

移动机器人编队自修复的切换拓扑控制

张 飞, 陈卫东

(上海交通大学 自动化系, 上海 200240)

摘要: 针对机器人缺失后的移动机器人编队自修复问题, 构建了结合切换拓扑和交互动力模型的移动机器人编队模型, 通过分析机器人缺失后的拓扑变化情况, 提出了网络切换拓扑控制, 该算法利用递归实现自修复, 并且是收敛的. 通过设计相应的分布式算法, 本文将拓扑控制转化为基于局部交互的递归自修复个体控制, 证明了编队自修复个体控制的稳定性. 最后针对编队任务, 通过仿真验证了切换拓扑控制的有效性, 和其他方法比较具有低恢复时间和低功率消耗的优点.

关键词: 移动机器人; 编队; 网络; 自修复; 切换拓扑

中图分类号: TP242 **文献标识码:** A

Switched topology control for self-healing of mobile robot formation

ZHANG Fei, CHEN Wei-dong

(Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: To solve the self-healing problem of mobile robot formation with failed robots, a formation model combining switched topologies and an interaction dynamics model for mobile robot is established. Based on the analysis of the network topology with failed robots, we propose a switched topology control using local rules. Furthermore, considering the distributed information obtained by individual robots, we transform the topology control into an individual control for recursive self-healing with local interactions. The stability of the system with the individual control is also proved. Simulation results demonstrate the validity of the switched topology control, and show that the proposed control method has the advantages of shorter recovery time and lower power consumption over previous methods.

Key words: mobile robots; formation, networks; self-healing; switched topology

1 引言(Introduction)

移动机器人编队任务^[1,2]是指大规模机器人群体在局部交互和有限感知的条件下能够保持给定队形. 编队是多机器人分布式协作的基本任务之一, 并广泛应用于实际多移动机器人系统. 自修复^[3], 作为群体智能关键组成部分, 是指移动机器人编队在缺失部分机器人的情况下能够恢复自身拓扑和系统性能的一种自组织行为. 在移动机器人编队过程中, 尤其是在人类无法达到的工作区域, 比如空间探索和军事侦察等, 自修复行为显得尤为重要, 它能自发地恢复整体编队. 自修复的关键问题在于如何设计一种分布式的控制方法以完成全局修复目标.

多机器人系统和传感器网络的自修复算法通常可分为3类. 第1类算法指派某个机器人去填补缺失机器人的空位, 被称为直接自修复方法^[4]. 这类算法需要获得所有机器人的信息, 因此不是分布式的, 并且恢复时间长. 第2类自修复方法主要用于传感器

网络的覆盖问题, 它使用一个全局参数(比如节点密度)来修复由缺失节点引起的感知盲区, 被称为密度自修复方法^[5,6]. 但它没有考虑网络拓扑的变化和由增大的通信范围引起的功率消耗增大问题. 第3类方法利用局部交互和切换拓扑来修复缺失机器人, 被称为递归自修复方法. 在前期工作中, Zhang等人^[3,7]提出一种递归自修复的拓扑控制方法并指出这种方法可以改善网络的同步性, 但是该方法并没有给出具体的个体机器人控制, 而且没有给出评价自修复算法的指标. Boonma等人^[8]提出一种分布式方法用于传感器网络中缺失传感器的修复, 但没有说明如何利用其移动性.

在一般网络中, 同步是一个基本概念^[9], 并且存在于许多研究领域, 比如传感器网络, 复杂动力网络和通信网络. 相对应地, 运动同步^[3]在移动机器人系统也是最基本的集体行为之一, 它的目标是在局部交互下使所有机器人的速度达到同步. 它主要源于

收稿日期: 2008-12-07; 收修改稿日期: 2009-07-21.

基金项目: 国家“863”计划资助项目(2006AA040203, 2007AA04z340); 国家自然科学基金资助项目(60775062); 新世纪优秀人才支持计划资助项目(NCET-07-0538).

生物行为的研究^[10]. 在多机器人学中, 许多复杂的实际任务都可以被分解成一系列的运动同步, 比如大型物体的搬运和操作, 队形形成和保持, 在未知环境探索和创建地图, 以及覆盖问题等.

本文基于复杂网络的同步理论, 构建了移动机器人编队模型, 它结合了切换网络拓扑和用于描述机器人运动的交互动力模型. 基于这一模型, 提出了一种全分布式递归自修复算法, 证明了自修复编队个体控制的稳定性. 最后, 针对编队任务, 通过仿真验证了该方法的有效性, 和其他算法比较而言, 该算法具有低恢复时间和低功率消耗的优点.

2 问题描述(Problem description)

2.1 移动机器人编队建模(Modeling of mobile robot formation)

假设编队中有 n 个移动机器人, 每个机器人的工作区域是 N 维的. 第 i 个机器人的位置状态变量和速度状态变量分别用 $p_i \in \mathbb{R}^N$ 和 $x_i \in \mathbb{R}^N$ 表示, $i = 1, 2, \dots, n$. 根据文献[3,7,9,11,12]中的运动模型, 每个机器人的连续时间动力模型可以用一个交互动力模型所表示:

$$\begin{cases} \dot{p}_i = x_i, \\ \dot{x}_i = f(p_i, x_i) + c_1 \sum_{j=1}^n a_{ij}(k) \cdot h_1(p_j) + \\ c_2 \sum_{j=1}^n a_{ij}(k) h_2(x_j), \\ i = 1, 2, \dots, r. \end{cases} \quad (1)$$

其中 $f(p_i, x_i)$ 表示外部环境对机器人 i 的作用, 或是机器人 i 的控制输入. 例如, 可以使用速度反馈来表示 $f(p_i, x_i)$:

$$f(p_i, x_i) = a_m(v(t) - x_i), \quad i = 1, 2, \dots, n. \quad (2)$$

其中: $f(p_i, x_i)$ 表示反馈系数, $v(t)$ 为给定速度. 模型(1)中, $c_1, c_2 > 0$ 表示系统的耦合强度, 函数 h_1 和 h_2 分别表示机器人位置和速度的输出函数, 用来描述系统中其他机器人对机器人 i 的作用. 切换耦合矩阵 $A(k) = (a_{ij}(k))$ 表示第 k (整数 $k > 0$)步机器人通信拓扑: 若机器人 i 和 j 之间存在通信连接, 则 $a_{ij}(k) = 1$; 否则 $a_{ij}(k) = 0$, $i \neq j$. 另外, $a_{ii}(k) = -\sum_{j=1, j \neq i}^n a_{ij}(k) = -d_i(k)$, 其中 $d_i(k)$ 表示第 k 步机器人 i 的度. 与机器人 i 有连接的机器人成为它的邻居. 令 $N_e(i)$ 表示机器人 i 的邻居所组成的索引集合, 即

$$N_e(i) \triangleq \{j | a_{ij} = 1\} \subseteq \{1, 2, \dots, n\}. \quad (3)$$

此外, 机器人 i 的度的大小等于它邻居节点的个数. 本文中机器人间的交互和通信分别基于以下两个假设.

假设 1 局部交互: 机器人仅能得到其邻居的信息.

假设 2 通信: 机器人间的通信无延时.

为了简化表达以及编队任务要求, 本文将 h_i 和 h_2 表示成一个线性耦合函数, 即

$$\begin{aligned} \dot{x}(t) = & f(p_i, x_i) + c_1 \sum_{j=1}^n a_{ij}(k) \cdot (p_j - q_{ij}) + \\ & c_2 \sum_{j=1}^n a_{ij}(k) x_j, \quad i = 1, 2, \dots, r. \end{aligned} \quad (4)$$

其中: $q_{ij} = p_j^d - p_i^d$ 表示机器人 j 和 i 之间的期望相对距离, p_j^d 表示机器人 j 的期望(目标)位置. 所有机器人的目标位置组成的集合表示该机器人系统的物理拓扑, 即队形, 记作 $Q = \{p_i^d | i = 1, 2, \dots, n\}$. 编队任务的目标是移动机器人个体达到给定队形, 即可以使用下列队形函数:

$$F_m = \sum_{i=2}^n \|(p_i - p_1) - (p_i^d - p_1^d)\|^2. \quad (5)$$

来描述移动机器人编队的结果^[13]. 若函数 F_m 始终等于零, 那么说明编队任务完成.

移动机器人的通信拓扑通常由 K 邻居模型^[14]所描述, 其中每个机器人的度等于或小于 K , 即 $d_i \leq K$, $i = 1, 2, \dots, n$, 例如, 图1分别显示了4种典型的机器人网络拓扑, 其中 $K = 3, 4, 6$ 和 8 .

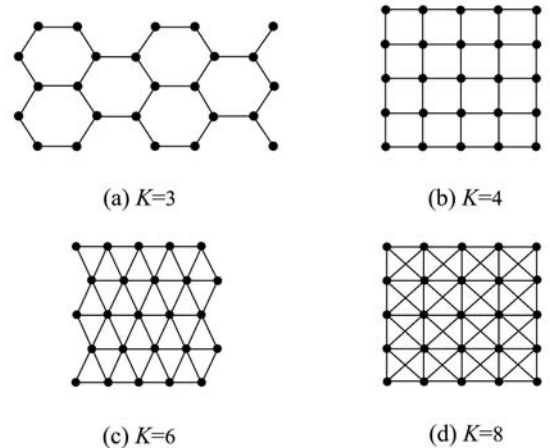


图1 4种典型的移动机器人编队 K 邻居模型

Fig. 1 Four typical K -neighbor model of mobile robot formation

2.2 机器人缺失后的网络拓扑分析(Topology analysis for robot network with failed robots)

假设编队中有一小部分机器人缺失, 缺失率为 Δ ($0 < \Delta \ll 1$), 那么其通信拓扑结构遭到破坏. 在文献[3,7,15]中, 耦合矩阵 A 的第二大特征值 λ_2 用于评价该系统同步的稳定性和鲁棒性. λ_2 的值越小表示该系统同步稳定性和鲁棒性越高. 此外, 当且仅

当 $\lambda_2 = 0$ 时, 整个系统破裂成两个或多个不相连的部分. 明显地, 系统(1)中的机器人缺失将改变系统的耦合矩阵. 如果其耦合矩阵的第二大特征值保持不变, 那么机器人缺失后的系统同步稳定性和鲁棒性也将不变.

分别令 $A_{rf} \in \mathbb{R}^{n \times n}$ 和 $\tilde{A}_{rf} \in \mathbb{R}^{(n-[\Delta n]) \times (n-[\Delta n])}$ 为 n 个机器人的初始系统和缺失 $[\Delta n]$ 个机器人后的新系统的耦合矩阵. 这里, $[\Delta n]$ 表示小于且最接近于实数 Δn 的整数. 再分别令 λ_{2rf} 和 $\tilde{\lambda}_{2rf}$ 表示 A_{rf} 和 $\tilde{A}_{rf}$ 的第二大特征值. 那么新耦合矩阵 $\tilde{A}_{rf}$ 可通过删除初始矩阵 A_{rf} 中的相关行与列, 并重新计算对角线上元素而得(具体步骤参见文献[3,15]).

最坏情况下, 极少量机器人缺失就会导致整个系统破裂. 例如, 一行或一列上的机器人全部缺失, 那么整个系统将会不相连^[3], 即 $\lambda_2 = 0$. 因此, 当机器人缺失时, 有必要自修复网络拓扑以阻止系统分裂成两个或多个不相连的部分.

2.3 自愈性及其评价指标(Self-healing ability and its performance metrics)

本文首先从缺失机器人的拓扑结构变化入手, 寻找移动机器人的拓扑位置变化对于整个群体性能变化的规律, 从而指导机器人个体控制的设计. 明显地, 建立机器人系统自愈性的评价指标是研究上述变化规律的第1步.

评价系统自愈性的指标可以分成2组: 一组用于评价网络拓扑变化情况, 包括度总和、连通性和收敛性; 另一组用于评价自修复的动态过程, 包括恢复时间和功率消耗.

1) 度总和.

度总和表示系统中所有机器人度的总和, 即

$$T = \sum_{i=1}^n d_i. \quad (6)$$

机器人缺失后, 度总和下降. 编队自修复的目标就是将系统拓扑尽量恢复成未缺失时的拓扑, 即度总和尽量与未缺失时保持一致.

2) 连通性.

连通性是移动机器人系统重要特性之一, 是指自修复方法阻止系统分裂成两个或多个不相连部分的性能, 即 $\lambda_2 \neq 0$.

3) 收敛性.

对于递归自修复方法, 每一步迭代都能使得式(6)的值不减, 那么称该算法是收敛的.

4) 恢复时间.

恢复时间是指从机器人缺失开始到系统完成自修复为止的时间长度, 记作 t_r . 若一种控制算法使得系统的恢复时间减小, 那么该系统的自愈性增大.

5) 功率消耗.

机器人的功率消耗可以分成4大部分: 通信、移动、感知和计算. 后面3项的功率消耗, 对于不同自修复算法而言, 大体相等. 因此, 不同自修复算法的功率消耗差异主要由通信功率所决定, 而通信功率与两个相连机器人距离有限方 α 成正比, 即

$$P_i = \sum_{j \in N_e(i)} \beta q_{ij}^\alpha. \quad (7)$$

其中: P_i 表示机器人 i 的通信功率, β 是功率和距离的系数. 指数 α 由环境条件所决定, 一般取2到5之间. 由此可见, 机器人功率消耗主要由通信机器人之间的距离所决定. 若自修复时通信机器人之间的距离较小, 则功率消耗也较小.

系统的总体通信功率是所有机器人通信功率的总和, 即

$$P = \sum_{i=1}^n P_i. \quad (8)$$

由此可见, 设计自修复控制的目标是要求系统在相同恢复结果(相同的度总和与连通性)下拥有较高的自愈性(较短的恢复时间和较小的功率消耗).

为了解决大规模群体全局目标和个体局部控制的矛盾, 基于移动机器人编队模型, 本文提出一种从系统网络到个体的控制设计方法.

3 网络切换拓扑控制(Switched topology control for network)

本节研究通过递归方法进行自修复的系统在保证相同恢复结果的前提下具有较高的自愈性, 提出了网络切换拓扑控制.

3.1 自修复设计原则(Principles of self-healing design)

文献[15]证明了当缺失率足够小, 即 $\Delta(0 < \Delta \ll 1)$, 网络规模足够大, 即 $n \gg 1$, 并且缺失机器人的度足够小, 即 $d_i < d_0, i = 1, 2, \dots, [\Delta n]$, 那么有

$$\tilde{\lambda}_{2rf} \approx \lambda_{2rf}. \quad (9)$$

这表明耦合矩阵的第二大特征值保持大致不变, 所以网络的同步性也保持不变. 假设采用 K 邻居模型的机器人系统规模足够大 $n \gg 1$, 那么式(9)也可以应用到本文的模型中, 因为每个机器人的度都是足够小的, 即 $d_i \leq K, i = 1, 2, \dots, n$.

通过以下仿真验证式(9)应用到 K 邻居模型并给出设计原则. 仿真采用一个规模为 $n = 1089$ 的移动机器人系统, 其中 $K = 6$. 比较两种机器人缺失方式, 观察 λ_2 的变化情况: 1是每次移除度最高的机器人(图2中带圈虚线), 2是每次移除度最低的机器人(图2中实线). 如图2所示, 第1种移除方式下第二大特征值随移除数量而增大, 而第2种方式下则减小.

这表明当度较低的机器人缺失时对于系统同步性影响小并使得 λ_2 有所改善. 因此, 将度较低的机器人替换度较高的缺失机器人会改善系统同步性, 这成为自修复方法的设计原则.

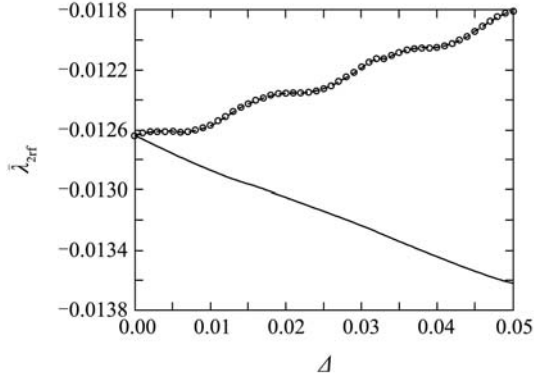


图2 耦合矩阵第二大特征值随缺失率变化关系图

Fig. 2 Second-largest eigenvalue of coupling matrix vs. failure ratio

3.2 递归自修复(Recursive self-healing)

虽然直接自修复方法可将度最小的机器人填补缺失机器人的空缺位置, 但是它需要得到所有机器人的信息. 本文关于自修复的基本思想是通过局部交互寻找一个合适机器人去填补空缺, 在下一步, 新出现的空缺又通过这个局部规则被另一个机器人所填补, 依次类推. 由此形成一种具有切换拓扑的递归自修复算法.

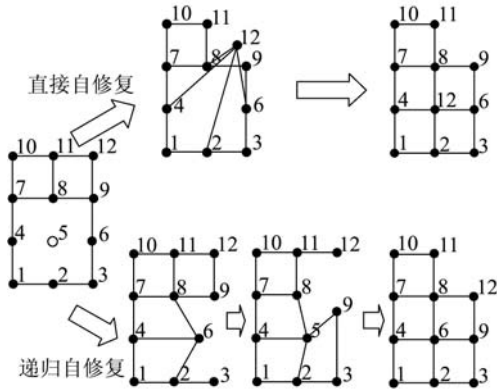


图3 递归自修复和直接自修复的过程

Fig. 3 Processes of recursive self-healing and direct one

图3显示了一个移动机器人系统分别采用直接自修复和递归自修复的过程. 系统中机器人5缺失了, 采用直接自修复方法的系统由度最小的机器人12去填补该空位. 基于递归自修复方法, 机器人6, 机器人5的邻居之一, 填补了该空缺, 随后机器人6产生的新空缺由机器人9所填补. 最后机器人12填补了机器人9的空缺. 虽然这两种方法最后所形成的拓扑是一样的, 但是递归自修复方法是完全分布式的.

递归自修复方法相对于直接自修复方法, 填补机器人的移动是依次进行的, 且仅填补邻居位置的空缺, 因此通信机器人之间的距离变短了, 因此递归自修复的优势在于: 分布式, 较低的恢复时间和较低的功率消耗. 所以递归自修复的自愈性较高.

3.3 网络拓扑控制算法(Topology control algorithm)

通过控制网络拓扑, 系统可以自修复缺失机器人的空缺位置. 令 $A(1) = \tilde{A}_{rf}$ 为机器人缺失之后的初始耦合矩阵. 令 i_f 表示缺失机器人, 那么 $N_e(i_f)$ 为 i_f 的邻居集合. 本文的自修复拓扑控制可以通过下列规则所描述.

规则1(较低度的候选者) 缺失机器人的邻居 $j \in N_e(i_f)$, 若满足它的度小于 i_f 的度, 即 $d_j \leq d_{i_f}$, 那么它成为填补机器人 i_f 空缺的候选者, 它们所组成的索引集合表示为

$$C_a(i_f) \triangleq \{j | d_j \leq d_{i_f}, j \in N_e(i_f)\}. \quad (10)$$

规则2(最低度) 集合 $C_a(i_f)$ 中最低度的机器人填补 i_f 的空缺, 称作填补机器人, 记作 i_s .

规则3 如果集合 $C_a(i_f)$ 中存在两个或多个机器人拥有相同的最低度, 那么随机选择一个作为填补机器人 i_s .

为了避免由规则3随机选择所引起的同一机器人的重复填补, 本文使用一标志来凸显拥有优先填补空缺的机器人, 该标志记作 M . 出现随机选择之后, 该标志用缺失机器人和填补机器人的相对距离所表示, 即 $M = q_{i_s i_f}$. 随后递交给具有相同相对距离的机器人 i'_s , 即 $q_{i'_s i_s}, i'_s \in N_e(i_s)$. 在下一步, 机器人 i'_s 拥有填补缺失机器人 i_s 新空缺的优先权, 如果它属于集合 $j \in C_a(i_s)$ 并且是度最低的机器人之一. 如果只有一个度最低的机器人, 那么 i'_s 的标志消失, 并根据规则2由度最低的机器人填补空缺. 若标志不消失, 则继续传递给下个符合条件的机器人, 直到找不到相对应的机器人.

4 个体控制(Individual control)

本节通过分布式算法设计, 将网络切换拓扑控制转化为基于局部交互的个体控制算法, 从而实现了算法的分布式特性.

4.1 分布式算法设计(Distributed algorithm design)

假设编队中每个机器人仅能获得其邻居的信息, 并且不考虑通信延时. 那么上节所述的3条规则中, 只有第1条满足分布式条件, 因为它仅需知道缺失机器人的信息就可作出判断. 对于剩余2条, 需要在候选机器人集合内实现信息共享. 对于第 $k(k > 1)$ 步,

所需要填补的空缺正是第 $k-1$ 步的填补机器人, 它拥有所有邻居的信息, 因此可以作为决策者决定哪个邻居填补它的空缺位置, 决策结果记作 T , 由该机器人发送给下一任填补机器人. 对于第1步拓扑切换, 由于机器人缺失是随机的, 无法指定哪个机器人作为决策者, 只能让候选机器人之间互相建立临时通信连接, 选出临时决策者, 再由它根据上述规则决定填补机器人.

假设每个填补机器人都会发送给其邻居信息, 以区别第1步空缺填补和其他填补, 该信息记作 W . 若一个机器人发现其邻居缺失, 并且它没有收到信息 W , 那么表明这是第1步空缺; 否则表示填补机器人已存在, 该机器人不需要产生动作.

4.2 个体控制算法(Individual control algorithm)

这根据上述规则和分布式设计, 形成了一种递归的, 分布式的自修复拓扑控制算法. 下面的算法根据机器人 i 而制定, 并适用于编队中所有机器人.

自修复算法:

Step 1 如果机器人 i 得知其邻居 i_f 的缺失信息, 并收到了信息 T , 那么转到Step 5; 若收到信息 W , 则转到Step 7; 若没有收到信息, 则转到Step 1.

Step 2 根据规则1, 机器人 i 比较它的度与缺失机器人度 d_{i_f} 的大小, 若小于, 则属于候选者集合 $C_a(i_f)$; 否则, 返回Step 1.

Step 3 机器人 i 扩大通信范围, 与其他属于 $C_a(i_f)$ 的机器人建立临时连接, 并从中选出一名决策者.

Step 4 若机器人 i 是决策者, 那么根据规则2和3决定填补机器人 i_s .

Step 5 若机器人 i 是填补机器人, 即 $i_s = i$, 它根据邻居信息决定下一步填补机器人, 并发送信息 T 给它, 同时发送信息 W 给其他邻居机器人.

Step 6 填补机器人 i_s 与缺失机器人的邻居 $N_e(i_f)$ 建立新连接, 同时切断与原先邻居 $N_e(i_s)$ 的通信连接, 同时耦合矩阵中相应的元素作如下变化:

$$\begin{cases} a_{i_s j}(k) = 0, j \in N_e(i_s), \\ a_{i_s j}(k) = 1, j \in N_e(i_f), \\ a_{ii}(k) = - \sum_{j=1, j \neq i}^n a_{ij}(k). \end{cases} \quad (11)$$

返回Step 1.

Step 7 机器人 i 与填补机器人 i_s 建立新连接:

$$\begin{cases} a_{i i_s}(k) = 1, \\ a_{ii}(k) = - \sum_{j=1, j \neq i}^n a_{ij}(k). \end{cases}$$

返回Step 1.

5 拓扑控制稳定性分析(Stability analysis of topology control)

移动机器人编队(1)达到同步速度, 如果满足

$$x_1(t) = x_2(t) = \cdots = x_n(t) \rightarrow v(t), t \rightarrow \infty, \quad (12)$$

其中 $v(t) \in \mathbb{R}^N$ 可以是团队均衡速度, 或是公式(2)给定的速度. 通过将模型(1)表示成复杂网络理论中的一般表达形式, 本节研究编队自修复运动同步的稳定性.

首先, 为简化证明, 令 $c = c_1 = c_2$, 那么模型(1)可表达为

$$\begin{bmatrix} \dot{p}_i \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} x_i \\ f(p_i, x_i) \end{bmatrix} + c \sum_{j=1}^n a_{ij}(k) \begin{bmatrix} 0 & 0 \\ I_N & I_N \end{bmatrix} \begin{bmatrix} h_1(p_j) \\ h_2(x_j) \end{bmatrix}, \quad n=1, 2, \dots, n, k=1, 2, \dots, k_s. \quad (13)$$

其中 I_N 表示 N 维单位矩阵.

令 $y_i = [p_i \ x_i]^T \in \mathbb{R}^{2N}$, $F(y_i) = \begin{bmatrix} x_i \\ f(p_i, x_i) \end{bmatrix}$ 和 $H(y_i) = \begin{bmatrix} 0 & 0 \\ I_N & I_N \end{bmatrix} \begin{bmatrix} h_1(p_i) \\ h_2(x_i) \end{bmatrix}$, $i = 1, 2, \dots, n$, 重新改写式(13)为

$$\begin{cases} \dot{y}_i = F(y_i) + c \sum_{j=1}^n a_{ij}(k) H(y_j), \\ i=1, 2, \dots, n, k=s(t), A(k) \in \Omega. \end{cases} \quad (14)$$

这是一个带有连续状态 y_i 和离散状态 $A(k)$ 的混合系统, 其中 $A(k)$ 属于切换耦合矩阵的有限集合 $\Omega \in \{A(k), k = s(t)\}$. 令 $s(t) : \mathbb{R}_{\geq 0} \rightarrow \Lambda_\Omega$ 表示切换信号, 其中 $\Lambda_\Omega \subset \mathbb{N}$ 表示与 Ω 中元素相应的索引集合.

注意到当机器人编队达到同步状态(12)时, 机器人之间的距离也是保持不变的. 令 $q_i = p_i^d - p_1^d$ 表示机器人 i 和1之间的期望相对距离. 设 $p'_i = p_i - q_i$, 那么状态 p'_i 在稳定状态时也达到同步:

$$p'_1(t) = p'_2(t) = \cdots = p'_n(t) = p_1(t), t \rightarrow \infty. \quad (15)$$

令 $Y(t) = [p_1(t) \ v(t)] \in \mathbb{R}^{2N}$ 和 $y'_i = [p_i - q_i \ x_i]^T \in \mathbb{R}^{2N}$. 这样, 编队(14)的同步状态变为

$$y'_1(t) = y'_2(t) = \cdots = y'_n(t) = Y(t), t \rightarrow \infty. \quad (16)$$

注意 q_i 仅简化了上述分析, 而没有影响自修复算法的分布式特性和模型(14).

假设编队中没有孤立的机器人, 也就是说连接矩阵 A 是对称的且不可约的. 这样, 矩阵 A 的特征值由

零和负数组成^[9]:

$$0 = \lambda_1 > \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_n, k = s(t). \quad (17)$$

令 $\omega_i(t) = y_i(t) - Y(t)$, $i = 1, 2, \cdots, n$, 并关于 $Y(t)$ 对式(14)进行线性化, 可得^[9]

$$\begin{cases} \dot{\omega}_i = [DF(s) + c\lambda_i(k)DH(s)]\omega_i, \\ i = 2, \cdots, n, k = s(t). \end{cases} \quad (18)$$

其中 $DF(s)$ 和 $DH(s)$ 分别为 $F(s)$ 和 $H(s)$ 关于 s 的Jacobian阵. 编队(14)的一个常用稳定性判据是要求公式(18)的横截Lyapunov指数^[16]全为负数. 对于每个 $\lambda_i(k)$, $i = 2, \cdots, n$, $k = s(t)$, 式(18)中对应的横截Lyapunov指数是 $\lambda_i(k)$ 的函数, 记作 $L_j(\lambda_i(k))$, $j = 1, 2, \cdots, N$.

为保证编队自修复过程中切合拓扑下系统稳定性, 切换时间应满足下面假设.

假设3 编队每次切换拓扑的时间足够长^[17], 因为在自修复算法中编队花费决定时间 t_{di} , $i = 1, 2, \cdots, k_s$ 来收集信息.

定理1 考虑移动机器人编队(14). 当机器人发生缺失时, 编队进行递归自修复. 那么整个编队达到稳定状态, 如果满足

$$L_j(\lambda_i(k)) < 0, i = 2, \cdots, n, k = s(t). \quad (19)$$

证 为了达到稳定状态, 要求对于每个 $\lambda_i(k)$, $i = 2, \cdots, n$, $k = s(t)$, $L_j(\lambda_i(k))$ 均为负数^[11], 即 $L_j(\lambda_i(k)) < 0, i = 2, \cdots, n, k = s(t)$.

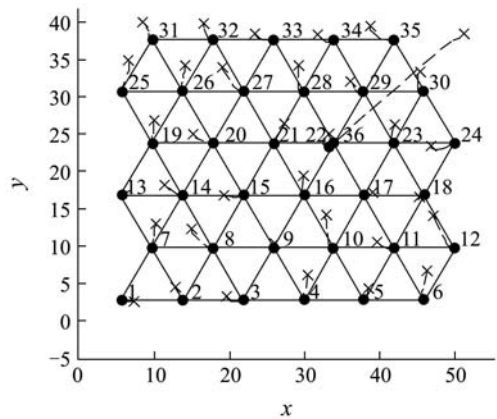
模型(1)(2)和(4)中, 计算可得 $DF(s) = -a_m < 0$ 和 $DH(s) = 1$. 又根据引理1和2, 可得: $\lambda_i(k) \leq 0, i = 2, \cdots, n, k = s(t)$. 那么每步自修复切换状态都是稳定的, 即 $L_j(\lambda_i(k)) < 0, i = 2, \cdots, n, k = s(t)$.

由于编队在每个切换状态都是稳定的, 且集合 Ω 是有限的, 切换时间足够长, 则整个编队稳定.

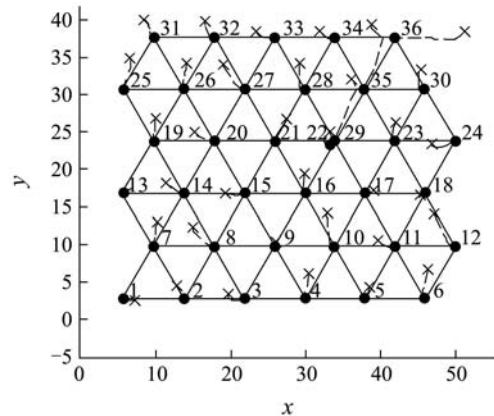
6 仿真(Simulation)

采用MATLAB程序对编队自修复拓扑控制进行仿真. 机器人工作空间为平面 $\mathbb{R}^2$. 仿真步长为0.05 s. 仿真中采用 K 邻居模型, $K = 6$. 针对编队任务, 比较直接自修复和递归自修复方法. 仿真中采用规模为 $n = 36$ 的移动机器人编队. 在 $t = 1.0$ s时刻机器人22缺失. 整个系统在每个切换状态是一个线性系统(2)和(4), 其中 $a_m = 1$ 和 $v(t) = 0$. 根据稳定性判据, 个体机器人的最大Lyapunov指数为0, 那么满足 $c > 0$, 整个系统将达到稳定. 仿真时间为6 s. 仿真结果见图4~7

图4中, (a)和(b)分别表示采用直接自修复^[4]和递归自修复方法的编队结果, 其中“ $\times$ ”表示机器人的初始位置, “ $\cdot$ ”表示最终位置, 实线表示机器人之间的通信连接, 虚线表示机器人的运动轨迹. 机器人的初始位置在编队目标位置为中心、 $[-5, 5]$ 为区间中随机选取, 初始速度在区间 $[-2.5, 2.5]$ 中随机选取. 两种方法的初始条件是相同的. 虽然两种方法最终的自修复结果(网络拓扑结构)是相同的, 直接自修复方法中填补机器人36的轨迹长度要远远大于递归自修复方法中的任一填补机器人. 所以, 图5所显示的递归自修复方法队形函数的恢复时间(虚线)要明显小于直接自修复方法(实线), 其中队形函数取作(5). 在通信功率公式(7)中, 本文取 $\alpha = 2$. 图6显示了两种方法的系统编队总功率关于时间的变化, 在修复过程中采用递归自修复的系统总功率(虚线)要小于采用直接自修复的系统(实线), 因为前者的通信距离要明显小于后者.



(a) 直接自修复结果



(b) 递归自修复结果

图4 队形生成任务直接自修复和递归自修复的仿真结果
Fig. 4 Simulation results of formation generation for direct self-healing and recursive one

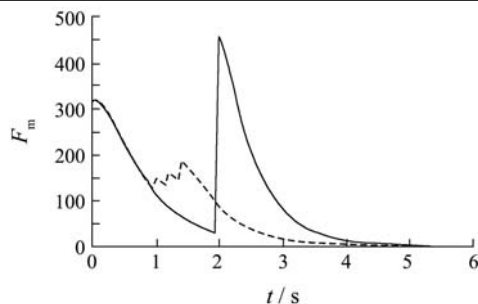


图5 队形函数随时间变化曲线

Fig. 5 Formation function vs time

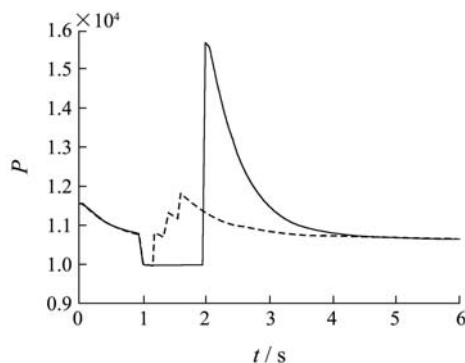


图6 网络总功率随时间变化曲线

Fig. 6 Total power consumption of network vs time

图7显示了两种方法耦合矩阵第二大特征值的变化情况, 观察可知递归自修复系统拓扑(虚线)的最终结果与直接自修复系统(实线)相同, 但是它分成了3步变化. 由此可见, 递归自修复方法是有效的, 且相对直接自修复而言其自愈性较高.

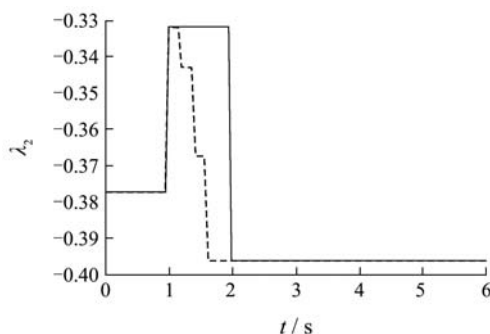


图7 耦合矩阵第二大特征值随时间变化曲线

Fig. 7 Second-largest eigenvalue of coupling matrix vs time

7 结论(Conclusion)

针对移动机器人编队自修复问题, 首先建立了结合切合拓扑结构和交互动力模型的移动机器人编队模型, 通过分析机器人缺失的拓扑变化情况, 提出了递归自修复的网络拓扑控制算法, 并将上述算法转化为基于局部交互的个体控制算法, 从而实现了算法的分布式特性. 这种从系统到个体的控制设计方法为解决大规模群体全局目标和个体局部控制的矛盾提供了一般思路. 最后针对编队任务, 通过仿真验证了递归自修复控制算法的有效性, 和其他方法比较具有较高的自愈性.

参考文献(References):

- [1] FAX J A, MURRAY R M. Information flow and cooperative control of vehicle formations[J]. *IEEE Transactions on Automatic Control*, 2004, 49(9): 1465 – 1476.
- [2] ZHENG Y F, CHEN W D. Robot team forming for crystallization of proteins[J]. *Autonomous Robots*, 2007, 23(1): 69 – 78.
- [3] ZHANG F, CHEN W D, XI Y G. Motion synchronization of mobile robot networks: robustness[C] // *IEEE/RSS Conference on Intelligent Robots and Systems*. Beijing: IEEE, 2006: 5570 – 5575.
- [4] TOMITA K, MURATA S, KUROKAWA H, et al. Self-assemble and self-repair method for a distributed mechanical system[J]. *IEEE Transactions on Robotics and Automation*, 1999, 15(6): 1035 – 1045.
- [5] GANERIWAL S, KANSAL A, SRIVASTAVA M B. Self aware actuation for fault repair in sensor networks[C] // *Proceedings of IEEE Conference on Robotics and Automation*. New Orleans: IEEE, 2004, 5: 5244 – 5249.
- [6] CHENG J, CHENG W, NAGPAL R. Robust and self-repairing formation control for swarms of mobile agents[C] // *Proceedings of 20th National Conference on Artificial Intelligence*. Menlo Park: AAAI Press, 2005: 59 – 64.
- [7] ZHANG F, CHEN W D. Self-healing for mobile robot networks with synchronization[C] // *Proceedings of IEEE/RSS Conference on Intelligent Robots and Systems*. San Diego: IEEE, 2007: 3107 – 3112.
- [8] BOONMA P, CHAMPASERT P, SUZUKI J. BiNET: A biologically-inspired architecture for wireless sensor network[C] // *Proceedings of IEEE International Conference on Autonomic and Autonomous Systems*. California: IEEE, 2006: 54 – 60.
- [9] LI X, CHEN G. Synchronization and desynchronization of complex dynamical networks: an engineering viewpoint[J]. *IEEE Transactions on Circuits System I*, 2003, 50(11): 1381 – 1390.
- [10] SHAW E. The schooling of fishes[J]. *Scientific American*, 1962, 206(6): 128 – 138.
- [11] GAZI V, PASSINO K M. Stability analysis of swarms[J]. *IEEE Transactions on Automatic Control*, 2003, 48(4): 692 – 697.
- [12] OLFATI-SABER R, MURRAY R M. Consensus problems in networks of agents with switching topology and time-delays[J]. *IEEE Transactions on Automatic Control*, 2004, 49(9): 1520 – 1533.
- [13] EGERSTEDT M, HU X. Formation constrained multi-agent control[J]. *IEEE Transactions on Robotics and Automation*, 2001, 17(6): 947 – 951.
- [14] PODURI S, SUKHATME G S. Constraint coverage for mobile sensor networks [C] // *Proceedings of IEEE Conference on Robotics and Automation*. New Orleans: IEEE, 2004: 165 – 172.
- [15] WANG X F, CHEN G. Synchronization in scale-free dynamical networks: Robustness and fragility[J]. *IEEE Transactions on Circuits System I*, 2002, 49(1): 54 – 62.
- [16] RANGATAJAN G, DING M Z. Stability of synchronized chaos in coupled dynamical systems[J]. *Physics Letters A*, 2002, 296(4/5): 204 – 209.
- [17] BRANICKY M S. Multiple Lyapunov functions and other analysis tools for switched and hybrid systems[J]. *IEEE Transactions on Automatic Control*, 1998, 43(4): 475 – 482.

作者简介:

张 飞 (1980—), 男, 博士研究生, 研究方向为多机器人系统;

陈卫东 (1968—), 男, 教授, 博士生导师, 机器人与智能信息处理研究所所长, 主要研究方向为自主移动机器人、多机器人系统等, E-mail: wdchen@sjtu.edu.cn.