# 一种团队自治实体路径规划算法 TAEPP

郑延斌 赵沁平

(北京航空航天大学 计算机学院, 北京 100191)

**摘 要**: 对团队自治实体的路径规划问题进行了分析,提出了一种团队自治实体路径规划算法 TAEPP(Team Autonomous Entity Path Planning). 该算法是一种集中式的全局规划方法,使用改进的 A* 算法得到团队中 Leader成员的路径,通过对 Leader路径的优化得到关键点表和队形转换表,利用关键点表和队形转换表得到团队中非 Leader成员的路径. 队形转换表可以使行进中的队形变换更合理,减少了队形维护时需要的通信量. 通过调整算法中的 2 个影响因子,可以分别得到侧重于队形和侧重于距离的路径. 试验结果验证了 TAEPP算法的合理性和有效性.

**关 键 词**: 团队自治实体; 路径规划; 队形; A* 算法

**中图分类号**: TP 391

**文献标识码**: A **文 章 编 号**: 1001-5965(2008) 06-0656-05

## Team autonomous entity path planning algorithm TAEPP

Zheng Yanbin Zhao Qinping

(School of Computer Science and Technology, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

**Abstract**: Through analyzing of team autonomous entity path planning problem, a path planning algorithm TAEPP (Team Autonomous Entity Path Planning) was proposed. It is a centralized global planning method. Using mended A* algorithm, can get path for team leader; and through optimizing, the table of key points and the table of formation changing were achieved, so paths for all non-leader members could be found. Formation changing table made the formation change more reasonable, and decreased the required communication numbers in formation maintaining. Through adjusting the two parameters in the algorithm, different paths emphasized on formation and emphasized on distance would be found. The results of experiment show that TAEPP algorithm can provide more reasonable and more efficacious paths for every member in team, according to the environment and team formations.

**Key words**: team autonomous entity; path planning; formation; A* algorithm

自治实体是分布式虚拟环境中一类非常重要的实体,其行为由计算机控制而不需要人的干预. 自治实体包括个体和群体等类型,团队自治实体是群体自治实体的一种存在形式,其成员以个体的形式存在,各个个体自主决策并服从整个团队的控制[1]. 路径规划问题是团队自治实体研究的主要问题之一.

与个体自治实体行进不同,团队自治实体在行进过程中,成员之间要保持一定的几何形状,即队形. 按照队形行进具有如下优点[2]: ①充分获取工作环境信息,有利于执行侦察、搜索、排雷、安全巡逻等任务; ②增强团队的工作及抵抗外来侵扰的能力; ③提高团队的工作效率,简化任务的执行; ④增加系统的鲁棒性.

团队自治实体路径规划的任务是为团队中每个成员指定一条从起始地点到目标地点的路径,满足: ①与环境中的静态障碍物不碰撞; ②相互之间不碰撞; ③成员位置满足队形要求; ④成员按照

收稿日期: 2008-01-04

基金项目: 国家 863计划资助项目 (2006AA01Z333)

作者简介: 郑延斌 (1964- ), 男, 河南内乡人, 博士生, zybcg@163.com.

给定的路径可以达到目的地.

团队自治实体路径规划问题是多 Agent路径规划问题. 目前研究者提出的方法可以分为 2类: ①基于单 Agent的规划方法, 把多 Agent规划问题转换为单 Agent规划, 通过某种协调机制来修改各 Agent的路径、速度、方向等[3]; ②基于队形的规划方法, 如基于 L-F (Leader-Follower) 的方法[4]、基于行为的方法[5]、基于虚拟结构的方法[6]等.

第 1类方法, 由于规划的顺序对 Agent的路径影响非常大, 因此不同的顺序可能得到不同的规划结果, 不能得到最优解, 同时简单地把其它 Agent看作动态障碍物可能导致 Agent的行为不一致[7]. 第 2类方法在复杂的环境中队形的维持效率和路径最优都不能保证, 并且不能事先对可能的队形变化进行预测, 导致过多的等待和通讯.

为了解决这些问题, 文献 [7]首先建立了一条道路走廊, 把多 Agent运动限制在该走廊中, 虽然能够保证 Agent的运动一致性, 但是没有考虑队形问题. 文献 [8]提出了一种规划方法, 能够很好地维持队形, 但是没有考虑静态障碍物、Agent的目标等问题. 文献 [9]把团队形成的队形看作是一个整体, 设队形宽度为 $D$, 每个 Agent的宽度为 $L$, 利用 $A^*$ 算法从 $D$ 到 $L$ 循环求解. 该算法在规划中要队形宽度不变, 因此最终得到的很可能是队形的宽度为 $L$ 的路径.

本文提出一种团队自治实体路径规划算法 TAEPP(Team Autonomic Entity Path Planning), 该算法是一种全局规划算法, 首先使用改进的 $A^*$ 算法得到 Leader路径; 其次, 根据环境特性生成的队形转换表, 可以使队形转换更合理, 队形维持效果更好, 并且能够减少队形维护中需要的通信数量.

## 1　TAEPP算法

对具体的队形, 在团队中选择一个成员作为 Leader, 并且 Leader一经选定, 在行进的过程中不会改变. TAEPP算法通过下面 2步来实现:

1) Leader成员的路径规划. 使用改进的 $A^*$ 算法为团队中的 Leader成员规划出一条从起始点到目标点的最优路径.

2) 非 Leader成员的路径规划. 根据 1)得到的 Leader的路径, 为团队中每个非 Leader成员生成一条满足要求的路径.

图 1给出了 TAEPP算法的框图, 初始化包括地形栅格化、二值化、调入地形等工作. 根据行进中对队形的要求不同, 团队任务分为 3类: 队形在行进的过程中不能改变; 行进过程中尽量保持队形, 队形可根据具体的地形灵活改变; 行进中只要求距离最短, 队形不作要求. 第一类任务 Flag= 0, 第二、第三类任务 Flag = 1. $k_1$, $k_2$ 的定义在 1.1.1节给出.

### 1.1　Leader成员的路径

$A^*$ 算法的评价函数定义为: $f(n) = g(n) + h(n)$, 其中: $g(n)$是状态空间中从初始节点到当前节点 $n$的实际代价, $h(n)$是从当前节点 $n$到目标节点的最佳路径的估计代价. $A^*$ 确保使用该评价函数可以得到最优路径. 但是 Leader作为团队中的一个成员, 其路径的选择不能只考虑自身的因素, 还需考虑团队的队形.

#### 1.1.1　改进的 $A^*$ 算法

由于团队中的自治实体个数和团队的队形是已知的, 定义如下评价函数:

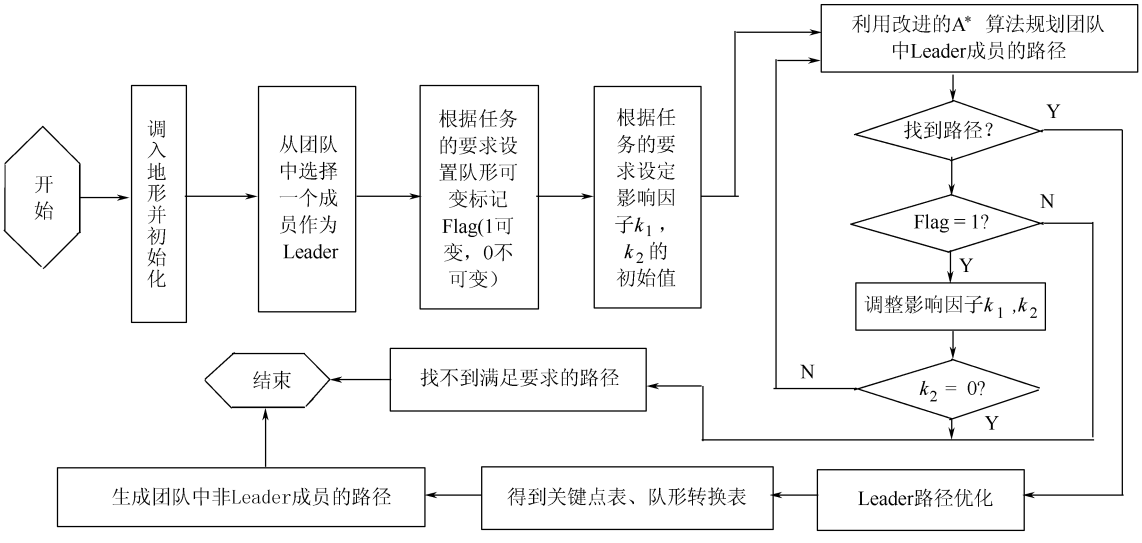



图 1　规划算法框图

$f'(n) = k_1 \times (g(n) + h(n)) + k_2 \times f_p(n, H, m)$

该评价函数包括2部分: ①距离评价函数; ②队形评价函数. $k_1 > 0$, $k_2 \geqslant 0$, 且 $k_1 + k_2 = 1$, 称 $k_1$ 为距离影响因子, $k_2$ 为队形影响因子, 当 $k_2 = 0$ 时, $k_1 = 1$, 即评价函数只考虑距离, 就变成 A* 算法的评价函数. $g(n)$ 和 $h(n)$ 的含义与 A* 算法的定义相同, $f_p$ 是队形效果评价函数, 定义如下:

$$f_p(n, H, m) = 1 \Big\backslash \sum_{i=1}^{m} a_i$$

式中, $n$ 为当前栅格点; $H$ 为要保持的队形函数; $m$ 为团队中自治实体的个数. $\sum_{i=1}^{m} a_i$ 表示当 Leader 位置为 $n$, 队形函数为 $H$ 时, 团队中所有成员在非障碍点的个数. $\boldsymbol{a}(n) = (a_1, a_2, \cdots, a_m)$ 表示当 Leader位置为 $n$ 时的队形向量, 如果成员 $i$ 的期望位置 (符合队形要求的位置) 上无障碍物, 则 $a_i = 1$, 否则 $a_i = 0$. 选择某个节点后, 除保存它的位置坐标外, 还要保存该节点的队形向量.

1.1.2 Leader路径的优化

设使用改进的 A* 算法求得的 Leader路径为 $(x_i, y_i, \boldsymbol{a}^i)$, $i = 1, 2, \cdots, p$. 其中, $(x_1, y_1, \boldsymbol{a}^1)$ 为起始点; $(x_p, y_p, \boldsymbol{d}^p)$ 为目标点. $x_i, y_i$ 为位置坐标, $\boldsymbol{a}^i$ 为队形向量. 由于改进的 A* 算法输出的是离散栅格节点, 故需要对离散的节点做连续化处理, 即路径的优化, 算法如下.

路径优化算法:

1) 把开始节点作为关键点, $i = 1$;

2) 从第 $i+1$ 个节点开始, 往后找出第一个满足 $\boldsymbol{a}^k \neq \boldsymbol{a}^i$ 点节点, 第 $k$ 个节点作为关键点, 插入到关键点表中. 若 $\| \boldsymbol{a}^i \| > \| \boldsymbol{a}^k \|$, 则 $(x_{k-d}, y_{k-d}, \boldsymbol{a}^{k-d})$ 为队形开始变换位置, $(x_k, y_k, \boldsymbol{a}^k)$ 为结束变换位置; 若 $\| \boldsymbol{a}^i \| < \| \boldsymbol{a}^k \|$, 则 $(x_k, y_k, \boldsymbol{a}^k)$ 为队形开始变换位置, $(x_{k+d}, y_{k+d}, \boldsymbol{a}^{k+d})$ 为结束变换位置, 将队形变换开始位置和结束位置插入到队形变换表中;

3) 在节点 $i$ 和节点 $k$ 之间做一条直线;

4) 用 LOS算法检查该直线上是否有障碍;

5) 如果没有障碍物, 并且直线上所有栅格点上队形向量相同, 则把该节点作为新的关键点, GOTO 7);

6) $k = [(k+i)/2]$, GOTO 3);

7) 如果 $k = p$, End;

8) $i = k$, GOTO 2).

算法中, $\| \boldsymbol{a} \| = \sum_{j=1}^{m} a_j$, $d \in [1, 2, 3, 4]$, 当地形栅格的宽度 ≫ 自治实体的宽度时, $d = 1$. 该算法在为 Leader路径优化的过程中得到了两个表: 关键点表和变换表. 关键点表示是经优化后的连续路径节点集合, 变换表中记录了队形变化的位置. 变换表中记录了队形变换开始位置和队形变换结束位置的队形向量, 可以使用某种转换机制或者方法把一种队形转换为另外一种队形. 当相应位置上的队形向量替换为队形函数后, 变换表就变成队形转换表, 队形转换表的结构见图 2

| $x_0, y_0, H_0$ | $x_1, y_1, H_0$ | $x_2, y_2, H_1$ | $x_3, y_3, H_1$ | ... | $x_{s-1}, y_{s-1}, H_g$ | $x_n, y_n, H_g$ |
|---|---|---|---|---|---|---|

图 2 队形转换表结构

其中: $(x_i, y_i)$ 表示位置坐标, $H_i$ 为队形函数, $i = 0, 1, \cdots, n$, $(x_0, y_0)$ 是 Leader的起始位置, $(x_n, y_n)$ 是 Leader的目标位置, $H_0$ 为初始队形, $H_g$ 为目标队形. 队形变化表的含义是从 $(x_0, y_0)$ 到 $(x_1, y_1)$ 队形一直是 $H_0$, 队形从 $(x_1, y_1)$ 开始变化, 到 $(x_2, y_2)$ 后变化结束, 队形变为 $H_1$, …,

1.2 团队中非 **Leader**成员的路径

根据前面得到的 Leader路径、队形转换表就可以生成团队中非 Leader成员的路径. 当然要得到非 Leader成员的路径首先要计算队形变换时这些成员的路径, 根据队形转换表, 利用文献[10]中给出的各个成员在队形转换中的位置变换方法就可以得到每个成员的转换路径.

非 Leader成员的路径算法:

1) $k = 1$;

2) IF $k = n$ THEN 8;

3) 根据 Leader的位置 $(x_k, y_k)$、队形函数 $H_k$ 计算每个成员的位置 $(x_i, y_i)$;

4) 把 $(x_i, y_i)$ 插入到该成员的路径列表中;

5) 如果 $(x_k, y_k)$ 在转换表中, 则把该成员的转换路径插入到该成员的路径列表中;

6) $k = k + 1$; GOTO 2).

## 2 试 验

在 DVENET[11]环境中对 5个自治实体构成的团队进行两组试验, 第 1组为验证 TAEPP算法, 如图 3、图 4所示. 图 3给出的是 5个自治实体按照三角形队形行进, 在经过障碍物时各个成员的运动轨迹. 由于 2个障碍物之间的距离小于队形的宽度要求, 队形发生变换.

为了验证影响因子对规划结果的影响, 图 3中给出了 5个自治构成的团队在经过复杂障碍物时, 不同的影响因子对规划结果的影响, 当 $k_1 = 0$ 时, 该算法就是标准的 A* 算法, 得到距离最短路径, 当 $k_1$ 比较小, $k_2$ 较大时, 可以得到一个最大限

度维持队形的路径规划 (如果该路径存在的话 ).

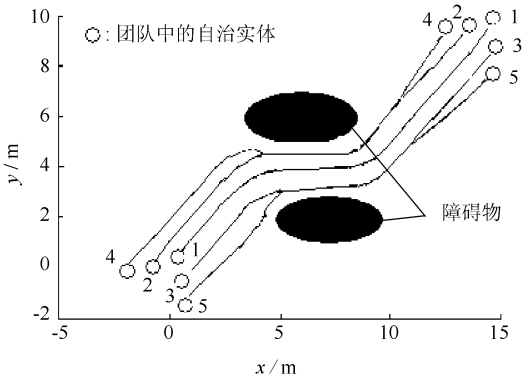



图 3　团队自治实体通过障碍物时的运动轨迹

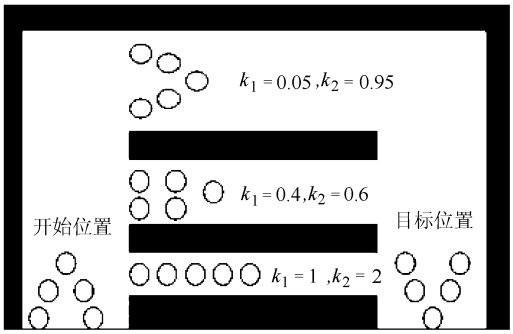



图 4　参数对规划路径的影响

第 2组为比较试验, 与文献 [9]的方法相比, TAEPP算法具有如下特点: ①Leader是团队的一个成员, 利用改进的 A* 算法来规划团队中 Leader成员的路径, 通过调整 2个影响因子可以得到不同的路径, 而文献 [9]的方法把 Leader看作是宽度为队形宽度的特殊成员; ②根据具体环境得到的队形转换表使队形的变换更符合环境要求、更合理, 文献 [9]的方法中的队形转换策略是固定的, 图 5给出了团队经过特殊障碍区时两种算法的结果比较, $a$ 和 $b$ 是 TAEPP算法的结果, $a'$和 $b'$ 是文献 [9]的方法的结果. 从图中可以看出, 在 $a$ 和 $a'$的区域中, 障碍物只影响到队形一侧的成员, 因此本文的算法只变换受影响的一侧, 在 $b$和 $b'$的区域中, 障碍物对整个队形没有影响, 因此本文算法得到结果是队形不变, 文献 [9]的方法需要将队形变为列队形.

从队形维持的角度, TAEPP算法属于 L-F方法, 与文献 [4]的方法相比, TAEPP算法具有: ①减少队形维持过程中需要的通信量, 因为当生成 Leader路径后, 同时得到队形转换表, Leader只需要将队形转换表发送给团队中的非 Leader成员, 非 Leader成员就可以根据队形转换表计算自己的路径, 因此只需要一次通信即可. 文献 [4]的方法要求 Leader在队形转换开始和结束时把自己的位置告诉每个非 Leader成员, 需要的通信次数为: Number(队形转换表中元素的个数 ); ②队形转换表使团队中的非 Leader成员清楚了解队形转换的结果, 从而使成员的行为达到一致, 文献 [4]的算法中非 Leader成员根据局部环境信息来调整自己的位置, 不知道队形的最后转换结果, 容易引起混乱.

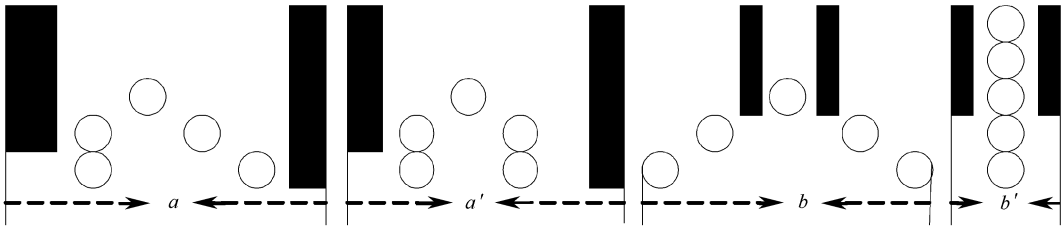



图 5　经过特殊障碍区的结果比较

## 3　结　论

本文给出了一个团队自治实体路径规划算法 TAEPP. 该算法是一种集中式的全局规划方法, 通过两步得到团队中所有自治实体的路径规划, 第 1步使用改进的 A* 算法得到团队中 Leader的路径, 并通过对 Leader路径的优化得到关键点表和队形转换表; 第 2步, 利用关键点和队形转换表得到团队中非 Leader成员的路径, 试验结果表明该算法是合理和有效的. 通过调整影响因子 $k_1$, $k_2$, 可以分别得到侧重距离和侧重队形的路径规划, 因此用户可以根据任务的不同要求选择不同的影响因子. 队形转换表是根据环境特性得到的, 用它来指导团队行进中的队形变换更合理, 非 Leader成员在队形变换时知道队形转换的结果, 故可以使队形维护的效果更好. 利用队形转换表可以减少队形维持中需要的通信数量. TAEPP算法中队形表示函数是 Leader路径及非 Leader位置计算的一个重要依据, 如何根据用队形函数来准确地描述不同的队形及如何根据 Leader路径规划中得到的队形向量来最优地进行队形转换, 是进一步的研究工作.

## 参考文献 (References)

[1] 梁晓辉. 计算机生成角色行为建模研究[D]. 北京: 北京航空航天大学, 计算机学院, 2002
Liang Xiaohui. The research on human behavior representation of computer generated actor[D]. Beijing: School of Computer, Beijing University of Aeronautics and Astronautics, 2002(in Chinese)

[2] Cao Y U, Fukunage A S, Kahng A B. Cooperative mobile robotics: antecedents and directions[C]//Published in Autonomous Robots, 1997, 4(1): 7-27

[3] Bennewitz M, Burgad W, Thrun S. Optimizing schedules for prioritized path planning of multi-robot systems[J]. Proc Int Conf on Robotics and Automation, 2001, 1: 271-276

[4] Hogg R W, Rankin A L, Romelitis S I, et al. Algorithms and sensors for small robot path following[C]//Proc 2002 IEEE Int Conf on Robotics and Automation. Washington D C: [s n.], 2002: 3850-3857

[5] Reynolds C W. Flocks, birds and schools: a distributed behavioral model[J]. Computer Graphics, 1987, 21(4): 25-34

[6] Anthony Lewis M, Tan K H. High precision formation control of mobile robots using virtual structures autonomous[J]. Autonomous Robots, 1997, 4: 387-403

[7] Kamphuis A, Overmars M H. Finding paths for coherent groups using clearance[C]//In Eurographics/ACM SIGGRAPH Symposium on Computer Animation, 2004: 19-28

[8] Tabuada P, Pappas G J, Lima P. Motion feasibility of multi-agent formations[J]. IEEE Transactions on Robotics, 2005, 21(3): 387-391

[9] Hao Yongxing, Agrawal S K. Planning and control of UGV formations in a dynamic environment: a practical framework with experiments[J]. Robotics and Autonomous Systems, 2005, 51: 101-110

[10] 王凤林. 坦克兵营连战术教程[M]. 北京: 国防工业出版社, 1991
Wang Fenglin. The course for tank camp tactics[M]. Beijing: National Defence Industry Press, 1991(in Chinese)

[11] 赵沁平. DVENET分布式虚拟现实应用系统运行平台与开发工具[M]. 北京: 科学出版社, 2005
Zhao Qinping. DVENET distributed virtual environment application system runtime platform and development tools[M]. Beijing: Science Press, 2005(in Chinese)


(责任编辑: 张 嵘)

(上接第 655页)

对比图 3中的 2条曲线、图 4和图 5,可以看出采用智能去抖动处理后的滑模变结构控制具有很强的鲁棒性,且输出的抖动现象明显减小.

### 3.2 实 验

对上述算法进行离散化处理,在工控机上实现. $p_p = 25 + 5\sin(10t)$ MPa, 实测速度曲线如图 6所示. 数据显示系统在干扰的作用下速度的变化小于 0.5%.

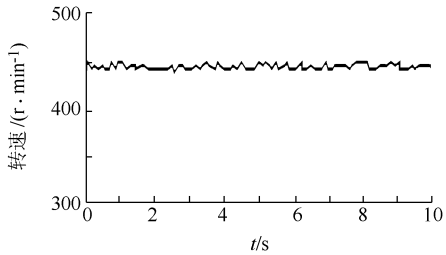



图 6 干扰作用下的速度曲线

## 4 结束语

对于地面试验采用变频调速驱动的航空泵试验台, 为使试验时和飞机上工作状态相同, 采用模糊控制监督下结合 SMC和 PDF的控制方法, 系统具有响应速度快, 与干扰和参数变化无关的优点, 且能降低抖振. 仿真与实验结果表明这种方法能有效地消除由于负载变化等因素对泵速度的影响.


## 参考文献 (References)

[1] Young K D, Utkin V I. A control engineer's guide to sliding mode control[J]. IEEE Transaction on Control Systems Technology, 1999, 7(3): 328-342

[2] Phelan R M, Chen L. Subvariable control 1983, technical report MSD-84-02[M]. Ithaca, NY: Sibley School of Mechanical and Aerospace Engineering, Cornell University, 1984

[3] Utkin V I. Variable structure systems with sliding modes[J]. IEEE Transactions Automatic Control, 1977, 22(2): 212-222

[4] Barrero F, Gonzalez A, Torralba A, et al. Speed control of induction motors using a novel fuzzy sliding-mode structure[J]. IEEE Transactions on Fuzzy Systems, 2002, 10(3): 375-383

[5] 王岩, 付永领. 模糊滑模迭代学习控制算法在液压系统中应用[J]. 北京航空航天大学学报, 2007, 33(1): 87-89
Wang Yan, Fu Yongling. Application of fuzzy sliding mode iterative learning control algorithm in hydraulic servo system[J]. Journal of Beijing University of Aeronautics and Astronautics, 2007, 33(1): 87-89(in Chinese)


(责任编辑: 赵海容)