智能机器人可变参数群体控制模型的多目标优化方法

王原¹, 马力¹, 王凌², 何永明¹, 齐伟华¹, 邢立宁^{1*}

1. 国防科技大学系统工程学院, 长沙 410073;

2. 清华大学自动化系, 北京 100084

* E-mail: xln_2002@nudt.edu.cn

收稿日期: 2019-08-12; 接受日期: 2019-10-31; 网络版发表日期: 2020-04-15

国家自然科学基金(批准号: 61773120, 61873328)和国家杰出青年科学基金(批准号: 61525304)资助项目

摘要 近年来, 针对自组织群体智能机器人的研究一直是智能机器人研究的热点问题. 在这些研究中, 提升群体机器人控制模型的泛用性一直是一个热点问题. 为了提升群体机器人控制模型的泛用性, 提高群体机器人在不同类型的环境下的功能表现, 使用可配置的控制模型, 利用仿真的方法求解控制模型在不同环境下的参数配置, 是一种较为常见的方法. 本研究提出了一种基于规则系统模型的多目标群体智能机器人控制模型优化方法, 并针对该方法测试了多种不同的多目标优化方法的表现. 使用包含凸障碍、非凸障碍以及通道型障碍的多个不同的仿真场景验证了该模型的效能. 仿真验证结果表明该方法在各种环境下均具有较好的表现.

关键词 群体智能机器人, 控制模型, 多目标优化, 进化算法, 粒子群优化

1 引言

近年来, 针对群体机器人(swarm robotics, SR)的研究迎来了一轮新的高潮. 其原因是群体机器人在仿生学、人工智能、控制论等领域均具有重要应用. 群体机器人通常被视为一种特殊的多机器人系统(multi-robot system). 在群体机器人系统中, 一个复杂的任务被分解到多个完全自主控制的同类型机器人上执行. 群体机器人和经典多机器人系统的主要区别如下:

(1) 群体机器人系统包含的机器人数量最少不能少于10;

(2) 群体机器人系统中的单个机器人不能有过于

复杂的结构或功能;

(3) 群体机器人系统中的机器人必须具备相似的类型和功能.

在针对群体机器人的研究中, 如何设计合适的群体机器人控制系统一直是一大挑战. 其原因是在群体机器人控制系统中, 考虑到群体机器人设计的原则, 通常不允许存在一个或几个具有领导者角色的机器人. 因此, 当设计针对群体机器人中的个体的控制系统时, 该控制系统需要保证能够仅通过个体与个体、个体与环境之间的信息交互, 就能够使个体以及整个群体的行动表现出足够的群体性.

以此为目的, 通常设计群体机器人控制系统的方法

引用格式: 王原, 马力, 王凌, 等. 智能机器人可变参数群体控制模型的多目标优化方法. 中国科学: 技术科学, 2020, 50: 526–537

Wang Y, Ma L, Wang L, et al. A multi-objective optimization method for intelligent swarm robotic control model with changeable parameters (in Chinese). Sci Sin Tech, 2020, 50: 526–537, doi: [10.1360/SST-2019-0280](https://doi.org/10.1360/SST-2019-0280)

式有两种: (1) 基于行为(behavior-based)的设计方式; (2) 自动化设计(autonomic design)方式. 在基于行为的设计方式中, 通常又可以分为三大类: (1) 概率有限状态机模型(probabilistic finite state machine design); (2) 基于虚拟物理规则的模型(virtual physical-based design); (3) 其他模型. 基于自动化设计的方式又包括: (1) 基于进化的(evolutionary robots, ER)机器人系统; (2) 基于强化学习(reinforcement learning methods, RL)的机器人系统. 关于这些不同类型的无人机集群控制系统的设计原则可见Brambilla等人^[1]的综述文章. 其中, 基于虚拟物理学规则的设计方式被广泛地应用于各类需要同时维持无人机群形态并完成避障、导航等多种用途的无人机集群控制设计中. 就我们所知, 最先提出虚拟物理规则概念的是Khatib^[2]. 文献[2]介绍了一种无人机集群控制的概念——人工势力场(artificial potential field, APF). 基于APF设计的无人机集群控制器通常满足如下准则: 集群中的任何无人机个体至少受到两种力, 一种是来自目标定向的引力, 一种是来自其他个体或障碍物的排斥力. 基于虚拟物理规则设计的无人机集群控制器通常具有如下优点^[3]:

(1) 不同于基于有限状态机的控制方式, 基于虚拟物理规则的设计使用唯一物理学规则替代多种不同的行为规则用于满足使无人机集群出现多种不同行为的要求;

(2) 无人机个体行为能够通过简单地向量操作求得;

(3) 能够通过一些理论工具对控制模型进行定量分析.

自Khatib^[2]以来, 针对虚拟物理规则的研究主要集中于扩展不同的物理原则上. Spears等人^[4]提出了后来获得广泛认可并被广泛地用于描述基于虚拟物理规则的无人机群控制器的控制模型. Fiorini和Shillert^[5]在已有规则的基础上扩展出了一类重要的规则: 障碍速度(velocity obstacle, VO). Berg等人^[6]通过考虑临近个体对VO规则进行了进一步扩展. 最终, Santos等人^[7]将VO规则扩展到为virtual group velocity obstacle (VGVO). 通过使用VGVO概念, 能够保证无人机集群能够更好地在空间内分布为几个小的集群并更好地完成一类使命目标. La等人^[8]将虚拟物理规则和强化学习相结合, 从而提高了控制器的效能. 其他扩展包括Woods和La^[9]将相对速度纳入了模型的考虑范围. 为了测试该

模型的效能, 他们使用了一个室内的实机场景, 一台基于Parrot ARDrone 2.0无人机平台被用于执行3D环境下的目标定向飞行. Ha等人^[10]和Meier等人^[11]将速度的同一性纳入控制模型的考量范围. Scott和Leal^[12]提出了基于虚拟物理规则的一般化无人机集群控制模型, 该模型可广泛地应用于各类队形维持任务并支持不同规模的无人机集群.

尽管目前针对虚拟物理规则的无人机集群控制模型已经进行了广泛的研究, 但是至今该模型仍需要解决一个至关重要的问题: 如何使简单的基于数学规则的控制模型具有更加广泛的应用并在各种不同的问题环境下均保证其表现的稳定性.

为了解决这一问题, 一类重要的解决方案是允许模型使用可变参数. 使用可变参数的无人机集群控制模型的研究最早可追溯至Hettiarachchi^[13]. 在本文中, 基于进化算法(evolutionary algorithm, EA)的算法被用于离线地求解基于Lennard-Jones potential function的无人机集群控制模型中的参数. Pugh和Martinoli^[14]使用基于粒子群算法的优化方法用于求解无人机集群控制模型中的参数, 并获得了比遗传算法具有更好种群多样性的算法表现. Folino等人^[15]将自适应无人机集群控制模型用于完成去中心化的无人机集群聚合任务. 该算法针对不同规模的无人机集群均有较好表现. Yang等人^[16]改进了基于经典Vicsek模型的无人机集群控制模型以完成无人机聚合任务. 在该研究中, 一个可变的参数用于控制集群中不同个体之间互相影响的能力. Zhao等人^[17]研究了自适应无人机集群控制模型在高速环境下的表现.

在2018年, Vásárhelyi等人^[18]提出了一种新的无人机集群控制模型. 该文章描述了一种基于11个可变参数的虚拟物理规则无人机个体控制系统. 作者使用了基于Atlasz超级计算机的仿真场景和实机飞行试验验证了该控制器的效能. 刨除巨大的计算资源消耗(在Atlasz计算机环境下运行一个仿真场景的单次试验通常需要消耗2~6天), 该模型在设定的试验场景(包括无障碍任意飞行以及有障碍环境下的任意飞行)中均获得了较好的表现.

本文描述了一个基于可变参数的无人机集群控制系统, 该研究的灵感来源于Vásárhelyi等人^[18]. 在该系统的设计中, 我们结合了基于规则的控制系统(Ueyama等人^[19])和基于虚拟物理规则的控制系统. 在该模

型中, 基于规则的控制系统的用于使无人机个体针对不同的环境观测结果做出不同反应, 而带可变参数的虚拟物理规则系统用于解决不同场景下实现不同目标的问题. 多种不同的多目标优化算法被用于进行控制模型的参数调整. 仿真试验场景包括各种典型障碍(凸障碍、非凸障碍、通道障碍等)下的目标定向运动. 本文的主要贡献包括: (1) 描述了一种新的带可变参数的无人机集群控制模型; (2) 针对该模型设计了相应的多目标优化算法; (3) 通过多个典型场景(包括多障碍、通道障碍、凸障碍等)对算法和模型的效能进行了分析.

2 控制模型

目前针对无人机集群的控制模型主要分为三种. 第一种是源自Vicsek模型的控制模型, 该模型首先提出于1995年^[20]. 在该模型中, 每一个个体下一步的移动速度取决于该个体能够观察到的种群的整体移动速度以及一个随机扰动. 该模型仅在种群密度较大且通信噪声较小的情况下能够保证较好的群体性. 第二种模型来源于Reynolds三控制原则的Coulzin Model^[21]. 在该模型中, 主要考虑如下几个控制原则: (1) 个体间总是倾向于保持一个较小的距离; (2) 个体总是倾向于维持和周边个体相同的移动速度和移动方向. 第三种模型方法为基于势力场法(potential field)^[22]的控制规则. 在该规则下, 目标相对于无人机个体存在引力, 而场景内的障碍物和其他个体则对当前无人机个体存在斥力. 斥力和引力的大小取决于当前个体和目标物之间的距离. 该规则主要用于确定场景下的单个或多个机器人的目标定向和障碍规避活动.

本文所使用的基于规则系统的控制模型来源于经典的Reynolds控制模型. 该模型在群智能机器人队形维持、避障以及导航等现实任务中存在广泛的应用. 本文中使用的控制模型在Reynolds模型的基础上, 增加了新的基于规则系统的速度更新方式, 并设计了相应的多目标优化策略对模型在不同环境下的速度更新规则进行了优化. 以下对该模型的相关细节进行介绍.

2.1 基本假设

本模型在满足SR的基本定义的前提下, 同时考虑如下基本假设:

(1) 无人机集群中的每一个个体都必须穿过一个

正方形的场地并到达指定地点;

(2) 场地大小为一个边长为250 m的正方形;

(3) 无人机中的每一个个体均应尽量避免与其他个体或障碍物发生碰撞, 若发生碰撞则视为该个体死亡;

(4) 任何一台无人机均不允许飞出场地边界, 触碰场地边界同样视为死亡;

(5) 无人机有最大速度限制;

(6) 无人机的通信范围和探测范围均有限;

(7) 无人机间的通信存在延迟.

基本假设(5)~(7)为考虑到现实世界中无人机的能力约束加入的假设, 以使本文所研究的模型更加贴近现实.

2.2 无人机速度更新模型

本文所考虑的无人机速度更新模型主要分为三部分: 来自目标定向的定向力; 用于防止与障碍物发生碰撞的防撞力; 用于维持队形的队形维持力. 以下分别描述这三种不同的力对无人机速度更新的作用.

(1) 定向力. 定向力是吸引群体无人机中的个体向既定目标运动的力. 该力可以通过如下公式表示:

$$\Delta v_i^{\text{tar}} = \frac{p_t - p_i}{r_{it}}, \quad (1)$$

式中, Δv_i^{tar} 表示定向力的大小, p_t 为用向量表示的目的地所处位置, p_i 为用向量表示的个体当前所处的位置, r_{it} 为目的地和个体间的距离.

(2) 防撞力. 防撞力为避免无人机群中的个体和障碍物相撞的力. 当在设定好的探测范围里无人机个体探测到障碍物时, 会在障碍物表面生成虚拟个体. 每个虚拟个体会像无人机个体输出一个斥力. 通过这些斥力的合力可以求得针对单个无人机的防撞力. 该力可以通过如下公式表示:

$$\Delta v_i^{\text{obs}} = \sum_m^M (r_0^{\text{obs}} - r_{im}) \times \frac{p_i - p_m}{r_{im}}, \text{ for } r_{im} < r_0^{\text{obs}}, \quad (2)$$

式中, Δv_i^{obs} 为无人机个体受到的防撞力, r_0^{obs} 为个体最远可探测到障碍物的距离, r_{im} 为个体探测到的位于障碍物表面的虚拟个体距离当前个体的距离, p_m 为用向量表示的障碍物 m 的空间位置, M 为当前个体 i 能够探测到的障碍物的总数.

(3) 队形维持力. 队形维持力用于保证群体无人机表现出足够的群体性(包括聚集在通常为无人机最大

可通信范围的指定范围内, 保持无人机运动速度和方向的一致性). 在本研究中, 队形维持力同样包括三部分. 个体接收到的来自其他个体的队形维持力共有三种可能: 斥力、同向力以及引力. 当无人机探测到其他个体时, 根据其他个体距离当前无人机的距离, 分别表现出不同的队形维持力模式. 具体的判断方式如下: 当探测到的个体与当前无人机的距离小于 R_0 时, 队形维持力表现为斥力; 当探测到的个体与当前无人机的距离大于 R_0 且小于 R_1 时, 队形维持力表现为同向力; 当探测到的个体与当前无人机的距离大于 R_1 且小于 R_2 时, 队形维持力表现为斥力. 队形维持力的判断模式如图1所示. 以下分别详细介绍这三种力的作用方式.

当探测到另一个个体和当前个体的距离小于最大排斥距离 r_0^{rep} 时, 则对当前个体施加斥力. 斥力的大小通过以下公式决定:

$$\Delta v_{ij}^{\text{rep}} = (r_0^{\text{rep}} - r_{ij}) \times \frac{p_i - p_j}{r_{ij}}, \quad (3)$$

式中, $\Delta v_{ij}^{\text{rep}}$ 代表个体 i 受到另一个个体 j 的斥力, r_{ij} 为个体 i 和个体 j 之间的距离, p_i 和 p_j 为用向量表示的个体 i 和个体 j 在当前空间之中的位置. 通过公式(3), 可以得出个体在某一时刻所受到的来自其他个体的斥力如下:

$$\Delta v_i^{\text{rep}} = \sum_{j=1, j \neq i}^N \Delta v_{ij}^{\text{rep}}, \text{ for } r_{ij} < r_0^{\text{rep}}, \quad (4)$$

图1 斥力、引力 and 同向力的判断
Figure 1 Repulsion, attraction and alignment.

式中, Δv_i^{rep} 为当前个体 i 受到的来自其他个体的全部斥力之和. N 代表整个群体中全部无人机的集合. 通过引入斥力, 能够保证无人机在群体行进的过程中尽可能不与其他个体发生碰撞.

在本研究所描述的模型中, 同向力用于缩小在 R_1 范围内的无人机之间速度之差(包括速度的大小和方向), 进而使整个无人机群表现出更好的群体特性. 同向力可以通过如下公式表示:

$$\Delta v_i^{\text{ali}} = \frac{1}{n_{\text{ali}}} \sum_{j=1, j \neq i}^N \frac{v_j}{|v_j|}, \text{ for } r_0^{\text{rep}} < r_{ij} < r_0^{\text{ali}}, \quad (5)$$

式中, Δv_i^{ali} 为个体 i 受到的同向力的大小, v_j 是通过向量表示的当前无人机的速度, n_{ali} 为位于 R_1 范围内的无人机的数量, r_0^{ali} 为 R_1 的最大值.

最终, 当个体探测到其他个体位于 R_1 到 R_2 之间时, 则该被探测到的个体会对当前个体产生引力, 从而使这两个个体靠拢. 这是为了保证无人机群之间的个体距离能够维持在一个合理的范围内, 从而使无人机群保持基本的群体形态. 单个个体对其他个体施加的引力作用如公式(6)所示:

$$\Delta v_{ij}^{\text{att}} = \frac{1}{r_0^{\text{att}} - r_{ij}} \times \frac{p_j - p_i}{r_{ij}}, \quad (6)$$

式中, $\Delta v_{ij}^{\text{att}}$ 代表个体 j 对个体 i 释放的引力, r_0^{att} 为 R_2 的最大值. 通过公式(6)可得出, 当个体 j 在 R_2 范围内距离个体 i 越远时, 个体 j 对个体 i 施加的引力则越大. 通过公式(6), 可得个体 i 在某一时刻受到的引力总量的大小如下:

$$\Delta v_i^{\text{att}} = \sum_{j=1, j \neq i}^N \Delta v_{ij}^{\text{att}}, \text{ for } r_0^{\text{ali}} < r_{ij} < r_0^{\text{att}}, \quad (7)$$

其中, Δv_i^{att} 为个体 i 所受到的引力.

通过公式(1)~(7), 最终可得到个体下一阶段的速度如公式(8):

$$v_i(t+1) = v_i(t) + \Delta v_i, \quad (8)$$

$$\Delta v_i = a\Delta v_i^{\text{rep}} + b\Delta v_i^{\text{ali}} + c\Delta v_i^{\text{att}} + d\Delta v_i^{\text{tar}} + e\Delta v_i^{\text{obs}}.$$

由于本研究考虑到针对不同的环境情况, 设计了相应的基于规则的速度更新公式, 因此在公式(8)中,

不同于经典的Reynolds模型, 我们引入了 $\{a, b, c, d, e\}$ 共5个参数用于描述在不同场景和任务下, 个体对环境探测的不同可能导致的规则变化. 需要说明的是, 考虑到在现实情况下, 无人机的速度不可能无限增加, 因此设定了一个无人机速度的上限 $v^{\max}$. 因此本研究最终的速度更新公式如下:

$$v_i(t+1) = \frac{v_i(t+1)}{|v_i(t+1)|} \times \min(|v_i(t+1)|, v^{\max}). \quad (9)$$

由于本文使用了基于规则系统的速度更新模型, 因此以下继续介绍本文中的规则系统部分.

2.3 速度更新规则系统

本研究通过一个速度更新规则系统用于解决无人机群在不同环境下的不同行为模式问题, 为介绍该速度更新规则系统, 首先需要明确判断条件 zor_{obs} 如图2所示.

当无人机的可探测范围 zor_{obs} 中存在障碍物时, 则启用有障碍物规则, 否则采用一般规则. 因此本文最终的个体速度更新公式如下:

$$\begin{aligned} \text{if } zor_{\text{obs}} \neq \emptyset: \\ \Delta v_i = a_1 \Delta v_i^{\text{rep}} + b_1 \Delta v_i^{\text{ali}} + c_1 \Delta v_i^{\text{att}} + d_1 \Delta v_i^{\text{tar}} + e_1 \Delta v_i^{\text{obs}}, \\ \text{if } zor_{\text{obs}} = \emptyset: \\ \Delta v_i = a_2 \Delta v_i^{\text{rep}} + b_2 \Delta v_i^{\text{ali}} + c_2 \Delta v_i^{\text{att}} + d_2 \Delta v_i^{\text{tar}} + e_2 \Delta v_i^{\text{obs}}. \end{aligned} \quad (10)$$

因此, 本文最终共需要针对十个参数 $\{a_1, b_1, c_1, d_1, e_1, a_2, b_2, c_2, d_2, e_2\}$ 进行参数调整.

3 优化方法

本研究采用基于NSGA II的多目标优化方法, 对模型参数进行优化. NSGA II是一种被广泛应用于各类多目标优化场景的基于种群的启发式算法. 关于NSGA II的算法结构和用于进行多目标优化的非支配排序机制可见文献[23]. 本节主要介绍在本研究中针对无人机集群控制模型的参数调整问题对NSGA II算法做的特定改变, 具体如下所述.

3.1 问题分析

在本文所针对的无人机集群任务场景中, 需要考虑如下三个无人机集群表现的优化目标: (1) 尽可能减小无人机和其他物体(包括障碍、其他无人机个体

图2 基于障碍的判断规则 zor_{obs}
Figure 2 Rule of objective detection.

和场地边线)间的碰撞; (2) 无人机集群应尽可能保持合理队形; (3) 无人机集群应尽可能早地到达目标地点. 其中, 设计目标(2)是因为通常在实践中, 单个无人机的通信范围有限, 保持合理队形有利于实现无人机集群内部的互联互通和信息共享. 为实现以上目标, 本文设计如下三个模型优化目标.

$$r^{\text{death}} = \frac{N^{\text{death}}}{N}, \quad (11)$$

式中, r^{death} 为死亡率, N^{death} 为仿真结束时死亡个体的数量, N 为集群中无人机的数量. 该公式代表仿真结束时死亡的无人机的数量与无人机总数的比值.

$$d^{\text{agg}} = \frac{\sum_{t=1}^T \sum_{i=1}^N \|p_c^t - p_i^t\|_2}{NT}, \quad (12)$$

式中, d^{agg} 为聚集系数, T 为仿真总时长, p_i^t 为个体 i 在仿真时刻 t 的所处位置. p_c^t 为无人机集群的虚拟质心在仿真时刻 t 的所在位置, p_c^t 可通过公式(13)求得:

$$p_c^t = \frac{\sum_{i=1}^N p_i^t}{N}. \quad (13)$$

聚集系数 d^{agg} 实际上衡量了在整个仿真过程中整个无人机集群中的每一台个体距离整个种群的虚拟质

心的距离的均值.

$$\overline{TC} = \frac{\sum_{i=1}^N TC_i}{N}. \quad (14)$$

公式(14)用于计算整个种群到达目标区域所需时间的均值 $\overline{TC}$. 其中, TC_i 代表个体 i 到达目标区域所消耗的时间.

根据公式(10)可知, 本研究所需要优化的10个参数 $\{a_1, b_1, c_1, d_1, e_1, a_2, b_2, c_2, d_2, e_2\}$ 为在两种不同情况下的10个不同的用于调节个体方向的力的重要度的系数, 因此在本研究中, 这10个参数的取值范围均设定为(0, 1].

NSGA II算法是一种基于种群进化规则的多目标优化方法, 在大量多目标优化问题中均具有广泛的应用. 本文中使用NSGA II作为优化方法的主要依据有: (1) 本研究所面向的问题需要优化10个无约束参数, 能够较好地使用1维顺序编码进行解的表达; (2) NSGA II算法在问题领域的相关知识均不是很明确的条件下, 也能够通过“解的评价→迭代寻优”的方式对解的质量进行优化; (3) 相对于其他类型的优化算法, NSGA II能够在保证解的质量的同时, 兼顾解在帕累托前沿上的均匀分布.

3.2 编码

根据以上问题分析, 我们决定采用1维顺序编码, 对问题的解进行编码, 具体编码方式如图3所示, 即每条DNA代表完整的一组10个待优化参数, 一条DNA中的每个基因位按顺序代表每个待优化参数. 基于图3编码规则, 在种群初始化时, 针对单个DNA中的每个基因位产生(0, 1]的随机数并赋值, 即完成了种群的初始化.

3.3 交叉

在本研究中, 我们使用如图4的方法进行交叉操作. 在进行交叉操作时, NSGA II算法首先使用锦标赛选择法随机选择两个父代, 然后随机选择一个父代的最多3个基因位, 与另一个父代的同位基因进行交换, 最终产生2个子代加入种群.

3.4 变异

在进行变异操作时, 针对种群中的每一个个体, 设

置变异系数为30%. 当一个个体被选定进行变异时, 通过如图5的操作完成个体变异. 算法首先随机选择一个个体中的最多3个基因位, 用随机生成的(0, 1]之间的随机数代替该基因位上的原数值, 从而完成个体的变异操作.

4 仿真实验及结果

4.1 仿真场景设计

本文设计的仿真场景主要为包括各种典型障碍的目标定向任务. 障碍类型主要包括: 凸障碍、非凸障碍以及通道障碍. 仿真场景如图6所示.

在该场景中, 左下角为无人机起始位置, 右上角四分之一圆代表无人机需要到达的位置, 阴影区域为障碍物. 场景从左到右分别为多障碍场景、带凸障碍的

图 3 种群编码规则

Figure 3 Encoding.

图 4 交叉操作

Figure 4 Crossover operator.

图 5 变异操作

Figure 5 Mutation operator.

图6 仿真场景示意图

Figure 6 Simulation experiment scenarios.

多障碍场景以及带通道障碍的场景。实验的基本设定如下: (1) 每个场景中的每个算法均运行10次; (2) 场地的边长为250 m; (3) 无人机的最大速度设定为5 m/s; (4) 无人机集群的规模为20台; (5) 通信延迟为1 s (即仿真实验中的1步); (6) 考虑到现实问题中无人机的通信范围通常具有一定的波动, 因此设定单个无人机的最大通信范围为29~31 m中的一个随机值; (7) 同样考虑到现实问题中无人机机载载荷的不确定性, 设定单个无人机的最大探测范围为95~105 m中的一个随机数。

本文的对比算法包括四种不同的经典多目标优化算法: MOEA/D, PSO, SPEA2, ICA。其中, MOEA/D (Zhang等人^[24])和SPEA2 (Zitzler等人^[25])为两种经典的基于种群的多目标优化算法。PSO (Kennedy和Eberhart^[26])为模拟鸟类飞行和觅食行为的一种多目标优化算法。ICA (imperialist competitive algorithm)为一种基于种群行为和竞争规则的优化算法, 该方法的详细介绍可见Esmail和Lucas^[27]。最终, 我们将该模型的优化结果和基于经典Reynolds控制模型设计的BREAM控制器以及基于势力场模型设计的APF控制器¹⁾进行了比较。本文描述的所有算法和仿真实验均运行在一台使用Intel® Core™ i5-4460(3.2 GHz), 4 GB内存的计算机上。本文中的所有算法均在每个场景中运行10次, 算法得到的10个帕累托前沿被用于分析算法效能。算法的最大运行代数被设定为100代, 种群规模设定为30。本文所描述的仿真平台代码已被放置于<https://github.com/Downloadmarktown/Flockingexperiment-platform>。

4.2 优化效能分析

本节从两个方面展示本文所描述的优化算法在不同场景下的表现。首先我们展示基于帕累托前沿的基本数据分析, 如表1所示。

在该表中, 我们展示了五种不同的优化算法针对本文所描述的模型在不同类型场景下的优化效果。在表1中, MO (multi obstacles)代表多障碍场景, MOC (multi obstacles with convex obstacle)代表带凸障碍的多障碍场景, T (tunnel)代表通道障碍场景。DR (death rate)为死亡率, AI (aggregation index)为聚集系数, TC (time cost)为完成目标的总时间。需要注意的是, 本表格中TC部分使用百分比的方式展示, 本研究中所有仿真场景统一仿真总长度为200步。从结果可以看出, 本文所使用的NSGA II算法在多障碍场景以及包含凸障碍的多障碍场景下具有较为优秀的表现(包括相对更低的死亡率、相对更好的聚集性表现以及完成目标定向任务消耗的时间较少)。

为了更加全面地评价多目标优化算法的效能, 本文引入了针对多目标优化算法的超体积分析(hyper-volume analysis)方法, 用于评价该算法在不同场景下的具体表现。超体积分析模型通过分析算法求得的帕累托前沿在特定区域内的覆盖率, 分析不同算法之间表现的优劣。由于本文所描述的优化问题的真实上下界未知, 因此采用基于蒙特卡罗仿真方法的超体积分析方法。该分析程序可从<https://ww2.mathworks.cn/ma->

1) Kala R. Code for Robot Path Planning using Artificial Potential Fields. Indian Institute of Information Technology, Allahabad, 2014. Available at: <http://rkala.in/codes.html>

表 1 各场景帕累托前沿分析

Table 1 Pareto Frontier analysis in each scenario

	MOEA/D			PSO			NSGA II			SPEA2			ICA		
	DR	AI	TC	DR	AI	TC	DR	AI	TC	DR	AI	TC	DR	AI	TC
	MO			MO			MO			MO			MO		
Ave.	0.07	0.32	0.38	0.03	0.38	0.36	0.08	0.27	0.39	0.03	0.31	0.35	0.03	0.39	0.39
Min	0.00	0.27	0.33	0.00	0.28	0.33	0.00	0.23	0.33	0.00	0.25	0.33	0.00	0.27	0.37
Max	0.25	0.38	0.50	0.25	0.52	0.50	0.25	0.31	0.50	0.25	0.44	0.50	0.25	0.71	0.40
	MOC			MOC			MOC			MOC			MOC		
Ave.	0.04	0.36	0.42	0.01	0.40	0.40	0.02	0.39	0.41	0.01	0.47	0.40	0.01	0.40	0.43
Min	0.00	0.30	0.39	0.00	0.29	0.38	0.00	0.25	0.35	0.00	0.28	0.37	0.00	0.28	0.35
Max	0.25	0.51	0.55	0.20	0.79	0.52	0.15	0.82	0.49	0.25	0.92	0.55	0.15	0.53	0.82
	T			T			T			T			T		
Ave.	0.09	0.34	0.40	0.03	0.36	0.36	0.03	0.38	0.36	0.02	0.38	0.36	0.03	0.40	0.36
Min	0.00	0.30	0.34	0.00	0.29	0.34	0.00	0.24	0.33	0.00	0.27	0.34	0.00	0.38	0.34
Max	0.25	0.70	0.51	0.25	0.52	0.51	0.25	0.80	0.51	0.25	0.90	0.51	0.25	0.42	0.48

[tlabcentral/fileexchange/30785-hypervolume-computation](https://github.com/tylabcntral/fileexchange/30785-hypervolume-computation)获得. 该仿真方法的图形化描述如图7所示.

该程序在参考上/下界之间随机生成虚拟结果, 通过统计被不同帕累托前沿支配的结果的数量, 分析不同帕累托前沿对当前区域的覆盖能力, 从而进一步比较不同帕累托前沿的优劣. 如图7所示, 帕累托前沿1共支配2个虚拟结果, 因此帕累托前沿1的得分为0.5; 帕累托前沿2支配了3个虚拟结果, 因此帕累托前沿2的得分为0.75. 由于本研究所描述的三个场景的真实上下界均未知, 因此设定参考上界为{0, 0, 0}, 参考下界为不同算法针对不同场景找到的结果的最大值. 三个场景的参考下界如表2所示. 使用上文所述的分析方式最终得到的超体积分分析结果如表3所示.

表3可以得出如下结论: 本文所描述的NSGA II算法在三个场景中相对于其他算法均具有更好的覆盖率表现, 即该算法在不同场景下均表现出更好的种群多样性特征. 同时应注意到, 在不同场景下, NSGA II算法的超体积评价值的变异系数分别为3.07%、2.29%和1.72%, 可以得出结论NSGA II在不同场景下均具有较为稳定的覆盖率. 因此可以得出该算法具有较好鲁棒性的结论. 需要特别指出的是, 由于本文所描述的优化问题的真实上下界未知, 因此超体积分分析中的百分比数值不能直接用于评价算法效能.

图 7 超体积分分析
Figure 7 Hyper-volume analysis.

表 2 场景参考下界
Table 2 Scenario lower bound

	DR	AI	TC
MOC	0.25	0.92	0.55
MO	0.25	0.52	0.5
Tunnel	0.25	0.9	0.51

4.3 控制模型分析

为了研究该控制模型的具体表现, 我们将该模型在不同场景下的表现和基于经典Reynolds模型设计的BREAM控制器以及基于人工势力场法的APF控制器的表现进行了对比. 首先, 为了选出针对不同场景的具

体参数集, 我们引入了基于线性加权聚合法的方法将该多目标优化问题转化为单目标优化问题, 具体操作方法如下:

$$F(s) = w_1 f_{DR}^s + w_2 f_{AI}^s + w_3 f_{TC}^s, \quad (15)$$

其中, $F(s)$ 为该组参数集的评分. $f_{DR}^s, f_{AI}^s, f_{TC}^s$ 为归一化的死亡率、聚集系数和任务完成时间评分. $f_{DR}^s, f_{AI}^s, f_{TC}^s$ 可通过如下公式求得:

$$f_{DR}^s = \frac{r_{max}^{death} - r_{min}^{death}}{r_{max}^{death} - r_{min}^{death}}, \quad (16)$$

$$f_{AI}^s = \frac{d_{max}^{agg} - d_{min}^{agg}}{d_{max}^{agg} - d_{min}^{agg}}, \quad (17)$$

表 3 算法超体积表现分析

Table 3 Hyper-volume analysis of each algorithm

	MOEA/D			PSO			NSGA II			SPEA2			ICA		
	MO	MOC	T	MO	MOC	T	MO	MOC	T	MO	MOC	T	MO	MOC	T
Ave.	0.134	0.183	0.209	0.135	0.182	0.200	0.166	0.207	0.232	0.141	0.190	0.224	0.151	0.193	0.225
Max	0.144	0.192	0.219	0.159	0.195	0.220	0.174	0.215	0.242	0.152	0.203	0.228	0.157	0.209	0.229
Min	0.101	0.169	0.186	0.111	0.142	0.136	0.159	0.200	0.228	0.132	0.178	0.221	0.150	0.183	0.222
Div.	9.01%	4.29%	4.30%	8.74%	10.1%	13.3%	3.07%	2.29%	1.72%	3.96%	4.12%	1.04%	1.43%	4.65%	1.06%

表 4 控制器效能测试结果

Table 4 Performances of each controller

	CTP-NSGA II			BREAM			APF		
	DR	AI	TC	DR	AI	TC	DR	AI	TC
MO	0.00	0.31	0.33	0.20	0.82	0.48	0.20	5.63	0.46
MOC	0.00	0.33	0.40	0.15	1.04	0.50	0.00	4.32	1.00
T	0.00	0.29	0.34	0.10	0.77	0.41	0.00	2.77	1.00

图 8 BREAM控制器仿真测试结果

Figure 8 Performances of BREAM controller in each scenario.

图 9 CTP-NSGA II控制器仿真测试结果

Figure 9 Performances of CTP-NSGA II controller in each scenario.

图 10 APF控制器仿真测试结果

Figure 10 Performances of APF controller in each scenario.

无人机集群质心的平均距离衡量), 使用CTP-NSGA II控制器的无人机集群相对于使用BREAM控制器和APF控制器的无人机集群具有更大的优势. 另外需要注意的是, 从实验结果分析, 基于Reynolds模型设计的CTP-NSGA II控制器和BREAM控制器相对于APF控制器具有不同的种群行为特征. CTP-NSGA II控制器和BREAM控制器在多障碍场景中的表现均倾向于将无人机个体聚集在一定范围内形成一个较大群体共同活动. 而APF控制器多障碍环境下则倾向于将较大群体分裂为多个较小的群体继续完成任务. 因此相比于BREAM控制器, APF控制器表现出更小的死亡率的同时有更大的聚集系数以及目标任务完成时间.

5 结论与讨论

本文研究了一种基于多目标优化的群体智能机器人控制模型优化方法. 本文首先描述了一种基于规则系统的群体智能机器人控制模型. 该模型是一种经典

Reynolds机器人控制模型的变种. 将Reynolds模型和规则系统进行组合的目的在于通过结合基于规则的控制模型和基于虚拟物理规则的控制模型的优点, 使该控制模型能够适应不同的环境. 然后, 我们使用了多个多目标优化算法对该模型进行了优化, 并对比了这些优化算法在不同条件下针对该模型的优化效能. 仿真实验的场景包括各种不同障碍类型下的目标定向任务.

本文的后续研究可包含以下方向: (1) 在本文中, 仅考虑了复杂环境下的目标定向任务, 考虑到无人机集群面临的任務还包括更丰富的形式(如多编队飞行、组成特定队形、障碍物随机出现或可移动等), 因此本文描述的控制系統仍有较大的扩展空间; (2) 通常情况下, 多目标优化算法的优化效果和对问题的了解程度紧密相关, 在未来的研究中, 可以尝试使用其他工具对该控制模型在不同情况下的表现进行分析, 了解模型在不同条件下的运行特征, 从而设计有针对性的优化算子, 对该模型进行进一步的优化; (3) 本文所

描述的仿真场景, 和机器人现实应用问题仍存在一定程度的差异, 如未考虑到机器人可能的最大加速度限

制. 因此, 可以针对性地设计考虑更加全面的仿真场景, 以达到更加贴合实际的仿真效果.

参考文献

- 1 Brambilla M, Ferrante E, Birattari M, et al. Swarm robotics: A review from the swarm engineering perspective. *Swarm Intell*, 2013, 7: 1–41
- 2 Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. *Int J Robot Res*, 1986, 5: 90–98
- 3 Gazi V, Passino K M. Stability analysis of social foraging swarms: Combined effects of attractant/repellent profiles. In: *Proceedings of the 41st IEEE Conference on Decision and Control*. Piscataway: IEEE Press, 2002. 3: 2848–2853
- 4 Spears W M, Spears D F, Hamann J C, et al. Distributed, physics-based control of swarms of vehicles. *Autonomous Robots*, 2004, 17: 137–162
- 5 Fiorini P, Shiller Z. Motion planning in dynamic environments using velocity obstacles. *Int J Robotics Res*, 1998, 17: 760–772
- 6 Berg J V, Lin M, Manocha D. Reciprocal velocity obstacles for real-time multi-agent navigation. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2008. 1928–1935
- 7 Santos V G, Campos M F, Chaimowicz L. On segregative behaviors using flocking and velocity obstacles. In: Hsieh M A, Chirikjian G, eds. *Distributed Autonomous Robotic Systems*. Distributed Autonomous Robotic Systems, Springer Tracts in Advanced Robotics 104. Berlin, Heidelberg: Springer-Verlag, 2014
- 8 La H M, Lim R S, Sheng W, et al. Cooperative flocking and learning in multi-robot systems for predator avoidance. In: *Proceedings of the 2013 IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*. Nanjing: IEEE, 2014
- 9 Woods A C, La H M. Dynamic target tracking and obstacle avoidance using a drone. In: Bebis G, et al, eds. *Advances in Visual Computing*. ISVC 2015. Lecture Notes in Computer Science, vol 9474. Cham: Springer
- 10 Ha S Y, Ha T, Kim J H. Emergent behavior of a cuckoo-smale type particle model with nonlinear velocity couplings. *IEEE Trans Automat Contr*, 2010, 55: 1679–1683
- 11 Meier L, Tanskanen P, Fraundorfer F, et al. PIXHAWK: A system for autonomous flight using onboard computer vision. In: *Proceedings of the 2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011. 2992–2997
- 12 Scott A H, Leal K L. Formation specification for control of active agents using artificial potential fields. *J Intell Robot Syst*, 2018, 95: 279–290
- 13 Hettiarachchi S D. Distributed evolution for swarm robotics. Dissertation for Doctoral Degree. Laramie: University of Wyoming, 2007
- 14 Pugh J, Martinoli A. Parallel learning in heterogeneous multi-robot swarms. In: *Proceedings of the IEEE Congress on Evolutionary Computation*. Piscataway: IEEE Press, 2009. 3839–3846
- 15 Folino G, Forestiero A, Spezzano G. An adaptive flocking algorithm for performing approximate clustering. *Inf Sci*, 2009, 179: 3059–3078
- 16 Yang H X, Zhou T, Huang L. Promoting collective motion of self-propelled agents by distance-based influence. *Phys Rev E*, 2014, 89: 032813
- 17 Zhao M, Su H, Wang M, et al. A weighted adaptive-velocity self-organizing model and its high-speed performance. *Neurocomputing*, 2016, 216: 402–408
- 18 Vászárhelyi G, Virágh C, Somorjai G, et al. Optimized flocking of autonomous drones in confined environments. *Sci Robot*, 2018, 3: eaat3536
- 19 Ueyama A, Isokawa T, Nishimura H, et al. A comparison of grouping behaviors on rule-based and learning-based multi-agent systems. In: Suzuki Y, Hagiya M, eds. *Recent Advances in Natural Computing*. Mathematics for Industry. Tokyo: Springer, 2016
- 20 Vicsek T, Czirók A, Ben-Jacob E, et al. Novel type of phase transition in a system of self-driven particles. *Phys Rev Lett*, 1995, 75: 1226–1229
- 21 Couzin I D, Krause J, James R, et al. Collective memory and spatial sorting in animal groups. *J Theor Biol*, 2002, 218: 1–11
- 22 Ge S S, Cui Y J. Dynamic motion planning for mobile robots using potential field method. *Autonomous Robots*, 2002, 13: 207–222
- 23 Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Computat*, 2002, 6: 182–197
- 24 Zhang Q F, Li H. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Computat*, 2007, 11: 712–731
- 25 Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: *Proceedings of the Evolutionary Methods for Design, Optimisation, and Control*. Barcelona, 2001. 95–100
- 26 Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of International Conference on Neural Networks*. Perth: IEEE, 1995. 1942–1948
- 27 Esmaeil A G, Lucas C. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. In: *Proceedings of the 2007 IEEE Congress on Evolutionary Computation*. Singapore: IEEE, 2007

A multi-objective optimization method for intelligent swarm robotic control model with changeable parameters

WANG Yuan¹, MA Li¹, WANG Ling², HE YongMing¹, QI WeiHua¹ & XING LiNing¹

¹ College of Systems Engineering, National University of Defense Technology, Changsha 410073, China;

² Department of Automation, Tsinghua University, Beijing 100084, China

In recent years, research into self-organized swarm robotics has received much attention. Within this research, development of a general control model that functions under different conditions has been a hot topic. One significant approach is to use a control model with tunable parameters determined through simulation. In this article, we propose a rule-based swarm robotics control model with tunable parameters. To improve the performance, a modified non-dominated sorting genetic algorithm (NSGA II) is embedded into this control model. Multiple scenarios including convex obstacles, square obstacles, and tunnels are used to test the performances of the proposed control model. Comparison is made with three widely applied population-based multi-objective optimization algorithms. Experimental results show that our control model has good performance and is robust under different scenarios.

swarm robotics, control model, multi-objective optimization, evolutionary algorithm, particle swarm optimization

doi: [10.1360/SST-2019-0280](https://doi.org/10.1360/SST-2019-0280)