

动态二叉树表示环境的 A* 算法及其在 足球机器人路径规划中的实现

唐平¹, 杨宜民²

(1. 广东工业大学计算机学院; 2. 广东工业大学自动化学院, 广州 510090)

[摘要] 提出采用二叉树表示二维空间的方法, 对全局路径规划和局部路径规划进行综合考虑, 设计移动机器人在复杂环境下对动态障碍物进行避障的 A* 算法。在足球机器人系统中进行仿真, 将二叉树动态地表示球场的机器人与目标为对角线的矩形环境, 使搜索范围随搜索进程动态地减小, 实现了路径规划的整体优化。

[关键词] 动态二叉树; A* 算法; 路径规划

[中图分类号] TP301.6; TP242 **[文献标识码]** A **[文章编号]** 1009-1742 (2002) 09-0050-04

1 引言

对移动机器人的路径规划研究, 近年来已发展许多新技术, 包括神经网络 (NN), 进化算法 (GA) 和模糊控制等。对静态障碍物的路径规划已取得较好的解决方法, 而对动态障碍的避障, 仍然是路径规划中的难题, 文献 [1~3] 中提出用带有时标的四叉树表示环境, 对二维空间轴作递归分解, 对时间轴引入匀速度的概念, 试图解决在具有动态障碍物环境下移动机器人的路径规划问题。但当智能机器人的工作环境较大时, 结点分裂的增多使表示环境模型的四叉树过于庞大, 从而增加了存储空间和计算时间, 不同程度地扩大了搜索空间, 不能满足动态环境下智能机器人路径规划的实时性。笔者提出采用二叉树表示二维空间的方法, 利用二叉树遍历方法, 设计移动机器人在复杂环境下对动态障碍物进行避障的 A* 算法^[4]。在足球机器人系统中进行仿真, 将二叉树动态地表示球场的机器人与目标为对角线的矩形环境, 使搜索范围随搜索进程动态地减少。采用带有时标的动态二叉树表示方法, 时间信息中增加了加速度, 即考虑了相对的全局路径规划, 又考虑了局部路径规划的智能控制, 从整体上减少了搜索时间, 提高了实时性, 实

现了路径规划的整体优化。

2 动态二叉树表示二维空间方法

二叉树是树结构中最简单的一种结构^[5]。在复杂环境下, 障碍物是动态的。当某时刻动态障碍物处于已知环境中, 将环境简化为一个矩形, 根据障碍物的大小简化为一个矩形或多个矩形的叠加。

设矩形环境 E 的参数结构为 $\{0, X, 0, Y\}$ 。将矩形环境 E 进行 K 层二分递归。根结点表示整个矩形环境平面, 根结点的二个子结点表示矩形环境平面上等分的二个子区间, 依此类推, 得到了二维空间的二叉树的表示。其中, K 为二叉树的深度, 最小的叶结点为一个矩形障碍物的大小。

定义 1 结点 n 表示某矩形区域内无障碍物, 称结点 n 为可规划结点 A 。

定义 2 结点 n 表示某矩形区域内存在障碍物, 称结点 n 为可转换结点 B 。

定义 3 结点 n 表示某矩形区域是一个障碍物, 称结点 n 为不可规划结点 C 。

定义 4 设环境 $E = \{0, X, 0, Y\}$, 其空间面积为 S_E , 则任意结点 n 的空间面积

$$S_n = S_E / 2^i,$$

其中 i 为结点 n 在二叉树中的层数, $1 \leq i \leq K$, K

[收稿日期] 2001-11-08; **修回日期** 2002-05-09

[基金项目] 国家自然科学基金资助项目 (69874024); 广东省科技攻关资助项目 (2KM00502G)

[作者简介] 唐平 (1958-), 女, 湖南桃江县人, 广东工业大学副教授

为二叉树的深度。

二叉树的最多结点为 $2^K - 1$, 由定义4可知, K 的值取决于最大障碍物的矩形空间 S_o , 也就是叶结点的空间, 即: $S_o = S_E / 2^K$,

$$K = \text{Ln}(S_E / S_o)。$$

当 $K = 2, 4, 6, \dots$ 时, 其结点的分裂是在 X 轴进行的;

当 $K = 3, 5, 7, \dots$ 时, 其结点的分裂是在 Y 轴进行的。

设矩形障碍物是一个四元组 $O_s = \{x_{o1}, x_{o2}, y_{o1}, y_{o2}\}$, 表示障碍物的数据结构

$$O = \{(x_{o1}, x_{o2}), (y_{o1}, y_{o2}), (v_{ox}, v_{oy}), (a_{ox}, a_{oy})\}。$$

其中: x_{o1} 表示其左边界的 x 坐标, x_{o2} 表示其右边界的 x 坐标, y_{o1} 表示其下边界的 y 坐标, y_{o2} 表示其上边界的 y 坐标, v_{ox} 表示其速度的 x 分量, v_{oy} 表示其速度的 y 分量, a_{ox} 表示其加速度的 x 分量, a_{oy} 表示其加速度的 y 分量。

设结点 i 是一个四元组 $n_i = (x_{i1}, x_{i2}, y_{i1}, y_{i2})$, 其数据结构表示为

$$\{x_{i1}, x_{i2}, y_{i1}, y_{i2}, llink, rlink, flag\}。$$

其中: x_{i1} 为结点 i 在 t 时刻的左边界 x 坐标, x_{i2} 为结点 i 在 t 时刻的右边界 x 坐标, y_{i1} 为结点 i 在 t 时刻的下边界 y 坐标, y_{i2} 为结点 i 在 t 时刻的上边界 y 坐标。llink 为指向左邻子结点的指针, rlink 为指向右邻子结点的指针,

$$flag = \begin{cases} 1 & \text{结点 } i \text{ 为 } A, \\ 0 & \text{结点 } i \text{ 为 } B, \\ -1 & \text{结点 } i \text{ 为 } C。 \end{cases}$$

很明显, 当矩形环境中没有障碍物时, 表示环境的二叉树只有一个可规划结点 A 结点; 当矩形环境中充满障碍物时, 表示环境的二叉树只有一个不可规划结点 C 结点; 其他情况, 表示环境的二叉树点数都小于等于 $2^K - 1$ 。

根据以上定义, 把机器人和目标的连线, 作为当前考虑的环境模型的矩形对角线。对此环境建立一棵二叉树。当搜索一个时间周期后, 机器人的位置发生了变化, 搜索空间变小 (除非矩形环境中充满障碍物), 必须重新建立一棵新的二叉树。依此类推, 随着机器人位置的变化, 矩形环境变小, 而表示矩形环境的动态二叉树也越来越小。这样, 把一个类似于栅格法描述全局路径规划的二叉树表示环境方法, 演变成既考虑全局路径规划, 又考虑了

局部路径规划的动态二叉树表示环境模型。

当障碍是静止状态时, 即障碍物的速度 v_{ox}, v_{oy} , 加速度 a_{ox}, a_{oy} 均为零, 其路径规划只需对此二叉树进行遍历。例如按层次遍历二叉树对 flag 进行判断, 当 flag = -1 时, 放弃此结点 (即可绕过障碍物), 继续遍历其它结点, 从而找到目标。

3 动态二叉树表示动态环境的A*算法

动态二叉树表示的环境模型, 当障碍物是动态时, 其环境模型也是动态的。当遍历二叉树时, 由于障碍的移入, 原来的 A 结点可能会变成 B 结点, 从而可二分递归分解为 A, B, C 三类结点, 原来是 B 结点, 可能会由于障碍物的移出, 而变为 A 结点。所以移动机器人的路径取决于机器人和障碍物对节点 n 的访问时间, 而访问时间的计算由它们各自的速度、加速度及机器人、障碍物与节点 n 的直线距离计算得到。

定义5 对结点 $i, i \in A$ 结点集, 障碍 O_i 的移入时间, 称对结点 i 的访问时间, 记为

$$V_i(i, O_i) = \bigcup_{i \in A} t(i, O_i),$$

其中 $t(i, O_i)$ 是某个障碍物对结点 i 的访问时间, 而 $V_i(i, O_i)$ 是多个障碍物对结点 i 访问时间中最短的一个访问时间。

定义6 对结点 $i, i \in A \cup B$ 集合, 移动机器人 R 的移入时间称对结点 i 的访问时间, 记为

$$V_i(i, R) = \bigcup_{i \in A \cup B} t(i, R),$$

其中 $t(i, R_i)$ 是某个移动机器人对结点 i 的访问时间, 而 $V_i(i, R_i)$ 是多个移动机器人对结点 i 访问时间中最短的一个访问时间。

访问时间的计算描述如下:

设 $\Delta L_x, \Delta L_y$ 分别为障碍物 O_i 在 t 时间内运行在 X 轴和 Y 轴上的距离坐标分量, 当速度 $v_o \neq 0$, 加速度为 a_o 时, 则访问时间 $t(i, O_i)$ 可由下式得到:

$$\Delta L_x = V_{ox} t_{ox} + a_{ox} t_{ox}^2 / 2,$$

$$\Delta L_y = V_{oy} t_{oy} + a_{oy} t_{oy}^2 / 2。$$

其中: $\Delta L_x = x_{n1} - x_{o1}$, 表示障碍 O_i 在结点 i 的左边; $\Delta L_x = x_{o2} - x_{n2}$, 表示障碍 O_i 在结点 i 的右边; 且

$$y_{n1} \leq y_{o2} \leq y_{n2} \text{ or } y_{n1} \leq y_{o1} \leq y_{n2} \text{ and } v_{ox} \neq 0。$$

$\Delta L_y = y_{n1} - y_{o1}$, 表示障碍 O_i 在结点 i 的下边; $\Delta L_y = y_{o2} - y_{n2}$, 表示障碍 O_i 在结点 i 的上

边;且

$$x_{n1} \leq x_{o2} \leq x_{n2} \text{ or } x_{n1} \leq x_{o1} \leq x_{n2} \text{ and } v_{oy} \neq 0.$$

同理, 设 $\Delta J_x, \Delta J_y$ 分别为机器人 R 在 t 时间内运行在 X 轴和 Y 轴上的距离坐标分量, 当速度 $v_R \neq 0$, 加速度为 a_R 时, 根据下式求得机器人 R 访问 i 结点的时间 t_i :

$$\Delta J_x = v_{Rx}t_{Rx} + a_{Rx}t_{Rx}^2/2;$$

$$\Delta J_y = v_{Ry}t_{Ry} + a_{Ry}t_{Ry}^2/2.$$

定义 7 设 $f = g(n) + h(n)$ 为估计函数, 其中: $g(n)$ 为移动机器人 R 从初始结点 S 到 n 结点的时间估计, $h(n)$ 为移动机器人 R 从 n 结点到目标结点 G 的时间估计。

在动态二叉树表示环境模型的基础上, 对二叉树进行遍历, 筛选掉某时刻具有障碍物的结点 C , 利用人工智能启发式搜索算法 A^* 算法, 经过计算机对当前结点 n 的访问时间, 障碍物对当前结点 n 的访问时间, 并进行比较, 决定结点 n 的 A, B, C 属性。然后进行估价函数的计算, 利用估价函数进行最优路径规划, 设机器人的当前位置为 A_1 , 目标位置为 G , A^* 算法如下:

Step 1 设 $close = \emptyset, open = \{A_1\}, f(s) = h(A_1) + g(A_1), g(A_1) = 0.$

Step 2 建立表示以 A_1 与 G 为对角线的矩形环境的二叉树, 如果 $A_1 = G$, 则成功。

Step 3 遍历二叉树, 将 $flag = 1$ 的结点移入 $open$ 表中。

Step 4 计算 $open$ 表中结点的 $f()$ 值。

Step 5 在 $open$ 表中选 $f()$ 之最小的一个结点 $n, open = open - \{n\}, close = close + \{n\}.$

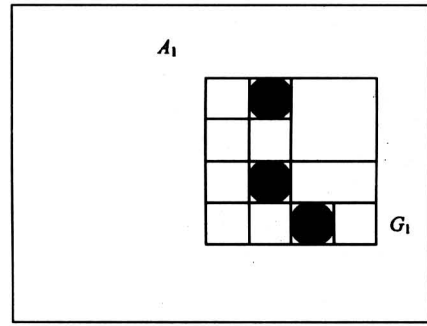
Step 6 若 $V_r(n, R) < \alpha V_r(n, O_t)$, 则 n 为路径规划结点, 令 $A_1 = (n \text{ 的当前位置}), open = \{A_1\}, g(A_1) = 0$, 转 step 2; 否则: 若 $S_n \leq 2S_E/2^k$, 则放弃 n 结点, $close = close - \{n\}$, 转 Step 4; 否则, 递归二分结点 n , 并将 $flag = 1$ 的后继结点移入 $open$ 表中, 转 Step 4。

算法中 α 为访问时间的权值, 一般取 $\alpha \geq 2$ 。

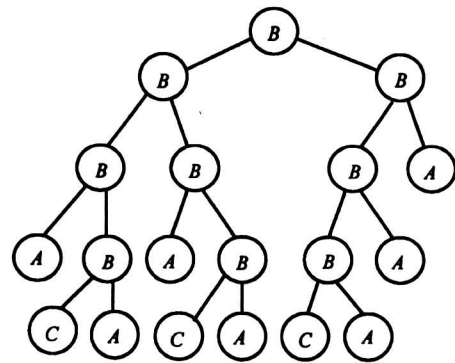
4 仿真及结果

足球机器人系统中的足球机器人的路径规划是一个复杂环境下对动态障碍的避障过程。进行路径规划时, 除了本足球机器人以外, 认为其他足球机器人都是障碍物, 而且是动态障碍物。由于其球场是一个 $150 \text{ mm} \times 130 \text{ mm}$ 的二维平面, 且足球机

器的速度和加速度已知, 把此算法灵活地应用到足球机器人的路径规划中, 将任何一个机器人 A_1 与目标 G_1 的连线, 作为一个动态环境矩形 E_1 的对角线。对此矩形建立一个动态二叉树。图 1 为动态环境 E_1 , 图 1a 表示具有 3 个障碍物的动态环境 E_1 , 图 1b 为二叉树表示的动态环境 E_1 。建立动态二叉树算法的流程图如图 2 所示。



a. 具有三个障碍物的动态环境 E_1



b. 二叉树表示的动态环境 E_1

图 1 动态环境 E_1

Fig.1 Dynamic environment E_1

在仿真算法中, 将球场视为 $C -$ 空间环境, 机器人 A_1 、目标 G_1 和障碍均视为一个点, 在一个时间周期内, 任何一个机器人和目标均可构成动态矩形环境, 都用一棵动态二叉树表示, 采用二叉树表示环境的 A^* 算法, 估价函数为: $f(n) = g(n) + h(n)$, 其中: $g(n)$ 为机器人 R 对结点 n 的访问时间, $h(n)$ 为机器人 R 从结点 n 到目标 G 的时间。

该算法与栅格法相比, 既考虑了全局路径规划, 又考虑了局部路径规划, 是栅格法的一种动态改进。与四叉树表示环境的算法相比, 结点分裂明显减少。在 Mirosot 平台的仿真中, 因为它是递归算法, 算法的大部分时间都花在建立动态二叉树。

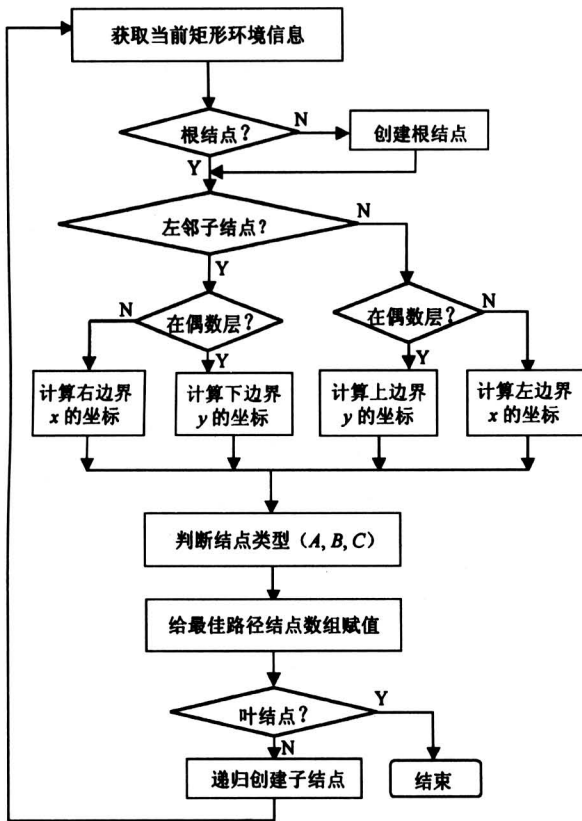


图 2 建立动态二叉树的流程图

Fig.2 The flow chart of dynamic binary tree

动态二叉树的大小决定机器人思考的速度，同时，当障碍物与机器人的距离大于 $4r$ (r 是机器人的平面直径) 时，碰撞的几率为零，但当障碍物与机器人的距离小于 $4r$ 时，机器人必须加速。当机器人多于 3 个并且障碍物与机器人的距离小于 $4r$ 时，可能产生碰撞。

5 结语

移动机器人的动态避障是一个智能搜索问题，其中环境建模技术在智能搜索中显得非常重要。Fujimura 提出用八叉树表示环境模型和文献 [1,2] 中提出用四叉树表示环境模型，其环境被分割的节点繁多而使计算复杂化。笔者提出的动态二叉树表示环境模型的 A* 算法，从全局路径规划和局部路径规划综合考虑，利用人工智能的智能搜索技术 A* 算法，在时间信息中考虑了加速度因素，提出基于动态二叉树表示动态环境的群体障碍物的智能路径规划。在足球机器人系统仿实现中，将二叉树动态地表示球场的机器人与目标为对角线的矩形环境，使搜索范围随着搜索过程动态缩小。从整体上减少了搜索空间，提高了实时性，实现了智能的路径规划。

参考文献

- [1] Noborio H, Naaniwa T, Arimoto S. A feasible motion-planning algorithm for a mobile robot on a quad tree representation [A]. Proc IEEE Int Conf Robot and Automation [C], 1989. 327~332
- [2] 李 强, 林良民, 颜国正. 一种新的移动机器人环境模型 [J]. 机器人, 1999, 21 (5): 379~385.
- [3] 张 育. 移动机器人路径规划算法研究 [D]. 上海: 上海交通大学, 1996
- [4] 蔡自兴, 徐光佑. 人工智能及其应用 [M]. 北京: 清华大学出版社, 1996
- [5] 严慧敏, 吴伟民. 数据结构 [M]. 北京: 清华大学出版社, 1996

Study on Algorithm A* Based on Dynamic Representation of Binary Tree for Environment and Robotic Path Planning

Tang Ping¹, Yang Yimin²

- (1. Faculty of Computer, Guangdong University of Technology, Guangzhou 510090, China ;
- 2. Faculty of automation, Guangdong University of Technology, Guangzhou 510090, China)

[Abstract] This paper deals with a way of representing dynamic environment based on binary-tree. A new algorithm A*, which plans the path of soccer robots in complicated environments is presented. With the soccer game's environments represented by dynamic binary-tree, optimal results were obtained in a simulation of the soccer game.

[Key words] dynamic binary-tree; algorithm A*; path planning