

文章编号: 1002-0446(2003)01-0066-06

多机器人任意队形分布式控制研究*

韩学东 洪炳熔 孟 伟

(哈尔滨工业大学计算机科学与技术学院 哈尔滨 150001)

摘 要: 本文针对多智能体协作完成特定任务时难以在全自主控制的前提下协作形成任意队形和队形向量不易确定的问题, 通过由各智能体自主简单的确定自己的队形向量, 从理论上扩展基于队形向量的队形控制原理以生成任意队形, 改进机器人的运动方式以提高收敛速度, 提出一种快速收敛的机器人部队任意队形分布式控制算法. 为了解决智能体机器人之间的冲突问题, 提出了一个通信协调模型. 仿真实验和实际机器人实验均表明了算法的可行性和有效性.

关键词: 多机器人系统; 队形; 分布式控制; 机器人足球

中图分类号: TP24 **文献标识码:** B

DISTRIBUTED CONTROL FOR GENERATING ARBITRARY FORMATION OF MULTIPLE ROBOTS

HAN Xue-dong HONG Bing-rong MENG Wei

(Department of Computer Science and Technology, Harbin Institute of Technology Harbin 150001)

Abstract To generate arbitrary shape cooperatively and determine formation-vectors autonomously for multiple mobile robots when multiple mobile robot are to achieve a common task, an autonomous formation-vector determination method is first proposed for every agent. Then formation-vector based control algorithm is extended theoretically as so to generate a sort of given arbitrary formation shape and robot motion strategy is improved to reduce convergence time. A quick converging distributed algorithm for generating arbitrary shape of multiple mobile robots is proposed. To prevent the robot from collision, a communication model is presented. The simulation results and the real robot experiments show the feasibility and validity of the algorithm proposed in this paper.

Keywords: multiple robots system, formation, distributed control, robotic soccer

1 引言(Introduction)

在机器人技术中, 常需要多个机器人协调合作完成由单一机器人难以完成的复杂任务, 因而, 基于多智能体概念的多机器人合作问题正受到越来越多的学者的注意. 多智能体系统(Multi-Agent System)是由多个可计算的智能体(Agent)组成的集合, 它能协调一组自主体的行为(知识、目标、方法和规划等), 以协同地动作来完成的任务. 多智能体机器人系统(Multi-Agent Robotic System)中的每个机器人都被视为一个具有基本行为的自主体, 一般只处理与自身相关的局部目标和本地信息, 进行自主运动, 同

时又可充分发挥其智能和自主性来与其他机器人进行合作, 体现创现(Emergent)的群体行为. 目前, 多智能体机器人协作完成给定任务是多智能体系统(MAS)研究的一个热点问题, 其中, 多机器人编队控制是多机器人系统中一种常见的协作问题. 所谓编队控制, 是指多个机器人在到达目标的过程中, 保持某种队形, 同时又要适应环境约束(例如存在障碍物或者空间的物理限制)的控制技术. 多机器人编队控制是一个具有典型性和通用性的多机器人协作问题, 是许多多机器人协作问题的基础, 在很多领域有着广泛的应用. 这些领域包括形成固定队形搬运大的物体^[1,2]、形成弧形队形包围/捕获入侵

* 基金项目: 国家 863 高科技发展计划资助项目小型组机器人足球关键技术(2001AA422270); 国家自然科学基金资助项目(69985002).

收稿日期: 2002-08-26

者^[3,4]、完成空间任务等. 多机器人组成的机器人部队目前已经出现, 为完成巡逻、包围、侦察、杀敌等任务, 部队中各机器人间的协作是必不可少的, 而各机器人协作形成各种队形就成为机器人部队投入实际战争之前非常重要的一步.

本文研究了多个智能机器人编队包围捕获“入侵者”的合作问题, 以队形控制算法^[3,4]为基础, 给出各智能体独立确定其队形向量的方法, 扩展了队形控制原理, 改进机器人的运动方式以提高收敛速度, 提出一种快速收敛的机器人部队任意队形分布式控制算法. 本文提出的通信机制和协调机制保证系统不会产生冲突. 文中着眼于分散式的情形, 即系统不存在中心智能机器人, 各个机器人的地位相同. 各个机器人可以互相通信, 但只是局部的, 而不是全局的.

2 系统模型(System model)

实际机器人是有一定尺寸的, 故本文将环境定义为由若干正方形单元组成的二维世界(如图 1 所示), 每个单元视为智能机器人系统的共享资源, 或者是空的、或者被一个且仅有一个机器人占领, 机器人可以自由移动到任一单元. 在这个世界里有 n 个机器人 $R_i(i=1, 2, \dots, n)$, 一个目标 TARGET 和 m 个障碍 $O_j(j=1, 2, \dots, m)$, 机器人、目标和障碍物构成集合 OBJs. 每个机器人可检测到与其它机器人、目标、障碍的相对距离. 定义每个机器人本身的坐标系统为 Σ_i , 公共的基坐标为 Σ_0 , 为表征 R_i 的速度, 定义坐标系统 Σ_i^v , 它和基坐标系统的坐标原点相同, 和 Σ_i 的坐标轴方向一致, R_i 仅知道自己的坐标系统 Σ_i . 每个机器人相互间保持一定的距离(为此引入“队形向量”), 形成一定队形向目标移动以包围之, 当然必须避免和其它机器人、目标和障碍的碰撞.

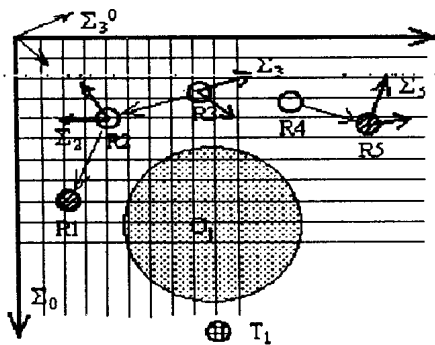


图 1 机器人工作空间模型及坐标系统
Fig. 1 Workspace and reference frame of robots

为描述部队队形 θ , 定义 θ 的前 $n-1$ 个分量为 n 个机器人之间的 $n-1$ 个相对位置向量

$$\begin{cases} \theta_{xi} \\ \theta_{yi} \end{cases} = \begin{vmatrix} x_{i+1} & x_i \\ y_{i+1} & y_i \end{vmatrix} \quad i = 1, 2, \dots, n-1 \quad (1)$$

这些相对位置向量和全部机器人构成了一个有向连接图 $G = \{V, E\}$, G 中节点 $v \in V$ 表示机器人, 有向路径 $e \in E$ 表示机器人间的相对位置向量. 为了确定部队的位置, θ 的最后一个分量定义为部队的重心

$$\begin{cases} \theta_n = (x_1 + x_2 + \dots + x_n)/n \\ \theta_n = (y_1 + y_2 + \dots + y_n)/n \end{cases} \quad (2)$$

一旦目标队形 θ 给定之后, 随后的任务主要就是完成如下两个映射: (1) 从 R_i 周围的环境(包括目标 TARGET、相邻机器人 $L_i = \{R_{i-1}, R_{i+1}\}$ 和自己周围的障碍物 O_{R_i}) μ_i 到其队形向量 d_i 的映射 $S_i: \mu_i \rightarrow d_i$, 主要解决各机器人如何自主确定队形向量的问题; (2) 从 $d_i, i=1, 2, \dots, n$ 到目标队形 θ 的映射 $F: d_i \rightarrow \theta$, 提供队形的控制算法.

3 队形控制算法(Formation control algorithm)

根据上述分析可以看出多机器人要协作形成任意指定队形需要确定队形向量(S 映射)和确定队形控制原理(F 映射)两步.

3.1 确定队形向量

队形向量 d_i 是 R_i 用来控制自身与部队中其它机器人协作形成指定队形的向量, 表示 R_i 在当前应产生的驱动力的大小和方向. 因自主分布式的要求, R_i 只能根据自己周围的环境 μ_i 在 Σ_i 中确定其队形向量 d_i . 机器人 R_i 确定其队形向量 d_i 的算法(S_i)描述如下:

记 R 为所有机器人组成的集合, 设集合 $q_0 = \{R_i \in R \mid R_i \text{ 看到目标 TARGET}\}$, $q_1 = \{R_i \in R - q_0 \mid R_i \text{ 看到 } R_j, R_j \in q_0\}$, $q_2 = \{R_i \in R - q_0 \cup q_1 \mid R_i \text{ 看到 } R_j, R_j \in q_1\}$, $q_3 = \{R_i \in R - q_0 \cup q_1 \cup q_2 \mid R_i \text{ 看到 } R_j, R_j \in q_2\}$, \dots , $q_{L-2} = \{R_i \in R - q_0 \cup q_1 \cup q_2 \dots \cup q_{L-3} \mid R_i \text{ 看到 } R_j, R_j \in q_{L-3}\}$, $q_{L-1} = \{R_i \in R - q_0 \cup q_1 \cup q_2 \dots \cup q_{L-2} \mid R_i \text{ 看到 } R_j, R_j \in q_{L-2}\}$, $q_L = R - q_0 \cup q_1 \cup q_2 \cup q_3 \dots \cup q_{L-2} \cup q_{L-1}$.

显然 $L+1 \leq n$ (机器人的个数). 不难理解, q_0, q_1, \dots, q_L 是 R 的一个划分, q_0 表示直接(亲眼)看到目标的机器人集合; q_1, \dots, q_{L-1} 表示间接看到目标的机器人集合; q_L 表示无法看到(包括直接和间接)目标的机器人集合.

(1) 确定 R_i 的跟踪对象 $t(R_i)$

$$t(R_i) = \begin{cases} g(\text{TARGET}, \theta, R_i) & R_i \in q_0 \\ R_j, & R_j \in q_s, R_j \in q_{s-1} \\ & \text{并且 } 0 < s < L - 1 \\ \varnothing & R_i \in q_L \end{cases} \quad (3)$$

其中 \varnothing 为空集, $g(\text{TARGET}, \theta, R_i)$ 是机器人 R_i 根据指定目标队形 θ 和目标 TARGET 的位置计算出的自己的目的地。

(2) 确定机器人 R_i 的队形向量 d_i

记从 $R_i (R_i \in q_0 \cup q_1 \cup q_2 \cup q_3 \dots \cup q_{L-2} \cup q_{L-1})$ 出发到其跟踪对象 $t(R_i)$ 终止的向量为 Q_i , 其相应的单位向量记为 Q_i . 阻挡 $R_i (R_i \in q_L)$ 的障碍物表面上距离 R_i 最近的点为 h , 障碍物上点与机器人中心连线的垂线方向的单位向量记为 Q_i , 则 d_i 可按下面的公式确定

$$d_i = \begin{cases} K Q_i & R_i \in q_0 \cup q_1 \cup q_{L-1} \\ K_p Q_i & R_i \in q_L \end{cases} \quad (4)$$

其中 K, K_p 是算法的控制参数。

在图 1 中, $q_0 = \{R_1, R_5\}$, 用阴影圆圈表示; $q_1 = \{R_2, R_4\}$, $q_2 = \{R_3\}$, 用空白圆圈表示; $q_3 = \varnothing$ 因此, R_1 和 R_5 的跟踪对象分别为 $g(\text{TARGET}, \theta, R_1)$ 和 $g(\text{TARGET}, \theta, R_5)$; R_2, R_3, R_4 则选择一个符合要求的机器人为跟踪对象。图中的细箭头表示跟踪关系, 即 R_2 跟踪 R_1, R_3 跟踪 R_2, R_4 跟踪 R_5 。

图 2 给出上述三种情况下队形向量的确定方法实例。

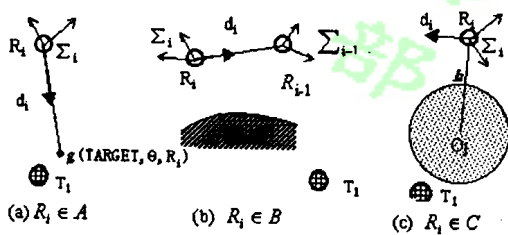


图 2 队形向量的确定示例

Fig. 2 The demonstration of determining formation vector

图 2(a) 表示在 R_i 能看见目标 TARGET 时, R_i 沿着通过其目标位置 $g(\text{TARGET}, \theta, R_i)$ 和自身连线的方向确定其队形向量; R_i 产生的驱动力可理解为其目标位置对 R_i 的吸引力。图 2(b) 表示 R_i 看不见目标 TARGET 但能看见机器人 R_{i-1} , 并且它们的关系满足式(3)时, R_i 沿着通过自身和 R_{i-1} 连线的方向确定其队形向量; R_i 产生的驱动力可理解为 R_{i-1} 对 R_i 产生的导向力。图 2(c) 表示(1) R_i 属于集合 q_L

时, R_i 沿障碍物表面上距自己最近的 h 点与自身连线的垂线方向确定其队形向量; R_i 产生的驱动力可理解为该障碍物对 R_i 产生的排斥力。

当 $R_i \in q_1 \cup q_2 \cup q_3 \dots \cup q_L$ 时, 上述方法确定的 d_i 方向的力可使 R_i 直接或间接绕开障碍物而看见目标 TARGET , 当所有机器人都看到 TARGET 后, 就可以在它的吸引下到达各自的目标点 $g(\text{TARGET}, \theta, R_i)$ 形成指定队形 θ . 因而对应这种队形向量确定方法 S , 必定存在一个 Γ , 使得给定的队形 θ 能够形成。

3.2 队形控制原理

目标 TARGET 和指定队形 θ 作为各智能体机器人的知识给定后, 机器人 R_i 就能计算出自己的目的地 $g(\text{TARGET}, \theta, R_i)$, 所有机器人都运动到其目标点后形成指定队形 θ . 为了和其它机器人协作形成指定队形完成类似包围目标 TARGET 的任务, R_i 要和 TARGET 保持一定的相对位置关系. 同时, 为了和其它机器人协作, R_i 还要和其它机器人保持一定的相对位置关系. 由于机器人工作中存在障碍物, 因此机器人还要受到来自障碍物的排斥力的作用, 从而避免与障碍物发生碰撞. 由 3.1 可知, 这三种作用力的方向就是队形向量的方向, 因此这三种作用力统称为队形向量驱动力, 它们都是 R_i 利用本地信息得出的. 当 R_i 距离一个物体表面足够近时, R_i 就会受到来自这个物体表面的斥力, 它使 R_i 控制自己到其它机器人、障碍物和目标 TARGET 的距离以免碰撞. 根据文献[3, 4]中的算法, 改进队形向量的确定方法和机器人的运动方式, 将原算法的极限收敛过程变成了匀速收敛过程并改进队形控制原理以生成任意队形. R_i 的运动状态由它受到的合力决定, 一般地, 部队中 R_i 的队形控制方法可表示如下:

$$\begin{aligned} \begin{bmatrix} \dot{x}_i^i \\ \dot{y}_i^i \end{bmatrix} &= \alpha \tau_i K \begin{bmatrix} x_i^i \\ y_i^i \end{bmatrix} / \left\| \begin{bmatrix} x_i^i \\ y_i^i \end{bmatrix} \right\| + \\ &\beta \tau_i \sum_{j \in L_i} K \begin{bmatrix} x_{j \in L_i}^i \\ y_{j \in L_i}^i \end{bmatrix} / \left\| \begin{bmatrix} x_{j \in L_i}^i \\ y_{j \in L_i}^i \end{bmatrix} \right\| + \\ &\gamma K_p \begin{bmatrix} x_b^i \\ y_b^i \end{bmatrix} / \left\| \begin{bmatrix} x_b^i \\ y_b^i \end{bmatrix} \right\| + \\ &\sum_{j \in \text{OBJS}} \delta_{ij} \left[\begin{bmatrix} x_j^i \\ y_j^i \end{bmatrix} - L \begin{bmatrix} x_j^i \\ y_j^i \end{bmatrix} / \left\| \begin{bmatrix} x_j^i \\ y_j^i \end{bmatrix} \right\| \right] \end{aligned} \quad (5)$$

式中当 $R_i \in q_0$ 时, $\alpha = 1, \beta = 0, \gamma = 0$; 当 $R_i \in q_1 \cup q_2 \cup q_3 \dots \cup q_{L-1}$ 时, $\alpha = 0, \beta = 1, \gamma = 0$; 当 $R_i \in q_L$ 时, $\alpha = 1, \beta = 0, \gamma = 1$. $(x_i^i, y_i^i)^T$ 是 R_i 的速度在 Σ_i^0 中的表示; $(x_b^i, y_b^i)^T$ 是自 R_i 出发到其目标位置 $g(\text{TARGET}, \theta, R_i)$ 终止的向量在 Σ_i 中的表示, $(x_j^i, y_j^i)^T /$

$(x_i^i, y_i^i)^T$ 是该向量的单位向量; $(x_j^i, y_j^i)^T, j \in L_i$ 是从 R_i 出发到 R_j 终止的向量在 Σ_i 中的表示, $(x_j^i, y_j^i)^T / \|(x_j^i, y_j^i)^T\|$ 是该向量的单位向量; $(x_b^i, y_b^i)^T / \|(x_b^i, y_b^i)^T\|$ 是避障向量在 Σ_i 中的表示, 其大小和方向同 Q_i . τ 是目标位置对 R_i 的吸引系数, τ_{ij} 是 R_i 和 R_j 之间的吸引系数. 当其它物体表面到 R_i 的距离小于 D 时, $\delta_{ij} = \delta > 0$, 否则, $\delta_{ij} = 0$. 式中右边前三项分别对应图 3(a)(b)(c) 三种情况下的队形向量驱动力. 就其物理意义而言, 公式右边第一项表示目标位置 $g(\text{TARGET}, \theta, R_i)$ 对 R_i 的吸引力; 第二项表示相邻机器人 $R_j \in L_i$ 对 R_i 的吸引力; 第三项表示避障力对 R_i 的作用; 第四项表示其它物体对 R_i 的排斥, 该排斥力只有在 R_i 到它们的距离小于 D 时才发挥作用.

根据队形 θ 的定义可知

$$\begin{bmatrix} \theta_x \\ \theta_y \end{bmatrix} = \begin{bmatrix} P & 0 \\ 0 & P \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \quad (6)$$

其中, $\theta_x = (\theta'_{x1}, \theta'_{x2}, \dots, \theta'_{xn})^T, \theta_y = (\theta'_{y1}, \theta'_{y2}, \dots, \theta'_{yn})^T, X = (x_1, x_2, \dots, x_n)^T, Y = (y_1, y_2, \dots, y_n)^T,$

$$P = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & \dots & 0 & 0 \\ M & M & \dots & 0 & M & M \\ 0 & 0 & \dots & 1 & -1 & 0 \\ 1/n & 1/n & 1/n & \dots & 1/n & 1/n \end{bmatrix}$$

如果所有机器人之间、任意机器人到障碍物之间和目标 TARGET 之间的相对距离都大于 D (如大多数实际应用中一样), 则有 $\delta_{ij} = 0, i = 1, 2, \dots, n, j \in OBJ_S$ 成立, 即斥力不发生作用, 而且, 如果所有机器人都能看见目标 TARGET, 从而计算出其目标位置 $g(\text{TARGET}, \theta, R_i)$, 则公式(3)可简化为

$$\begin{bmatrix} x_i^{\&} \\ y_i^{\&} \end{bmatrix} = \tau_i K \begin{bmatrix} x_i^i \\ y_i^i \end{bmatrix} / \left\| \begin{bmatrix} x_i^i \\ y_i^i \end{bmatrix} \right\| = \tau_i K \begin{bmatrix} d_{xi}^i \\ d_{yi}^i \end{bmatrix} \quad (7)$$

化成全局坐标系中的表示形式为

$$\begin{bmatrix} x_i^{\&} \\ y_i^{\&} \end{bmatrix} = \tau_i K \left[\begin{bmatrix} x_t \\ y_t \end{bmatrix} - \begin{bmatrix} x_i \\ y_i \end{bmatrix} \right] / \left\| \begin{bmatrix} x_t \\ y_t \end{bmatrix} - \begin{bmatrix} x_i \\ y_i \end{bmatrix} \right\| = \tau_i K \begin{bmatrix} d_{xi} \\ d_{yi} \end{bmatrix} \quad (8)$$

将公式(8)改写成矩阵的形式, 可得

$$\begin{bmatrix} X^{\&} \\ Y^{\&} \end{bmatrix} = \begin{bmatrix} \tau_i I_n & 0 \\ 0 & \tau_i I_n \end{bmatrix} \begin{bmatrix} d_x \\ d_y \end{bmatrix} = tK I_{2n} \begin{bmatrix} d_x \\ d_y \end{bmatrix} \quad (9)$$

由公式(6)和(9)可得出映射 $\theta = Fd$ 中的映射 F .

上述两部分表明在任意初始条件下, 都能在 3.1 给出的队形向量的映射的情况下找到从 d_i 到目标队形 θ 的映射 $F: d_i \rightarrow \theta$, 即能在任意初始条件下形成任

意的队形. 公式中涉及到的参数均为其它物体到 R_i 的相对方位在 Σ_i 中的表示形式, 这些参数由 R_i 本身的感知系统在自己的坐标系中就可以测出. 其中队形向量的确定可视为智能体本身的知识, 其相应的运动就是智能体的行为. 因此, 该控制算法就是智能体根据自己周围环境自主决定其行为的一个分布式自主控制算法.

3.3 队形控制算法描述

根据上述描述, 本文所提出的队形控制算法的形式描述如下:

Begin

Step1 [初始化]给出 R, O, TARGET 的初始位置和目标队形 θ .

Step2 [结束条件]各机器人是否到达 $g(\text{TARGET}, \theta, R_i)$, 如果是, 结束(指定队形 θ 形成), 否则, 转 Step3.

Step3 各机器人根据公式(3)自主确定跟踪对象 $t(R_i)$.

Step4 各机器人根据公式(4)计算出自己的队形向量 d_i .

Step5 根据公式(5)计算出各机器人下一时刻的位置 Pos_i .

Step6 如果 $|Pos_i - g(\text{TARGET}, \theta, R_i)| < \epsilon$, 则 R_i 调整自己的速度使其在下一时刻恰好到达 $g(\text{TARGET}, \theta, R_i)$.

Step7 机器人 R_i 运动到 $Pos_i (i = 1, 2, \dots, n)$, 转 Step2.

End

有上述描述可知, 队形控制算法的基本运算主要是 Step3、Step4 或 Step5, 其时间复杂度均为 $O(n^2m^2)$, 故整个控制算法的时间复杂度仍为 $O(n^2m^2)$.

4 冲突问题(Colliding)

由于 R_i 在检测其它机器人、障碍物和目标的位置时, 检测值与实际值存在误差, 误差的存在会造成机器人之间发生碰撞, 产生冲突, 这种情况主要发生在相邻机器人之间. 为解决这个问题, 我们将环境划分为由若干方形单元组成的二维世界, 如图 1 所示, 每个单元视为多机器人系统的共享资源, 或者是空的、或者被一个且仅有一个机器人占领, 如果有多个机器人同时抢占同一单元(冲突单元), 则认为产生了冲突.

为解决这个问题, 我们建立了局部通信和协调

的模型,使机器人之间能够交换信息,相互协调.下面简单介绍一下我们使用的基于消息的通信机制.

消息是由一个任意的字符串—消息名和一组任意的消息参数所组成.其语法为 $\langle \text{msg_name} \rangle [(P_0, P_1, \dots, P_n)]$, 其中 $\langle \text{msg_name} \rangle$ 是消息名,它描述消息的类型和含义, (P_0, P_1, \dots, P_n) 是消息参数,它具体说明此消息的内容.消息发送的原语为: $\text{Send} \langle \text{Receiver} \rangle \langle \text{msg_name} \rangle [(P_0, P_1, \dots, P_n)]$, 为此我们定义了如下四种消息:

INFORM: 告诉其它机器人一些信息

REQUEST: 请求其它机器人完成某些任务

ACCEPT: 接受请求

REJECT: 拒绝请求

解决冲突的最有效的方法是优先权法^[5],其基本思想是给每个机器人分配一个级别不同的优先权.分配机器人的优先权通常有两种方法:固定优先权法和动态优先权法.固定优先权法是在机器人作业以前为所有的机器人分配不同的优先权,一般不轻易改动.动态优先权法是在发生冲突的机器人之间根据一定的原则分配优先权.这里我们采用动态优先权法,让离冲突单元近的机器人优先权高,先通过,离冲突单元远的机器人优先权低,等待,用这种方法指导机器人的运动.每个机器人可以通过通信来获得其它机器人的优先权级别.

5 实验与分析(Experiment and analysis)

实验一:为了测试本文所提出的算法在收敛时间方面的改进,采用与文献[3,4]相同的仿真环境进行形成相同两个目标队形的仿真.文献[3,4]的仿真环境简述如下:8个机器人开始时以2的间距位于 $y = 16$ 的直线上, $R_1 = (3, 16)$, 两个半径为4的圆形障碍物分别位于 $(5, 16)$ 和 $(15, 12)$ 处,目标位于 $(10, 3)$ 处,队形向量的模为40, $\tau_{ij} = \tau_{ji} = 2, i = 1, 2, \dots, 7, j = i + 1, \tau = 10, D = 1, \delta = 150$.文献[3,4]采用龙格-库塔法,采样时间为0.001秒,机器人部队最终收敛位置如下: $R_1 = (8.14, 2.23), R_2 = (7.86, 3.74), R_3 = (8.41, 4.96), R_4 = (9.42, 5.61), R_5 = (10.58, 5.61), R_6 = (11.59, 4.96), R_7 = (12.14, 3.74), R_8 = (11.86, 2.23)$.我们用MATLAB重现仿真过程,队形最后收敛时循环次数为 $N_{old} = 2104$,用本文所提出的队形控制算法完成同样的仿真,最后收敛到这个队形时循环次数 $N_{MRF C} = 245$,仿真的过程如图3所示.

实验二:为了测试本文所提出的队形控制算法在功能上的改进(可以形成任意队形),在图3所示仿真环境下控制机器人部队进行绕过两个障碍物形成箭头形状队形的仿真,给定的队形如下: $R_1 = (9, 2.2), R_2 = (8, 3.2), R_3 = (9, 3.2), R_4 = (9, 4.2), R_5 = (11, 4.2), R_6 = (11, 3.2), R_7 = (12, 3.2), R_8 = (11, 4.2)$.仿真的初始条件和其它参数的值相同,控制参数 $K = K_p = 10$,算法在循环了284次后收敛到指定位置,形成了这个预先指定的队形,仿真的过程如图4所示,最终队形在图中用红色表示.

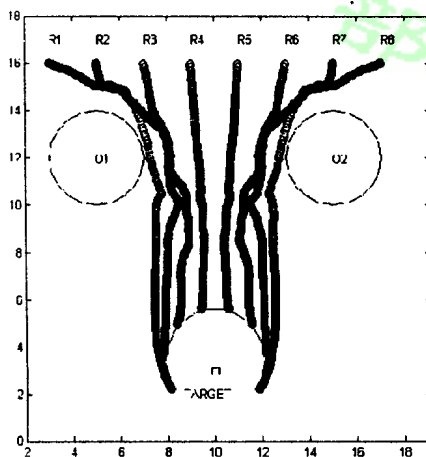


图3 弧形队形仿真过程
Fig. 3 The arc formation simulation

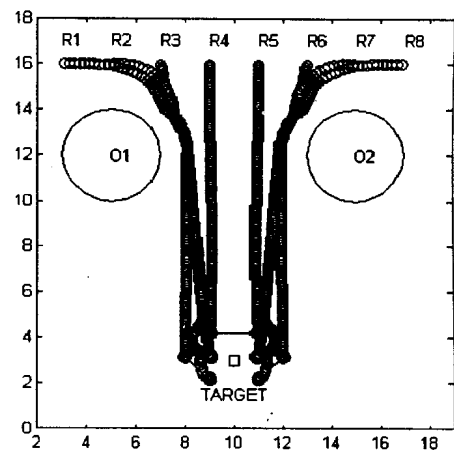


图4 箭头队形仿真过程
Fig. 4 The arrow formation simulation

实验三:机器人足球比赛是一个有趣并且复杂的新兴的人工智能研究领域,它是一个典型的多智

能体系统^[6,7].为了验证本文所提算法的有效性,将该所算法应用到机器人足球比赛中,并把它作为机

机器人球队的一种防守行为,为了简单起见,本文中仅给出包围静态目标时的情况,实验结果如图 5 所示.

图 5 中橙色的高尔夫球为目标,两个圆形的黄色物体为障碍物.

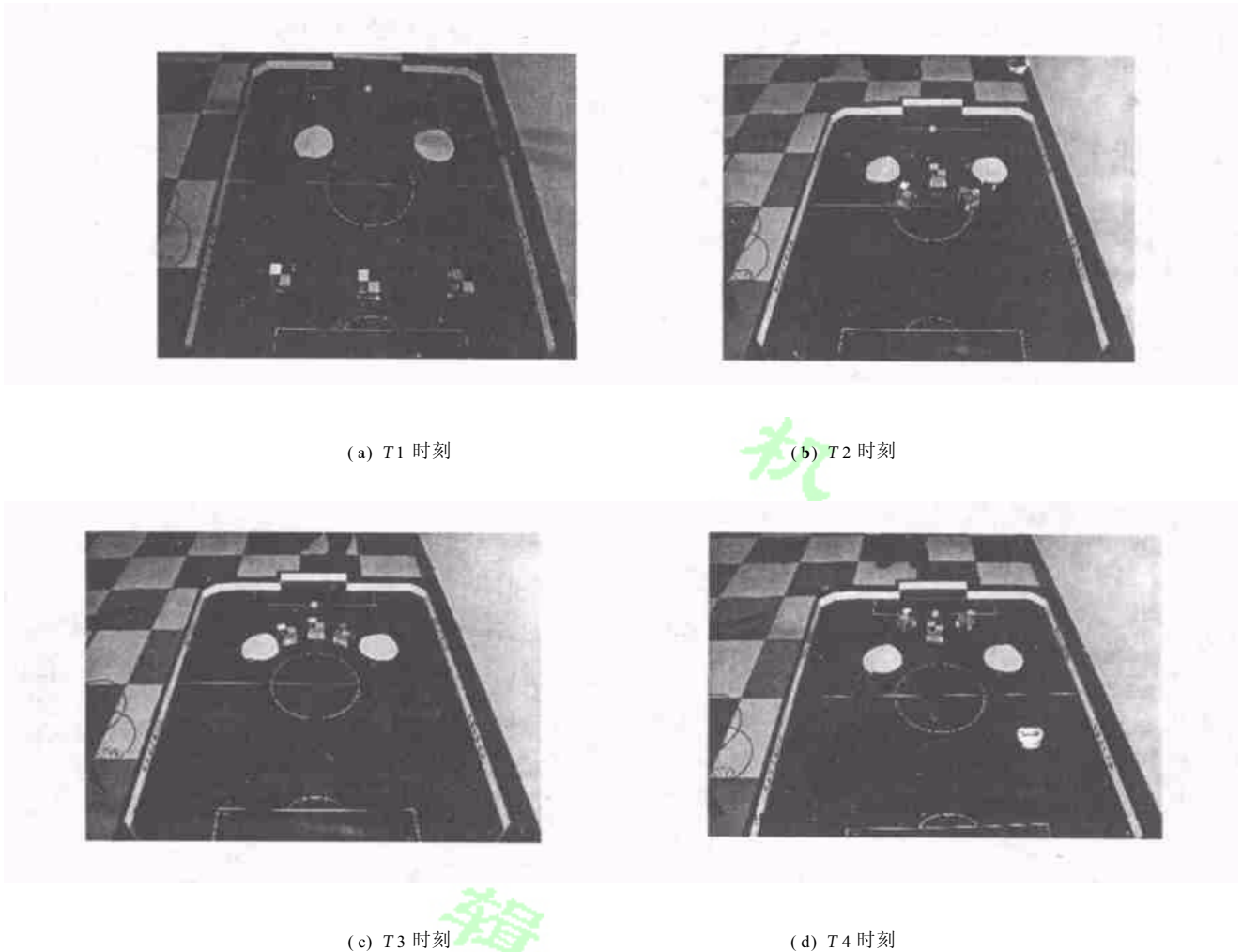


图 5 多机器人包围静态目标实验

Fig. 5 The experiment of capturing static target for multiple robots

6 结论(Conclusion)

本文讨论了机器人部队中多机器人协作形成任意队形的分布式自主队形控制算法.在文献[3,4]提出的多机器人协作捕获猎物的弧形队形控制算法的基础上,针对多智能体协作完成指定任务时各智能体自主控制条件下难以协作形成任意队形且队形向量不易确定的问题,给出各智能体自主确定其队形向量的方法,扩展了队形控制原理,改进了机器人的运动方式缩短收敛时间,提出一种快速收敛的静态已知环境下机器人部队任意队形分布式控制算法.相对于文献[3,4]的算法,本文提出的算法做了三点改进:(1)给出了一种自主确定队形向量的方法;(2)各智能体可协作形成任意指定的队形;(3)将文献[3,4]队形控制算法中极限逼近方式改为等速运动方式,缩短了队形收敛时间,改进了算法的实时性.

实验结果表明,本文所提出的队形控制算法收敛到指定队形的速度快,且能使所有机器人最终绕开障碍物,无碰撞的形成任意指定的队形.本文的仿真是在非常简化的条件下讨论的.但尽管如此,在目标和障碍物的某些特定安排下,发生冲突现象.为此,提出了一个局部通信协调模型以解决系统的冲突问题.

参考文献 (References)

- [1] Ota J, Miyata N, Arai T, Yoshida D, Kurabayashi, Sasaki J. Transferring and regrasping a large object by cooperation of multiple mobile robots. In: Proceeding of 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1995, 543- 548
- [2] Yamashita A, Fukuchi M, Ota J, Arai T, and Asama H. Motion planning for cooperative transportation of a large object by multiple mobile robots in a 3d environment. In: Proceedings of

- IEEE International Conference on Robotics and Automation, 2000, 3144- 3151
- [3] Yamaguchi H. A cooperative hunting behavior by mobile-robot troops. The International Journal of Robotics and Research, 1999, 20(9): 931- 940
- [4] Yamaguchi H. A cooperative hunting behavior by mobile-robot troops. In: Proceedings of the 1998 IEEE International Conference On Robotics and Automation, Washington, DC IEEE, 1998, 3204- 3209
- [5] M Erdmann, T Lozano-Perez. On Multiple Moving Objects. Proc. 1986 IEEE Int Conf On Robotics and Automation, 1986 : 1419- 1424
- [6] Jong-Hwan Kim. Third Call for Participation: Micro-robot World Cup Soccer Tournament 1996 (MiroSot'96). Accessible

from <http://vivaldi.kaist.ac.kr>.

- [7] Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, Eiichi Osawa. RoboCup: The Robot World Cup Initiative. In IJ-CAI-95 Workshop on Entertainment and AI/AI-life, Montreal, Quebec, August 1995

作者简介:

- 韩学东 (1973-), 男, 博士生. 研究领域: 多智能体系统, 智能控制, 机器人足球.
- 洪炳熔 (1937-), 男, 教授, 博士生导师. 研究领域: 空间机器人, 虚拟现实, 机器人足球.
- 孟 伟 (1974-), 女, 博士生. 研究领域: 多智能体系统, 系统建模.

(上接第 65 页)

考虑了系统的可扩充性、重用性和可移植性, 将机器人的正逆运动学、正逆动力学、各种控制算法、轨迹规划封装成通用模块, 并提供用户接口, 供用户设计自己的模型和控制算法. 基于 VB 和 MATLAB 设计了图形化的人机接口, 通过 DDE 进行数据交换, 解决了 dSPACE 的 TRACE 软件不能在一个图形窗口显示多个变量的问题.

本文所示的处理器任务分配是静态分配, 没有考虑处理器的效率及最优策略, 这将是本文进一步研究的问题.

参考文献 (References)

- [1] P I Corke. A robotics toolbox for MATLAB. IEEE Robotics and Automation Magazine, 1996, 3(1): 24- 32

- [2] Z Q Sun. Robot simulation system and its application. J of System Simulation, 1995, 7(3): 23- 29
- [3] 姜山, 程君实等. 面向对象的机器人实时仿真系统实现. 上海交通大学学报, 1999, 33(11): 1386- 1390
- [4] 陈亮. 基于多 DSP 的机器人实时仿真系统及控制方法研究. 同济大学硕士学位论文, 2001
- [5] dSPACE implementation guide. dSPACE Inc, 2001

作者简介:

- 陈启军 (1966-), 男, 36 岁, 工学博士, 教授. 研究领域: 机器人控制, 智能控制.
- 王月娟 (1935-), 女, 教授, 博士生导师. 研究领域: 机器人控制, 智能控制.
- 陈辉堂 (1933-), 男, 教授, 博士生导师. 研究领域: 机器人控制, 智能控制.