

doi: 10.3969/j.issn.1006-1576.2011.12.022

基于蚁群算法的仿真机器鱼动作决策策略

张纯, 邓彦松

(西南民族大学电气信息工程学院, 成都 610041)

摘要: 针对仿真机器鱼非对抗赛和对抗赛情况, 为使求解结果在既不依赖初始路线的选择, 也不需要外界的稳定干预的情况下, 实现鱼快速、准确的调整, 分别提出 2 种基于蚁群算法的动作决策策略。基于蚁群算法的分支界限法, 判断机器鱼关键物理量所在分支, 自主确定当前时刻的鱼的速度和角速度档位的最优组合; 而基于蚁群算法的动态规划法, 在每个周期内, 根据机器鱼反馈回来的动态变量及时进行自主调整。以上 2 种方法经 2D 仿真平台验证结果表明: 机器鱼可根据该策略调整路径, 实现速度和方向的组合优化, 以最短的时间和距离找到目标点。这说明基于蚁群算法的 2 种动作决策策略具有很强的适应能力, 满足仿真机器鱼对于动作决策的要求。

关键词: 仿真机器鱼; 蚁群算法; 动作决策; 分支界限; 动态规划

中图分类号: TP301.6 **文献标志码:** A

Action Decision Strategy of Simulation Robot Fish Based on Ant Colony Algorithm

Zhang Chun, Deng Yansong

(College of Electronic & Information Engineering, Southwest University for Nationalities, Chengdu 610041, China)

Abstract: Aiming at the non-match and match situations of simulation robot fish, in order to solve the result is neither dependent on the choice of the initial line does not need outside intervention-specific circumstances, to achieve the fish fast, accurate adjustment, this paper proposed two kinds of ant colony algorithms action decision strategy. Ant colony algorithm based on branch bound method judges the key physical machine where the fish branch, self-determined speed and angular velocity in the moment and optimal combined speed of the fish with angular velocity of the fish; In each cycle, the ant colony algorithm based on dynamic programming make self adjustment according to the dynamics of the robot fish immediate feedback. Examples of the above two methods used by the 2D simulation platform validation results showed, robot fish can be adjusted based on the policy path, to achieve optimal combination of speed and direction, the shortest time and distance to find the target point. This shows that based on ant colony algorithm's two kinds of action decision strategy has a strong ability to adapt effectively to meet the simulation of robot fish for action decision-making.

Keywords: simulation robot fish; ant colony algorithm; action decision-making; branch boundaries; dynamic programming

0 引言

动作决策策略要求机器鱼根据给予的指令及环境信息自主地做出动作变量的决断。一个精确可行的动作决策对实现准确可靠的路径规划是必不可少的。岳富占提出基于改进蚁群算法的星球探测机器人路径规划技术^[1], 针对机器人的动作控制对蚁群算法中蚂蚁的个体行为进行改进, 并在蚁群算法规划的基础上提出紧绳算法, 对蚁群算法的结果进行处理, 最后得到最优路径; 王鹏飞提出一种改进的蚁群算法^[2]用于路径规划, 其中的动作控制部分使得系统在遇到可能引起“任务死锁”的任务时, 快速跳出该任务, 并顺利迅速地完成任务中其它可完成的任务。以上 2 种算法应用于机器人均取得了满意的结果, 但由于水下环境的复杂性、流动性和不确定性, 使机器鱼和水球很难稳定在水中的某一

点, 给机器鱼准确到达目标点参与进攻带来很大干扰, 因此这些算法不完全适合机器鱼, 需要进一步改进^[3]。柴仲明提出基于遗传算法的机器鱼路径规划方法^[4], 采用栅格法对机器鱼工作空间进行划分, 充分考虑到了水下环境对机器鱼造成的影响。只是基于遗传算法的规划空间栅格法建模还存在缺陷, 若栅格划分过粗则精度较低; 若栅格划分过细则数据量又过大, 造成动作决策的局限性^[5]。因此, 笔者提出了基于蚁群算法的机器鱼动作决策策略。

1 蚁群算法与机器鱼动作决策

蚁群算法是一种用来在图中寻找优化路径的概率型算法。它由 M.Dorigo 等人受到真实蚁群集体行为研究成果的启发而提出来^[6]。在蚁群世界里, 每只蚂蚁根据其观察到的范围即一个方格世界内的信息, 利用移动规则、避障规则、觅食规则等几条简

收稿日期: 2011-08-26; 修回日期: 2011-09-19

项目资金: 西南民族大学中央高校基本科研业务费专项资金资助(10NZYZJ05)

作者简介: 张纯(1988—), 女, 河北人, 满族, 本科, 从事教育软件开发方向研究。

通信作者: 邓彦松(1976—), 男, 博士, 讲师, 从事信号处理和非线性控制研究。

单的规则进行决策，从而使其移动路径不断趋向最优路径。蚁群算法在解决组合优化问题的求解方面表现了突出的适用性^[7]。该方法具有正反馈、分布式计算和富于建设性的贪婪启发式搜索的特点，是一种求解组合最优化问题的新型通用启发式方法。

在 2D 仿真机器鱼竞赛中，一般从路径规划、动作控制、协作分工、误差分析 4 个模块着手。路径规划就是按照某一性能指标搜索一条从起始状态到目标状态的最优或近似最优的无碰壁路径^[8]。而动作控制是控制机器鱼在水中按照预定轨迹运动的关键模块，它关系到比赛中决策能否得到准确的贯彻和实施，因此动作水平的高低直接影响到机器人完成任务的质量^[9]。

借鉴以上思想，根据机器鱼比赛中的非对抗性项目和对抗性项目的不同性质和特点提出 2 种动作决策策略——分支界限法和动态规划法。其思想是：首先确定每个机器鱼的动作执行范围，即蚁群算法中的方格世界；其次，建立数据储存区并储存上一周期鱼的动作决策，即“信息素”；第三，根据当前方格世界内返回的结果，在方格世界中加入目标点的对应位置，确定机器鱼所在位置和目标点的关系，利用蚁群算法，把路径规划问题转化为组合优化问题，根据决策规则进行第一次决策；最后，根据返回结果结合信息素进行二次决策，并用本次决策数据替换上周信息素数据，提供给下周决策调用，直到机器鱼位置与目标位置满足阈值范围。

2 基于蚁群算法的分支界限法

2.1 算法的基本思想

2D 仿真机器鱼的比赛场地^[10]实际上是一个二维坐标系，并给出相应的 Math 库来确定各个对象之间的几何参数。基于蚁群算法的分支界限法做出的策略首先以比赛场地的中心点作为圆的圆心，根据各种比赛的具体情况以不同的半径做出多个同心圆，一般情况下，越靠近中心点同心圆越密集，越远离中心点，同心圆越稀疏。这样把比赛场地划分为不同的方格世界，见图 1。

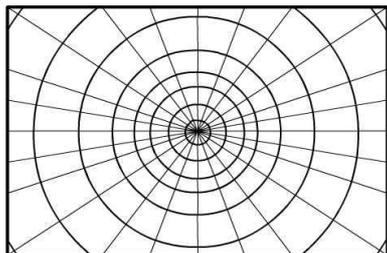


图 1 分支图

根据仿真机器鱼的中心点和目标点的相对位置，包括夹角度数和距离，判断鱼中心点到场地中心点的距离以及鱼中心点与目标点的夹角值所在分支，从而自主确定当前时刻的鱼的速度和角速度档位的最优组合。

2.2 算法的实现步骤

1) 根据规划的最优路径，选定机器鱼要到达的目标点 $D(X_2, Z_2)$ 。

2) 确定机器鱼中心点 $F(X_1, Z_1)$ 和目标点 $D(X_2, Z_2)$ 的相对位置，求出偏航角（机器鱼的前进方向与 F 点到 D 点连线的夹角为偏航角 θ ， θ_2 为负角） $\theta = \theta_1 - \theta_2$ ，鱼中心点到场地中心点的距离 R ，如图 2。

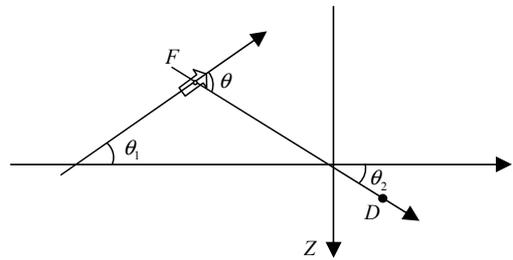


图 2 分支界限法分析图

3) 根据 R 和夹角判断所在分支。2D 仿真机器鱼控制的实现主要依赖于 2 个参数： $Vcode$ 、 $Tcode$ ，即机器鱼速度和角速度的档位。在仿真平台中通过 $VCodeAndVelocityTable$ 数组决策结果的速度档位，它的取值范围 $\{0, 1, \dots, 15\}$ 。用 0~15 这 16 个元素依次确定 0~15 档的速度值，mm/s；其中 0 最慢，15 最快。具体表示为：`public static float[] VCodeAndVelocityTable = new float[16] { 0, 0, 0.091f, 67, 112, 112, 112, 154, 175, 227, 273, 291, 298, 294, 307, 317 }` 同样，通过 $TCodeAndAngularVelocityTable$ 数组来决策结果的转弯档位，取值范围 $\{0, 1, \dots, 15\}$ 。用 0~15 这 16 个元素依次确定 0~15 档的角速度值，rad/s；其中 0 左转最急，7 直游，15 右转最急。具体表示为 `public static float[] TCodeAndAngularVelocityTable = new float[16] { -3.000f, -0.788f, -0.634f, -0.535f, -0.400f, -0.356f, -0.088f, 0, 0.088f, 0.134f, 0.289f, 0.356f, 0.400f, 0.535f, 0.634f, 3.000f }`。因为每个档位之间的 $Vcode$ 和 $Tcode$ 变化并不均匀，所以影响到所划分的分支范围以及 $Vcode$ 和 $Tcode$ 的增减幅度。下面给出在实际测试中经过多次重复测试和大量统计分析在 $R=400$ mm 得到的一组有效数据，见表 1。

表 1 最佳速度与角速度组合表

分支范围	Toode	Voode	分支范围	Toode	Voode
(-360, -355)	7	7	(-5,5)	7	15
(-355, -270)	10	6	(5,20)	9	14
(-270, -150)	3	5	(20,30)	12	12
(-150, -120)	2	3	(30,50)	13	11
(-120, -90)	4	5	(50,70)	4	10
(-90, -70)	1	8	(70,90)	13	8
(-70, -60)	0	9	(90,120)	4	6
(-60, -40)	2	10	(120,270)	12	3
(-40, -20)	5	12	(270,355)	3	4
(-20, -5)	5	13	(355,360)	7	7

4) 根据表格 1 进行第一次决策, 记录在数组 aa 中。

5) 对信息素进行处理。在实际处理中, 信息素是一组二维数组 bb, 记录上周期鱼的速度和角速度, 由于相邻 2 个周期间隔很短, 可看作匀变速运动, 由积分得出鱼在状态转换过程中所经过的距离 L 和角度 θ 。将 L 和 θ 的影响加入到表 1 中, 进行二次决策。

6) 执行速度 Vcode 和角速度 Tcode 的最优组合。由于信息素的生存周期:

$$\tau_{ij}(t-1) = (1-\sigma) \times \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (1)$$

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad (2)$$

式 (1) 中, σ 表示信息素蒸发系数, 则 $1-\sigma$ 表示信息素残留因子, σ 的取值范围是 $[0,1)$, 实际操作中取 $\sigma=0$, 即信息素一个周期后就会全部挥发掉。同时, 每个周期会有新的信息素 T 产生, 即鱼的二次决策产生的速度和角速度值。

3 基于蚁群算法的动态规划法

3.1 算法的基本思想

动态规划法是把机器鱼将要寻找下一个目标点的动作看作是在上一次决策的基础上做出下一次自组织决策的方法。首先要确定一个较小的周期, 在每个周期内根据机器鱼反馈回来的动态变量即时进行自主调整。

3.2 算法的实现步骤

1) 为了实现实时路径规划, 就必须每隔一个很短的时间周期为机器人规划下一时刻的路径点。首先确定一个较小的周期, 定为 0.1 s 。一般情况下, 鱼始终都是处在运动的状态中, 必然经历速度/角速度档位由 i 切换到 j 的一个过渡过程, 这个过程所

需的周期数具体表示如下:

```
Public static int[,] VCodeSwitchCycleCntTable = new
int[16, 16]{
{0,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10},
{10,0,10,10,10,10,10,10,10,10,10,10,10,10,10,10},
{10,10,0,10,10,10,10,10,10,10,10,10,10,10,10,10},
{10,10,10,0,10,10,10,10,10,10,10,10,10,10,10,10},
{10,10,10,10,0,10,10,10,10,10,10,10,10,10,10,10},
{10,10,10,10,10,0,10,10,10,10,10,10,10,10,10,10},
{10,10,10,10,10,10,0,10,10,10,10,10,10,10,10,10},
{10,10,10,10,10,10,10,0,10,10,10,10,10,10,10,10},
{10,10,10,10,10,10,10,10,0,10,10,10,10,10,10,10},
{10,10,10,10,10,10,10,10,10,0,10,10,10,10,10,10},
{10,10,10,10,10,10,10,10,10,10,0,10,10,10,10,10},
{10,10,10,10,10,10,10,10,10,10,10,0,10,10,10,10},
{10,10,10,10,10,10,10,10,10,10,10,10,0,10,10,10},
{10,10,10,10,10,10,10,10,10,10,10,10,10,0,10,10},
{10,10,10,10,10,10,10,10,10,10,10,10,10,10,0,10},
}
```

根据实际情况的不同, 周期也会做出相应的调整。

2) 确定机器鱼中心点 $F(X_1, Z_1)$ 、目标点 (X_2, Z_2) 以及机器鱼的速度方向 θ_1 。

3) 计算鱼和目标点之间的距离 L 、鱼与目标点的夹角 $\theta = \theta_1 - \theta_2$ 。

4) 根据实际情况设定合适的速度档 Vcode, 得到速度 V 。

5) 对距离 L 和角度进行判断, 返回同时过当前点 F 与目标点 D 的圆的半径 $R=L/2\sin\theta$ (见图 3), 且当前鱼的速度方向是该圆的切线, 切点为当前点 F (仅限于二维平面中)。

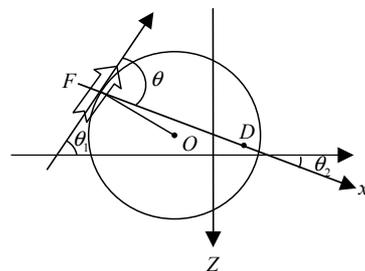


图 3 动态规划法分析图

6) 计算角速度 $\omega = 2V\sin\theta/L$ 。若速度为正则半径为正, 反之亦然。至此, 第一次决策完成, 将所得数据记录在数组 T 中。图 4 为所述方法求半径的流程图。

7) 同分支界限法, 带入信息素数据, 进行第二次决策并执行, 同时更新信息素信息。

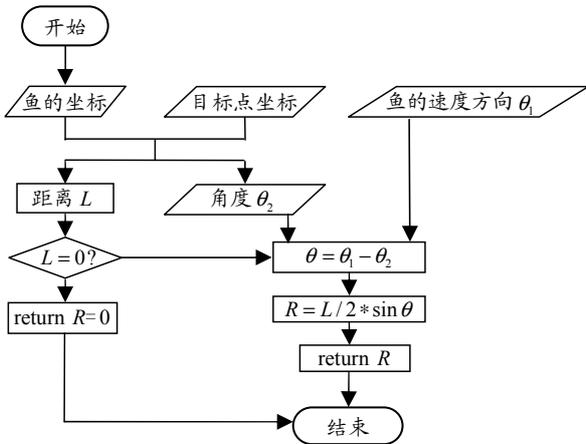


图 4 动态规划法求半径流程图

4 仿真结果及分析

4.1 分支界限法仿真结果分析

分支界限算法不是盲目的穷举搜索, 而是在搜索过程中通过限界, 可以中途停止对某些不可能得到最优解的子空间进一步搜索(类似于人工智能中的剪枝), 因此效率更高。它不仅能很好地实现仿真机器鱼从所在位置到目标点的自主决策, 而且能够精确地完成边界位置的寻点动作。作为静态决策策略方法的分支界限算法适应能力很强, 可在分布式 CPU 计算机结构下比较准确地执行决策, 受分布式条件下服务器和客户端的通信延时影响非常小。这个决策策略比较适合在避障、协作带球等非对抗性项目中实施, 还可以通过不断地细化夹角范围来提高机器鱼的灵活度。仿真结果见图 5、图 6。

从图 5、图 6 可以看出: 相对于未采取分支界限法的策略, 本策略能快速稳定地使机器鱼到达预定位置, 而不会出现“S”型的路线。

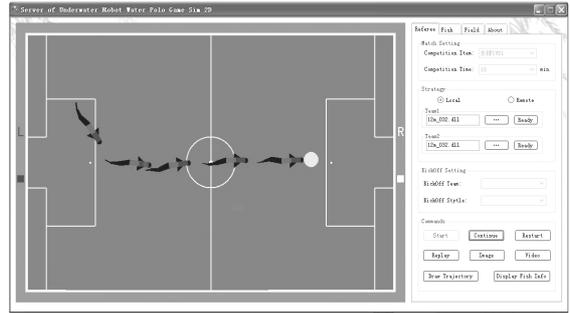


图 5 分支界限法仿真结果图

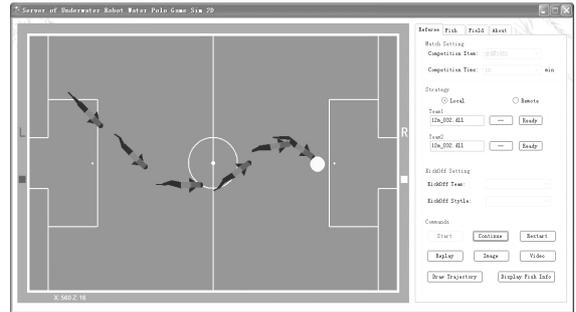


图 6 未采用分支界限法仿真结果图

4.2 动态规划法仿真结果分析

动态规划法是一种根据仿真平台中各个对象上一时刻各个动态变量的数据即时做出决策的动态自组织决策策略。它执行周期短, 每 0.1 s 进行一次调整, 可以看成机器鱼基本处在动态的变化之中, 而且调用方便, 对抗性较强, 需要辅助的速度策略进行配合, 在水球 1vs1、抢球大作战等对抗性项目中有很强的优势。应指出的是在决策中, 速度档位赋值的阈值需要根据实际情况来操作。

表 2 是按鱼的中心点和目标点的夹角和距离的不同情况, 对采用动态规划法和未采用动态规划法进行比较得出的结果, 每个项目都是 20 次测试后的平均结果。通过比较表明, 在相同的角度或距离情况下, 采用动态规划法用时较短、偏移角度较小、所走路程较少。

表 2 动态规划与非动态规划的比较

物理量	采用动态规划	未采用动态规划	物理量	采用动态规划	未采用动态规划	物理量	采用动态规划	未采用动态规划
夹角/(°)	45	45	夹角/(°)	45	45	夹角/(°)	90	90
距离/mm	1 000	1 000	距离/mm	500	500	距离/mm	1 000	1 000
用时/s	7.32	7.41	用时/s	3.35	4.23	用时/s	8.92	9.13
偏移角度/(°)	18	24	偏移角度/(°)	21	46	偏移角度/(°)	29	78
路程/mm	1 123	1 201	路程/mm	512	554	路程/mm	1 304	1 345

5 总结

基于蚁群算法的 2 种动作决策策略具有很强的适应能力, 有效地增加了系统熵, 能满足仿真

机器鱼对于动作决策的要求。仿真结果表明: 机器鱼能够快速到达预定点, 而且具有很高的稳定性, 说明以上 2 种方法是可靠、可行的。