

doi: 10.7690/bgzdh.2013.12.026

盾形洄游算法

何洋, 李淑琴

(北京信息科技大学计算机学院软件工程系, 北京 100020)

摘要: 为解决机器鱼因高速带球失误造成的绕球和逆向顶球行为, 对盾形洄游算法进行研究。该算法通过栅格建模, 在球周围一定范围内进行布线确定栅格, 使得机器鱼在超出目标点后快速寻找最短洄游路径, 并找到折返初始位置的最佳方式。实验结果证明: 盾形洄游算法能有效地改善带球失误, 极大地提高了机器鱼在比赛中的稳定性。

关键词: 2D 仿真; 机器鱼; 盾形洄游算法; 带球策略

中图分类号: G642 **文献标志码:** A

Peltate Migration Algorithm

He Yang, Li Shuqin

(Department of Software Engineering, College of Computer, Beijing Information Science & Technology University, Beijing 100020, China)

Abstract: To solve the robotic fish due to high speed ball error around the ball and reverse ball behavior, the peltate migration algorithm is researched. Through the grid model, the algorithm carried out within a certain scope around the ball determine grid wiring, make the fish after beyond the target quickly find migratory path, shortest turn-back initial position and find a best way. Experimental results show that peltate migration algorithm can effectively improve the ball error, greatly improved the stability of robotic fish in game.

Key words: 2D simulation; robotic fish; peltate migration algorithm; dribbling strategies

0 引言

机器鱼水球比赛^[1]主要涉及 2 类核心策略: 一类为路径规划; 一类为机器鱼顶球算法^[2]。其中, 顶球算法一直是策略实施的核心。文献[2]实现了基于动作决策的顶球算法, 文献[3]中作者通过对鱼位姿的控制提升了顶球算法的稳定性。该顶球算法主要是首先通过机器鱼预计的行进目标点和球心点, 来确定机器鱼顶球的球上目标点, 再由球上目标点和鱼身体点来确定鱼顶球的行进方向, 当两向量积为负时, 则会引起鱼的归位行为, 常使得鱼逆向顶球或产生连续绕球行为, 即洄游现象。另外, 机器鱼带球过快或指令信息在传输中延迟甚至丢包时, 鱼会失去稳定性而导致带球产生偏移, 难以快速返回失误点重新击球, 甚至造成反复绕球的隐患。

基于此, 笔者主要对带球算法进行研究, 通过盾形洄游算法实现机器鱼带球失误后快速顺利归位, 从而提高带球时速度和带球稳定性, 避免带球失误后长时调整, 解决当目标球行进向量和顶球向量积为负时引起的逆向顶球和绕球行为, 产生最适归位路径算法。

1 盾形洄游策略

1.1 洄游难题

水球斯诺克比赛由单支队伍参与, 每支队伍拥有 1 条仿真机器鱼, 用到 10 个仿真水球, 需要编写策略使鱼带球进球洞。与现实斯诺克比赛规则类似, 首先需要进一颗红球, 之后进一颗彩球, 来回交替直到红球全部入洞后, 按彩球分数大小自低向高依次进球。其中红球 1 分, 黄绿棕粉球依次为 2, 3, 4, 5 分, 进球则加入相应分数, 进错球则扣去进错的球色分数。每支队伍拥有 10 min 时间, 时间结束时按队伍已进球得分进行成绩排名。

在机器鱼水球比赛中, 最常见的一个难题就是当鱼偏离原本的目标击球点后, 从球侧冲出一段距离, 由于机器鱼不能直接后退, 也不能原地旋转, 从而使得偏离击球点的鱼需要花费大量时间重新回到初始目标击球点^[4], 而在这一过程中, 稍有不慎则会造成如图 1 所示的绕球行为, 一旦出现绕球行为, 则会花费更长的时间进行调整, 最终回到初始击球位置。

收稿日期: 2013-06-30; 修回日期: 2013-07-12

基金项目: 市教委“PXM2013_014224_000058 教育教学-本科生科研训练项目”; 北京信息科技大学 2013 年大学生科技创新计划项目

作者简介: 何洋(1991—), 男, 安徽人, 本科在读, 从事软件工程研究。

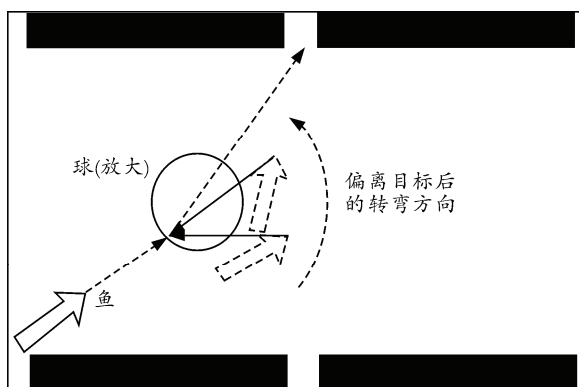


图 1 偏离击球点产生的绕球行为示意图

机器鱼水球比赛中的另一个难题如图 2 所示，当击球方向与鱼寻找球上目标击球点的方向角度很小时，为了最快到达目标击球点，极易产生逆向顶球行为，此行为风险极大，不仅需要花费很长时间调整，同时也会造成球偏离球洞更远，甚至于整个比赛结束仍未到达击球点位置。

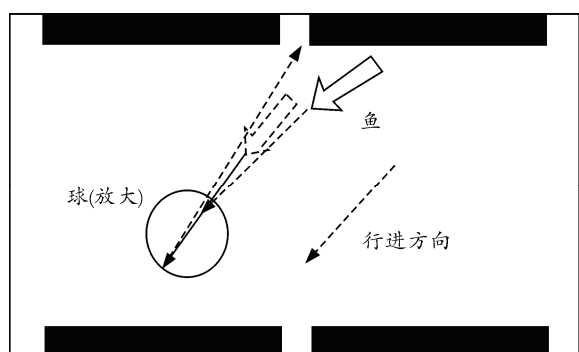


图 2 追寻击球点产生的逆向顶球示意图

1.2 洄游难题的一般解决思路

通常情况下，处理洄游难题的算法都是当鱼身偏离目标击球点一定距离后，在球后一定距离位置重新定位新的目标击球点^[5]，使鱼绕一个较远路径转弯，从而较安全地回归原始击球点。这样的算法虽然一定程度上保证了鱼击球避免绕球和逆向顶球的稳定性，但仍然非常耗时。

1.3 盾形洄游策略

盾形洄游策略的思想是在球周围一定范围内进行布线确定栅格，使得机器鱼身中点处于每个栅格区域内时都能自由配置角速度、线速度及以布线交点为目标点的下一行进点。这些行进点连接后在球身四周构成一个盾形区域，因此命名盾形洄游算法。

由于在球身四周始终动态分布着多区域栅格，每个区域映射一个行进目标点，因此鱼带球时一旦偏移目标点后能根据所处区域立即调整相应速度和

行进目标点，在这个洄游过程中不断更换区域，同时也不断映射新的行进目标点和速度，使鱼最终快速稳定地返回原带球点，有效避免高速带球较大偏移目标点后难以快速回退的难题，更能有效避免鱼绕球隐患。

2 盾形洄游算法的实现

2.1 盾形栅格建模的基本布线规则

盾形洄游算法将球周围区域通过布线划分成 15 个区域，使每个区域有独立的控制逻辑，每个区域可映射布线相交的目标点。

如图 3 所示，其中以球心点到球洞目标点为基线 C_3 ，以球半径 r 为布线间垂直距离，分别建立 C 组 C_1 、 C_2 、 C_4 、 C_5 直线，同时作直线 A_2 经过球心点与 C_3 相交，以球半径 r 为 A 组布线间垂直距离建立 A_1 、 A_3 直线，并在 A_1 下方以相同规则扩展建立 2 条虚拟直线。

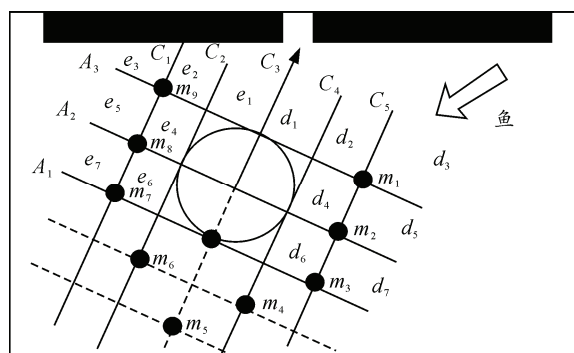


图 3 盾形洄游算法栅格模型示意图(球放大)

2.2 栅格区域及区域映射目标点的确立

取栅格模型中 10 个交点 $m_1 \sim m_9$ 以及球上目标点，如图 3 所示。作为区域映射目标点，同时将栅格区域划分成 15 块，分别以 $d_1 \sim d_7$ ， $e_1 \sim e_7$ 以及 A_1 直线下部全部区域标识。

区域与目标点映射关系为式 (1)：

$$\begin{aligned}
 & d_1 - m_1, \quad e_1 - m_9, \\
 & d_2 - m_2, \quad e_2 - m_8, \\
 & d_3 - m_4, \quad e_3 - m_6, \\
 & d_4 - m_3, \quad e_4 - m_7, \\
 & d_5 - m_5, \quad e_5 - m_5, \\
 & d_6 - m_5, \quad e_6 - m_5, \\
 & d_7 - m_5, \quad e_7 - m_5.
 \end{aligned} \tag{1}$$

2.3 盾形洄游算法布线与结点函数的确立

设球心坐标为 x_1, y_1 ；目标点坐标 x_2, y_2 ，球

半径 $r=58\text{ mm}$, 建立 C 组与 A 组直线函数, 可求得:

$$C\text{ 组直线函数斜率 } k_1 = \frac{y_2 - y_1}{x_2 - x_1};$$

$$A\text{ 组直线函数斜率 } k_2 = -\frac{x_2 - x_1}{y_2 - y_1};$$

$$C_3\text{ 基线: } \left(\frac{y_2 - y_1}{x_2 - x_1}\right)x - y - \left(\frac{y_2 - y_1}{x_2 - x_1}\right)x_1 + y_1 = 0;$$

$$A_2\text{ 基线: } -\left(\frac{x_2 - x_1}{y_2 - y_1}\right)x - y + y_1 - \left(\frac{x_2 - x_1}{y_2 - y_1}\right)x_1 = 0;$$

$$C\text{ 组垂直分量: } b_1 = \sqrt{\frac{r^2}{k_1^2} + r^2} \times k_1 = \sqrt{r^2 + k_1^2 r^2};$$

$$A\text{ 组垂直分量: } b_2 = \sqrt{r^2 + k_2^2 r^2};$$

易求得

$$C_1: k_1x - y + 2b_1 - k_1x_1 + y_1 = 0$$

$$C_2: k_1x - y + b_1 - k_1x_1 + y_1 = 0$$

$$C_4: k_1x - y - b_1 - k_1x_1 + y_1 = 0$$

$$C_5: k_1x - y - 2b_1 - k_1x_1 + y_1 = 0$$

$$A_1: k_2x - y - b_2 - k_2x_1 + y_1 = 0$$

$$A_3: k_2x - y + b_2 - k_2x_1 + y_1 = 0$$

则可由上述直线方程求得 $m_1 \sim m_9$ 目标点, m_1 如式 (2) 为:

$$\left(\frac{2b_1 + b_2 + k_1x_1 - k_2x_1}{k_1 - k_2}, \left(\frac{2b_1 + b_2 + k_1x_1 - k_2x_1}{k_1 - k_2}\right) \times k_1 - 2b_1 - k_1x_1 + y_1\right) \quad (2)$$

通过最终得到的 m 坐标, 获得其在盾形洄游算法中的映射区域 d , 当鱼在不同区域时, 拥有不同的 m 目标点, 这样便可找到最短路径, 并以最快速度返回原始击球位置。

机器鱼在每个栅格中依照区域 d 到行进目标点 m 的映射关系, 使在每个区域 d 中的鱼被赋予新的目标行进点 m 及线速度角速度, 而在鱼驶向新的目标点 m 的过程中, 更换了区域 d , 则又会确立新的行进目标点及速度, 由此最终返回原始行进目标点, 即为洄游结束。

3 实验及分析

笔者以 2D 仿真机器鱼平台为基础, 通过水球斯诺克项目进行实验。故意提高机器鱼的顶球速度, 造成其带球失误而偏移原始击球位置, 来测试其洄游时间和效果。实验结果如表 1 和图 4 所示。

表 1 斯诺克比赛中是否采用洄游算法的得分实验记录

策略次数	进球剩余时刻		策略次数	进球剩余时刻	
	采用	不采用		采用	不采用
1	9:10	8:56	8	4:34	1:08
2	8:28	8:01	9	3:12	—
3	7:46	7:00	10	2:28	—
4	7:03	5:30	11	1:29	—
5	6:13	4:32	12	0:37	—
6	6:20	3:31	13	—	—
7	5:25	2:24			

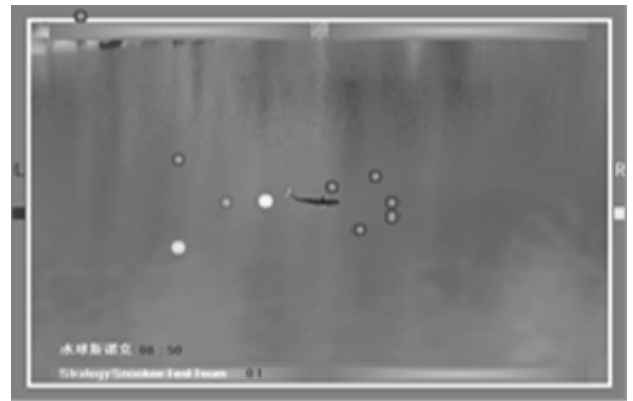


图 4 偏移击球点后盾形洄游示意图

根据实验数据可见, 采用盾形洄游算法进行带球控制, 能提高带球的稳定性, 减少丢球绕球行为的发生, 同时栅格的运用提高了带球的精细粒度, 提高了带球进球的准确率, 也就大幅缩短了进球时间。这是因为盾形洄游算法一方面找出了回归原始击球位置的最短路径与最短时间, 另一方面避免了绕球行为和逆向顶球行为, 使得机器鱼带球控制更加稳健, 降低了带球失误造成的时间损耗。

4 结语

实验结果表明: 盾形洄游算法能极大地提高机器鱼归位的速度, 找寻最短归位路径, 有效地改善由于机器鱼不能直接后退和原地转弯造成的带球失误, 解决因高速带球失误造成的绕球行为和逆向顶球行为, 极大地提高了机器鱼在比赛中的稳定性。

参考文献:

[1] 机器人水球比赛项目推介书[M]. 北京: 北京大学工学院, 2012.
 [2] 陶金, 孔峰, 谢广明. 基于动作决策的机器鱼顶球算法[J]. 兵工自动化, 2010, 29(11): 70-73.
 [3] 刘傲, 蔡勇, 谢广明. 一种机器鱼的位姿镇定控制算法[J]. 兵工自动化, 2012, 31(11): 46-50.
 [4] 彭非, 李卫京. 基于 PID 算法的水中机器鱼方向档位控制方法[J]. 兵工自动化, 2012, 31(11): 80-82.
 [5] 安永跃, 李淑琴, 龙海楠, 等. 机器鱼仿真水球斯诺克比赛策略[J]. 兵工自动化, 2012, 31(11): 51-55.