

动态环境中移动机器人多状态转换自主导航

曹其新 黄先群 蒋宇捷 朱笑笑

(上海交通大学机械与动力工程学院, 上海 200240)

摘要 针对已有的移动机器人自主导航方案在通过狭小区域及动态障碍物区域时, 容易陷入局部极值, 导致移动机器人无法通过的问题, 提出一种多状态转换的自主导航方案. 移动机器人根据当前传感器信息, 在初始运动、全局规划、局部规划及终止运动四种状态之间转换, 不仅能够应对动态环境, 而且能改善移动机器人速度震荡的问题, 在复杂环境中表现得更加具有鲁棒性. 实验验证表明: 在狭小环境及动态的超市环境中, 多状态转换自主导航方案的表现优于 ROS(机器人操作系统)开源的 Navigation 导航方案.

关键词 移动机器人; 动态环境; 自主导航; 多状态转换; 局部规划

中图分类号 TP301 **文献标志码** A **文章编号** 1671-4512(2019)10-0022-06

Autonomous navigation of mobile robot with multi-state transition in dynamic environment

CAO Qixin HUANG Xianqun JIANG Yujie ZHU Xiaoxiao

(School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract As the existing mobile robot autonomous navigation scheme is easy to fall into local extremes when passing through narrow areas and dynamic obstacle areas, resulting in the inability of mobile robots to pass, a multi-state transition autonomous navigation scheme was proposed. The mobile robot converted between initial motion state, global planning state, local planning state and termination motion state according to the current sensor information, which could not only work well in dynamic environment, but also improve the velocity oscillation of mobile robot, making the mobile robot more robust in complex environment. Result of experiments shows that the autonomous navigation scheme of multi-state transition has better performance compared with ROS (robot operating system) open source Navigation scheme in the narrow and dynamic supermarket environment.

Key words mobile robot; dynamic environment; autonomous navigation; multi-state transition; local plan

随着机器人技术的发展和人力成本的提高, 机器人技术在越来越多的领域得到应用, 而超市购物机器人就是其中一种. 购物机器人吉塔可携带 19 kg 工具、杂货等物品, 帮助人们解决拎重物的烦恼^[1]. 法比奥是英国首名受雇于超市的购物机器人^[2]. 广东今甲机器人科技有限公司推出了一款针对零售终端的新型定制化机器人——美超人智能售卖机器人, 可以向顾客主动售卖商品. 超市中购物机器人可以代替人类进行导购和拣选商品的任务.

购物机器人依赖于自主导航技术在超市环境中自由移动, 关键在于如何避开超市环境中的障碍物, 本研究重点关注自主导航技术中的移动机器人路径规划技术. 移动机器人的路径规划包括全局规划和局部规划^[3-4], 全局规划算法包括 Dijkstra 算法^[5]、A*算法^[6]和人工势场法^[7-8]等. Dijkstra 算法将路径规划问题转化为图搜索问题, 由起始节点逐层向外扩散直到搜索到目标节点. A*算法在 Dijkstra 算法的基础上添加目标点对扩散节点的距离启发, 加快

收稿日期 2019-03-15.

作者简介 曹其新(1960-), 男, 教授, E-mail: qxcao@sjtu.edu.cn.

基金项目 国家自然科学基金资助项目(61673261, 61703273).

了 Dijkstra 算法的搜索速度. 人工势场法创建虚拟的力, 目标点对机器人产生吸引力, 障碍物对机器人产生排斥力, 最终机器人受到的合力将机器人引导至目标点. 其中 Dijkstra 算法和 A*算法在静态障碍物环境中(无动态物体)表现良好, 但是无法应对包含动态物体的环境; 人工势场法可以应对包含动态物体的环境, 但是容易陷入局部极小值. 局部规划算法主要用于未知环境的探索, 机器人依靠当前局部的环境信息进行规划, 包括人工势场法和动态窗口法^[9]等, 动态窗口法通过对当前机器人速度空间进行采样, 获取相应速度对应的路径, 然后选出最优路径对应的速度. 局部规划算法的优点在于可以应对动态环境, 但是由于缺失全局信息, 因此很难获取最优路径甚至无法到达目标点.

基于全局规划和局部规划的特点, 近年来学者大都采取全局规划结合局部规划的方式^[10-11]. 文献[10]中的自主导航方案(后称 Navigation)能够应对大多数室内环境, 但是当面对狭窄区域及动态障碍物时, 局部规划器产生的速度会出现明显的震荡. 因此, 本研究针对超市拥挤、动态的环境提出了多状态转换的自主导航方案, 通过状态之间的转换使移动机器人能够快速采取正确的决策, 避免了动态窗口法中容易出现的震荡效果, 大大提高了移动机器人在超市环境中的执行速度.

1 移动机器人平台

图 1 为自主研发的移动机器人综合实验平台, 采用四个麦克纳姆轮作为驱动轮, 实现全向移动, 无转弯半径, 可以在狭小区域自由移动, 悬挂机构保证移动机器人能够应对不同的地面, 对角布置的激光传感器用来感知环境, 并且搭载有机械臂. 移动机器人的自主导航算法是基于开源机器人框架 ROS(机器人操作系统)进行开发, 移动机器人的建图、定位方案采用了文献[10]开源的程序, 而移动机器人的路径规划方案是本研究提出的多状态转换的自主导航方案.

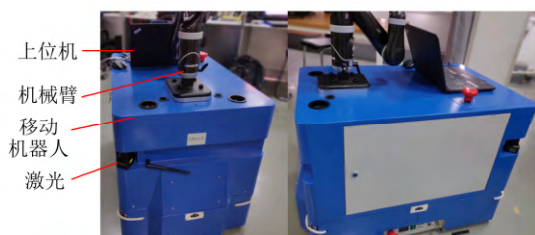


图 1 移动机器人平台

2 多状态转换的自主导航方案

2.1 多状态转换的自主导航方案框架

图 2 为多状态之间的转换, 多状态转换的自主导航方案中包含初始运动状态、全局规划状态、局部规划状态以及终止运动状态四种状态. 在初始运动状态中, 移动机器人根据当前激光传感器信息判断是否在障碍物附近, 并且通过适当运动保证机器人与障碍物保持一定距离; 在全局规划状态中, 移动机器人根据全局静态地图以及当前传感器信息规划出一条从当前位置到目标位置的无障碍路径; 在局部规划状态中, 计算速度使得移动机器人能够成功跟随全局路径, 并且不与环境中障碍物发生碰撞; 在终止运动状态中, 移动机器人减缓速度并且调整最终姿态.

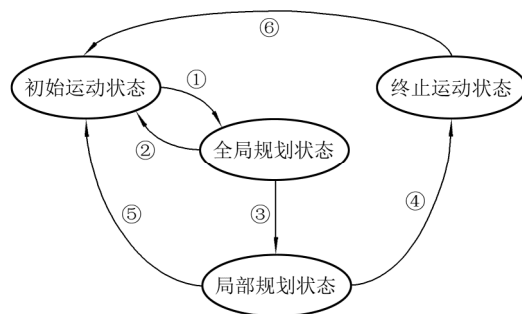


图 2 多状态之间的转换

移动机器人导航任务开始在初始运动状态中, 移动机器人根据传感器信息, 进行运动, 保证一定范围内无障碍物, 然后由过程 ① 进入全局规划状态. 移动机器人将传感器检测的动态障碍物信息叠加到静态全局地图, 用 A*算法规划得到全局路径, 若规划失败, 则由过程 ② 返回初始运动状态, 再次进行初始位置的调整, 若规划成功, 则由过程 ③ 进入局部规划状态, 用路径跟踪算法计算移动机器人的速度, 同时依据实时的传感器信息避障, 若跟随路径长时间被障碍物占据, 则由过程 ⑤ 进入初始运动状态, 调整位置, 重新规划. 当局部规划状态运动到目标位置附近时, 由过程 ④ 进入终止运动状态, 移动机器人减速并调整最终姿态. 结束一次导航任务后, 移动机器人处于终止运动状态中, 当新的任务加入时, 由过程 ⑥ 进入初始运动状态. 各状态需要的输入输出信息如图 3(a)所示.

如图 3 所示, 本研究自主导航方案与 Navigation 导航方案都需要输入的传感器信息、地图以及任务, 输出移动机器人实时的速度, 内部包括了全局规划

及局部规划. 本研究的自主导航方案参考了 Navigation 的导航框架, 两者不同点在于应对动态障碍物的方式. 如图 3(b)所示, 红色箭头表示状态的转换, 可以看到 Navigation 中状态之间的转换是单向的, 因此 Navigation 不能通过状态转换来实现动态避障, 事实上是通过局部规划器中的动态窗口规划算法来实现避障; 而如图 3(a)所示的本文的导航方案, 状态之间切换是双向的, 可以通过状态切换来实现动态避障, 放弃了局部规划器中的避障策略, 使本文导航方案能够更好地应对狭窄环境及动态障碍物, 避免了震荡.

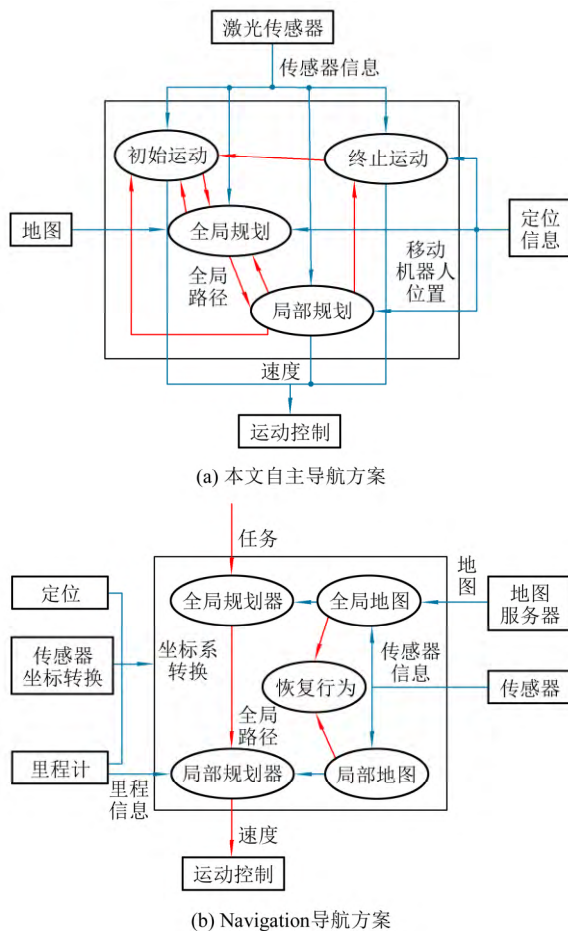


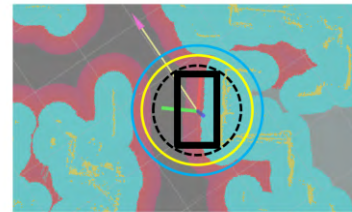
图 3 本文导航方案与 Navigation 导航方案对比

2.2 初始运动状态

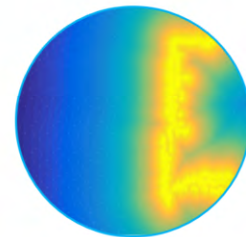
如图 3(a)所示, 初始运动状态读取激光传感器信息, 然后输出移动机器人的运动. 初始运动状态中调整移动机器人在规划前的位置, 防止移动机器人距离障碍物过近, 导致全局规划状态中规划失败以及局部规划状态中发生碰撞.

图 4 为初始运动状态中的人工势场法. 图 4(a)为导航中的部分耗费地图, 障碍物周围按照移动机器人的内切圆半径(蓝色区域)和外接圆半径(红色区域)进行膨胀. 黑色虚线圆表示移动机器人的外接

圆, 半径为 r_1 ; 移动机器人能够原地旋转, 选取以移动机器人中心为圆心, 半径为 r_2 的圆($r_2 > r_1$), 为图 4(a)中黄色的圆; 初始运动状态中的运动属于局部运动, 因此选取以移动机器人中心为圆心, 半径为 r_3 的圆(势场圆), 为图 4(a)中蓝色的圆, 只考虑势场圆内的障碍物.



(a) 移动机器人的外接圆及势场圆



(b) 势场圆中的势场图

图 4 初始运动状态中的人工势场法

在初始运动状态中, 采用人工势场法的思想, 但是只考虑障碍物产生的排斥力和势场圆内的障碍物. 设 $q(x, y)$ 为势场圆内任意一点, $q_0(x_0, y_0)$ 为障碍物位置, 那么由障碍物产生的势场为

$$U(q) = \begin{cases} (k/2)(1/\delta(q) - 1/r_2)^{1/2} & (\delta(q) < r_2) \\ 0 & (\delta(q) > r_2) \end{cases}$$

$$\delta(q) = [(x - x_0)^2 + (y - y_0)^2]^{1/2}$$

式中: $U(q)$ 为势场; k 为常数系数. 移动机器人收到的排斥力为

$$F(q) = -\nabla U(q)$$

如图 4(b)所示, 越亮的区域表示排斥力越大, 处于障碍物附近的移动机器人会被排斥力推离障碍物. 初始运动状态中通过运动使半径为 r_2 的圆内无障碍物, 同时跳出当前状态.

2.3 全局规划状态

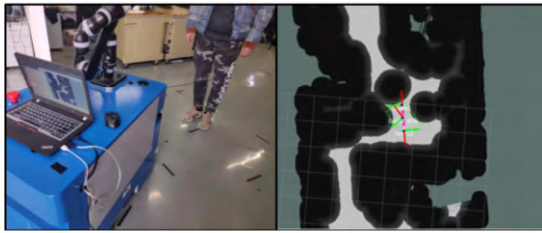
如图 3(a)所示, 全局规划状态读取激光传感器信息、全局静态地图及定位信息, 输出全局路径. 状态中包含地图更新与路径规划两部分, 由于超市是动态环境, 因此须全局静态地图上叠加实时传感器信息得到实时的规划地图, 再利用路径规划算法进行规划.

图 5 为静态地图与实时规划地图. 图 5(a)为环境的静态地图及膨胀后的耗费地图, 将静态地图与实时的传感器信息融合得到实时的规划地图(如图

5(b)所示), 地图更新的频率为 1 Hz, 路径规划采用 A* 算法, 在实时地图中进行规划. 若规划成功, 则进入局部规划状态, 否则返回初始运动状态, 进行位置调整.



(a) 静态地图与静态耗费地图



(b) 动态障碍物与实时规划地图

图 5 静态地图与实时规划地图

2.4 局部规划状态

Navigation 方案采用动态窗口法进行局部规划, 对速度空间采样, 最终选取最优的速度, 虽然能够应对动态物体, 但是在靠近障碍物及狭小区域处容易出现震荡的效果. 本研究的全局规划状态中采用实时地图, 具有动态避障的能力, 局部规划状态中采用路径跟踪算法计算速度. 如图 3(a)所示, 局部规划状态读取全局路径、激光传感器信息及定位信息, 输出移动机器人的运动. 状态中包括碰撞检测及路径跟踪两部分, 碰撞检测根据当前传感器信息计算与周围障碍物距离, 判断是否会发生旋转碰撞或者直行碰撞; 路径跟踪输出运动使移动机器人沿着全局路径移动.

首先考虑移动机器人的角速度, 如图 6 所示, 全局规划状态给定一条路径, 图中: p_1 为路径中距离移动机器人最近的点; p_2 为 p_1 点之后路径上的一点; θ_1 为移动机器人在世界坐标系下的偏航角; θ_2 为当移动机器人位置到 p_2 点时的向量与世界坐标

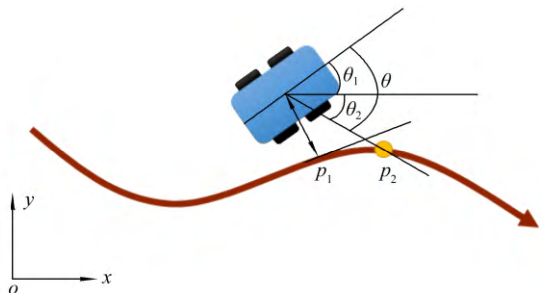


图 6 局部规划状态中的路径跟踪

系 x 轴正向的夹角. 设 $\theta = \theta_2 - \theta_1$, 那么移动机器人当前的角速度

$$\omega = \begin{cases} \dot{\theta} & (r_{\text{obs}} \text{ 为假}); \\ 0 & (r_{\text{obs}} \text{ 为真}), \end{cases}$$

式中: 当 r_{obs} 为真时, 表示移动机器人在当前位置旋转会导致碰撞; 当 r_{obs} 为假时则不会.

其次考虑移动机器人的线速度, 本研究在局部规划状态中不允许移动机器人侧向移动, 只考虑移动机器人前后方向上的速度. 移动机器人的线速度 v 满足

$$v = \begin{cases} v_{\text{max}} & (0 < |\omega| < \omega_t, l_{\text{obs}} \text{ 为假}); \\ v_{\text{max}} - k_v(|\omega| - \omega_t) & (|\omega| > \omega_t, l_{\text{obs}} \text{ 为假}); \\ 0 & (l_{\text{obs}} \text{ 为真}), \end{cases}$$

式中: v_{max} 为移动机器人在路径跟踪时的最大速度; ω_t 为最大线速度下允许的最大角速度值; k_v 为正常数; 当 l_{obs} 为真时, 表示移动机器人直行会导致碰撞. 当移动机器人角速度绝对值大于 0 小于 ω_t 时, 移动机器人采用最大线速度 v_{max} ; 当角速度绝对值大于 ω_t 时, 移动机器人进行适当减速; 当移动机器人前进会导致障碍时, 则应立即停止.

在局部规划状态中, 当移动机器人到障碍物的距离小于 d_f 时, 机器人进入终止运动状态; 在跟踪路径过程中, 遇到障碍物被迫停止后, 移动机器人原地等待时间 t_{local} , 若 t_{local} 内障碍物离开, 则继续跟踪原有路径, 否则返回初始运动状态, 调整机器人位置, 进行重新规划.

2.5 终止运动状态

如图 3(a)所示, 终止运动状态读取激光传感器信息和定位信息, 输出移动机器人的运动. 状态中包括初调姿态和精调位姿两部分. 初调位姿是移动机器人在刚进入终止运动状态时, 就进行姿态的调整(不考虑位置), 初调阶段只考虑是否会发生旋转碰撞, 则

$$\omega = \begin{cases} e_{\theta} \omega_{\text{max}} / |e_{\theta}| & (|e_{\theta}| > \theta_t, r_{\text{obs}} \text{ 为假}); \\ \frac{e_{\theta}}{|e_{\theta}|} \frac{\omega_{\text{max}}}{\theta_t} e_{\theta} & (|e_{\theta}| > \theta_t, r_{\text{obs}} \text{ 为假}); \\ 0 & (r_{\text{obs}} \text{ 为真}), \end{cases}$$

式中: e_{θ} 为当前偏航角的误差; ω_{max} 为初调姿态中的最大速度; θ_t 为常数, $e_{\theta} / |e_{\theta}|$ 为方向.

精调位姿阶段对移动机器人的位置和姿态进行精确调整, 须同时考虑直行碰撞和旋转碰撞, 直至移动机器人到达目标点(位姿误差允许范围内). 那么移动机器人的线速度 v_x , v_y 和角速度为

$$\omega = \begin{cases} e_{\theta} k_{f\omega} / |e_{\theta}| & (r_{\text{obs}} \text{ 为假}), \\ 0 & (r_{\text{obs}} \text{ 为真, 或 } |e_{\theta}| < e_{\theta a}); \end{cases}$$

$$v_x = \begin{cases} e_x k_{fv} / |e_x| & (r_{\text{obs}} \text{ 为假}), \\ 0 & (r_{\text{obs}} \text{ 为真, 或 } |e_x| < e_{xa}); \end{cases}$$

$$v_y = \begin{cases} e_y k_{fv} / |e_y| & (r_{\text{obs}} \text{ 为假}), \\ 0 & (r_{\text{obs}} \text{ 为真, 或 } |e_y| < e_{ya}); \end{cases}$$

式中: $k_{f\omega}$ 和 k_{fv} 分别为角速度和线速度的调速系数; $e_{\theta a}$, e_{xa} , e_{ya} 分别为偏航角、 x 方向位移及 y 方向位移的停止精度。

在终止运动状态中, 移动机器人会一直保持运动, 直到距离目标点的距离在允许范围内。在移动机器人接收新的任务后, 跳转到初始运动状态, 开始下一次的自主导航。若目标点周围障碍物过多, 导致终止运动状态中停留时间过长, 则会使效率低下, 因此设置时间 t_1 , 移动机器人进入终止运动状态时间超过 t_1 后, 会自动结束本次导航, 开始接收下一个任务。

3 室内导航实验

为了验证多状态转换的移动机器人自主导航方案的有效性以及在超市环境中的优越性, 本研究进行了两种实验。两种实验均采用图 1 中的移动机器人, 算法运行的计算机平台是联想 E450, 内存 4 GiB, 算法开发依赖于开源的机器人框架 ROS 的 indigo 版本, 移动机器人最大线速度为 0.3 m/s, 最大角速度为 0.5 rad/s。

第一种实验是在作者的实验室环境中进行多状态转换的自主导航方案的可行性验证。

如图 7 所示, 环境中包含狭窄的弯道及动态的人, 实验中随机指定 5 个任务点, 人为走动制造动态的障碍物, 记录移动机器人的执行时间、路径长度以及平均速度。

如表 1 所示, 移动机器人采用多状态转换的自主导航方案能够很好地完成动态环境中的任务, 5 次随机生成的任务都能成功地完成, 并且在执行任务的平均速度上也较为稳定。表 1 中: L 为路径长度; T 为执行时间。

其次, 本研究对比了在狭窄弯道处本文的多状态转换自主导航方案与 Navigation 方案, 选取图 7 左下角的图片中的弯道, 在弯道头尾各选取一个位置, 运用两种导航方案各执行 5 次, 所得实验结果如表 2 所示。

如表 2 所示: 在狭窄的弯道处, 本文方案执行

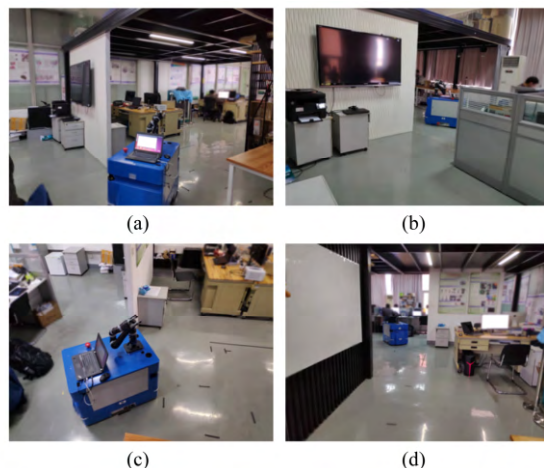


图 7 狭窄的实验室环境

表 1 随机任务下多状态转换自主导航方案测试数据

任务序号	L/m	T/s	平均速度/ $(\text{m}\cdot\text{s}^{-1})$
1	5.29	39.6	0.13
2	5.51	41.7	0.13
3	10.38	70.0	0.15
4	11.72	79.1	0.15
5	5.72	50.7	0.11

表 2 狭窄弯道处两种导航方案花费时间对比

方案	执行次数					平均时间
	1	2	3	4	5	
Navigation	68.3	45.1	104.0	88.6	76.5	76.5
本文	47.8	43.5	43.0	42.2	45.7	44.5

相同任务花费的平均时间明显小于 Navigation 方案, 并且平均执行时间仍小于 Navigation 方案的最短时间。因为在狭窄的弯道处, Navigation 方案的局部规划采用的动态窗口法常常会导致机器人震荡, 难以越过狭窄的弯道, 而本文方案能够很好地克服这一点, 并且在执行相同任务时花费的执行时间也很稳定, 基本与平均执行时间一致, 这一点也优于 Navigation 方案。

第二种实验是在模拟超市搭建的环境中进行实验。模拟的超市环境尺寸为 $3\text{ m} \times 6\text{ m}$, 环境中包含 3 个货架及走动的人, 将本文方案与 Navigation 方案进行对比, 随机执行 5 个任务点, 记录两种方案各自的执行时间和路径长度。

两种导航方案的实验数据如图 9 所示, 在 5 次任务中, 两种导航方案都能够成功地执行任务, 表明两种导航方案都适用于动态环境。但是, 只有在第一次任务中, Navigation 方案的路径长度(3.72 m)和执行时间(32.57 s)略小于本文方案的路径长度(3.86 m)和执行时间(34.33 s)。在其余 4 次任务中, Navigation 方案的路径长度和执行时间都大于本文方案, 尤其是执行时间差别较大。在执行任务过

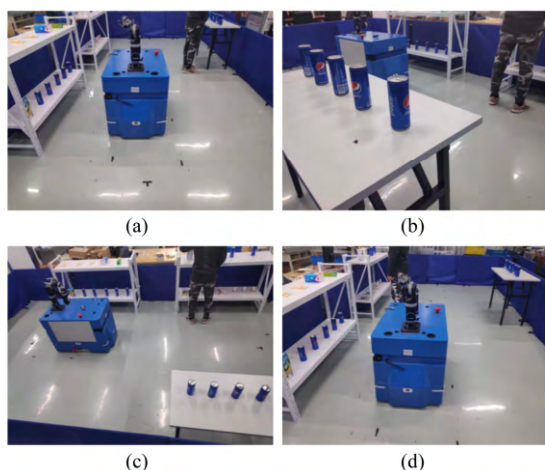
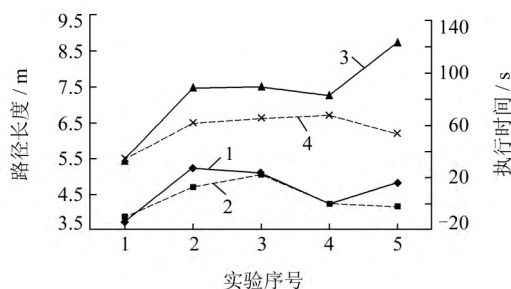


图 8 模拟超市环境



1—Navigation 方案路径长度；2—本文方案路径长度；
3—Navigation 方案执行时间；4—本文方案执行时间。

图 9 模拟超市环境中两种导航方案对比

程中, 当面对动态障碍物时, Navigation 方案通过速度采样、评分的方式获取最佳速度, 但是常常会产生震荡, 虽然能够避开障碍物, 但是越过障碍物的能力较差. 而当本文方案面对动态障碍物时, 通过状态的转换来实现避障以及规划, 重要的是移动机器人一直试图沿着全局路径靠近目标点, 不是为了避障而在障碍物附近产生震荡, 因此如图 9 中数据显示, 本文方案的执行时间明显小于 Navigation 方案, 因此本文方案更加适用于动态的超市环境.

4 结语

针对超市拥挤的环境特点, 本研究创新地提出了一种多状态转换的自主导航方案, 通过状态之间的转换应对狭窄的动态环境. 通过实验验证, 本文中的自主导航方案在超市环境中其表现优于 Navigation 方案. 本研究在 4 种状态中引入了局部规划状态原地等待时间、终止状态中角速度和线速

度的调速系数等多个参数, 目前都是根据经验进行选取, 以后希望通过机器人自主学习的方式进行获取, 以进一步提高导航的效率.

参 考 文 献

- [1] WIKIPEDIA . Gita[EB/OL] . [2019-01-13] . [https://en.wikipedia.org/wiki/Gita_\(mobile_carrier\)](https://en.wikipedia.org/wiki/Gita_(mobile_carrier)) .
- [2] PATRICIA MARX . Learning to love robots[EB/OL] . [2019-01-13] . <https://www.newyorker.com/magazine/2018/11/26/learning-to-love-robots> .
- [3] 李敏 . 基于视觉的扫地机器人导航系统设计与实现[D] . 合肥: 中国科学技术大学图书馆, 2018 .
- [4] CHERNI F , BOUTEREAAY , REKIK C , et al . Autonomous mobile robot navigation algorithm for planning collision-free path designed in dynamic environments[C]// Proc of 2015 9th Jordanian International Electrical and Electronics Engineering Conference (JIEEEEC) . New York : IEEE , 2015 : 1-6 .
- [5] DIJKSTRA E W . A note on two problems in connexion with graphs[J] . Numerische Mathematik , 1959 , 1(1) : 269-271 .
- [6] HART P E , NILSSON N J , RAPHAEL B . A formal basis for the heuristic determination of minimum cost paths [J] . IEEE transactions on Systems Science and Cybernetics , 1968 , 4(2) : 100-107 .
- [7] 宋建辉, 代涛, 刘砚菊 . 基于改进人工势场法的移动机器人路径规划[J] . 计算机工程与科学, 2017(7) : 1328-1332 .
- [8] 程金勇, 王钦钊 . 基于改进人工势场法的机器人路径规划研究[J] . 科技与创新, 2017(22) : 49-50 .
- [9] FOX D , BURGARD W , THRUN S . The dynamic window approach to collision avoidance[J] . IEEE Robotics & Automation Magazine , 1997 , 4(1) : 23-33 .
- [10] MARDER-EPPSTEIN E , BERGER E , FOOTE T , et al . The office marathon : robust navigation in an indoor office environment[C]// Proc of 2010 IEEE International Conference on Robotics and Automation . New York : IEEE , 2010 : 300-307 .
- [11] 吕太之 . 移动机器人路径规划和地图创建研究[D] . 南京: 南京理工大学图书馆, 2017 .