

基于 Voronoi 图法的移动机器人路径规划

许松清, 吴海彬, 林 宜, 高洪张, 陈天炎

(福州大学 机械工程及自动化学院, 福建 福州 350002)

摘要: 研究了一种在随机静态环境条件下移动机器人的路径规划方法. 在环境地图信息的基础上, 运用 Voronoi 图理论, 建立了移动机器人运行的无碰撞路径网络. 按特定的算法将给定的移动机器人起始点和目标点连入该路径网络, 然后采用 Dijkstra 最短路径搜索算法, 找出一条从起点到终点的最短路径. 最后控制机器人沿着找到的最短路径运行到目标点位置, 实现移动机器人的智能避障. 整个系统以 TI 公司生产的 TMS320LF2407A 型号数字信号处理器(DSP) 为控制器的核心.

关键词: 移动机器人; 路径网络; Voronoi 图; Dijkstra 算法

中图分类号: TP 24

Path planning of mobile robot based on voronoi diagram method

XU Song-qing, WU Hai-bin, LIN Yi, GAO Hong-zhang, CHEN Tian-yan

(College of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350002, China)

Abstract: A path planning method of mobile robot based on random static environments is studied. Based on environmental map information and use the Voronoi diagram theory, the collision-free path for mobile robot's running is built. The given start point and object point of mobile robot are linked to this path network according to a special formula, then the Dijkstra shortest path search algorithm was used to find out the shortest path from the start point to the object point. Finally, the mobile robot was controlled running along the found shortest path to get to the object position, so, the intelligent obstacle avoidance of mobile robot was achieved. Whole system uses the TI Co.'s TMS320LF2407A digital signal processor(DSP) chip as controller.

Key words: mobile robot; path network; Voronoi diagram; Dijkstra algorithm

路径规划是移动机器人研究领域的一个主要研究课题, 机器人难免要在有各种障碍物的环境中运行, 路径规划的目标就是为机器人寻找一条最优无碰撞路径. 路径规划可以分为全局路径规划和局部路径规划两大类. 全局路径规划是按照给定的环境信息, 在机器人运行前就已经为机器人规划好一条运行路径, 也称为静态路径规划; 局部路径规划是指机器人在运行过程中, 通过自身安装的传感器不断感知周围环境信息, 对路径进行实时调整, 因此也称为动态路径规划. 文献[1]提出了一种运用连通图的路径规划方法, 文献[2]研究了静态不确定性环境下非完整轮式移动机器人路径规划问题, 提出了一种实时路径规划方法. 本文基于 Voronoi 图理论的路径规划方法属于全局路径规划范畴, 但可适用于随机静态环境, 也就是说在机器人运行过程中环境是静态的, 环境中的各种障碍物是静止不动的, 但在机器人进行路径规划之前环境是不确定的, 环境中的各种障碍物的大小、位置等参数是可变的. 因此采用在随机静态环境条件下移动机器人的路径规划方法的机器人具有一定的适应环境的能力, 更符合实际应用的要求.

Voronoi 图是俄国数学家 G. Voronoi 最早发现的, 由于具有独特的几何特性, 从发现以来其应用已涉

及自然科学的多个领域, 在计算机辅助几何设计、气象学、生态学以及计算机科学等领域有着广泛的应用, 文献[3] 和文献[4] 研究了 Voronoi 图在潜艇自主导航系统和刀具走刀路线规划方面的应用. 在障碍物环境中运用 Voronoi 图理论进行移动机器人的运动规划, 其优点已被人们所发现^[5], 如果把障碍物近似成质点, 那么机器人沿着环境障碍物的 Voronoi 边行走碰到障碍物的几率是最小的, 如果障碍物不能近似成质点, 可运用扩展的 Voronoi 图理论进行路径规划. 虽然 Voronoi 图理论在机器人路径规划方面具有突出的优点, 但在中小型机器人上由于受到控制器的运算能力等限制, 其实际应用受到很大限制. 本文不仅从理论高度上阐述了 Voronoi 理论在机器人路径规划中的实现算法, 更重要的是把该算法在自主研制的以 DSP 为控制器的小型移动机器人上加以实现. 从起始点到目标点的最优路径问题属于单源最优路径搜索问题, 文中采用 Dijkstra 算法实现移动机器人的最优路径搜索.

1 基于 Voronoi 图理论的环境建模

1.1 Voronoi 图算法的数学建模

Voronoi 图构建的基本原理^[3]为: 在一个已确定了尺度的量度平面上, 对该平面上分布的离散点集进行区域划分, 使划分区域中的点到点集中某一点的距离比其到点集中所有其他点的距离小. 具体的数学描述为:

设平面 B 是一个已确定了尺度的量度平面, 设 P 为平面上的离散点集, $p_1, p_2, p_3, \cdots, p_n \in P$ 且对应的坐标为 $(x_i, y_i) (i = 1, 2, 3, \cdots, n)$, 取 B 平面上任一点 (x, y) , 建立公式:

$$\sqrt{(x-x_i)^2+(y-y_i)^2}<\sqrt{(x-x_j)^2+(y-y_j)^2}$$

(1)

使公式(1) 对于选定点 P_i , 任取点 $P_j \in P$ 且 $P_j \neq P_i$ 成立的点 (x, y) 形成的轨迹称为离散点 P_i 的 Voronoi 区域. 由此可得, 最终把平面 B 划分成 n 个相互邻接的多边形, 并且每个多边形中有且只有包含一个离散点. 因此如果点 (x, y) 位于相邻多边形的公共边上, 则有

$$\begin{cases} \sqrt{(x-x_i)^2+(y-y_i)^2}=\sqrt{(x-x_j)^2+(y-y_j)^2} \\ \sqrt{(x-x_j)^2+(y-y_j)^2}<\sqrt{(x-x_k)^2+(y-y_k)^2} \end{cases}$$

(2)

其中 P_i, P_j 为相邻两多边形包含的离散点, 任取 $P_k \in P$ 且 $P_i \neq P_j \neq P_k$, 式(2) 成立. 称由式(2) 确定出的点 (x, y) 形成的轨迹为离散点集 P 的 Voronoi 边.

1.2 机器人运行环境预处理

显然在机器人运行环境中, 障碍物是有大小的, 而且形状也各不相同, 要把障碍物近似成离散的质点就需要对运行环境进行一些预处理. 首先, 移动机器人作二维平面运动, 障碍物不考虑高度信息; 其次, 小间距障碍物合并处理, 即如果两个障碍物相距太近, 障碍物之间距离小于机器人通过的最小安全距离, 则将两个障碍物合并作为一个障碍物处理; 再次, 作出障碍物的外接圆, 并对障碍物外接圆进行径向扩张且对环境边界作径向扩张, 移动机器人即可退化运动质点处理. 因此路径规划问题就转化为在预处理后的地图平面上, 对点机器人进行路径规划, 简化了问题的复杂性. 预处理后机器人运行环境如图 1 所示.

经过预处理后的运行环境比原始环境附加了一些预处理信息, 环境信息更加复杂, 为了便于无碰撞路径网络生成, 需要对预处理后的运行环境进行适当的简化, 并提取障碍物的离散中心作为 Voronoi 建模的生成元. 由于障碍物形状大小各不相同, 显然如果要精确描述环境信息, 要占用运动机器人微控制器的大量资源, 因此进行如下简化: 把具体的障碍物从环境图中去除, 只留下由障碍物外接圆径向扩张和边界径向扩张后留下的抽象区域, 把这些抽象区域定为移动机器人运行的危险区域, 机器人只有在危险区域外行走才认为是安全的. 最后提取 Voronoi 生成元, 显然障碍物危险区域为圆形区域, 取圆形危险区域的圆心作为障碍物的近似质心, 并以此

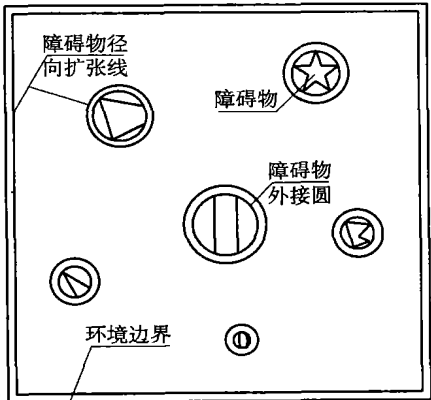


图 1 经预处理后环境示意图

Fig. 1 Environment map after pretreatment

作为 Voronoi 生成元. 简化结果如图 2 所示.

经简化过后的环境地图只由简单的几何图形构成, 采用几何信息表示法来表示环境信息具有存储量小, 便于处理的优点. 基于几何信息的地图表示法^[6]是指从环境的感知信息中提取抽象的几何特征, 并使用这些抽象的几何信息来描述机器人运行环境的一种方法, 这种地图表示方法显得比较紧凑, 便于位置估计和目标识别. 因此对于简化后的环境地图, 只需存储环境范围 x_{\max} 和 y_{\max} 以及危险区域的圆心坐标 (x_i, y_i) 和危险区域半径 R_i 就可以完全描述整个环境的地图信息.

1.3 基于 Voronoi 图的路径规划仿真

众所周知当机器人沿着路径行走时, 由于系统总是存在误差, 行走过程中难免会偏离运行路径, 并且偏移量随着运行时间的增长而不断累积. 如果规划好的路径虽然可以达到行走距离最短, 但是却距离障碍物很近, 那么机器人在该路径行走时就容易碰上障碍物而使整个系统不能正常运行. Voronoi 图法是保证机器人运行路径安全的一种常用方法, 由公式 (2) 可知 Voronoi 边是自由空间的中轴线, 通常能保证机器人在运行过程中最大限度地远离障碍物.

经过环境预处理后提取了 Voronoi 生成元, 即可由公式 (2) 计算出具有 n 个障碍物的 Voronoi 边, 并把得出的 Voronoi 边作为移动机器人的运行路径. 算法的仿真程序用 VC++ 高级语言编写, 取障碍物的个数为 6 个即经过预处理后可得生成元 $n=6$, 整个机器人运行环境为 $10\text{ m}\times 10\text{ m}$, 仿真结果如图 3 所示.

分析算法的仿真结果可知, 由于把障碍物近似成质点, 生成的无碰撞路径网络只是近似最大限度地远离障碍物, 而且路径网络远离障碍物的距离与障碍物的大小有关, 障碍物越小, 无碰撞路径网络规划的效果就越好.

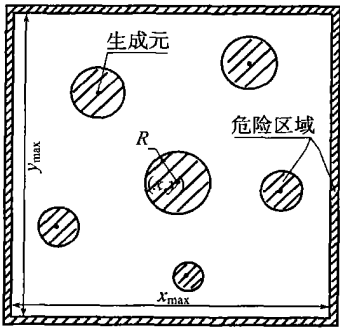


图 2 简化的环境地图

Fig. 2 Environment map after simplify

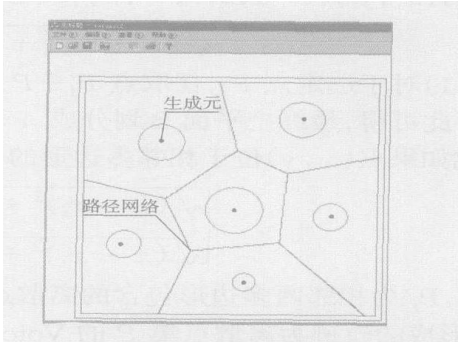


图 3 算法仿真结果

Fig. 3 Emulator result of algorithm

2 Dijkstra 最短路径搜索算法(Dijkstra shortest path search algorithm)

路径规划的目标就是为机器人寻找一条无碰撞路径. 无碰撞路径网络建立了以后, 接下来就必须按给定的起始点和目标点, 为机器人找出一条行走路径. 由于采用 Dijkstra 算法进行最短路径搜索, 首先必须建立路径图, 并计算出路径图中各边的权值. 显然, 由 Voronoi 算法得出的无碰撞路径网络中, 并没有包含给定的起始点和目标点. 为了实现路径的自主搜索, 对已建立的无碰撞路径网络按一定规则生成包含起始点和目标点的路径图:

规则 1: 无碰撞路径网络节点如果在环境边界上, 则取该节点所在边的中点作为路径图的新节点, 否则保留原节点作为路径图的新节点, 称由新节点组成的路径图为固定路径.

规则 2: 分别把起始点和目标点与距其最近的新节点相连接, 称该连接为临时路径, 因此搜索由固定路径和临时路径组成的包含起始点和目标点的图, 就可以找到一条从起始点到目标点的路径.

规则 3: 计算路径图中边的权值, 对于临时路径取边的长度 d 作为边的权值, 而对于固定路径权值计算公式如下:

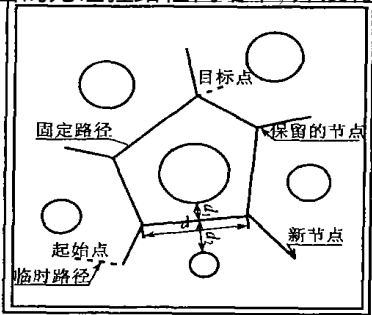


图 4 路径图

Fig. 4 Path diagram

$$t = k_1/\min(d_1, d_2) + k_2d \tag{3}$$

式中: t 为权值; k_1 为危险代价系数; k_2 为长度代价系数; d_1 和 d_2 为边到边两侧危险区域的最小距离; d 为边的长度. 按设定的规则建立的路径图如图 4 所示.

2.1 Dijkstra 算法的基本思想

求两点之间的最短路径, 可以应用图论中的 Dijkstra 算法来求解. Dijkstra 算法是戴克斯特拉最早提出的, 该算法的基本思想就是: 逐步地把节点集合分为两类, 第一类是已被测试的同源节点有最短距离的节点, 第二类是留下有待进一步测试的节点. 测试过或未测试过的节点都保留着值, 对第一类其值就是到源节点的最小距离; 对第二类其值表示当前找到的到源节点的路径最小距离, 随着算法的执行, 当找到较短路径时, 其值就改为后者. 在第二类节点值调整之后, 再找其中具有最小值的一个节点, 将它重新分类到第一类中. 称刚刚被分类到第一类里的节点为主节点, 算法就以其为轴心, 用其值来调整留下的第二类节点的值, 一直继续下去, 当目标点被分类到第一类时, 算法结束.

2.2 算法仿真结果

算法的仿真程序是在 VC++ 环境下用 C 语言编写, 取起始点 start 为(0, 0), 目标点 end 为(450, 720), 公式 3 中 k_1 取 0, k_2 取 1, 仿真结果如图 5 所示. 由于临时路径随着给定的起始点和目标点不同而不同, 因此赋权图的邻接矩阵为动态邻接矩阵, 在采用 Dijkstra 算法搜索最短路径之前应有计算当前邻接矩阵的过程. 由该算法搜索得到的最短路径为 start→H→G→I→B→end, 如图 5 中粗黑线标记.

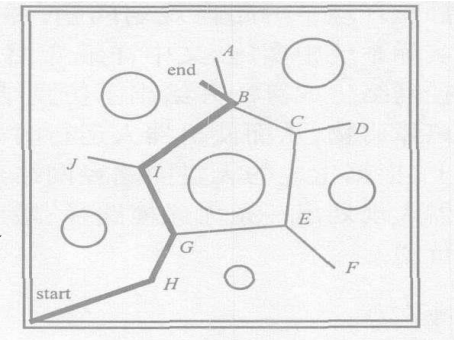


图 5 路径搜索结果

Fig. 5 Path search result

3 算法在移动机器人系统上的实现

3.1 移动机器人系统构成

本移动机器人系统机械结构采用典型的三轮式机器人, 其中两个单独驱动的固定式后轮, 前轮为万向轮, 驱动电机为直流电机, 通过齿轮传动带动光电编码器输出驱动轮的反馈信息, 构成对直流电机的闭环控制. 机器人小车的微控制器采用 TMS320LF2407A 型号 DSP 芯片, 该芯片运行频率可达 40 MHz, 适合作一些算法的微运算器. 整个系统以 F2407A 为控制器的核心, 扩展了机器人本体的人机接口(液晶平面显示器(liquid crystal display, LCD)显示及按键)模块、环境摄像模块、驱动电机模块、光电编码器反馈模块以及通用分组无线业务(general packet radio service, GPRS) 人机交互模块等. 移动机器人本体的一些模块如图 6 所示.

3.2 算法的移植

为了实现机器人的智能避障, 需要把上述两算法的仿真程序结合起来并改写为 DSP 能够执行的程序代码. TI 公司的 TMS320LF2407A 型号的 DSP 芯片软件设计支持 C 语言、汇编语言以及两者的混合编程, 采用混合编程的方法, 可以较好地结合 C 语言和汇编语言的优点, 编写出效率较高的程序代码. 因为算法仿真程序的编制都是在 VC++ 环境下用 C 语言编写的, 所以可以方便地把程序段移植到 DSP 程序中. 但是程序段的移植得针对 DSP 程序设计的特点作相应的变动, 因为该型号的 DSP 芯片属于定点系列, 程序中涉及变量类型、数学运算以及负数问题等和上位机有所不同. 此外, DSP 系统的程序设计要注意与硬件电路相匹配.

在 DSP 程序编制过程中, 公式 3 中取 $k_1 = 0, k_2 = 1$, 即均以路径长度作为权值, 减少了 DSP 计算量, 简化了程序编写. 同时为了使系统避障功能更加智能化, 引入了障碍物预测机制. 所谓障碍物预测是指在给定的起始点和目标点之间直接用直线连接, 判断走直线是否会碰上危险区域, 如果不会则扩张机器人直

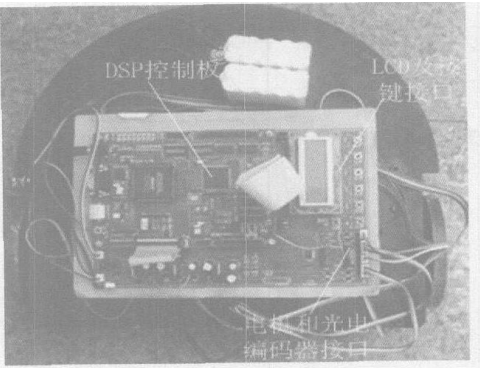


图 6 机器人系统

Fig. 6 Robot system

接从起始点走直线到达目标点, 否则 DSP 转入执行路径生成算法, 最后控制机器人沿着算法规划出的最短无碰路径, 通过走折线到达目标点.

3.3 试验结果

机器人试验时, 选取 6 个障碍物并通过 LCD 和按键设定环境中障碍物半径 R_i 及位置坐标 (x_i, y_i) , 并给定目标点位置 $(450, 720)$, 然后让机器人自主运行, 实验取得了良好的结果, 实现了移动机器人的智能避障. 移动机器人实际运行路径如图 7 所示.

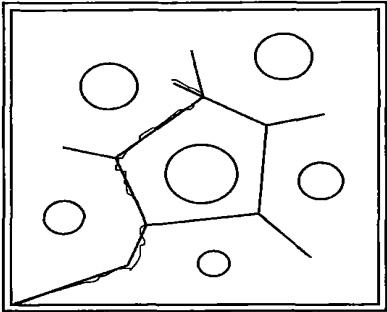


图 7 实际行走路径
Fig 7 Real moving path

4 结语

通过理论分析结合实验结果, 应用 Voronoi 图理论可以解决随机静态环境下的路径规划问题, 并且经过适当的简化, 该理论可以在实际系统中实施. 文中详细介绍了运用 Voronoi 图理论的无碰撞路径网络生成算法并给出了仿真结果, 使所求路径近似最大限度地远离障碍物, 从而使机器人运行时碰上障碍物的可能性降低, 提高了移动机器人运行的可靠性; 在这个基础上, 根据生成的无碰撞路径网络建立含有起始点和目标点的路径图, 运用 Dijkstra 最优路径搜索算法为机器人规划出一条无碰撞路径; 最后, 用 DSP 实现了算法的程序编制. 因此该路径规划方法具有一定的实用价值.

参考文献:

[1] Peter D H, Erland R A J. Symbolic and geometric connectivity graph method for routing planning in digitalized map[J]. IEEE Tran on Pattern Analysis and Machine Intelligence, 1992, 14(5): 549— 563.

[2] 贾艳华, 虞 钢. 静态不确定性环境中非完整轮式移动机器人路径规划研究[J]. 机器人, 2003, 12: 661— 664.

[3] Gold C M. The use of the dynamic Voronoi data structure in autonomous marine navigation[EB/ OL]. http://www.voronoi.com/pdfs/1995-1999/The_use_of_the_dynamic_Voronoi_data_structure.pdf, 2005— 10— 20.

[4] FU Zhang, LIU Cheng-liang, YIN Yue-hong, et al. Tool path planning using Voronoi diagram and three stacks[J]. Chinese Journal of Mechanical Engineering, 2001, 14: 314— 318.

[5] Aurenhammer F. Voronoi diagrams—— A survey of a fundamental data structure[J]. ACM Computing Surveys, 1991, 23: 345— 405.

[6] Chong K S, Kleeman Lindsay. Mobile robot map building from an advanced sonar array and accurate odometry[J]. International Journal of Robotics Research, 1999, 18(1): 20— 26.

(上接第 319 页)

6 结语

计算机辅助设计在油路块开发中的应用, 大幅度地提高了开发效率和精度. 同时也为设计人员提供了直观、形象的参考模型, 为产品的制造提供了更准确的参数信息. 当然, 随着计算机技术在机械设计领域里的更深入的应用, 油路块计算机辅助设计的功能也将会逐渐完善和强大. 尤其最优化设计功能的实现将为油路块开发提供更强大的支持.

参考文献:

[1] 江雪松. AutoCAD 2000 二次开发指南[M]. 北京: 国防工业出版社, 2000.

[2] 张树兵. Visual Basic 6. 0 中文版入门与提高[M]. 北京: 清华大学出版社, 1999.

[3] 张晋西. Visual Basic 对 AutoCAD2000 二次开发[M]. 北京: 清华大学出版社, 2002.

[4] 王 钰. 用 VBA 开发 AutoCAD2000 应用程序[M]. 北京: 人民邮电出版社, 1999.