

# 分布式多机器人运动控制的离散事件系统方法

周 远<sup>1</sup>, 胡核算<sup>1,2†</sup>, 刘 杨<sup>1</sup>, 林尚威<sup>1</sup>

(1. 南洋理工大学 计算机科学与技术学院, 新加坡 639798; 2. 西安电子科技大学 机电工程学院, 陕西 西安 710071)

**摘要:** 传统多机器人系统的运动控制主要依赖于机器人的动力学方程或运动学方程, 通过求解微分方程组来获得机器人的输入控制信号. 随着系统中机器人数量的增加和运行环境的复杂化, 动力学方程很难描述多机器人系统的运动行为, 且无法很好地解决诸如死锁等逻辑故障. 本文简略综述了国内外的研究现状, 重点介绍笔者所在研究组开展的关于离散事件系统方法在多机器人运动控制方面的应用性研究工作. 其动机在于: 1) 基于离散事件系统方法的运动控制能够有效地解决系统运行过程中产生的诸如死锁等逻辑故障. 首先, 利用离散事件系统模型对多机器人系统的运动进行建模, 从而降低计算复杂性; 其次, 基于所得离散事件系统模型, 设计分布式安全运动控制算法, 使各个机器人可以自主地、无碰撞地、无死锁地运动; 设计分布式鲁棒运动控制算法, 使得失效的机器人对系统的影响最小. 2) 基于离散事件系统方法的运动控制策略可以结合传统的基于运动学方程的运动控制方法, 从而使系统不但能够避免顶层的逻辑故障, 而且能够确定机器人执行器的输入信号.

**关键词:** 多机器人系统; 运动控制; 离散事件系统; 分布式算法

**引用格式:** 周远, 胡核算, 刘杨, 等. 分布式多机器人运动控制的离散事件系统方法. 控制理论与应用, 2018, 35(1): 110–120

**中图分类号:** TP273      **文献标识码:** A

## Distributed approaches to motion control of multiple robots via discrete event systems

ZHOU Yuan<sup>1</sup>, HU He-suan<sup>1,2†</sup>, LIU Yang<sup>1</sup>, LIN Shang-wei<sup>1</sup>

(1. School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, Singapore;  
2. School of Electro-Mechanical Engineering, Xidian University, Xi'an Shaanxi 710071, China)

**Abstract:** Classical motion control of multi-robot systems is dependent on the dynamic or kinematic equations, where the input control signals for robots are obtained by solving a set of differential equations. With the increase of the number of robots in a system and the more complexity of environment, it is hard to describe the behavior of robots exactly by only applying the dynamic equations and cannot deal well with some logical problems, such as deadlocks. This paper summarizes the state-of-the-art motion control technologies, especially emphasizes our research work with regard to the application of discrete event systems (DESs) on the motion control of multi-robot systems. The motivations are as follows. 1) DES-based motion control can deal with the logical problems, such as deadlocks, during the evolution of the system. First, use the DES models to formalize the motion of a multi-robot system; this can reduce the computation complexity. Second, based on the DES models, design distributed safe control algorithms for robots to avoid collisions and deadlocks, as well as guarantee that they can move independently; design distributed robust control algorithms such that the failures of robots have the least detrimental effect on the system. 2) DES-based motion control can be combined with the kinematic equation-based motion control such that the system can not only resolve high-level logical problems but also determine the input signals of the actuators of robots.

**Key words:** multi-robot systems; motion control; discrete event systems; distributed algorithms

**Citation:** ZHOU Yuan, HU He-suan, LIU Yang, et al. Distributed approaches to motion control of multiple robots via discrete event systems. *Control Theory & Applications*, 2018, 35(1): 110–120

## 1 引言(Introduction)

随着科学技术的发展, 移动机器人的应用越来越普遍, 人们研制了可在各种环境下作业的移动机器人

系统, 如无人驾驶飞机(unmanned aerial vehicle, UAV), 无人驾驶汽车(unmanned ground vehicle, UGV), 无人潜航器(unmanned underwater vehicle, UUV)等. 同时,

收稿日期: 2016–12–04; 录用日期: 2017–09–04.

†通信作者. E-mail: huhesuan@gmail.com; Tel.: +86 15291862950.

本文责任编辑: 奚宏生.

国家自然科学基金项目(61573265, 61203037, 51305321), 新加坡教育部Tier 1科研基金(2014–T1–001–147), 新加坡教育部Tier 2科研基金(MOE 2015–T2–2–049)资助.

Supported by the National Natural Science Foundation of China (61573265, 61203037, 51305321), the Academic Research Fund Tier 1 by Ministry of Education in Singapore (2014–T1–001–147) and the Academic Research Fund Tier 2 by Ministry of Education in Singapore (MOE2015–T2–2–049).

随着工业化的进展, 诸如自动制造、柔性生产、搜索营救、环境监测、安全健康等许多领域面临着大量操作复杂、规模庞大的任务. 因此, 单一的机器人已经无法很好地完成这些任务. 相比单机机器人系统, 多机器人系统由于多个机器人的相互协作而具有一系列显著的优点, 例如多机器人系统可以降低任务求解的复杂性、促进任务完成的高效性、增加系统的可靠性、简化系统的设计等<sup>[1]</sup>. 由于这些优越的特性, 多机器人系统越来越受到人们的关注, 吸引众多学者对其进行广泛而深入的研究. 例如, 美国工程院院士、宾夕法尼亚大学教授Vijay Kumar及其领导的研究小组在自主地面机器人和空中机器人的制造、机器人集群系统以及群集行为等领域进行了深入研究<sup>1</sup>, 例如文献[2-7]; 美国工程院院士、MIT教授Daniela Rus领导的研究小组致力于研发自配置、自组织机器人、机器人网络、协作潜航器系统等<sup>2</sup>, 如文献[8-14]. 同时, 机器人领域的两个国际顶级会议, ICRA(IEEE International Conference on Robotics and Automation)和IROS(IEEE/RSJ International Conference on Intelligent Robots and Systems), 每年都有大量关于多机器人系统的研究工作. 这些会议同时展示了大量工业界的相关应用, 为学术界和工业界的交流互动提供了良好而广泛的平台.

多机器人系统的运动控制是其中一个至关重要的研究问题. 从方法上看, 传统机器人的运动控制通常依赖于机器人的动力学方程(如速度、加速度、力矩等). 然而随着系统中机器人数量的增加, 基于动力学方程的系统模型将变得高度耦合、高度复杂, 从而使得系统不易求解. 同时, 当机器人运动在固定路径上时, 例如无人驾驶汽车行驶在固有的交通道路上时, 基于动力学方程的运动控制无法解决逻辑上的故障, 如死锁. 离散事件系统(discrete event system, DES)方法能够有效地解决这类问题, 如文献[15]. 其次, 当前的控制器结构主要可分为中心化(centralized)和分散化(decentralized)<sup>[16]</sup>. 中心化控制导致系统缺乏健壮性和可扩展性, 单个机器人的故障将会导致整个系统的故障, 且系统无法动态增加或减少机器人; 而分散化控制使系统中的各个机器人无法很好地进行协作.

本文的目的是介绍笔者团队如何利用离散事件系统方法来设计多机器人系统的运动控制策略. 因此, 它是一篇介绍性论文, 区别于一般的技术性论文和综述性论文. 文中首先概述了国内外关于机器人运动控制的代表性方法, 总结了他们的适用范围和优缺点; 然后详细介绍了通过离散事件系统方法来研究多机器人系统运动的思路 and 方向, 总结了当前获得的阶段性成果, 包括多机器人系统的碰撞和死锁避免分布式

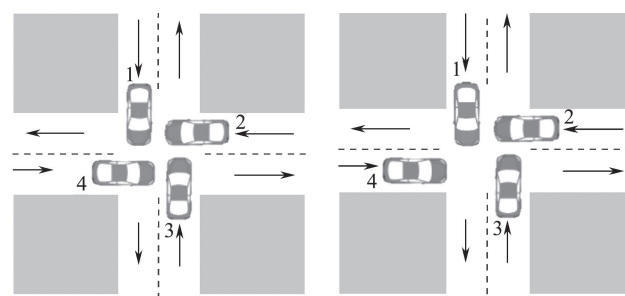
实时控制算法、鲁棒性实时控制算法.

## 2 问题描述(Problem statement)

多机器人系统可以通过多个简单的机器人的协作来完成复杂任务. 为完成相应任务, 机器人需要在不同环境中运动. 因此, 机器人运动规划研究的是如何确定机器人运动中的最优安全运动轨迹, 即每个机器人在运动过程中不会发生碰撞. 根据机器人可部署的轨迹情况, 多机器人系统可分为自由轨迹(trajjectory)系统和固定路径(path)系统.

前者适用于无结构化环境场景, 例如智能营救搜索. 在这类系统中, 每个机器人在给定目标位置后, 需要确定从初始位置到目标位置的最优安全运动轨迹. 尽管该问题的表述简单, 它是一个PSPACE-难问题<sup>[17-18]</sup>. 这意味着无法找到一个合适的、通用的算法. 因此, 人们当前主要致力于对部分运动规划问题的近似求解算法的探索, 提出了一些具有实际复杂度可行的近似算法<sup>[19-20]</sup>.

后者大量存在于智能交通系统中, 例如无人驾驶汽车需要在固定的道路上运行. 在这类场景中, 通常需保证路径上无静态障碍物. 因此, 任一个机器人在其运动过程中只需规避动态障碍物, 如系统中的其他机器人. 然而在这类系统中, 不恰当的碰撞避免控制将导致一些诸如死锁和不鲁棒等逻辑问题. 例如, 在图1中, 左图显示4辆UGV为避免碰撞而产生死锁; 右图显示车辆1的失效阻塞了整个系统的运动. 这类逻辑问题很难通过运动学方程来提前预测和规避.



(a) 四辆正常行驶的UGV 出现死锁 (b) 汽车1的失效阻塞了整个系统

图 1 不恰当的运动控制导致系统阻塞

Fig. 1 The inappropriate motion control causes the stagnation of the system

因此, 考虑一个多机器人系统, 其运动控制需实现的控制目标如图2所示. 碰撞规避是指机器人在运行过程中不能和环境中的障碍物以及其他机器人发生碰撞; 死锁规避是指机器人在运行过程中不会出现一组机器人为避免碰撞而产生循环等待的情况; 鲁棒性控制是指在系统运行过程中失效机器人对系统的危害

<sup>1</sup><https://www.kumarrobotics.org/research/>

<sup>2</sup><http://danielarus.csail.mit.edu/index.php/about-daniela-2/research-2/>

性最低,即失效机器人只会影响和它直接交互的机器人,而不会干扰其他机器人的运动.这3类目标关乎系统能否安全正常运行,是必然目标,在任何情况下都必须得到保障.最后一个是性能优化目标,即在系统运行中需要尽可能满足其他的优化目标,比如运行时间最短、运行路程最短、运行能耗最少等.此类目标是可选目标,是在上述3类安全目标满足的情况下对系统运行的进一步改进和优化.

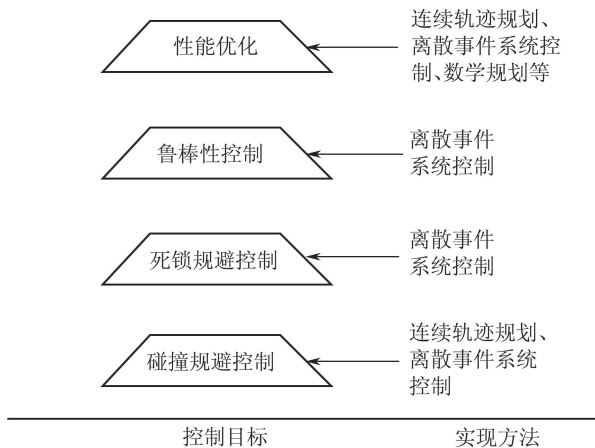


图2 多机器人系统的运动控制目标

Fig. 2 The objectives for the motion control of multi-robot systems

### 3 国内外研究现状(The state of the art)

在过去几十年里,研究者针对机器人的运动控制提出了许多控制算法以及相关的一些综述和总结<sup>[19-21]</sup>.这些方法大致可分为状态空间导向和机器人导向.前者通过分析状态空间来获得机器人运动的路径序列或者事件序列,但不涉及机器人的具体运动参数,如速度、加速度.后者根据机器人系统本身的行为来确定机器人的运动轨迹,包括位置、速度、加速度以及作用力等.

通过分析状态空间来部署机器人运动轨迹的方法主要有如下4类.第1种方法是状态空间离散化.这类方法将状态空间离散为一系列不重叠的区域,然后通过搜索算法来确定机器人运行的区域序列,如单元分解法(cell decomposition methods)<sup>[22-23]</sup>、状态格法(state lattices methods)<sup>[24-26]</sup>以及形式化方法(formal methods)<sup>[27-29]</sup>.单元分解法将状态空间分解为多个不重叠的无障碍物的区域,因此运动控制只需保证机器人在这些离散区域运行即可.传统的单元分解法如四边形分解法通常假设环境中的障碍物是多边形的<sup>[30]</sup>.考虑非多边形障碍物和球形机器人,Acar等在文献<sup>[23]</sup>中提出了一种莫尔斯分解的单元分解法,其中每个单元的边界可由莫尔斯函数的临界点来表示.状态格法通过将状态空间进行分解确定网格点,然后根据机器人的运动基元来确定网格点的网络连接图,从而

获得状态格.根据获得的状态格,机器人的运动路径可以通过图搜索算法来确定.有限状态机法根据状态空间构造用以描述机器人运动的状态机,然后利用诸如模型检查的方法来确定满足一定约束条件的执行序列.这类方法的不足在于需要事先知道全局环境并对其离散化.第2种方法是路标采样法.这类方法通过在运行环境中设置合适路标来部署机器人的运动轨迹,如路径图法(roadmap methods)<sup>[31-35]</sup>和随机采样法(random sampling methods)<sup>[35-40]</sup>.路径图法通常先构造无碰撞的路径图,然后通过搜索该路径图来确定运行轨迹.为了降低构造路径图的复杂度,有时也用随机采样法确定路径图中的各个路标<sup>[35]</sup>.随机采样法通过在状态空间中随机采样来进行路径规划,其中两种主要的采样方法有概率路径图法和快速搜索随机树.它们的区别在于概率路径图法需要构建路径图,而快速搜索随机树法(rapidly-exploring random tree)<sup>[38]</sup>是单次搜索法,它直接通过采样来确定机器人的后继路标,而无需构造路径图.准确的路径图法因性能原因而无法适用于复杂高维环境,而随机采样法的收敛速度无法保证,且碰撞检测需花费大量时间.因此,研究者也提出一些改进方法来降低样本检测时间,提高采样效率,如文<sup>[34,37]</sup>.第3种方法是Bug法<sup>[20,41]</sup>.在任何时候,Bug法将控制机器人朝目标位置作直线运动,期间一旦遇到障碍物,机器人将沿着障碍物边缘运动来绕过该障碍物,而后朝目标位置继续直线运动.Bug法简单易操作,然而无法对路径进行有效优化.第4种方法是人工势场法(artificial potential field methods)<sup>[42]</sup>.人工势场法通过引入障碍物的排斥函数和目标点的吸引函数来确定机器人的运动方向.然而对于拥挤复杂环境,人工势场法往往会陷入局部最小化.

通过机器人本身的运动学方程和系统约束来直接获取机器人的运动控制信号的方法主要有相对碰撞避免法(reciprocal collision avoidance methods)<sup>[43-44]</sup>、数学规划法(mathematical programming methods)<sup>[45-51]</sup>以及几何轨线法<sup>[52-53]</sup>.相对碰撞避免法通过搜索相对速度空间来确定机器人的速度,保证在此速度下该机器人不会和其他运动物体发生碰撞.数学规划法通过给定优化目标以及机器人的运动约束,包括机器人本身的动力学约束和无碰撞约束,构建和求解数学规划问题,如整数规划、混合整数规划、二次规划、凸规划等,从而获得机器人的运动控制信号.几何曲线法通过特定的几何曲线,如 $\beta$ 样条曲线<sup>[52]</sup>和贝塞尔曲线<sup>[53]</sup>,来近似机器人的运动学约束,并通过确定曲线的参数和机器人的运行速度来进行运动规划.

上述运动规划算法的适用前提是每个机器人可以自由地规划路径,即自由轨迹系统.然而,在诸如交通系统中,每个机器人的运动路径是固定的或者是有限



的. 因此, 上述轨迹规划算法无法继续使用. 针对这种场景, 碰撞避免的方法主要是通过优化各个机器人的运动时间来确保不同机器人在不同的时间通过相同的位置<sup>[54-56]</sup>. 其次, 当固定机器人运行路径后, 另一个重要的问题是在避免碰撞的过程中可能会发生死锁问题, 从而导致整个系统崩溃. 针对死锁问题, 研究者进行了广泛的研究, 提出了许多切实可行的技术, 如胡核算等通过在自动制造系统的Petri网模型中查找关键库所来预测和避免死锁状态的出现<sup>[57]</sup>. 文献 [58]对相关工作进行了详细的总结. 针对多机器人系统中的死锁问题, 研究者也进行了一些研究<sup>[59-62]</sup>. 传统的死锁规避采用的是中心化控制算法, 通过遍历整个系统的状态空间或者系统结构来有效地避免所有可能的死锁状态<sup>[59-60]</sup>. 然而遍历整个系统的状态空间的时间以及空间计算复杂度都是指数规模的. Reveliotis等<sup>[61-62]</sup>提出了多机器人系统死锁避免的分散化控制策略, 即在当前可用资源条件下, 每个机器人通过判断是否存在足够资源供其到达最近的无碰撞状态来确定其能否迁移到下一个共享资源的状态. 当机器人的运行轨迹包含很多连续且共享资源的离散状态时, 该方法过于保守而影响系统性能.

总而言之, 中心化控制算法能够最大性能地避免系统中的所有死锁, 但是其扩展性差, 且随着机器人数量的增加和路径的复杂化, 系统将产生状态爆炸问题. 分散化控制算法能够有效地避免系统中的死锁而不产生状态爆炸, 但是由于各个机器人之间缺乏协作, 从而导致系统的性能因控制算法的保守性而降低. 因此, 提出了基于离散事件系统方法的多机器人系统的分布式控制算法. 基于离散事件系统方法的分布式控制算法, 一方面可以避免中心化控制中的状态爆炸问题, 另一方面通过机器人间的相互协商提高系统的性能. 接下来将介绍当前已取得的一些研究成果.

#### 4 分布式实时运动控制算法设计(Design of distributed and real-time motion control algorithms)

离散事件系统的监督控制理论在处理诸如死锁等顶层逻辑问题上具有显著的优势和便利性. 但是, 离散事件系统的监督控制理论获得的是一个可执行的事件序列, 无法产生可直接作用于机器人的输入控制信号. 因此, 在离散事件系统方法的基础上, 还需考虑机器人的连续轨迹控制. 同时, 为满足上述两类不同的场景, 拟提出一个统一的混合运动控制架构, 同时包含离散事件控制和连续运动控制, 如图3所示.

该架构包含3个不同抽象程度的运动控制层面: 顶层是抽象程度最高的离散事件系统方法控制, 它利用离散事件系统模型对多机器人系统的运动进行建模, 然后获得激活事件序列, 使得系统在该事件序列下的

状态变迁不会产生碰撞、死锁、阻塞等问题; 底层是物理层的基于运动学方程的轨迹规划, 它通过对运动学方程的优化求解, 获得最优轨迹, 使得各机器人的运动不会产生碰撞, 同时能优化给定的目标; 中间层的混合控制层通过离散事件系统控制和轨迹规划的结合, 既能获得无逻辑故障的事件序列, 又能确定底层的输入控制信号以实现机器人在各状态的具体物理运动. 机器人可以根据其需求自主决定以何种方式执行运动控制. 如当机器人在固定轨迹上运动时, 基于离散事件系统方法的控制方法被选择用来控制机器人的运动; 当机器人进行短距离的轨迹规划时, 连续轨迹方法因能够获得优化路径而被用来规划运动路径; 当机器人需要在无固定路径的不同区域环境中进行循环运动时, 混合控制方法首先通过离散方法获得区域变迁序列, 然后针对每个区域确定最优路径, 从而既解决死锁等问题, 又保证获得较优路径. 针对这三个层面, 具体研究内容如下.

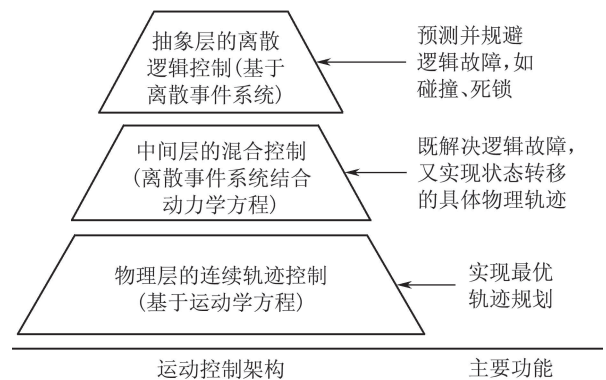


图 3 多机器人系统的运动控制架构

Fig. 3 The control architecture for the motion planning of multi-robot systems

#### 4.1 离散事件系统方法的运动控制研究(Motion control based on discrete event systems)

首先, 需对机器人的运动进行离散化建模. 这里使用标签变迁系统(labeled transition system, LTS)进行建模. 具体地, 假设有两个机器人 $r_i$ 和 $r_j$ , 它们的路径为 $P^i$ 和 $P^j$ , 安全半径为 $\rho$ .  $r_i$ 能与 $r_j$ 发生碰撞的路径可表示为一组互不相交的路径片段, 即 $CR^{i:j} = P^i \cap \{y \mid \|y - x\|_2 < 2\rho, x \in P^j\}$ . 同理,  $r_j$ 关于 $r_i$ 的碰撞片段为 $CR^{j:i} = P^j \cap \{y \mid \|y - x\|_2 < 2\rho, x \in P^i\}$ .  $CR^{i:j}$ 和 $CR^{j:i}$ 中的片段是一一对应的, 即对任意的片段 $seg_k^i \in CR^{i:j}$ , 存在唯一的 $seg_{k'}^j$ , 使得 $d(seg_k^i, seg_{k'}^j) < 2\rho$ . 根据以上所述,  $r_i$ 的所有碰撞片段为 $CR^i = \cup_j CR^{i:j}$ , 其中任意相邻的两个碰撞片段不会和同一个机器人发生碰撞;  $P^i \setminus CR^i$ 包含 $r_i$ 的无冲突片段. 因此, 根据安全半径和实时通信获得的其他机器人的路径, 每个机器人 $r_i$ 可以将自己的路径离散为一组路径片段. 每个碰撞片段抽象为一个离散状态, 称为冲

突状态;每个无碰撞片段亦离散为一个状态,称为无冲突状态.因此,可以获得机器人运动的离散状态空间.其次,每个机器人或者暂停在当前状态,或者从当前位置沿运动方向移动到邻近的路径片段,即从当前状态迁移到唯一的后继状态.因此,机器人 $r_i$ 的LTS模型为

$$\mathcal{T}_i = \langle S^i, \Sigma_i = \{\text{move}, \text{stop}\}, \rightarrow_i \rangle,$$

其中 $S^i$ 和 $\rightarrow_i$ 分别为 $r_i$ 的状态空间和状态变迁集合.

例如,图4所示的系统中包含三个机器人 $r_1 - r_3$ ,它们的路径分别是 $P^1 - P^3$ .通过观测其他机器人的路径,每个机器人可以获得其路径上的碰撞路径片段,如图4(a);基于碰撞路径片段,机器人可以构建对应的离散状态空间,如图4(b);最后,机器人根据其运动方向构建LTS模型,图4(c)给出了 $r_2$ 的LTS模型.

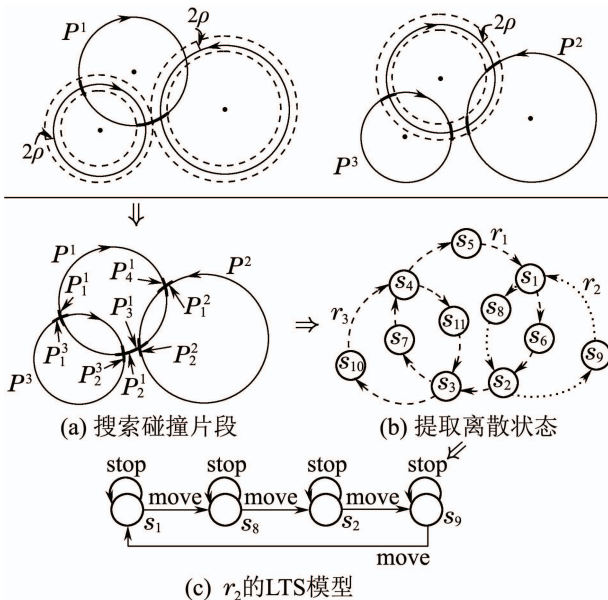


图4 机器人运动的LTS模型的建立过程

Fig. 4 The process to build LTS models

基于LTS模型,提出了碰撞和死锁规避的分布式实时控制算法,以及鲁棒控制算法.这些算法是MPC(model predictive control)思想在离散事件系统中的创新性应用.传统的MPC以时间作为滚动域(receding horizon)来进行优化预测,而基于离散事件系统方法的MPC以状态作为滚动域进行实时预测.

#### 4.1.1 碰撞和死锁规避控制算法(Collision and deadlock avoidance algorithms)

基于LTS的碰撞避免指的是任何时刻每个状态至多只有一个机器人.因此,单个机器人通过检测其直接后继状态来确定它的一步迁移是否会产生碰撞.为了确保互斥(mutual exclusion)性,即在同一时刻只有一个机器人可以迁移到一个相同的空闲状态,需迁移到同一个状态(或者状态区域)的不同机器人通过协商来确定最终可以迁移的机器人.

由于每个机器人自主地进行碰撞避免决策,多个机器人之间可能会出现死锁.如图5(a)所示,为避免碰撞, $r_1$ 需要停留在 $s_1$ 并等待 $r_2$ 离开 $s_2$ ;  $r_2$ 停留在 $s_2$ 并等待 $r_3$ 离开 $s_3$ ;同时, $r_3$ 停留在 $s_3$ 以等待 $r_1$ 离开 $s_1$ .因此, $r_1 - r_3$ 因循环等待而陷入死锁状态.一组机器人构成死锁状态指的是多个机器人因循环等待而无法迁移的状况.死锁检测的难点在于系统中存在高阶死锁.所谓高阶死锁,指的是系统当前状态是活的,然而随着系统的演化,在有限次迁移之后必然产生死锁状态.一组机器人 $\mathcal{R}$ 构成一阶死锁如果其中任何一个可迁移的机器人 $r$ 迁移一步后,都存在一组机器人 $\mathcal{R}'_r, \mathcal{R}'_r \subset \mathcal{R}$ ,使得 $\{r\} \cup \mathcal{R}'_r$ 构成一个死锁状态.递归地,一组机器人 $\mathcal{R}$ 构成 $k$ 阶死锁如果: 1) 其中任何一个可迁移机器人迁移一步后将和这组机器人的其他机器人(部分)构成更低阶的死锁; 2) 存在一个可迁移的机器人,它的一步迁移将和这组机器人中的其他机器人(部分)构成 $k - 1$ 阶死锁.图5(b)展示了一个2阶死锁.显然,唯一可迁移的机器人为 $r_3$ 和 $r_4$ .其中, $r_3$ 迁移一步将构成一个1阶死锁状态,而 $r_4$ 迁移一步将构成一个死锁状态.

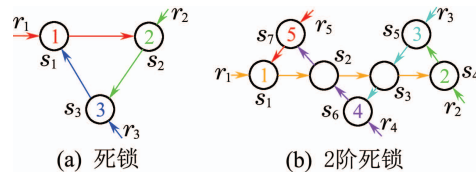


图5 死锁和高阶死锁

Fig. 5 Deadlocks and higher-order deadlocks

针对此类系统中的高阶死锁问题,笔者提出了分布式实时死锁检测方法.每个机器人通过搜索自己路径上的部分状态,并通过和其他邻近机器人通信来预测和预防死锁和高阶死锁.该方法有如下优点: 1) 能够预测高阶死锁; 2) 每个机器人只需搜索各自状态空间的部分状态,且搜索的状态数最少; 3) 分布式算法,每个机器人自主进行判断,无需搜索系统的全局状态空间,从而避免了状态爆炸问题.下面将给出详细的算法设计.

首先,根据多机器人系统的特殊性,研究了高阶死锁的性质.

- 1) 如果一组机器人构成了高阶死锁状态,则这组机器人的当前状态构成了一个闭合回路.
  - 2) 如果一组机器人构成了高阶死锁状态,则构成闭合回路的状态都是冲突状态,且这些冲突状态均属于这组机器人的状态空间.
  - 3)  $n$ 个机器人最多可构成 $n - 3$ 阶死锁状态.
  - 4)  $k$ 阶死锁构成的闭合回路中的冲突状态个数为 $k$ .这里计数的状态不包含各个机器人的当前状态.
- 根据上述性质,如果一个系统包含 $n$ 个机器人,则



每个机器人在任何时刻最多只需往前检索  $n - 1$  个状态, 其中直接后继状态用以判断是否会发生碰撞, 剩余  $n - 2$  个状态用来判断当其位于直接后继状态时是否会产生高阶死锁. 如果在这  $n - 1$  状态中存在无冲突状态, 则只需判断从当前状态到第一个无冲突状态之间的状态序列. 如果这些检测的状态都是空闲的, 即它们不被其他机器人占据, 则无高阶死锁, 所以可以向前迁移一步; 否则需进行死锁检测. 每个机器人通过和其他机器人通信进行死锁检测. 下面以  $r_i$  为例介绍死锁检测和避免过程.

**Step 1** 搜索闭合回路. 该过程可以通过  $r_i$  启动机器人之间的相互通信来实现.

**Step 1.1** 机器人  $r_i$  向前搜索最多  $n - 1$  个连续冲突状态. 假设搜索到的位于这些状态的机器人集合为  $\mathcal{R}_{r_i}$ . 如果  $\mathcal{R}_{r_i} = \emptyset$ , 则无死锁,  $r_i$  可迁移; 否则对任意的  $r_{j_1} \in \mathcal{R}_{r_i}$ , 记录从  $r_i$  到  $r_{j_1}$  的状态序列, 记为  $w_{ij_1}$ .  $r_i$  将信息  $(r_{j_1}, \langle r_i, s_i \rangle, RW, \text{Index})$  发送给  $r_{j_1}$ , 其中  $RW = \{w_{ij_1}\}$ ,  $\text{Index} = \mathcal{R}$ .

**Step 1.2** 当  $r_{j_1}$  接收到消息后,  $r_{j_1}$  检索其路径上的最多  $n - 2$  个状态并判断是否有  $\text{Index}$  中的机器人在这些状态里. 假设检索到的机器人集合为  $\mathcal{R}_{r_{j_1}}$ . 如果  $\mathcal{R}_{r_{j_1}} = \emptyset$ , 则无回路, 将结果返回给  $r_i$  并结束; 否则, 对任意的  $r_{j_2} \in \mathcal{R}_{r_{j_1}}$ , 记录从  $r_{j_1}$  到  $r_{j_2}$  的状态序列, 记为  $w_{j_1j_2}$  并令  $RW = RW \cup \{w_{j_1j_2}\}$ . 如果  $r_{j_2} = r_i$ , 将  $RW$  返回给  $r_i$ , 结束; 否则,  $\text{Index} = \text{Index} \setminus \{j_1\}$ , 然后将消息  $(r_{j_2}, \langle r_i, s_i \rangle, RW, \text{Index})$  发送给  $r_{j_2}$ .

**Step 1.3** 当  $r_{j_2}$  接收到消息时,  $r_{j_2}$  检索其路径上的最多  $n - 2$  个状态并判断是否有  $\text{Index}$  的机器人位于这些状态里.  $r_{j_2}$  重复  $r_{j_1}$  的判断并将消息传递给下一组机器人.

**Step 1.4** 重复上述消息传递. 由于机器人的个数有限, 最终将会以如下两个情况之一结束:  $\mathcal{R}_{r_{j_m}} = \emptyset$ , 这表示无回路; 或者  $\mathcal{R}_{r_{j_m}} = \{r_i\}$ , 存在回路, 并将对应的回路返回给  $r_i$ .

**Step 2**  $r_i$  决定是否可以进行迁移. 这需要判断每个回路是否构成高阶死锁. 传统的检测方法需要遍历所有这个回路可能演化出的状态. 虽然搜索空间远远降低了, 但是时间代价依旧很高. 为此, 提出了一个相对保守的判断准则. 对任意一个回路  $RW$  进行如下两个阶段检测.

1) 统计回路中的机器人个数和中间状态. 如果中间状态的个数  $>$  机器人的个数  $- 3$ , 或者存在属于其他不在该回路中的机器人的状态, 则该回路不构成高阶死锁;

2) 如果 1) 里的条件不满足, 回路中的机器人自主判断一步迁移构成的回路是否满足阶段 1) 的条件, 如果满足, 则该回路不构成高阶死锁. 如果 1) 或 2) 满足,

$r_i$  可以迁移一步, 否则  $r_i$  不能迁移.

在该算法中,  $w_{ij}$  表示从  $r_i$  的当前状态到  $r_j$  的当前状态的一条路径, 且路径上所有状态都是  $r_i$  的冲突状态;  $RW$  为当前已获得的这些路径的集合;  $\text{Index}$  表示还未被搜索到的机器人.

在 **Step 1** 中, 每个机器人只需搜索前  $n - 2$  个状态, 并将相关信息发送给位于这些状态上的机器人; 收到通信请求的多个机器人可以同时进行搜索. 因此, 此过程的复杂度为  $O(n^2)$ . 图 6 给出了上述回路检索的流程. 由于各个机器人是自主检测的, 因此多个机器人可以进行同步检测. 也就是说, 任意机器人  $r_j$  可以同步地将对应的消息发送给  $\mathcal{R}_{r_j}$  里对应的机器人.

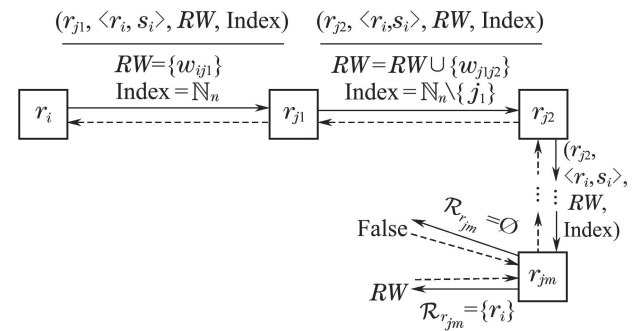


图 6 回路检索过程

Fig. 6 The process of circuit detection

在 **Step 2** 中,  $r_i$  需对所有的回路进行判断. 假设  $\mathcal{C}_i$  是  $r_i$  检测到的所有回路集合, 则死锁判断的复杂度为  $O(|\mathcal{C}_i|)$ . 当系统中存在大量连续的冲突状态时,  $r_i$  检测到的回路将会随着机器人数量的增加而增加. 但是在实际中, 一方面, 在一个多机器人系统中, 机器人的数量通常是有限的, 且很少会在系统运行的时候剧增; 另一方面, 一个合理的系统中, 机器人的运动路径不会频繁相交. 因此, 实际构成的回路个数往往不会太多, 这意味着该算法在实际中是可接受的.

#### 4.1.2 鲁棒控制算法(Robust control algorithms)

在上述研究中, 假设机器人一直能够完好地运动, 然而这通常与实际不合. 在实际中, 由于同一个系统里的机器人可能来自不同的生产厂家, 因此有的机器人可靠性高, 有的机器人可靠性低; 或者有的机器人需要经常和系统外的设备通信, 更容易被入侵而变得不可靠. 因此, 在一个系统中有的机器人是可靠的, 有的是不可靠的. 假设不可靠的机器人有可能在任意状态失效. 即便保证碰撞和死锁后, 失效的机器人依旧有可能阻塞整个系统. 因此, 多机器人系统还需鲁棒控制算法来抵抗不可靠机器人的失效. 不同于自动制造系统中假设部分资源是不可靠的<sup>[63-64]</sup>, 多机器人系统研究的是机器人(过程)的不可靠性.

鲁棒控制的目标为失效的机器人对系统的影响最小, 即在某状态  $s$  失效的机器人只能阻塞那些路径必

须通过状态 $s$ 的机器人. 首先, 定义几类关键路径和状态.  $r_i$ 的一个状态序列 $(s_1, \dots, s_m)$ 称为一个冲突状态域如果 $s_1 - s_m$ 都是冲突状态, 且 $s_1$ 的直接前继状态和 $s_m$ 的直接后继状态都是非冲突状态(分别称它们为前继关键状态和后继关键状态). 如果 $r_i$ 位于某个冲突状态, 则从该状态到对应的后继关键状态之间的状态序列称为关键冲突状态域. 不可靠机器人的状态空间中的冲突状态称为风险状态. 为保证系统的鲁棒性, 在任何时刻, 需要保证: 1) 任何一个机器人不能进入有不可靠机器人的冲突状态域; 否则, 不可靠机器人的失效将阻塞该机器人; 反过来, 该机器人又将阻塞其他机器人. 2) 不可靠机器人不能进入其他机器人的关键冲突状态域; 否则, 一旦不可靠机器人进入其他机器人的关键冲突状态域并失效, 它将阻塞这些机器人. 为了防止不可靠机器人进入其他机器人的关键冲突状态域, 引入了虚拟机器人, 它们只影响不可靠机器人的运动. 下面分别介绍可靠机器人和不可靠机器人的鲁棒控制算法.

首先, 给出可靠机器人 $r_i$ 的鲁棒控制算法.

**Case 1** 如果下一个状态为非冲突状态,  $r_i$ 迁移一步; 同时修正当前状态的信息.

**Case 2** 如果 $r_i$ 位于某个前继关键状态, 则:

**Step 1** 判断当前对应的冲突状态域里有无不可靠机器人: 如果存在不可靠机器人, 则不能迁移; 否则 $r_i$ 标记可迁移, 转Step 2;

**Step 2** 在所有可迁移进入该区域的机器人中确定最终可迁移的机器人. 如果 $r_i$ 赢得协商过程并迁移一步, 则在冲突状态域里的所有风险状态中设置对应的虚拟机器人.

**Case 3** 如果 $r_i$ 当前位于冲突状态, 且下一个状态无其他机器人,  $r_i$ 可迁移一步, 同时修改当前状态信息, 包括释放该状态里 $r_i$ 的虚拟机器人(如果存在).

其次, 给出不可靠机器人 $r_i$ 的鲁棒控制算法.

**Case 1** 如果下一个状态为非冲突状态,  $r_i$ 迁移一步; 同时修正当前状态的信息.

**Case 2** 如果 $r_i$ 位于某个前继关键状态, 则:

**Step 1** 判断是否有其他不可靠机器人在对应的冲突状态域里: 如果有, 则不迁移; 否则转Step 2;

**Step 2**  $r_i$ 检测其直接后继状态是否存在虚拟机器人: 如果有, 则不迁移; 否则 $r_i$ 标记可迁移, 转Step 3;

**Step 3** 所有可迁移进入该区域的机器人进行协商确定最终可迁移的机器人. 如果 $r_i$ 最终迁移一步, 则在冲突状态域中的所有风险状态设置对应的虚拟机器人.

**Case 3** 如果 $r_i$ 当前状态为冲突状态, 则:

**Step 1**  $r_i$ 检测其直接后继状态是否存在虚拟机

器人: 如果有, 则不迁移; 否则 $r_i$ 标记可迁移, 转Step 2;

**Step 2** 如果 $r_i$ 赢得协商过程, 则迁移一步, 同时修改当前状态信息, 包括释放该状态里 $r_i$ 的虚拟机器人(如果存在).

该算法中, 每个机器人只需检测从当前冲突状态到最近私有状态的路径上的各个状态信息, 并和这些状态上的机器人进行通信来判断它们是否为不可靠机器人, 从而决策它是否可以迁移. 在最坏情况下, 一个机器人需检索路径上的所有状态. 这意味着该机器人只有一个无冲突状态, 其余都是冲突状态. 在这种情况下, 往往需要重新制定该机器人的运行路径.

## 4.2 基于运动学方程的轨迹规划研究(Study on the kinematic equation based trajectory planning)

前文介绍了基于离散事件系统的多机器人系统运动控制, 提出了实时分布式碰撞规避、死锁规避和鲁棒性控制算法. 然而, 由于顶层的抽象, 该算法无法直接获得底层的控制输入变量, 如速度、加速度、力矩等. 因此, 还需研究如何从底层连续控制部署机器人的运动轨迹. 本小节将讲述如何基于机器人的运动学方程(kinematic equations)进行轨迹规划.

机器人的运动学方程通过机器人的位置变量及由位置变量导出的其他变量, 如速度和加速度, 来描述机器人的运动行为. 由于环境的动态变化, 机器人必须不断地检测其周围的环境, 并根据实时环境变化进行局部轨迹规划. 基于运动学方程, 学者提出了一些局部轨迹规划算法<sup>[48, 66]</sup>. 这些工作通过将全局耦合的优化问题解耦为多个局部优化问题, 从而每个机器人可以自主地求解相应的优化问题. 然而每个机器人构建及求解对应的局部优化问题需要等待其他机器人的求解结果. 因此, 机器人的运动是同步进行的, 而无法实现真正的分布式运动.

基于机器人的运动学方程和MPC策略, 笔者提出了实时的完全分布式轨迹规划算法<sup>[68]</sup>. 其一般过程如下. 首先, 在任一时刻, 每个机器人自主地搜索探测距离内的实时环境, 确定环境中的障碍物以及其他机器人. 其次, 确定系统中静态障碍物的边界. 第三, 通过通信, 获得系统中动态障碍物的相关信息, 从而预测它们在预测时域(prediction horizon)内的可能运动轨迹. 第四, 根据上述信息和运动学方程, 构建当前时刻的局部优化问题. 第五, 求解所得优化问题. 第六, 将求解获得的结果作为输入控制变量控制机器人的运动. 最后, 随着时间更新, 机器人重新搜索环境并计算轨迹.

给定时间区间 $[0, T_i]$ , 机器人 $r_i$ 需从 $x_0^i$ 运动到 $x_f^i$ , 其中 $i \in \{1, 2, \dots, n\}$ , 为第 $i$ 个机器人, 系统中共有 $n$

个机器人. 将时间区间等步长 $h_i$ 离散:  $0 = \tau_0, \tau_1, \dots, \tau_{K_i} = T$ , 其中 $\tau_k = kh_i$ . 基于MPC策略, 由于传感器的搜索距离有限, 每个机器人的预测时域为 $H$ . 已知当前时刻 $\tau_{k_0}$ 的速度 $\mathbf{v}_i[k_0]$ 和位置 $\mathbf{x}_i[k_0]$ , 机器人的运动方程可表示为:

$$\mathbf{x}_i[k+1] = \mathbf{x}_i[k] + \mathbf{v}_i[k]h_i + \mathbf{a}_i[k]h_i^2/2, \quad (1)$$

$$\mathbf{v}_i[k+1] = \mathbf{v}_i[k] + \mathbf{a}_i[k]h_i, \quad (2)$$

$$\mathbf{x}_i[T_i] = \mathbf{x}_i^f, \mathbf{v}_i[T_i] = \mathbf{0}, k = k_0, \dots, T_i - 1, \quad (3)$$

其中 $\mathbf{x}_i[k]$ ,  $\mathbf{v}_i[k]$ 和 $\mathbf{a}_i[k]$ 分别表示 $r_i$ 在时刻 $\tau_k$ 的位置、速度和加速度.

其次,  $r_i$ 在时刻 $\tau_{k_0}$ 观察到的障碍物为 $\mathcal{O}_i[k_0]$ . 它可以分为两类: 静态障碍物和动态障碍物. 每个静态障碍物用其外接多面体或者外接球来表示, 所有静态障碍物集合记为 $\mathcal{O}_i^\alpha[k_0]$ ; 每个动态障碍物用球体来建模, 所有动态障碍物集合记为 $\mathcal{O}_i^\beta[k_0]$ ; 因此,  $\mathcal{O}_i[k_0] = \mathcal{O}_i^\alpha[k_0] \cup \mathcal{O}_i^\beta[k_0]$ . 为避免碰撞, 机器人需保证其在预测时域内的位置不能位于这些障碍空间内. 同时, 由于动态障碍物的位置变化, 每个机器人还需根据动态障碍物的当前信息预测它们在预测时域内的位置. 当前, 采用匀速直线运动来预测动态障碍物的运动, 原因有二: 第一, 运动物体的速度无法突变; 第二, 虽然考虑了动态障碍物在预测时域内所有位置, 然而真正需要避免的是下一时刻的障碍物所在的位置, 一旦运动到下一时刻, 该机器人需重新搜索环境, 重新预测其他动态障碍物的位置. 因此,  $\forall o \in \mathcal{O}_i^\beta[k_0], \tilde{\mathbf{x}}_o[k] = \mathbf{x}_o[k_0] + (k - k_0)\mathbf{v}_o[k_0]$ , 其中 $k = k_0 + 1, \dots, \min\{k_0 + H, K_i\}$ ,  $\mathbf{v}_o[k_0]$ 为障碍物 $o$ 的当前速度. 综上所述, 碰撞避免约束为:

$$\forall k \in \{k_0 + 1, \dots, \min\{k_0 + H, K_i\}\},$$

$$\mathbf{x}_i[k] \notin \mathcal{O}_i^\alpha[k_0] + \rho; \quad (4)$$

$$\forall o \in \mathcal{O}_i^\beta[k_0], \|\mathbf{x}_i[k] - \tilde{\mathbf{x}}_o[k]\|_2 \geq \rho. \quad (5)$$

因此,  $r_i$ 在时刻 $\tau_{k_0}$ 所建立的优化问题为:

$$P_i(k_0) \min_{\mathbf{a}_i[k_0:K_i-1]} \sum_{k=k_0}^{K_i-1} \|\mathbf{a}_i[k]\|_2^2,$$

subject to (1)–(5).

$P_i(k_0)$ 通常是一个高度非凸的优化问题, 因此可以通过序列化凸规划<sup>[65–66]</sup>求解. 根据得到的加速度 $\mathbf{a}_i[k_0]$ ,  $r_i$ 可以运动到下一个位置 $\mathbf{x}_i[k_0 + 1]$ . 一旦达到 $\mathbf{x}_i[k_0 + 1]$ ,  $r_i$ 重新搜索环境, 重构优化问题.

根据上述过程, 每个机器人只需通过和邻近机器人进行实时通信以获得其他机器人的当前和/或历史信息, 而无需等待其他机器人的预测结果. 因此, 每个机器人能够自主地规划轨迹、自主地运动, 从而实现

了系统的完全分布式控制.

### 4.3 多机器人系统运动控制的混合框架(A hybrid framework for motion control of multi-robot systems)

基于运动学方程的轨迹规划可直接作用于机器人的物理控制, 但有时无法规避如死锁之类的逻辑故障; 基于离散事件系统的运动控制实现的是由一系列事件激发的状态变迁, 但无法控制机器人物理层次的运动, 即无法给出机器人在每个状态里的具体物理运动控制. 显然, 对于混合了逻辑与物理指标的控制问题, 上述两个层面的独立执行都难以有效解决. 理想的运动控制需要结合这两个层面, 执行混合运动控制, 从而既能够产生无逻辑故障的事件序列, 又确保规划符合既定事件序列的物理运动轨迹.

图7给出了一个可能的机器人混合运动框架.

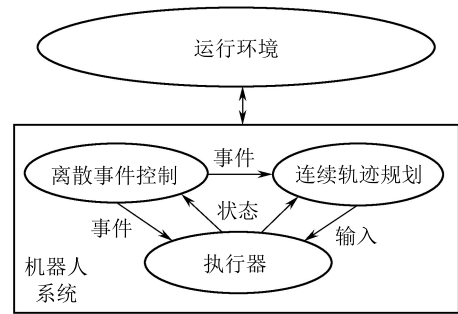


图7 机器人运动控制的混合架构

Fig. 7 A hybrid framework for robot motion control

该框架包含3个关键组件: 离散事件控制组件、连续轨迹规划组件, 以及执行器组件. 离散事件控制组件通过死锁规避算法和鲁棒控制算法实时产生事件序列并传递给连续规划组件和执行器组件; 连续轨迹规划组件通过外部环境和/或事件序列来确定输入物理变量, 例如加速度; 最后, 执行器根据输入信号来操作机器人的实际运动行为.

### 4.4 研究的创新性 (Advantages of this research work)

基于离散事件系统模型的多机器人系统运动控制方法研究主要有如下创新性.

第一, 离散事件系统理论在多机器人系统运动控制中的应用. 基于离散事件模型, 研究了多机器人系统的结构特性, 提出了多机器人系统中的高阶死锁的概念及其实时规避算法, 多机器人系统的鲁棒性控制算法. 通过系统地研究离散事件系统理论在机器人运动控制中的应用, 发展了离散事件系统的控制理论, 并拓宽了离散事件系统在实际中的运用.

第二, 离散事件系统的分布式实时控制算法. 扩展MPC思想到离散事件系统控制中, 将传统的时间变迁扩展到状态变迁. 每个机器人只需向前搜索其路径



上的部分状态, 并进行碰撞和死锁等逻辑故障检测确定能否迁移. 每个机器人通过自主运动控制实现整个系统的分布式控制. 每个机器人通过实时检测各自路径上的部分后继状态, 根据这些状态的信息, 如这些状态上的机器人的可靠性情况, 实时调整各自的运动情况以确保系统的活性并应对其他机器人的失效, 从而使系统具有良好的适应性和鲁棒性. 此过程无需求解系统的可达图, 从而有效地避免了状态爆炸问题, 降低运算和控制的复杂度.

第三, 连续物理轨迹规划的分布式实时控制算法. 基于MPC策略, 每个机器人实时探测局部环境, 通过和其他机器人通信获取它们的历史/当前运行状态并预测它们的运行轨迹, 然后自主地规划其在当前预测时域内的运行轨迹. 因每个机器人只需获得其他机器人的当前信息, 无需等待它们的预测结果, 故降低了不同机器人构建的优化问题的耦合性, 实现系统的完全分布式控制, 增加了系统的灵活性; 每个机器人只需构建预测时域内的碰撞约束, 无需规划整个运行时间内的轨迹, 从而降低了优化问题的规模, 减少了计算时间; 同时, 由于实时搜索环境, 系统能够及时地发现环境的动态变化, 从而增加系统对环境的适应性.

第四, 混合运动控制的原则性框架. 通过整合基于离散事件系统的运动控制和基于运动学方程的轨迹规划实现多机器人运动的混合控制. 该控制框架既能确保避免系统的逻辑故障, 如死锁等, 又能够提供底层的运动控制输入, 如加速度.

## 5 总结(Conclusions)

本文概述了国内外关于移动机器人运动控制的研究并着重介绍团队利用离散事件系统方法来研究多机器人系统的分布式运动控制的思路和阶段性成果, 如文献[67–68]. 首先, 基于离散事件系统, 提出了实时分布式控制算法来避免碰撞和死锁, 同时确保系统的鲁棒性, 即失效机器人对系统的影响最小. 通过分布式控制, 不需要一个中心化控制器, 从而避免了对整个系统的状态空间的搜索, 同时能够保证邻近机器人间的协调通信. 其次, 基于机器人的运动学方程, 提出了实时分布式控制算法来进行轨迹规划. 不同于现有算法中只能使机器人进行分布式轨迹预测而无法保证自主运动, 所提算法使得机器人既能够分布式地进行轨迹规划, 同时能够自主运动, 从而使得机器人具有更好的自主性, 能够增强系统的可扩展性和提高系统的性能.

根据已有的工作, 将来要把离散事件系统控制算法和完全分布式轨迹规划算法应用到更为复杂的多机器人系统中以便处理更为复杂的现实任务. 比如每个机器人有多条可行路径, 多机器人系统的最优队形维护等. 其次, 在本文所提原则性混合框架基础上, 将研究混合运动控制架构的具体实现方法, 并分析其在

实际系统中的应用效果.

## 参考文献(References):

- [1] KHAMIS A, HUSSEIN A, ELMOGY A. Multi-robot task allocation: a review of the state-of-the-art [C] //Koubâa A, Martínez-de Dios J R. *Cooperative Robots and Sensor Networks*. Switzerland: Springer, 2015: 31 – 51.
- [2] GUERRERO-BONILLA L, PROROK A, KUMAR V. Formations for resilient robot teams [J]. *IEEE Robotics and Automation Letters*, 2017, 2(2): 841 – 848.
- [3] PROROK A, HSIEH M A, KUMAR V. Formalizing the impact of diversity on performance in a heterogeneous swarm of robots [C] //Proceedings of IEEE International Conference on Robotics and Automation. Stockholm, Sweden: IEEE, 2016: 5364 – 5371.
- [4] LOIANNIO G, MULGAONKAR Y, BRUNNER C, et al. A swarm of flying smartphones [C] //Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems. Chicago, USA: IEEE, 2016: 1681 – 1688.
- [5] SASKA M, VONÁSEK V, CHUDOBA J, et al. Swarm distribution and deployment for cooperative surveillance by micro-aerial vehicles [J]. *Journal of Intelligent & Robotic Systems*, 2016, 84(1/2/3/4): 469 – 492.
- [6] TURPIN M, MICHAEL N, KUMAR V. Capt: concurrent assignment and planning of trajectories for multiple robots [J]. *The International Journal of Robotics Research*, 2014, 33(1): 98 – 112.
- [7] PIMENTA L C, PEREIRA G A, MICHAEL N, et al. Swarm coordination based on smoothed particle hydrodynamics technique [J]. *IEEE Transactions on Robotics*, 2013, 29(2): 383 – 399.
- [8] SCHWAGER M, DAMES P, RUS D, et al. A multi-robot control policy for information gathering in the presence of unknown hazards [C] //Christensen H I, Khatib O. *Robotics Research: The 15th International Symposium ISRR*. Cham, Switzerland: Springer, 2017: 455 – 472.
- [9] GIL S, KUMAR S, MAZUMDER M, et al. Guaranteeing spoof-resilient multi-robot networks [J]. *Autonomous Robots*, 2017, 41(6): 1383 – 1400.
- [10] ALONSO-MORA J, MONTIJANO E, SCHWAGER M, et al. Distributed multi-robot formation control among obstacles: a geometric and optimization approach with consensus [C] //Proceedings of IEEE International Conference on Robotics and Automation. Stockholm, Sweden: IEEE, 2016: 5356 – 5363.
- [11] ALONSO-MORA J, BAKER S, RUS D. Multi-robot navigation in formation via sequential convex programming [C] //Proceedings of 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems. Hamburg, Germany: IEEE, 2015: 4634 – 4641.
- [12] SOLTERO D E, SCHWAGER M, RUS D. Decentralized path planning for coverage tasks using gradient descent adaptive control [J]. *The International Journal of Robotics Research*, 2014, 33(3): 401 – 425.
- [13] ULUSOY A, SMITH S L, DING X C, et al. Optimality and robustness in multi-robot path planning with temporal logic constraints [J]. *The International Journal of Robotics Research*, 2013, 32(8): 889 – 911.
- [14] SMITH S L, SCHWAGER M, RUS D. Persistent robotic tasks: monitoring and sweeping in changing environments [J]. *IEEE Transactions on Robotics*, 2012, 28(2): 410 – 426.
- [15] CHEN Y, DING X C, STEFANESCU A, et al. Formal approach to the deployment of distributed robotic teams [J]. *IEEE Transactions on Robotics*, 2012, 28(1): 158 – 171.
- [16] YAN Z, JOUANDEAU N, CHERIF A A. A survey and analysis of multi-robot coordination [J]. *International Journal of Advanced Robotic Systems*, 2013, 10(12): 399:1 – 399:18.

- [17] HOPCROFT J E, SCHWARTZ J T, SHARIR M. On the complexity of motion planning for multiple independent objects; PSPACE-hardness of the “warehouseman’s problem” [J]. *International Journal of Robotics Research*, 1984, 3(4): 76 – 88.
- [18] REIF J H. Complexity of the mover’s problem and generalizations [C] // *Proceedings of the 20th Annual Symposium on Foundations of Computer Science*. Washington, DC, USA: IEEE, 1979: 421 – 427.
- [19] LATOMBE J C. *Robot Motion Planning* [M]. New York, USA: Springer, 2012.
- [20] CHOSET H, LYNCH K M, KAVRAKI L, et al. *Principles of Robot Motion: Theory, Algorithms, and Implementations* [M]. Boston, MA, USA: MIT Press, 2005.
- [21] HOY M, MATVEEV A S, SAVKIN A V. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: A survey [J]. *Robotica*, 2015, 33(3): 463 – 497.
- [22] KLOETZER M, MAHULEA C, GONZALEZ R. Optimizing cell decomposition path planning for mobile robots using different metrics [C] // *Proceedings of International Conference on System Theory, Control and Computing*. Cheile Gradistei, Romania: IEEE, 2015: 565 – 570.
- [23] ACAR E U, CHOSET H, RIZZI A A, et al. Morse decompositions for coverage tasks [J]. *The International Journal of Robotics Research*, 2002, 21(4): 331 – 344.
- [24] PIVTORAIKO M, KNEPPER R A, KELLY A. Differentially constrained mobile robot motion planning in state lattices [J]. *Journal of Field Robotics*, 2009, 26(3): 308 – 333.
- [25] PIVTORAIKO M, KELLY A. Kinodynamic motion planning with state lattice motion primitives [C] // *Proceedings of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. San Francisco, CA, USA: IEEE, 2011: 2172 – 2179.
- [26] PIVTORAIKO M, MELLINGER D, KUMAR V. Incremental micro-UAV motion replanning for exploring unknown environments [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Karlsruhe, Germany: IEEE, 2013: 2452 – 2458.
- [27] KLOETZER M, BELTA C. Temporal logic planning and control of robotic swarms by hierarchical abstractions [J]. *IEEE Transactions on Robotics*, 2007, 23(2): 320 – 330.
- [28] SAHA I, RAMAITHITIMA R, KUMAR V, et al. Automated composition of motion primitives for multi-robot systems from safe LTL specifications [C] // *Proceedings of 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Chicago, IL, USA: IEEE, 2014: 1525 – 1532.
- [29] KRNJAK A, DRAGANJAC I, BOGDAN S, et al. Decentralized control of free ranging AGVs in warehouse environments [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Seattle, WA, USA: IEEE, 2015: 2034 – 2041.
- [30] PREPARATA F P, SHAMOS M I. *Computational Geometry: An Introduction* [M]. New York: Springer, 1985.
- [31] CHEN Yan jie, WANG Yao nan, ZHONG Hang, et al. Improved geometrical learning planning for service robot in dynamic environment [J]. *Control Theory & Applications*, 2015, 32(2): 162 – 168.  
(陈彦杰, 王耀南, 钟杭, 等. 动态环境中服务机器人的改进型地图学习规划 [J]. *控制理论与应用*, 2015, 32(2): 162 – 168.)
- [32] BHATTACHARYA P, GAVRILOVA M L. Roadmap-based path planning — using the Voronoi diagram for a clearance-based shortest path [J]. *IEEE Robotics & Automation Magazine*, 2008, 15(2): 58 – 66.
- [33] LOZANO-PÉREZ T, WESLEY M A. An algorithm for planning collision-free paths among polyhedral obstacles [J]. *Communications of the ACM*, 1979, 22(10): 560 – 570.
- [34] ZHOU D, WANG Z, BANDYOPADHYAY S, et al. Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells [J]. *IEEE Robotics and Automation Letters*, 2017, 2(2): 1047 – 1054.
- [35] KAVRAKI L E, SVESTKA P, LATOMBE J C, et al. Probabilistic roadmaps for path planning in high-dimensional configuration spaces [J]. *IEEE Transactions on Robotics and Automation*, 1996, 12(4): 566 – 580.
- [36] ELBANHAWI M, SIMIC M. Sampling-based robot motion planning: A review [J]. *IEEE Access*, 2014, 2: 56 – 77.
- [37] ARSLAN O, BERNTORP K, TSIOTRAS P. Sampling-based algorithms for optimal motion planning using closed-loop prediction [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Singapore: IEEE, 2017: 4991 – 4996.
- [38] LAVALLE S M. Rapidly-exploring random trees: a new tool for path planning [R]. Ames, IA, USA: Iowa State University, 1998.
- [39] BRUCE J, VELOSO M. Real-time randomized path planning for robot navigation [C] // *Proceedings of 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Lausanne, Switzerland: IEEE, 2002: 2383 – 2388.
- [40] ZHANG Botao, LI Jiadong, LIU Shirong. Rapidly-exploring random trees motion planning for non-holonomic robot with collision-test and regression mechanism [J]. *Control Theory & Applications*, 2016, 33(7): 870 – 878.  
(张波涛, 李加东, 刘士荣. 采用碰撞测试和回归机制的非完整约束机器人快速扩展随机树运动规划 [J]. *控制理论与应用*, 2016, 33(7): 870 – 878.)
- [41] NG J, BRÄUNL T. Performance comparison of bug navigation algorithms [J]. *Journal of Intelligent & Robotic Systems*, 2007, 50(1): 73 – 84.
- [42] CETIN O, ZAGLI I, YILMAZ G. Establishing obstacle and collision free communication relay for UAVs with artificial potential fields [J]. *Journal of Intelligent & Robotic Systems*, 2013, 69(1/2/3/4): 361 – 372.
- [43] FIORINI P, SHILLER Z. Motion planning in dynamic environments using velocity obstacles [J]. *The International Journal of Robotics Research*, 1998, 17(7): 760 – 772.
- [44] VAN DEN BERG J, GUY S J, LIN M, et al. Reciprocal n-body collision avoidance [C] // *Proceedings of the 14th International Symposium on Robotics Research*. Berlin, Germany: Springer, 2011: 3 – 19.
- [45] LIU Yang, YU Shuyou, GUO Yang, et al. Receding horizon control for path following problems of wheeled mobile robots [J]. *Control Theory & Applications*, 2017, 34(4): 424 – 432.  
(刘洋, 于树友, 郭洋, 等. 基于滚动时域优化的轮式移动机器人路径跟踪问题研究 [J]. *控制理论与应用*, 2017, 34(4): 424 – 432.)
- [46] ZHAO Juanping, GAO Xianwen, FU Xiuhui, et al. Improved ant colony algorithm of path planning for mobile robot [J]. *Control Theory & Applications*, 2011, 28(4): 458 – 461.  
(赵娟平, 高宪文, 符秀辉, 等. 移动机器人路径规划的改进蚁群优化算法 [J]. *控制理论与应用*, 2011, 28(4): 458 – 461.)
- [47] ABICHANDANI P, FORD G, BENSON H Y, et al. Mathematical programming for multi-vehicle motion planning problems [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Saint Paul, MN, USA: IEEE, 2012: 3315 – 3322.
- [48] MORGAN D, CHUNG S J, HADAEGH F Y. Model predictive control of swarms of spacecraft using sequential convex programming [J]. *Journal of Guidance, Control, and Dynamics*, 2014, 37(6): 1725 – 1740.
- [49] HYUN N S P, VELA P A, VERRIEST E I. A new framework for optimal path planning of rectangular robots using a weighted  $L_p$  norm [J]. *IEEE Robotics and Automation Letters*, 2017, 2(3): 1460 – 1465.
- [50] CHEN Y F, LIU M, EVERETT M, et al. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Singapore: IEEE, 2017: 285 – 292.

- [51] LI Z, DENG J, LU R, et al. Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach [J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2016, 46(6): 740 – 749.
- [52] GÓMEZ-BRAVO F, CUESTA F, OLLERO A, et al. Continuous curvature path generation based on  $\beta$ -spline curves for parking manoeuvres [J]. *Robotics and Autonomous Systems*, 2008, 56(4): 360 – 372.
- [53] CHEN Cheng, HE Yuqing, BU Chunguang, et al. Feasible trajectory generation for autonomous vehicles based on quartic Bézier curve [J]. *Acta Automatica Sinica*, 2015, 41(3): 486 – 496.  
(陈成, 何玉庆, 卜春光, 等. 基于四阶贝塞尔曲线的无人车可行轨迹规划 [J]. *自动化学报*, 2015, 41(3), 486 – 496.)
- [54] WANG X, KLOETZER M, MAHULEA C, et al. Collision avoidance of mobile robots by using initial time delays [C] // *Proceedings of IEEE Conference on Decision and Control*. Osaka, Japan: IEEE, 2015: 324 – 329.
- [55] RODRIGUEZ-SEDA E J, TANG C, SPONG M W, et al. Trajectory tracking with collision avoidance for nonholonomic vehicles with acceleration constraints and limited sensing [J]. *International Journal of Robotics Research*, 2014, 33(12): 1569 – 1592.
- [56] SOLTERO D E, SMITH S L, RUS D. Collision avoidance for persistent monitoring in multi-robot systems with intersecting trajectories [C] // *Proceedings of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. San Francisco, CA, USA: IEEE, 2011: 3645 – 3652.
- [57] HU H S, SU R, ZHOU M C, et al. Polynomially complex synthesis of distributed supervisors for large scale AMS using Petri nets [J]. *IEEE Transactions on Control Systems Technology*, 2015, 24(5): 1610 – 1622.
- [58] LI Z W, WU N Q, ZHOU M C. Deadlock control of automated manufacturing systems based on Petri nets – a literature review [J]. *IEEE Transactions on Systems, Man, and Cybernetics: Applications and Reviews*, 2012, 42(4): 437 – 462.
- [59] LEE C C, LIN J T. Deadlock prediction and avoidance based on Petri nets for zone-control automated guided vehicle systems [J]. *International Journal of Production Research*, 1995, 33(12): 3249 – 3265.
- [60] YALCIN A, BOUCHER T O. Deadlock avoidance in flexible manufacturing systems using finite automata [J]. *IEEE Transactions on Robotics and Automation*, 2002, 16(4): 424 – 429.
- [61] REVELIOTIS S, ROSZKOWSKA E. Conflict resolution in free-ranging multi-vehicle systems: a resource allocation paradigm [J]. *IEEE Transactions on Robotics*, 2011, 27(2): 283 – 296.
- [62] ROSZKOWSKA E, REVELIOTIS S. A distributed protocol for motion coordination in free-range vehicular systems [J]. *Automatica*, 2013, 49(6): 1639 – 1653.
- [63] DU N, HU H S, LIU Y. Robust control of automated manufacturing systems with assembly operations using Petri nets [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Stockholm, Sweden: IEEE, 2016: 3632 – 3638.
- [64] CHENG Y, HU H S, LIU Y. Robust supervisor synthesis for automated manufacturing systems using Petri nets [C] // *Proceedings of IEEE International Conference on Automation Science and Engineering*. Gothenburg, Sweden: IEEE, 2015: 1029 – 1035.
- [65] SCHITTKOWSKI K, ZILLOBER C. Sequential convex programming methods [C] // MARTI K, KALL P. *Stochastic Programming: Numerical Techniques and Engineering Applications*. New York, USA: Springer, 1995: 123 – 141.
- [66] CHEN Y, CUTLER M, HOW J P. Decoupled multiagent path planning via incremental sequential convex programming [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. Seattle, WA, USA: IEEE, 2015: 5954 – 5961.
- [67] ZHOU Y, HU H S, LIU Y, et al. Collision and deadlock avoidance in multi-robot systems: a distributed approach [J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017, 47(7): 1712 – 1726.
- [68] ZHOU Y, HU H S, LIU Y, et al. A real-time and fully distributed approach to motion planning for multirobot systems [J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017. [Online] <http://ieeexplore.ieee.org/document/8055437/>.

#### 作者简介:

**周 远** (1989–), 男, 博士研究生, 研究领域为多机器人系统的自主运动控制、轨迹规划、协调控制, E-mail: y.zhou@ntu.edu.sg;

**胡核算** (1977–), 男, 教授, 研究领域为离散事件系统的监督控制及应用、Petri网、多机器人系统, E-mail: huhesuan@gmail.com;

**刘 杨** (1981–), 男, 副教授, 研究领域为形式化方法、网络安全、软件工程、网络物理系统, E-mail: yangliu@ntu.edu.sg;

**林尚威** (1979–), 男, 助理教授, 研究领域为形式化验证、软件工程、网络物理系统, E-mail: shang-wei.lin@ntu.edu.sg.