间的角度差。

$$S_{(v,\omega)} = \sigma(\alpha \cdot \text{heading}(v,\omega) + \beta \cdot \text{dist}(v,\omega) + \gamma \cdot \text{velocity}(v,\omega)) \quad (7)$$

机器人运动效果的关键是如何设定目标得分的系数 β 和路径得分的 γ ,如果 β 大于 γ ,机器人将总是倾向于移动到目标,当起始方向与路径方向有很大差异时,机器人将始终转动一个大圆圈,机器人运动轨迹如图3所示:

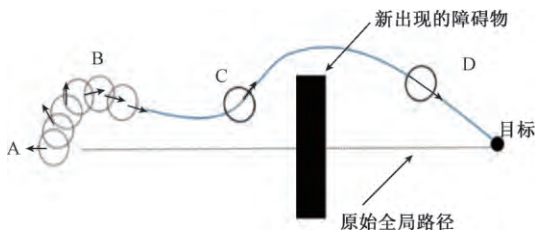


图3 $\beta > \gamma$ 时机器人的预估轨迹

该算法框架对于仅具有静态障碍的环境中的机器人似乎是可施行的。但是当机器人遇到原始地图中没有的新障碍时,它总是会失败。因为机器人只会请求一次全局路径。DWA的采样周期通常为1~2s。采样周期越长,机器人将花费更多的处理器来计算最佳轨迹。因此机器人如果以0.7m/s的速度,机器人就会停留在障碍物前面。在图3中的B点,对于大多数情况,机器人将尝试慢速一段时间然后它可以找到轨迹。或者机器人将进入清除状态,机器人将旋转以清除环境信息,然后它可以找到路径。但代价是用了很长时间。如果机器人的方向与全局路径的方向有很大的角度差,机器人将总是需要更多的时间来旋转调整位姿。

对于旋转轨迹,机器人倾向于首先移动并转向。如果除了机器人之外还有一堵墙,环境和机器人移动轨迹如图4所示,机器人将花费大量时间来转身,并且可能机器人将撞到墙上。

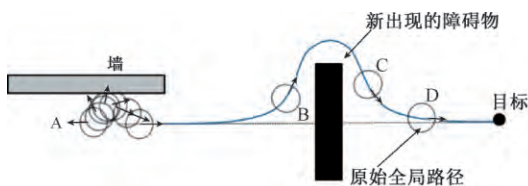


图4 当空间较小状态下的碰撞

2.2 改进的基于D-star和DWA的导航方法

针对上面的问题,我们已经讨论了很多move_base框架和这种导航方法的缺点。因此在本节中,基于move_base框架提出了改进的新导航方法。在这个新导航框架中,它有两个主要部分,一个是全局路径规划D-star算法部分,另一个是局部路径规划DWA算法部分。新的框架有两个主线程,全局部分线程(global thread,GT)和局部部分线程(local thread,LT)。新的导航框架的过程解释如下,且改进的导航框架图如图5所示:

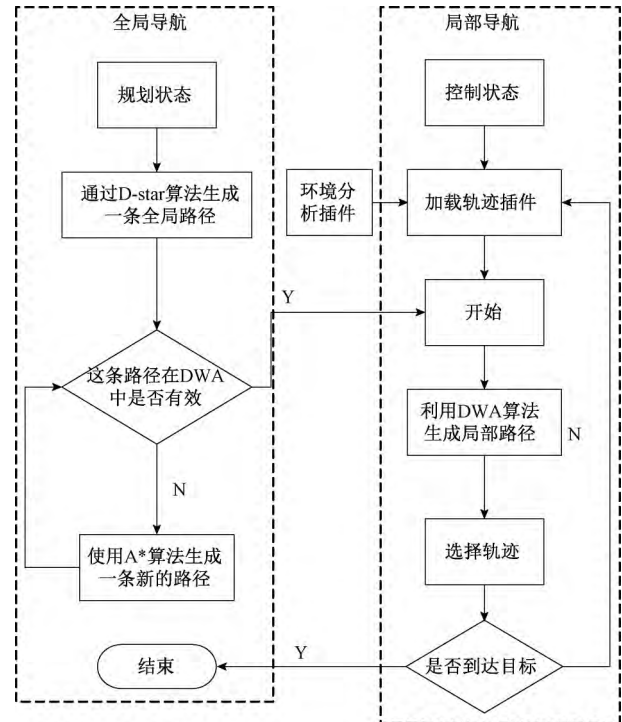


图5 改进的导航框架

具体实现步骤如下所示:

1)程序初始化GT和LT。然后GT进入等待目标的状态。LT加载环境分析插件,然后进入等待全局路径的状态;

2)在框架获得目标消息后,GT生成D-star的全局路径。然后GT将全局路径发送到LT。GT开始检查机器人动态窗口中路径的有效性。该动态窗口是机器人传感器的检测范围。在此动态窗口范围中,将更新障碍物信息。如果机器人移动时原始路径无效,GT将生成到当前目标的新A*路径。然后GT将发送到LT新路径;

3)LT将首先通过环境插件分析环境,并选择环境的情况并使用给定的全局路径执行运动轨迹。如果机器人已达到目标,则LT将END状态返回GT,否则返回CONTROLLING状态。

如果路径在局部动态窗口中无效,那么它将在全局路径规划中通过D-star算法生成到目标的新路径。DD算法

的关键点是机器人将始终分析路径在动态窗口中是否有效。虽然 DWA 可以避开障碍,但是对于更长的路径模拟,DWA 将花费更多的时间。DD 算法节省了机器人运动的时间和距离。

对于局部路径规划部分,它首先将加载环境分析插件。这些插件将分析机器人的环境功能。环境分析插件还提供与环境特征相对应的不同轨迹。然后机器人可以选择不同的轨迹。对于默认环境,局部路径规划算法现在是 DWA 轨迹。机器人的运动是一个非常复杂的问题。只有一种机器人轨迹不能适应所有环境条件。对于 DWA 的缺点已在第 2.1 节中进行了分析,不同的环境特征需要不同的轨迹。

本文还添加了用于机器人旋转的机器人插件。插件检查机器人开始移动时的角度,如果角度与路径方向有很大差异,则首先转向目标方向。在 move_base 中的原始 DWA 轨迹中,算法始终检查机器人得分的所有可能轨迹,然后选择最佳轨迹。在此插件中,已保存检查所有可能轨迹的时间。并且机器人的运动将显示出如图 6 所示的更好的结果。

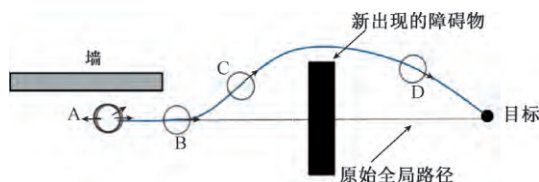


图 6 DD 导航法的机器人运动轨迹

3 实验结果与分析

3.1 实验平台搭建

用于研究开发定位方法的移动机器人如图 7 所示。机器人运动系统是由两个带有减速器和编码器的直流电机独立驱动独立轮子。通过安装附加层来安装必要的设备可测量驱动轮的旋转角速度,同时安装有惯性测量单元和二维激光雷达。机器人的移动平台半径约为 0.6 m,其质量约为 30 kg,载重高达 100 kg。

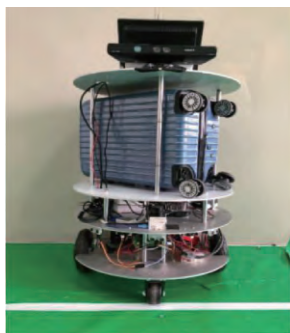


图 7 两轮机器人平台

3.2 导航实验与分析

环境地图的建立我们使用安装在机器人上的二维激

光雷达传感器,环境信息来自于激光雷达的环境“扫描”。激光雷达在 180° 的范围内扫描障碍距离数据,采用极坐标表示这些距离数据,进而得到激光雷达扫描得到视图。在机器人行驶过程中结合编码器距离信息进行自定位,从而生成增量式地图,进而生成机器人视图。

基于已经搭建好的机器人平台,下面的测试是对改进的 DD 导航框架进行测试,当机器人正在移动的时候,突然在机器人前面出现障碍物的时候,实验结果和过程如图 8 所示。

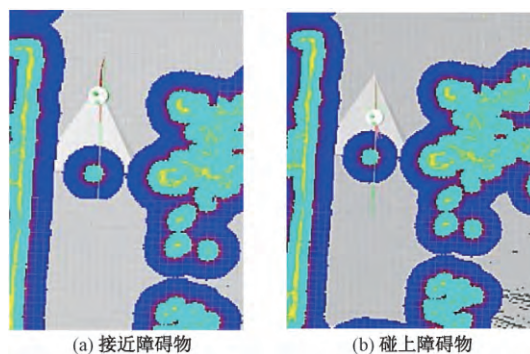


图 8 原始导航框架仿真

从图 8、9 可以看出:同样是在一条路径上,当机器人前方 1.5 m 处突然出现障碍物时,我们发现在传统的导航框架下(图 8),当机器人识别到障碍物以后,却并没有做出反应,依然沿着之前规划的路径前行,很快便会撞向障碍物。而本文提出的 DD 导航框架(图 9)却已提前规划好新的路径,并迅速的避开新出现的障碍物,完成后续的导航任务,达到了行进中避开突然障碍物的目标。

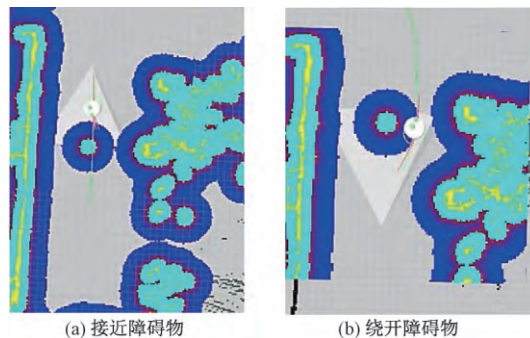


图 9 改进导航框架仿真

以上是在 rviz 中加载现有的地图,然后通过机器人模型进行的仿真效果演示,如图 10 所示为实验室现场环境,以及机器人和障碍物等。

下面来进行实际实验室环境中机器人的导航演示,我们选取实验室中的中间一个狭窄的通道进行实验,当在狭窄通道突然出现障碍物时,机器人在原始路径的基础上也能快速重新规划出一条新的路径来避开障碍物,如图 11 表示从起始点到目标点的全局路径规划,中间黄色的直线



图10 导航实验现场环境

表示规划好的路径,在没有障碍物的情况下,机器人将沿着路径向前移动。



图11 原始路径

如果障碍物突然出现在规划好的路径上(图12),在障碍物出现的一刻,机器人若尚未探测到障碍物且依然沿着路径行走,那样势必会碰撞到障碍物,所以机器人需要提前探测到障碍物,然后调用导航算法来完成运动。

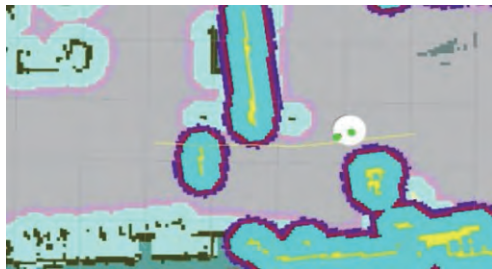


图12 新障碍物突然出现

调用了本文设计好的DD导航方法,在机器人探测到障碍物后,立即迅速的规划出一条新的路径来绕开障碍物,轨迹如图13所示,我们可以看到机器人的位置移动较



图13 DD导航框架规划出的新路径

小,体现了机器人规划算法的实时性和有效性,可以给机器人的导航运动提供有效的保障。

4 结 论

本文讨论了机器人导航算法。它将导航划分为全局路径规划部分和局部路径规划部分。全局路径规划部分不关注局部环境细节。局部路径规划部分不关心其传感器检测范围之外的环境,这将提高机器人导航的效率。为了更好的导航效果,benwen提出了DD导航方法和框架。它描述了DD导航框架的基本过程,因为DD基于move_base导航框架,因此DD导航框架的开发主要集中在move_base框架的问题上。它解决了move_base框架中的常见问题,并添加了环境分析插件。环境分析插件为通过图像处理进行障碍物检测提供了机会,开发人员可以为不同的机器人添加自己的插件,并且DD导航框架可以适应各种类型的机器人,为移动机器人导航提供了一个有效的方法。

参考文献

- [1] 赵国旗,杨明,王冰,等.基于智能终端的移动机器人室内外无缝定位方法[J].上海交通大学学报,2018,52(1):13-19.
- [2] NI J J, WU L Y, FAN X N, et al. Bioinspired intelligent algorithm and its applications for mobile robot control: a survey[J]. Computational Intelligence and Neuroscience, 2015, 2016(2):1.
- [3] BHATTACHARYA P, GAVRILOVA M L. Roadmap-based path planning-using the Voronoi diagram for a clearance-based shortest path [J]. Robotics & Automation Magazine IEEE, 2008, 15(2):58-66.
- [4] MONTIEL O, SEPÚLVEDA R, OROZCO-ROSAS U. Optimal path planning generation for mobile robots using parallel evolutionary artificial potential field[J]. Journal of Intelligent & Robotic Systems, 2015, 79(2):237-257.
- [5] ABIYEV R, IBRAHIM D, ERIN B. Navigation of mobile robots in the presence of obstacles [J]. Advances in Engineering Software, 2010, 41(10):1179-1186.
- [6] SOLTANI A R, TAWFIK H, GOULERMAS J Y, 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.
- [7] 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.
- [8] SHIM H S, KIM H S, JUNG M J, et al. Designing distributed control architecture for cooperative multi-agent system and its real-time application to soccer robot[J]. Robotics & Autonomous Systems, 1997, 21(2): 149-165.
- [9] MONTANER M B, RAMIREZ-SERRANO A. Fuzzy knowledge-based controller design for autonomous robot navigation [J]. Expert Systems with Applications, 1998, 14(1): 179-186.
- [10] PANIGRAHI P K, GHOSH S, PARHI D R. Navigation of autonomous mobile robot using different activation functions of wavelet neural network [J]. Archives of Control Sciences, 2015, 25(1): 21-34.
- [11] RIGELSFORD J. Introduction to autonomous mobile robots[J]. Industrial Robot, 2004, 2(6): 645-649.
- [12] 赵小川, 罗庆生, 韩宝玲. 机器人多传感器信息融合研究综述[J]. 传感器与微系统, 2008, 27(8): 1-4, 11.
- [13] THRUN S. Probabilistic robotics[J]. Communications of the ACM, 2002, 45(3): 52-57.
- [14] JETTO L, LONGHI S, VENTURINI G. Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots[J]. IEEE Transactions on Robotics and Automation, 1999, 15(2): 219-229.
- [15] GURUJI A K, AGARWAL H, PARSEDIYA D K. Time-efficient A* algorithm for robot path planning[J]. Procedia Technology, 2016, 23: 144-149.
- [16] Hernandez E, Carreras M, Ridao P. A comparison of homotopic path planning algorithms for robotic applications[J]. Robotics and Autonomous Systems, 2015, 64: 44-58.
- [17] BALLESTEROS J, URDIALES C, VELASCO A B M, et al. A biomimetical dynamic window approach to navigation for collaborative control [J]. IEEE Transactions on Human-Machine Systems, 2017, 47(6): 1123-1133.

作者简介

何壮壮, 硕士研究生, 研究方向为移动机器人定位与定位、智能优化算法。

丁德锐, 博士研究生导师, 研究方向为随机非线性系统控制与滤波、智能优化算法。

E-mail: zhuangzhuanghe302@163.com