

复杂环境下的机器人路径规划蚂蚁算法

朱庆保

(南京师范大学计算机科学系 南京 210097)
(E-mail: ffaqq@263.net)

摘 要 研究了全局静态环境未知时机器人的路径规划问题, 提出了一种新颖的滚动规划蚂蚁算法. 该方法将目标点映射到机器人视野域附近, 再由两组蚂蚁采用最近邻居搜索策略相互协作完成机器人局部最优路径的搜索, 机器人每前进一步, 都由蚂蚁对局部路径重新搜索, 因此, 机器人前进路径不断动态修改, 从而能使机器人沿一条全局优化的路径到达终点. 仿真实验结果表明, 即使在障碍物非常复杂的地理环境, 用本算法也能迅速规划出一条优化路径, 且能安全避碰, 效果十分令人满意.

关键词 机器人, 路径规划, 未知环境, 蚂蚁算法
中图分类号 TP24

Ant Algorithm for Path Planning of Mobile Robot in a Complex Environment

ZHU Qing-Bao

(Department of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097)
(E-mail: ffaqq@263.net)

Abstract The problem of path planning of mobile robot in an environment where the global information is unknown is studied, and a novel ant algorithm based on rolling planning is proposed. First, the object node is mapped to a node nearby the external of eyeshot of the mobile robot, then two groups of ants cooperatively execute a search for the local optimal path for the robot using the nearest-neighbor searching strategy. The ants will execute a local search again once the robot goes forward. So, the path for the robot is altered dynamically, which will make the robot move on a globally optimal path to the ending node. Simulation results indicate that the optimal path which the robot moves on can reach to the end safely and can be rapidly obtained even in complicated geographical environments, the effect being very satisfactory.

Key words Mobile robot, path planning, unknown environment, ant algorithm

1 引言

移动机器人路径规划属于研究机器人控制系统的重要应用基础问题, 也是机器人研究领域中的一个重要分支. 而环境完全未知时的机器人路径规划则是这一研究领域的难点. 已有不少学者提出了解决未知环境下机器人路径规划的方法和策略, 诸如滚动窗口规划方法、各类定位、导航方法等等^[1~3]. 然而, 由于这类路径规划属于 NP-hard 问题, 因

此, 不断寻求更佳的规划算法就成为国内外的一个研究热点, 特别是如何在复杂环境下避免死锁和振荡? 如何使机器人所走路径全局为最优或较优, 仍是有待解决的问题.

从机器人路径规划算法的发展看, 向智能算法、仿生算法发展是一个很明显的趋势. 由意大利学者 M.Dorigo 等人于上世纪 90 年代提出的蚁群优化算法^[4,5] 是一种新的仿生算法, 它的基本思想是模拟蚂蚁觅食运动过程来实现寻优. 近两年来, 已有学者探索性的将蚁群优化算法用于机器人路径规划^[6,7], 其中, 大多数文献主要研究不考虑环境信息时的多机器人协作方法, 诸如对需要协作的任务赋较强信息素, 以吸引其它机器人前来协作、分布式控制等; 也有部分文献研究了已知静态环境下的机器人的路径规划, 这些研究取得了一定的成绩. 但是, 在未知复杂环境下机器人路径规划蚁群算法仍是一个全新的课题.

根据目前的研究现状和不足, 本文研究了一种机器人工作空间中含有复杂未知静态物情况下的移动路径滚动规划蚂蚁算法. 该算法将目标点映射到机器人视野域附近, 再由两组蚂蚁采用最近邻居搜索策略相互协作完成机器人局部最优路径的搜索, 机器人每前进一步, 都由蚂蚁对局部路径重新搜索, 因此, 机器人前进路径不断动态修改, 在局部优化路径的引导下, 可使机器人沿一条全局优化的路径到达终点. 仿真实验结果表明, 即使在障碍物非常复杂的地理环境, 用本算法也能迅速规划出一条优化路径, 且能安全避碰, 效果十分令人满意.

2 环境描述

对于任意二维地形, 存在着有限个未知的静态障碍物. 规划的目的是使机器人由任意起点 g_{begin} , 安全避碰地沿一条较短路径达到任意终点 g_{end} .

在任意时刻, 机器人 Rob 能探测到以当前位置为中心, r 为半径区域内的环境信息, 该区域称 Rob 视野域 View; 记 AS 为 Rob 在二维平面上的凸多边形有限运动区域, 其内部分布着有限个未知的静态障碍物 $Sb_1, Sb_2, \dots, Sb_n$. 设 Rob 的行走步长为 δ , 能自由运动的活动范围为 $[0, \delta]$, 在 AS 中建立系统直角坐标系 Σ_0 , 以 AS 左上角为坐标 0 点, 以横向为 X 轴, 纵向为 Y 轴, 则有 AS 在 X, Y 方向的最大值分别为 $x_{\max}$ 和 $y_{\max}$. 以 δ 为步长将 X, Y 分别进行划分, 由此形成一个栅格, 如图 1 所示. 每行的栅格数 $N_x = x_{\max}/\delta$; 每列的栅格数 $N_y = y_{\max}/\delta$, 其中 Sb_i ($i = 1, 2, \dots, n$) 占一个或多个栅格, 当不满一个栅格时, 算一个栅格. 该划分策略从实用出发, 使场景描述与实际环境严格相符, 规划出的路径保证机器人畅通无阻.

记 $g \in AS$ 为任意栅格, A 为 AS 中 g 的集合, 记 $OS = \{o_1, o_2, \dots, o_m\} \subseteq A$ ($m \geq n$) 为静态障碍栅格集, $Sb_i \in OS$. $\forall g \in A$ 在 Σ_0 都有确定的坐标 (x, y) , 记作 $g(x, y)$, $x = row$, $y = col$, 其中, row 为 g 所在的行号, col 为 g 所在的列号 (为了和序号一致及方便, 定义左上角第一个栅格的坐标为 $(1, 1)$). 令 $C = \{1, 2, 3, \dots, M\}$ 为栅格序号集, $g(1, 1)$ 序号为 1, $g(2, 1)$ 序号为 2, $\dots, g(N_x, 1)$ 的序号为 N_x , $g(1, 2)$ 序号为 $(N_x + 1)$, $\dots$, 如图 1 所示.

根据上述约定, $g_i \in A$ 的坐标 (x_i, y_i) 与序号 $i \in C$ 构成互为映射关系, 序号 i 的坐标

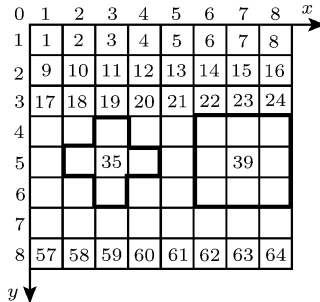


图 1 栅格坐标与序号关系

Fig. 1 Relation between rating coordinate and serial numbers

可由 (1) 式确定

$$x_i = ((i - 1) \bmod N_x) + 1, \quad y_i = (\text{int})((i - 1)/N_x) + 1 \quad (1)$$

式中, int 为舍余取整运算, $\bmod$ 求余运算.

Rob 在 AS 中的位置为记为 P_R , $\forall P_R$ 在 Σ_0 都有确定的坐标 (x, y) , t 时刻的位置表示为 $P_R(t)$, 其坐标为 $(x_R(t), y_R(t))$.

规划起始位置为任意位置 $g_{\text{begin}} \in A$, 且 $g_{\text{begin}} \notin OS$, 终止点 $g_{\text{end}} \in A$, $g_{\text{end}} \notin OS$ 也为任意位置, 其它约束条件为: $g_{\text{begin}}, g_{\text{end}} \in C$, $g_{\text{end}} \neq g_{\text{begin}}$.

3 滚动规划蚂蚁算法

3.1 算法的基本原理及问题定义

经研究发现: 蚂蚁在觅食过程中能够在所经过的路径上留下一一种称为信息素的物质, 而且蚂蚁在觅食过程中能够感知这种物质的存在及其强度, 并以此指导自己的运动方向, 它们倾向于朝着该物质强度高的方向移动. 因此, 由大量蚂蚁组成的集体觅食行为便表现出一种信息正反馈现象: 某一路径越短, 该路径上走过的蚂蚁就越多, 则留下的信息素强度就越大, 后来者选择该路径的概率就越大. 蚂蚁个体之间就是通过这种信息交流来选择最短路径并达到搜索食物的目的. 本文提出的滚动规划蚁群算法就是模拟蚁群这一觅食行为的优化算法.

算法首先将全局目标点映射到 Rob 的视野域 View 的边界外作为局部子目标点 g_{sub} . Rob 每前进一步, 都需用传感器探测其 View 内的环境信息, 在此基础上, 以 Rob 当前位置 $P_R(t)$ 为蚁穴, g_{sub} 为食物源或反之, 蚂蚁的觅食过程就是在 View 内从 $P_R(t)$ 或 g_{sub} 出发, 寻找食物的过程, 经过蚂蚁群体的反复寻食, 由于蚂蚁留下的信息素的正反馈作用, 最终绕开所有障碍物找到一条 Rob 在视野域内的最短局部路径. Rob 每行进一步, 都进行上述局部避碰规划过程, 因此, 其规划出的局部路径是实时动态修改的.

为了减少随机搜索时间, 在算法中增加了一个具有趋近引导作用的启发函数. 为了更好的适应非常复杂地形的路径规划, 提高搜索效率, 用两个蚂蚁家族共 $2m$ 只蚂蚁协作完成最优路径的搜索. 其中, 蚂蚁家族 ant_1 的 m 只蚂蚁以 $P_R(t)$ 为蚁穴位置 (出发点), 以 g_{sub} 为食物源; 蚂蚁家族 ant_2 的 m 只蚂蚁则以 g_{sub} 为为蚁穴位置 (出发点), 以 $P_R(t)$ 为食物源, 两组蚂蚁相向搜索, 从而协作完成规划工作.

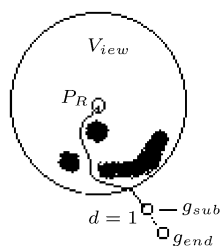


图 2 局部导航路径示意图
Fig. 2 Local navigation path

很显然, 本算法 Rob 总是以最终目标点的方向为前进目标, 更重要的是蚂蚁每次不是只走一步, 也不是计算 Rob 当前位置到子目标点的直线距离, 而是在复杂的障碍环境中找到一条通向子目标点的非常曲折的绕道局部避碰优化路径. 图 2 示意出了在 View 复杂边界环境下蚂蚁找到的局部优化路径, 其中, d 为 g_{sub} 与 View 任意边界间距离. 该路径上的信息素对下一步乃至未来一段时间的走向有明确的引导作用, Rob 每走一步, 修改的是一条局部路径而

不是一步, 加上由两组蚂蚁从两个端点进行相向并行搜索, 且局部路径较短, 因而可有效地避免每次向离目标点最近 (直线距离) 的方向滚动一步而导致的死锁和振荡等. 因此, 本算法即能保证机器人行走安全避碰, 又可使机器人所走路径全局最优或较优.

为了叙述方便, 进一步做出如下约定和定义.

定义 1. $ant_i = \{1, 2, \dots, k, \dots, m\}$ 表示一个蚂蚁家族所有蚂蚁的集合 ($i = 1, 2$), $k \in ant_i$ 表示某只蚂蚁, m 为第 i 蚂蚁家族的蚂蚁总数, $\tau_{ij}(t)$ 表示蚂蚁在 t 时刻残留在栅格 g_i, g_j 连线上的信息量.

定义 2. $\forall g \in A, g \notin OS$, 则称 g 为可行节点, 所有可行节点的集合称可行域, 记为 FS ; $\forall g \in A, g \in OS$, 则称 g 为禁入节点, 所有禁入节点的集合称禁入域, 记为 NFS .

定义 3. 蚂蚁 k 任意时刻所处的位置为 P , $\forall P$ 在 Σ_0 都有确定的坐标 (x, y) , 仅记 k 在 t_i 时刻处于某栅格的位置为 $P(x_i(t_i), y_i(t_i))$ 简记为 P_i 或 $P(t_i)$, 若它与 $g_j(x_j, y_j) \in A$ 的坐标相等, 则视 P_i 与 g_j 等价, 记作 $P_i \sim g_j$.

定义 4. 任意栅格间的距离指两栅格间的连线长度, 记作 $d(g_i, g_h)$ 或 $d(P_i, P_h)$, $i, h \in C$, 由 2 式计算, 特别的, 若有 $d(g_i, g_j)$, $i, j \in C$, 满足 $|j - i| = 1$ 或 $|j - i| = N_x$, (g_i, g_j) 在 AS 中的连线则称一个边 e_{ij} , 简称边 ij , $d(g_i, g_j)$ 则为边长, 记作 d_l .

$$d(g_i, g_h) = \sqrt{(x_i - x_h)^2 + (y_i - y_h)^2} \quad (2)$$

定义 5. $BR_i(g_i(x_i, y_i)) = \{g | g \in A, d(g, g_i) \leq S\}$, $i \in C$ 称 g_i 的邻域或蚂蚁 k 在 g_i 处的视野域. 其中, $S = \{1, \sqrt{2}\}$, 当只能走直线时, $S = 1$, 允许走斜格时, $S = \sqrt{2}$.

图 1 中的粗线十字范围示出了按本定义 $S = 1$ 时算出的 g_{35} 的邻域.

定义 6. 设 t_i 时刻, k 处于 g_i , $Wk_i(g_i(x_i, y_i)) = \{g | g \in BR_i(g_i(x_i, y_i)), g \notin OS, i \in C\}$ 称 t_i 时刻 k 在 BR_i 的可行域; 令 $tabu_k = \{P(t_0), P(t_1), \dots, P(t_i)\}$ 且有 $t_0 < t_1 < \dots < t_i$, 为 k 从 t_0 到 t_i 时刻已走栅格位置. $tabu_k$ 中各位置点的连线称 $P_R(t_i)$ 到局部子目标点 g_{sub} 的通道, 记作 $Path(P_R(t_i), g_{sub})$ (为简便, 也用 $Path(P_R(t_i), g_{sub})$ 表示该通道上的节点集合), 通道的长度记作路程 L , 由 (3) 式计算.

$$L = \sum_{l=1}^e d_l, \quad d_l = d(g_i, g_h), \quad g_i, g_h \neq OS, i, h \in C \quad (3)$$

定义 7. $\eta_{1i}(g_i) = D/d(g_i, g_{sub})$, $\eta_{2j}(g_j) = D/d(g_j, g_R)$, $\eta_{1i}(g_i), \eta_{2j}(g_j)$ 分别称蚂蚁家族 1、蚂蚁家族 2 的第 k 只蚂蚁选择栅格 g_i 或 g_j 的启发函数, D 为权重常数, g_R 为 k 所在栅格.

定义 8. $k_1 \in ant_1, k_2 \in ant_2$, k_1 从 g_R 出发, k_2 从 g_{sub} 出发, 经过 n 个时刻, k_1, k_2 的位置为分别为 P_{k_1}, P_{k_2} , 若有 $|d(P_{k_1}, P_{k_2})| \leq S$, 则称 k_1, k_2 相遇. S 的定义同定义 5.

3.2 算法步骤

根据以上原理和定义, 机器人局部路径滚动规划的蚁群算法步骤描述如下.

Step 1. 产生任意的起点 g_{begin} 和终点 g_{end} 并对有关参数进行初始化;

Step 2. 将 g_{end} 目标点映射到 Rob 视野域 View 边界外作为局部子目标 g_{sub} ; 若有 g_{sub} 与 g_{end} 相同, 规划结束; 否则转 Step3;

Step 3. Rob 在当前位置 $P_R(t)$ 进行环境探测, 测取障碍物 Sb_i 的坐标, 其中 $P_R(t) \sim g_R$.

Step 4. Rob 根据探测到的 $Sb_i, i = (1, 2, \dots, n)$, 用多蚂蚁协作局部路径规划算法迅速规划出从 g_R 到 g_{sub} 的局部优化路径 F_i ; 该路径对应的通道或结点的集合为 $Path(p_R, p_s)$, 其中, $p_s \sim g_{sub}, p_R \sim P_R(t)$;

Step 5. 机器人沿着规划出的局部最优路径向 g_{sub} 前进一步;

Step 6. 返回 Step 2 重复上述过程.

其中 Step 2 产生局部子目标的映射算法如下.

在 t_i 时刻 Rob 的视野域为 $View(P_R(t_i))$, 若有 $d(P_R(x_i, y_i), g_{end}(x_e, y_e)) = r + \delta$, 则取 $g_{sub} = g_{end}$; 此时即为 g_{sub} 与 g_{end} 相同. 否则令

$$d(P_R(x_i, y_i), g_{sub}(x_j, y_j)) = r + \delta \text{ 且满足 } \min d(g_{sub}(x_j, y_j), g_{end}(x_e, y_e)) \quad (4)$$

其中, r —Rob 的视野域 $View$ 的半径; δ —Rob 的行走步长, 两参数均为已知; 因为 g_{sub}, g_{end} 间的直线距离最短, 因此满足上述条件的 g_{sub} 点必定在过 P_R 与点 g_{end} 的直线上, P_R 与 g_{end} 点的坐标为已知, 由此直线方程并根据 (4) 式可求出 g_{end} 映射到子目标点 g_{sub} 的坐标 x_j, y_j .

很显然, 得到的 $g_{sub} \notin View(P_R(t_i))$. 为了在 $View$ 边界 g_{sub} 方向上的障碍物很多的情况下, 蚂蚁仍能找到一条导向目标点的路径, 假定 g_{sub} 与 $View$ 任意边界间的任何栅格均无障碍物, 且规定 g_{sub} 与 $View$ 任意边界的距离为 1 (1 个单位代表 δ), 如图 2 所示.

由于该算法是一个动态规划过程, 其路径动态修改, 因而每步规划出的局部路径对未来的行走只是一种引导 Rob 行走的趋势, 因此, 即使 g_{sub} 与 $View$ 间实际存在障碍物也不影响这一规划过程.

3.3 多蚂蚁协作局部路径规划算法

将两个蚂蚁家族的各 m 只蚂蚁分别放置在 $P_R(t)$ 和 g_{sub} , 对于两蚂蚁家族中的每一只蚂蚁, 以当前节点为中心, 按两族蚂蚁相向趋近原则和最近邻居选择策略选择并行走到下一可行节点. 为了达到一开始就能逐步趋近的目的, 两族蚂蚁分别采用定义 7 给出的趋近导引启发函数. 由于两族蚂蚁除了采用的启发函数不同、出发点不同外, 算法完全相同, 为了减少变量的下标, 以下以蚂蚁家族 1 的搜索算法为例进行说明, 并将表示蚂蚁家族 1 的所有下标省略.

考察机器人路径规划问题, 若机器人 Rob 在时刻 t_i 处于 g_R 位置, 它下一步要走的栅格范围必然是 g_R 邻域 BR_i 内的可行域中的可行点. 因此, 蚂蚁从当前节点 g_i 选择下一节点时, 只要在 $Wk_i(g_i)$ 内选择 g_j 即可. 算法步骤如下.

Step 1. 将 m 只蚂蚁放置在出发点 g_R , 并设置到禁忌表 $tabu_k$ 中 ($k = 1, 2, \dots, m$). 令初始信息素 $\tau_{ij}(0) = \tau_0$ (τ_0 为常数, 一般取 0.5 左右). 设置寻食代数计数器 $n = 0$, 最大代数为 MAX , 其中, 令 $m \leq 4$.

Step 2. $\forall k$, 以当前节点 $g_i \in FS$ 为中心, 按两族蚂蚁相向趋近原则和最近邻居选择策略选择并行走到下一节点 g_j , 且有 $g_j \in Wk_i(g_i)$, $g_j \notin tabu_k$, 节点选择算法如下.

Case 1. 若有 $g_j \in BR_R(g_R)$, 则 $\forall k$ 选择一个节点 $g_j^{k_i} \in Wk_R(g_R)$, 且令 $g_j^{k_i} \neq g_j^{k_j} \in Wk_R(g_R)$, $i, j = 1, 2, \dots, m$ 其中, 上标 k_i, k_j 表示不同蚂蚁选择的不同节点. 该约束是为了保证搜索多样性.

Case 2. 若 $g_i \notin BR_R(g_R)$, $\forall k$ 根据 (5) 或 (6) 选择下一个节点 $g_j \in Wk_i(g_i)$, $g_j \notin tabu_k$

$$j = \begin{cases} \arg \max \{ [\tau_{ij}(n)]^\beta \eta_j(g_i) \} & \text{若 } q \leq q_0 \\ S & \text{否则} \end{cases} \quad (5)$$

$$p_{ij}^k(n) = \frac{[\tau_{ij}(n)]^\beta \eta_j(g_i)}{\sum_{q \in |Z|} [\tau_{iq}(n)]^\beta [\eta_q(g_i)]}, \quad j \notin tabu_k \quad (6)$$

式中, $j \in C$, 蚂蚁 k 所选 g_j 的节点序号, 在此省略了上标 k ; S 由 (6) 式决定的随变量; q 随机数 ($0 < q \leq 1$); q_0 初始化时给定的阈值; $\eta_j(g_j)$ 由定义 7 给出的启发信息; β 在边 e_{ij} 上残留信息的重要程度; z 为蚂蚁 k 在当前节点邻域内的可行点集; $p_{ij}^k(n)$ 蚂蚁在 n 代寻食过程中, k 由节点 i 转移到节点 j 的概率, $i, j \in C$.

q 和 q_0 是为了防止出现停滞而设的随机搜索策略所需参数, 以增加搜索的多样性. 当 $q > q_0$ 时, 计算 $|Z|$ 个节点的转移概率 p_{ij}^k , 并根据赌轮盘规则选择节点 j . 需注意的是, 两族蚂蚁所用算法中, 蚂蚁所处的节点和所用的启发信息不同.

将 j 加入禁忌表 $tabu_k$.

Step3. 局部信息素更新

随着时间的推移, 以前留下的信息逐渐消逝, 用参数 $1 - \rho$ 表示信息消逝程度, 每一只蚂蚁选择完一个节点即走完一个边后, 按 (7) 式进行局部信息更新.

$$\begin{aligned} \tau_{ij}(n+1) &= (1 - \rho)\tau_{ij}(n) + \rho\Delta\tau_{ij}^k \\ \Delta\tau_{ij}^k &= \begin{cases} \frac{Q_1}{l_{jb}}, & \text{当蚂蚁 } k \text{ 走过边 } e_{ij} \\ 0, & \text{否则} \end{cases} \\ l_{jb} &= \sum_{l=1}^w d_l \end{aligned} \quad (7)$$

当 $\tau_{ij}(n+1) < \tau_{\min}$ 时, 令 $\tau_{ij}(n+1) = \tau_{\min}$. 式中, Q_1 为常数, $\tau_{\min}$ 是设定的最小值; d_l 是蚂蚁 k 已走过的边的边长, 按定义 4、5, $d_l = 1$ 或 $\sqrt{2}$, 可由 (2) 式计算. w 是蚂蚁 k 在本次寻食中已走过的边数; l_{jb} 是 k 在本次寻食中到当前时刻为止已走过的路程 (边的累加总长).

Step 4. $\forall k, k = 1, 2, \dots, m$, 选择完第 j 个节点后, 按定义 8 定义的条件, 检查两族蚂蚁中的所有蚂蚁是否已有蚂蚁相遇, 若有则转 Step 5; 否则, 返 Step 2 开始选择下一个节点, 直到有蚂蚁相遇或所有节点选择完毕.

Step 5. 当两族中的两只蚂蚁或多只蚂蚁满足定义 8 的相遇条件时, 将相遇蚂蚁所走通道连接, 并用 (3) 式计算其路程 L . 例如, $k_1 \in ant_1$, 所走的通道为 $Path(P_R, P_j)$, 所走的节点集合为 $tabu_{k_1} = \{1, 11, 12, 22, 23, 33, 43, 44\}$, $k_2 \in ant_2$ 所走的通道为 $Path(P_s, P_b)$, $P_s \sim g_{sub}$, $|b - j| \leq 1$ 或 $|b - j| \leq N_x$, $b, j \in C$, 所走的节点集合为 $tabu_{k_2} = \{88, 78, 68, 58, 57, 56, 46, 45\}$, 节点 44 和 45 满足定义 8 相遇条件, 则将两通道连接为 $Path(P_R, P_s)$, 通道的节点集合则为: $tabu_{k_{12}} = \{1, 11, 12, 22, 23, 33, 43, 44, 45, 46, 56, 57, 58, 68, 78, 88\}$, 所有相遇蚂蚁所走通道连接完成后, 用 (3) 式计算各连接路程 L_k , 并保存最短路程 $L_{k \min}$ ($L_{k \min} = \min L_k$).

将本次觅食得到的 $L_{k \min}$ 与已得到的历史最优长度 L_d 比较, 若有 $L_{k \min} < L_d$ 则用 $L_{k \min}$ 替换 L_d , 并记忆最佳通道的节点集合.

Step 6. 全局信息素更新

$$\begin{aligned} \tau_{ij}^{new} &= (1 - \alpha)\tau_{ij}^{old} + \alpha\Delta\tau_{ij} \\ \Delta\tau_{ij} &= \sum_{k=1}^m \Delta\tau_{ij}^k \\ \Delta\tau_{ij}^k &= \begin{cases} \frac{Q_2}{L_{k \min}}, & \text{若 } ij \in global - best - tour \\ 0, & \text{否则} \end{cases} \end{aligned} \quad (8)$$

本次觅食相遇并完成通道连接后, 将本次觅食最短通道上的信息素按 (8) 式调整: 式中, Q_2 常数; α 全局信息素挥发系数; $L_{k \min}$ 本次寻食最佳通道的路程; $ij \in global - best - tour$ 表示蚂蚁 k 所走的边 ij 属于最佳通道;

Step 7. 令寻食代数 n 加 1, 若不等于 MAX , 则清空禁忌表, 重复上述寻食过程, 直到 $n = MAX$ 为止. 最终记忆的最佳通道即为规划出的最优路径.

4 仿真实验

为了在不影响总体结果的前提下使仿真实验方便, 假定任意两相邻栅格的距离为 1; 并设机器人的视野域为 9×9 的方形. 为了实验算法的效果, 设计了图 1 所示的复杂地形, 并将其坐标设置在一个数据文件中, 随着 Rob 的行走, Rob 取出其视野域内的障碍格的坐标数据, 以模拟探测到的障碍物, 其中, 黑色为障碍格. 图 3 示出了用本文算法得到的从左上角 2 号栅格起到右下角 379 号格为终点的一条路径, 很显然, 这条路径不仅安全避碰, 且基本上全局最优. 这是目前许多算法所做不到的.

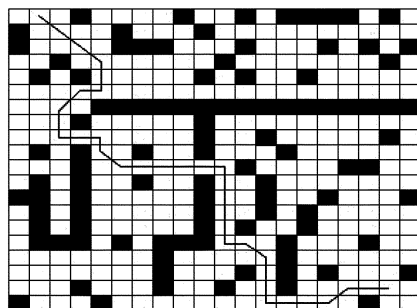
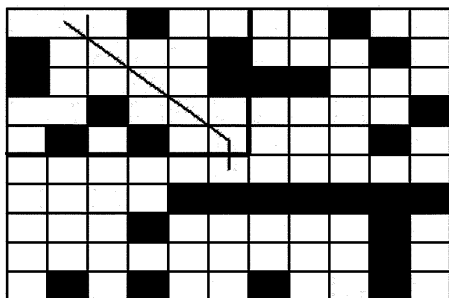


图 3 本算法得到的优化路径

Fig. 3 Global optimal path is from our algorithm

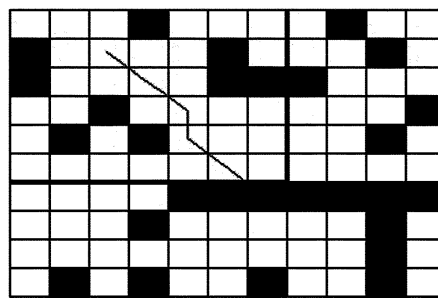
为了进一步分析本文算法能得到全局优化路径的原理, 图 4 给出了实验过程中的分解子过程, 其中, 粗线内表示 Rob 视野域范围. 图 4-A 示出了 Rob 在起点时的部分视野域及规划出的局部优化路径; 图 B、C 和 D 分别示出了 Rob 分别从起点前进一步、两步、六步后再次规划出的局部优化路径. 根据算法假定, 子目标点映射在视野域边界外, 且假定子目标点与视野域任意边界间为无障碍物, 因此, 子目标可以映射到视野域外的障碍格内, 如图 B 所示, 路径也可以穿越视野域外的障碍格如图 C 所示 (这些障碍格机器人还看不到, 把这些看不到的格均作自由格, 且与 View 边界距离为 1).

从图 4 可以看出, Rob 仅沿每次规划的局部最优路径走一步, 但它构成了全局的一种引导趋势, 从而使得不仅能避碰, 且所走路径全局最优或较优.



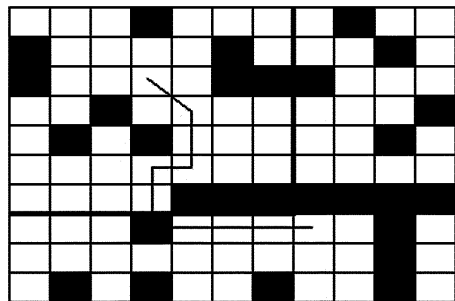
(a) 第一个局部路径

(a) First local path



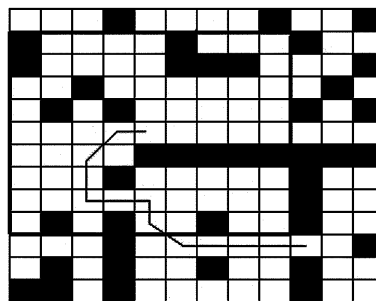
(b) 第二个局部路径

(b) Second local path



(c) 第三个局部路径

(c) Third local path



(d) 第六个局部路径

(d) Sixth local path

图 4 局部路径分解图

Fig. 4 Local path resolving drawing

5 结语

全局环境未知时的机器人路径规划的难点在于很难保证机器人所走路径全局最优或较优, 甚至容易引起死锁和震荡等. 本文将最终目标点映射到机器人视野域外部作为局部子目标, 并将该子目标与视野域边界视为无障碍物且其距离为 1, 然后用两组各 m 只蚂蚁分别置于机器人和子目标位置相向并行搜索, 实时规划出一条可能是曲折复杂的局部路径. 机器人每前进一步, 都要进行场景探测, 并重新进行局部路径的规划, 因而其前进路径动态修改. 由于在每个视野域内的局部路径的引导作用加上这些局部路径都是最优路径, 因此不仅使算法能免于死锁、震荡且能使机器人所走路径全局最优或较优.

References

- 1 Yi X, He Y, Guan X. Cooperative location model under the nearest neighbor criterion position location and navigation. In: Proceedings of Symposium, PLANS 2004, IEEE, 2004. 658~661
- 2 Zhang C Gg, Xi Y G. Mobile robot path planning based on rolling windows in the unknown environment. *Science in China (Series E)*, 2001, **31**(1): 51~58
- 3 Bruce J, Veloso M. Real-time randomized path planning for robot navigation. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and System, 2002. **3**: 2383~2388
- 4 Colorni A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies. In: Proceeding of European Conference of Artificial Life ECAL91, F.Varela and P.Bourgine (Eds), Paris, France: Elsevier Publishing, 1991. 134~144
- 5 Dorigo M, Di Caro G. Ant colony optimization: a new meta-heuristic. In: Proceedings of the 1999 Congress on Evolutionary Computation, Washington, DC, USA: IEEE Press, 1999. **2**: 1474~1477
- 6 Fan X Y, Luo X, Yi S, Yang S, Zhang H. Optimal path planning for mobile robots based on intensified ant colony optimization algorithm. In: Proceedings of 2003 IEEE on International Conference on Robotics, Intelligent Systems and Signal Processing, 2003. **1**: 131~136
- 7 Ding Y Y, He Y, Jiang J P. Multi-robot cooperation method based on the ant algorithm. In: Proceedings of the 2003 IEEE Swarm Intelligence Symposium SIS'03, 2003. 24-26 2003. 14~18

朱庆保 南京师范大学计算机系教授. 研究领域为人工智能与智能控制.

(ZHU Qing-Bao Professor in the Department of Computer Science at Nanjing Normal University. His research interests include artificial intelligence and intelligent control.)