arXiv:2303.04313v2 [cs.RO] 8 Sep 2023

用于分散式多智能体导航的在线控制屏障函数

Zhan Gao、Guang Yang 和 Amanda Prorok

摘要—**控制障碍函数（控制障碍函数）可在连续域中实现**有保证的安全**多智能体导航。然而，由此获得的导航性能对底层超参数高度敏感。传统方法采用**固定**控制障碍函数（参数预先调节），因此通常无法在障碍密集且动态性很强的环境中取得良好效果：保守的参数值可能导致智能体轨迹效率低下，甚至无法到达目标位置；而激进的参数值则可能导致控制不可行。为了解决这些问题，本文提出**在线控制障碍函数，根据智能体在其邻近区域内的感知信息实时调节超参数。由于控制障碍函数与导航性能之间的明确关系难以建模，我们利用强化学习以无模型方式学习控制障碍函数调节策略。由于策略采用图神经网络（图神经网络）进行参数化，我们能够构建分布式智能体控制器，在局部调整参数值，从而使各智能体表现出不同程度的保守与激进行为。仿真和真实世界实验表明，（*i*）在线控制障碍函数能够解决固定控制障碍函数无法处理的导航场景，并（*ii*）通过适应其他智能体及环境变化来提升导航性能。**

I. 引言

多智能体系统非常适合处理空间分布式任务，而安全、高效的运动规划是实现这些任务的关键基础 [1]–[3]。在多智能体导航领域，基于模型的方法通常假设能够全面了解环境和系统动力学，并要求在部署智能体之前设计明确的目标函数并调优超参数。数据驱动方法能够适用于部分可观测的环境以及难以建模的复杂系统动力学，但往往会牺牲安全性和收敛保证。本研究旨在找到一种折中方案，兼具两类方法的优势。

本文重点研究具有动力学约束的多智能体导航中的分散式控制器设计。不同于经典路径规划问题 [4]–[6],，我们在连续空间中生成反馈控制输入并执行碰撞避免。具体而言，利用控制障碍函数（CBF）和控制李雅普诺夫函数（CLF），可将具有收敛性和安全保证的多智能体导航问题表述为一系列实时优化问题：前者使智能体能够安全移动且避免碰撞，后者则引导它们到达目标状态。CBF 与 CLF 的结合已广泛应用于安全关键控制 [7], [8]。尽管 CBF 能够提供安全保证，传统方法仍要求手动设置控制屏障函数约束中的参数 [7]–[10]。控制屏障函数约束过强可能产生过于保守的轨迹，约束放宽则可能导致过于激进的轨迹；这两种情况都可能造成控制器不可行性，即不存在可行控制，从而使智能体无法驶向目的地。此外，当环境中移动智能体和障碍物数量不断增加、环境变得拥挤时，这类问题会更加频繁地出现，因为控制屏障函数约束的数量会随智能体和障碍物的数量而增加。许多现有研究在部署前预先设置控制屏障函数参数，并在整个导航过程中固定这些参数 [11]–[13]。这意味着每到一个新环境都需要重新设置控制屏障函数参数，也使多智能体系统无法在智能体配置和障碍物分布随时间变化的动态环境中运行。因此，我们旨在开发能够捕捉时变环境状态，并根据这些状态实时调整控制屏障函数参数的方法。

我们提出一种方法论，不再预先手动调参并固定控制障碍函数参数，而是根据智能体和障碍物状态，以实时、去中心化的方式对其进行调优。目标是找到一组随时间变化的最优控制障碍函数参数序列，使其能够适应新的环境配置，并在不同智能体之间调整保守与激进行为的程度（从而实现有助于轨迹冲突消解的平衡）。由于难以明确建模控制障碍函数与导航性能之间的关系，我们使用图神经网络（GNN）对控制障碍函数调优策略进行参数化，并通过无模型强化学习（RL）进行训练。得益于图神经网络固有的分布式特性 [14]–[17], , 所得策略能够以去中心化方式实现，即每个智能体仅凭邻域信息即可在本地执行该策略，从而得到高效且可扩展的解决方案。

相关工作。多智能体控制中的控制障碍函数技术主要分为两类：基于模型的方法 [18]–[21] 和数据驱动方法 [22]–[24]。基于模型的方法需要充分了解环境，并预先固定控制障碍函数参数。控制障碍函数约束关于控制变量是仿射的，因此可以构建二次规划（QP）控制器，为导航提供安全保证。相比之下，数据驱动方法直接使用神经网络近似控制障碍函数 [25]–[27]。然而，这些方法以牺牲安全保证为代价。文献 [28]通过学习备用控制障碍函数，将基于模型的方法与数据驱动方法相结合，以提升安全性。针对控制障碍函数的可行性，文献 [29]为交通汇流问题设计了一种可行性有保证的控制器。文献 [30] 研究了

Zhan Gao、Guang Yang 和 Amanda Prorok 均任职于剑桥大学计算机科学与技术系，地址为 CB3 0FD（电子邮箱：zg292@cam.ac.uk；gy268@cam.ac.uk；asp45@cam.ac.uk）。本研究得到 ERC项目949940（gAIa）资助。

离散时间情形下基于控制障碍函数的模型预测控制器（MPC）的可行性，而 [12] 则引入了与控制障碍函数配合使用的衰减项，以提高模型预测控制的可行性。此外， [31] 扩展了控制障碍函数的 $\mathcal{K}$ 类函数，以实现连续时间下的正向不变性。一项与本文更为相关的工作 [11], 开发了一种支持向量机分类器，用于筛除不可行的控制障碍函数参数，并缩小搜索最优控制障碍函数参数的范围。然而，该工作仅考虑单智能体场景，且所选控制障碍函数参数在导航过程中保持固定。据我们所知，上述工作均未根据智能体局部感知环境的变化在线更新控制障碍函数参数。

贡献。我们的贡献如下:

1) 我们提出了一种在线安全关键框架，能够以分布式方式使控制屏障函数适应动态环境。由于在线调节控制屏障函数参数，该框架继承了传统控制屏障函数的安全保证，并提升了控制器的可行性。

2) 我们使用图神经网络对控制障碍函数调优策略进行参数化，并采用无模型强化学习进行训练。前者支持分布式实现，后者则克服了显式建模控制屏障函数与导航性能之间关系的挑战。

3) 我们通过数值仿真和真实世界实验，在各种环境配置下验证了我们的方法。结果表明，在线控制屏障函数能够处理使用固定控制屏障函数时无法应对的导航场景（即使进行了穷举式参数搜索）。

II. 预备知识

我们介绍分布式多智能体导航中的系统动力学、控制李雅普诺夫函数和控制屏障函数的相关预备知识。

系统动力学 考虑一个包含 N 个智能体的多智能体系统，该系统位于一个含有 M 个静态障碍物 $\{O_j\}_{j=1}^{M}$ 的二维环境中。智能体的动力学形式为

$$\dot{\mathbf{x}}_i = f(\mathbf{x}_i) + g(\mathbf{x}_i)\mathbf{u}_i, \tag{1}$$

其中，$\mathbf{x}_i \in \mathbb{R}^n$ 表示内部状态，$\mathbf{u}_i \in \mathbb{R}^m$ 表示控制输入，$\dot{\mathbf{x}}_i$ 表示 $\mathbf{x}_i$ 对时间 t 的导数，$f(\mathbf{x}_i)$、$g(\mathbf{x}_i)$ 分别为 $i=1$、...、N 的流向量。每个智能体 A_i 都具有感知范围 $\sigma \in \mathbb{R}^+$，能够部分观测整个环境，即其他智能体的状态 $\{\mathbf{x}_j\}_{j\in\mathcal{N}_i}$ 以及半径为 σ 的邻域内障碍物的位置 $\{\mathbf{p}_{\ell,\,o}\}_{\ell\in\mathcal{N}_i}$，其中 $\mathcal{N}_i$ 为 A_i 的邻居集合 —— 见图 1。我们考虑分布式 控制策略

$$\pi_i\Big(\mathbf{u}_i \Big| \mathbf{x}_i, \{\mathbf{x}_j\}_{j\in\mathcal{N}_i}, \{\mathbf{p}_{\ell,o}\}_{\ell\in\mathcal{N}_i}\Big), \text{ for } i = 1,\dots,N \tag{2}$$

利用局部邻域信息，将智能体从初始状态 $\mathbf{X}^{(0)} := \{\mathbf{x}_i^{(0)}\}_{i=1}^{N}$ 引导至目标状态 $\mathbf{X}^{d} := \{\mathbf{x}_i^{d}\}_{i=1}^{N}$。**控制李雅普诺夫函数（控制李雅普诺夫函数）**。控制李雅普诺夫函数用于编码目标到达要求，即满足控制李雅普诺夫函数约束可保证智能体收敛至其目标状态。我们定义如下指数稳定控制李雅普诺夫函数，以确保指数收敛 [32]。

定义 1： 给定智能体A_i的系统动力学 (1)，如果存在一个正数 $\epsilon \geq 0$，使得对于任意 $\mathbf{x}_i \in \mathbb{R}^n$，正定的连续可微函数$V_i(\mathbf{x}_i)$：$\mathbb{R}^n \mapsto \mathbb{R}$是一个指数稳定控制李雅普诺夫函数，

$$\inf_{\mathbf{u}_i\in\mathbb{U}_i}[\mathcal{L}_f V_i(\mathbf{x}_i) + \mathcal{L}_g V_i(\mathbf{x}_i)\mathbf{u}_i + \epsilon V_i(\mathbf{x}_i)] \leq 0, \tag{3}$$

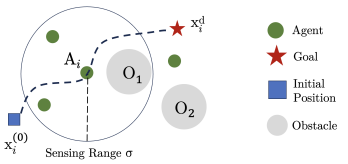

图 1. 智能体 A_i 与其他智能体通信，并感知其感知范围 σ内的障碍物。在此示意图中，有两个关于其他智能体的控制障碍函数约束，以及一个关于障碍物的约束 O_1。

其中 $\mathcal{L}_f V_i(\mathbf{x}_i) := \frac{\partial V_i(\mathbf{x}_i)}{\partial \mathbf{x}} f(\mathbf{x}_i)$ 是 $V_i(\mathbf{x}_i)$ 的李导数， [33] 而 $\mathbb{U}_i$是智能体 A_i的控制空间。

控制障碍函数（CBF）。CBF 旨在避开静态障碍物，并防止移动智能体之间发生碰撞。它确保状态轨迹的正向不变性，即如果智能体从安全集内开始运动，就始终保持在安全集内 [34]。具体而言，我们用一个光滑函数$h_i(\mathbf{x}_i) : \mathbb{R}^n \mapsto \mathbb{R}$ 来编码智能体 A_i 的安全要求，其对时间的导数为

$$\dot{h}_i(\mathbf{x}_i) = \mathcal{L}_f h_i(\mathbf{x}_i) + \mathcal{L}_g h_i(\mathbf{x}_i)\mathbf{u}_i, \tag{4}$$

其中 $\mathcal{L}_f h_i(\mathbf{x}_i) := \frac{\partial h_i(\mathbf{x}_i)}{\partial \mathbf{x}} f(\mathbf{x}_i), \mathcal{L}_g h_i(\mathbf{x}_i) := \frac{\partial h_i(\mathbf{x}_i)}{\partial \mathbf{x}_i} g(\mathbf{x}_i)$ 是 $h_i(\mathbf{x}_i)$ 的李导数。我们将相对阶为一的高阶控制障碍函数定义如下 [35]。

定义 2： 给定智能体 A_i的系统动力学 (1)，若存在严格递增函数 α_i：$\mathbb{R}^+ \to \mathbb{R}^+$，使得对于任意 $\mathbf{x}_i \in \mathbb{R}^n$，可微函数 $h_i : \mathbb{R}^n \mapsto \mathbb{R}$ 是相对阶为一的高阶控制障碍函数，

$$h_i(\mathbf{x}_i) \geq 0,\ \dot{h}_i(\mathbf{x}_i) + \alpha_i\big(h_i(\mathbf{x}_i)\big) \geq 0. \tag{5}$$

这里，$\alpha_i(\cdot)$称为智能体 A_i的 class $\mathcal{K}$ 函数，其由函数参数 η 和 ζ决定。更多细节见 IV-A节。

III. 问题表述

我们首先表述分布式多智能体导航问题，其中使用控制李雅普诺夫函数实现状态收敛，并使用控制障碍函数提供安全保证。随后，我们提出在线控制障碍函数优化问题，该问题根据即时感知状态生成时变控制障碍函数，以优化性能。

A. 分布式多智能体导航

假设智能体和障碍物均为圆盘形，半径分别为$\{R_i\}_{i=1}^{N}$和 $\{R_\ell\}_{\ell=1}^{M}$。令智能体 $\mathcal{A}$的位置、速度和目的地分别为 $\{\mathbf{p}_i\}_{i=1}^{N}$、$\{\mathbf{v}_i\}_{i=1}^{N}$ 和 $\{\mathbf{d}_i\}_{i=1}^{N}$，它们由内部状态 $\{\mathbf{x}_i\}_{i=1}^{N}$决定；令 $\{\mathbf{p}_{\ell,\,o}\}_{\ell=1}^{M}$为障碍物位置。目标是以分布式方式使智能体朝目的地移动，同时避免碰撞。目的地收敛等价于状态收敛，即当 T 为智能体 $i = 1, ..., N$的最大时间步长时，$\lim_{t\to T} \mathbf{x}_i^{(t)} = \mathbf{x}_i^{d}$。碰撞避免等价于对智能体状态施加安全约束，即

$$\mathcal{C}_{i,\mathrm{a}}^{(t)} = \{\mathbf{x}_i^{(t)} \in \mathbb{R}^n \,|\, \|\mathbf{p}_i^{(t)} - \mathbf{p}_j^{(t)}\| \geq \frac{R_i + R_j}{2},\ j \neq i\}, \tag{6}$$

$$\mathcal{C}_{i,\mathrm{o}}^{(t)} = \{\mathbf{x}_i^{(t)} \in \mathbb{R}^n \,|\, \|\mathbf{p}_i^{(t)} - \mathbf{p}_{\ell,\mathrm{o}}^{(t)}\| \geq \frac{R_i + R_\ell}{2},\ \ell = 1, ..., M\}. \tag{7}$$

其中，$\|\cdot\|$ 是向量范数。由此，我们可以将多智能体导航问题表述如下。

问题1（分散式多智能体导航）： 给定具有动力学方程（1）的多智能体系统 $\mathcal{A}$，

初始状态 $\{\mathbf{x}_i^{(0)}\}_{i=1}^N$和满足 $\{f(\mathbf{x}_i^d) = 0\}_{i=1}^N$的目标状态 $\{\mathbf{x}_i^d\}_{i=1}^N$，求分散式策略$\{\pi_i\colon \mathbb{R}^{|\mathcal{N}_i|\times n}\to\mathbb{R}^m\}_{i=1}^N$ [参见（2）]，使得对于 $i=1,...,N$,

$$\lim_{t\to T} \|\mathbf{x}_i^{(t)} - \mathbf{x}_i^d\| = 0, \tag{8}$$

$$\text{s.t. } \mathbf{x}_i^{(t)} \in \mathcal{C}_{i,\mathrm{a}}^{(t)} \cap \mathcal{C}_{i,\mathrm{o}}^{(t)} = \mathcal{C}_i^{(t)}. \tag{9}$$

条件 (8) 保证状态收敛，即导航；条件 (9) 保证状态安全，即碰撞避免。问题1具有挑战性，因为 (*i*) 分散式策略仅利用局部邻域信息生成控制输入；并且 (*ii*)，安全集可能是非凸且随时间变化的，这取决于移动智能体。

我们提出使用基于控制屏障函数-控制李雅普诺夫函数的二次规划（QP）控制器来求解问题1。具体而言，在每个时间步 t，我们可以将二次规划问题表述为

$$\min_{\mathbf{u}_i^{(t)}\in\mathbb{U}_i,\delta_i^{(t)}\in\mathbb{R}} \|\mathbf{u}_i^{(t)}\|_2 + \xi(\delta_i^{(t)})^2 \tag{10}$$

$$\begin{aligned}
\text{s.t. } & \pounds_f h_{i,j}(\mathbf{x}_i^{(t)}) + \pounds_g h_{i,j}(\mathbf{x}_i^{(t)})\mathbf{u}_i^{(t)} + \alpha_i(h_{i,j}(\mathbf{x}_i^{(t)})) \geq 0,\\
& \text{for all } j = 0,\dots,C_i,\\
& \pounds_f V_i(\mathbf{x}_i^{(t)}) + \pounds_g V_i(\mathbf{x}_i^{(t)})\mathbf{u}_i^{(t)} + \epsilon V_i(\mathbf{x}_i^{(t)}) + \delta_i^{(t)} \leq 0,\\
& \mathbf{x}_i^{(t)} \in \mathcal{C}_i^{(t)}, \text{ for all } i = 1,\dots,N,
\end{aligned}$$

其中，$\xi\in\mathbb{R}^+$ 是松弛变量 $\delta_i\in\mathbb{R}$的惩罚权重，根据对控制李雅普诺夫函数的约束需要执行得多严格来选取；C_i 是基于智能体 A_i感知到的智能体和障碍物所确定的控制屏障函数约束数量。例如，在图1中，针对其他智能体有两个控制屏障函数约束，针对障碍物有一个。根据智能体 A_i的物理约束，控制输入的范围为$\mathbf{u}_i^{(t)}\in\mathbb{U}_i$。每个时间步求解一次二次规划，以生成$\mathbf{u}_i^{(t)}$，直至任务完成。控制屏障函数约束中的类 $\mathcal{K}$ 函数 $\alpha_i(\cdot)$决定了我们希望以多严格的程度保证安全性，因此会改变智能体的行为，使其表现得保守或激进。

B. 在线控制障碍函数优化

CBF-CLF-QP 控制器求解问题 (10)，为每个智能体生成一系列控制输入 $\{\mathbf{u}_i^{(t)}\}_{t=0}^T$，从而保证到达目标的收敛性和安全性。然而，在导航过程中，若问题 (10) 在某个时间步 t 无法求解，即在时间步 t 给定控制障碍函数约束和控制李雅普诺夫函数约束时，问题 (10) 不存在可行解，则上述保证可能不再成立。在这种情况下，智能体会保持当前状态下的安全，但由于缺少可行控制输入而停止向目的地前进，最终导致多智能体导航失败。

具体而言，给定系统动力学 (1)、控制障碍函数约束 (5) 和控制李雅普诺夫函数约束 (3)，在时间步 t 满足问题 (10) 中约束的智能体 A_i 的超水平集为

$$\mathbb{U}_{i,\mathrm{CBF,CLF}}^{(t)} := \left\{ \mathbf{u}_i^{(t)} \middle| \begin{array}{l} \pounds_g h_{i,j}(\mathbf{x}_i^{(t)})\mathbf{u}_i^{(t)} \geq -\pounds_f h_{i,j}(\mathbf{x}_i^{(t)}) \\ \quad -\alpha_i\big(h_{i,j}(\mathbf{x}_i^{(t)})\big), \text{for all } j, \\ -\pounds_g V_i(\mathbf{x}_i^{(t)})\mathbf{u}_i^{(t)} \geq \pounds_f V_i(\mathbf{x}_i^{(t)}) \\ \quad +\epsilon V_i(\mathbf{x}_i^{(t)}) + \delta_i^{(t)} \end{array} \right\}. \tag{11}$$

结合式 (11) 与物理约束 $\mathbb{U}_i$，智能体 A_i 的可行解空间为

$$\mathbb{U}_i \cap \mathbb{U}_{i,\mathrm{CBF,CLF}}^{(t)}, \text{ for } i = 1,\dots,N. \tag{12}$$

对于保守的控制障碍函数，得到的约束较为严格，因此在 $\mathbb{U}_i^{(t)}$，控制障碍函数，控制李雅普诺夫函数 s.t. $\mathbb{U}_i \cap \mathbb{U}_i^{(t)}$，控制障碍函数，控制李雅普诺夫函数 $=\varnothing$ 中可能不存在可行解。对于激进的控制障碍函数，得到的约束较为宽松，智能体可能离障碍物或彼此过近。在这些情况下，二次规划控制器可能会生成需要突然变化的控制输入，超出智能体的物理能力 $\mathbb{U}_i$ s.t. $\mathbb{U}_i\cap\mathbb{U}_i^{(t)}$，控制障碍函数，控制李雅普诺夫函数 $=\varnothing$。这两种情况都会导致问题 (10) 不可行，从而造成导航失败，表明控制障碍函数约束存在内在权衡——示例见图 2d–2e。

随着智能体和障碍物数量不断增加，环境变得愈发拥挤，上述问题也随之加剧，这使得手动调节控制障碍函数颇具挑战。此外，在导航过程中固定控制障碍函数，可能无法有效应对存在移动智能体的环境的动态特性；环境发生变化时，即使调节良好的控制障碍函数也可能出现性能下降。这些观察促使我们采用基于即时感知状态的时变控制障碍函数，以调节智能体的保守和激进行为。我们将后者称为在线控制屏障函数。定义分散式控制障碍函数调节策略如下

$$\pi_{i,\mathrm{CBF}}\Big(\alpha_i\Big|\mathbf{x}_i,\{\mathbf{x}_j\}_{j\in\mathcal{N}_i},\{\mathbf{p}_{\ell,\mathrm{o}}\}_{\ell\in\mathcal{N}_i}\Big), \text{ for } i=1,...,N, \tag{13}$$

基于局部邻域信息生成函数类 $\mathcal{K}$ function $\alpha_i(\cdot)$，即控制障碍函数。在每个时间步 t，都会为智能体 A_i生成一个新的函数类 $\mathcal{K}$ function $\alpha_i^{(t)}(\cdot)$，并将其传入控制障碍函数中，以求解（10）并计算控制输入 $\mathbf{u}_i^{(t)}$。给定表示导航性能的任意目标函数 $F(\{\pi_{i,\mathrm{CBF}}\}_{i=1}^N, \mathbf{X}^{(0)}, \mathbf{X}^d)$、初始状态 $\mathbf{X}^{(0)}$ 和目标状态 $\mathbf{X}^d$，我们可以构造在线控制障碍函数优化问题。

问题2（在线控制障碍函数优化）： 给定初始状态 $\mathbf{X}^{(0)}$ 和目标状态 $\mathbf{X}^d$，寻找分散式控制障碍函数调节策略 $\{\pi_{i,\mathrm{CBF}}\}_{i=1}^N$ [参见（13）]，利用局部邻域信息生成在线控制障碍函数，以保证问题（10）的可行性，并最大化目标函数 $F(\{\pi_{i,\mathrm{CBF}}\}_{i=1}^N, \mathbf{X}^{(0)}, \mathbf{X}^d)$。

控制障碍函数调优策略根据动态环境的局部状态在线调整控制障碍函数，从而在固定控制障碍函数无法实现控制可行性的情况下实现控制可行性，同时维持安全保证。生成的时变控制障碍函数在不同智能体的保守与激进行为之间取得平衡。对于智能体轨迹相互冲突的情形（例如，多个智能体需要穿过狭窄空间），这会自然地在智能体之间形成优先级排序，并实现智能体轨迹的冲突消解——示例见图3a。

IV. 方法论

本节将介绍用于求解问题1的分散式CLF-CBF-QP控制器，并利用无模型强化学习和分散式图神经网络求解问题2。我们考虑每个智能体 A_i 对应一个线性系统，其系统动力学如下

$$\begin{bmatrix}\dot{p}_{i,1}\\ \dot{p}_{i,2}\end{bmatrix} = \begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}p_{i,1}\\ p_{i,2}\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_{i,1}\\ u_{i,2}\end{bmatrix}, \tag{14}$$

其中，$\mathbf{p}_i = [p_{i,1}, p_{i,2}]^\top$ 是智能体 A_i（$i=1,...,N$）的位置，$\mathbf{u}_i = [u_{i,1}, u_{i,2}]^\top$是其控制输入。

A. 控制李雅普诺夫函数-控制障碍函数-二次规划控制器

给定智能体 A_i 的目的地 $\mathbf{d}_i = [d_{i,1}, d_{i,2}]^\top$，将李雅普诺夫函数候选定义为 $V_i(\mathbf{x}) = (p_{i,1} - d_{i,1})^2 + (p_{i,2} - d_{i,2})^2$，控制李雅普诺夫函数约束为

$$2(\mathbf{p}_i - \mathbf{d}_i)^\top \mathbf{u}_i + \epsilon V_i(\mathbf{x}_i) + \delta_i \leq 0 \tag{15}$$

对于 $i = 1, \ldots, N$。将障碍函数候选定义为

$$h_{i,j,\mathrm{a}}(\mathbf{x}_i) = (p_{i,1} - p_{j,1})^2 + (p_{i,2} - p_{j,2})^2 - (R_i + R_j)^2, \tag{16}$$
$$h_{i,\ell,\mathrm{o}}(\mathbf{x}_i) = (p_{i,1} - p_{\ell,1,\mathrm{o}})^2 + (p_{i,2} - p_{\ell,2,\mathrm{o}})^2 - (R_i + R_\ell)^2,$$

其中，$h_{i,\,j,\,\mathrm{a}}$ 表示关于智能体A_i 与 A_j之间碰撞避免的函数，$h_{i,\,\ell,\,\mathrm{o}}$ 表示关于智能体 A_i 与障碍物 O_ℓ 之间碰撞避免的函数，其中$\mathbf{p}_{\ell,\,\mathrm{o}} = [p_{\ell,1,\mathrm{o}}, p_{\ell,2,\mathrm{o}}]^\top$ 为障碍物位置。由此得到的控制障碍函数约束为

$$\begin{aligned} 2(\mathbf{p}_i - \mathbf{p}_j)^\top \mathbf{u}_i - 2(\mathbf{p}_i - \mathbf{p}_j)^\top \dot{\mathbf{p}}_j + \zeta_{i,\mathrm{a}}\big(h_{i,j,\mathrm{a}}(\mathbf{x}_i)\big)^{\eta_{i,\mathrm{a}}} \geq 0, \\ 2(\mathbf{p}_i - \mathbf{p}_\ell)^\top \mathbf{u}_i + \zeta_{i,\mathrm{o}}\big(h_{i,\ell,\mathrm{o}}(\mathbf{x}_i)\big)^{\eta_{i,\mathrm{o}}} \geq 0, \end{aligned} \tag{17}$$

其中，$\zeta_{i,\,\mathrm{a}}$，$\eta_{i,\,\mathrm{a}}$ 是智能体 A_i 相对于其他智能体的控制屏障函数参数，而$\zeta_{i,\,\mathrm{o}}$，$\eta_{i,\,\mathrm{o}}$ 是相对于障碍物的参数。在此背景下，每个智能体分别针对其他智能体和障碍物拥有两组控制屏障函数参数。然后，将控制李雅普诺夫函数约束（15）和控制障碍函数约束（17）代入问题（10），即可确定CBF-CLF-QP 控制器。

B. 强化学习

控制障碍函数由类 $\mathcal{K}$ 函数 $\alpha_i(\cdot)$及其参数 $\zeta_{i,\,\mathrm{a}}$，$\eta_{i,\,\mathrm{a}}$（针对其他智能体）和$\zeta_{i,\,\mathrm{o}}$，$\eta_{i,\,\mathrm{o}}$（针对障碍物）确定。这表明，我们可以通过学习每个智能体 A_i的控制屏障函数参数 $\boldsymbol{\zeta}_i = [\zeta_{i,\mathrm{a}}, \zeta_{i,\mathrm{o}}]^\top$，$\boldsymbol{\eta}_i = [\eta_{i,\mathrm{a}}, \eta_{i,\mathrm{o}}]^\top$来学习控制障碍函数。由于难以显式建模控制屏障函数参数与导航性能之间的关系，我们在强化学习领域中构建问题2，并以无模型方式学习CBF调节策略。

我们首先定义一个部分可观测马尔可夫决策过程。在每个时刻 t，智能体由状态 $\mathbf{X}^{(t)} = \{\mathbf{x}_i^{(t)}\}_{i=1}^N$定义。每个智能体 A_i 观测其局部状态 $\mathbf{x}_i^{(t)}$，与其邻近智能体通信，并感知其邻近障碍物，以收集邻域信息 $\{\mathbf{x}_j^{(t)}\}_{j\in\mathcal{N}_i}$ 和 $\{\mathbf{p}_{\ell,\mathrm{o}}\}_{\ell\in\mathcal{N}_i}$。控制障碍函数调优策略 $\pi_{i,\mathrm{CBF}}$ 生成控制屏障函数参数 $\boldsymbol{\zeta}_i^{(t)}$ 和 $\boldsymbol{\eta}_i^{(t)}$，它们是在给定 $\mathbf{x}_i^{(t)}$，$\{\mathbf{x}_j^{(t)}\}_{j\in\mathcal{N}_i}$，$\{\mathbf{p}_{\ell,\,\mathrm{o}}\}_{\ell\in\mathcal{N}_i}$，的条件下关于 $\boldsymbol{\zeta}_i^{(t)}$，$\boldsymbol{\eta}_i^{(t)}$ 的分布。控制屏障函数参数 $\boldsymbol{\zeta}_i^{(t)}$，$\boldsymbol{\eta}_i^{(t)}$ 被输入二次规划控制器（10），后者根据智能体动力学（14）生成控制动作 $\mathbf{u}_i^{(t)}$，驱动局部状态 $\mathbf{x}_i^{(t)}$ 到 $\mathbf{x}_i^{(t+1)}$。奖励函数 $r_i(\mathbf{X}^{(t)})$表示智能体 A_i 在时刻 t的即时导航性能，由两部分组成：(i) 导航奖励$r_{i,\mathrm{nav}}$ 和 (ii) 二次规划的可行性奖励$r_{i,\mathrm{infs}}$，即

$$r_i^{(t)}(\mathbf{X}^{(t)}) = r_{i,\mathrm{nav}}^{(t)}(\mathbf{X}^{(t)}) + \beta_i r_{i,\mathrm{infs}}^{(t)}(\mathbf{X}^{(t)}), \tag{18}$$

其中，β_i 是正则化参数。第一项表示智能体 A_i与任务相关的性能，第二项则对应于二次规划控制器 (10) 的可行性，该控制器使用生成的控制屏障函数参数，例如，它会惩罚二次规划控制器因控制屏障函数过于保守或激进而无可行解的情况。智能体的总奖励为 $r^{(t)} = \sum_{i=1}^N r_i^{(t)}$。在考虑未来奖励的折扣因子 γ 后，期望折扣奖励可表示为

$$R\big(\mathbf{X}^{(0)}, \mathbf{X}^d, \{\mathbf{p}_{\ell,\mathrm{o}}\}_{\ell=1}^M | \{\pi_{i,\mathrm{CBF}}\}_{i=1}^N\big) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r^{(t)}\Big], \tag{19}$$

其中，$\mathbb{E}[\cdot]$ 是关于CBF调节策略的。式(19)中的期望折扣奖励对应于问题2中的目标函数，从而将该问题转化到强化学习领域。通过利用参数为 $\{\boldsymbol{\theta}_i\}_{i=1}^N$的信息处理架构 $\{\pi_{i,\mathrm{CBF}}\}_{i=1}^N$对策略 $\{\boldsymbol{\Phi}_i(\mathbf{x}_i^{(t)}, \{\mathbf{x}_j^{(t)}\}_{j\in\mathcal{N}_i}, \{\mathbf{p}_{\ell,\,\mathrm{o}}\}_{\ell\in\mathcal{N}_i}, \boldsymbol{\theta}_i)\}_{i=1}^N$进行参数化，目标是学习使 $R(\mathbf{X}^{(0)}, \mathbf{X}^d, \{\mathbf{p}_{\ell,\,\mathrm{o}}\}_{\ell=1}^M | \{\boldsymbol{\theta}_i\}_{i=1}^N)$最大化的最优参数 $\{\boldsymbol{\theta}_i^*\}_{i=1}^N$。我们通过策略梯度上升更新 $\{\boldsymbol{\theta}_i\}_{i=1}^N$ 来解决后者。

C. 图神经网络

我们使用图神经网络对CBF调节策略进行参数化，这使得去中心化执行成为可能。它们天然具有排列置换等变性（与智能体排序无关），因此能够泛化到未见过的智能体构型 [36]–[38]。

受控制障碍函数只需要相对信息（例如智能体与障碍物之间的相对位置）这一观察的启发， [参见（17）], 我们设计了一种平移不变图神经网络，利用消息传递机制生成包含相对信息的控制障碍函数参数。对于每个智能体 A_i 及其局部状态$\mathbf{x}_i$、邻近智能体的状态 $\{\mathbf{x}_j\}_{j\in\mathcal{N}_i}$和邻近障碍物的位置 $\{\mathbf{p}_{\ell,\,\mathrm{o}}\}_{\ell\in\mathcal{N}_i}$，该网络通过消息聚合函数$\mathcal{F}_{\mathrm{m},\,\mathrm{a}}$，$\mathcal{F}_{\mathrm{m},\,\mathrm{o}}$ 以及特征更新函数 $\mathcal{F}_{\mathrm{u}}$ 生成控制障碍函数参数，具体如下：

$$\begin{aligned} (\boldsymbol{\zeta}_i^{(t)}, \boldsymbol{\eta}_i^{(t)}) &= \boldsymbol{\Phi}_i(\mathbf{x}_i^{(t)}, \{\mathbf{x}_j^{(t)}\}_{j\in\mathcal{N}_i}, \{\mathbf{p}_{\ell,\mathrm{o}}\}_{\ell\in\mathcal{N}_i}, \boldsymbol{\theta}_i) \\ &= \mathcal{F}_{\mathrm{u}}\Big(\sum_{j\in\mathcal{N}_i} \mathcal{F}_{\mathrm{m,a}}(\mathbf{x}_j - \mathbf{x}_i) + \sum_{\ell\in\mathcal{N}_i} \mathcal{F}_{\mathrm{m,o}}(\mathbf{p}_{\ell,\mathrm{o}} - \mathbf{p}_i)\Big), \end{aligned} \tag{20}$$

其中 $\boldsymbol{\theta}_i$ 是$\mathcal{F}_{\mathrm{m},\,\mathrm{a}}$，$\mathcal{F}_{\mathrm{m},\,\mathrm{o}}$ 和 $\mathcal{F}_{\mathrm{u}}$的函数参数。通过在所有智能体之间共享$\mathcal{F}_{\mathrm{m},\,\mathrm{a}}$，$\mathcal{F}_{\mathrm{m},\,\mathrm{o}}$ 和 $\mathcal{F}_{\mathrm{u}}$，我们得到 $\boldsymbol{\theta}_1 = \cdots = \boldsymbol{\theta}_N$因此 $\boldsymbol{\Phi}_1 = \cdots = \boldsymbol{\Phi}_N$。

基于图神经网络的策略具有以下性质：

1）去中心化执行：函数$\mathcal{F}_{\mathrm{m},\,\mathrm{a}}$，$\mathcal{F}_{\mathrm{m},\,\mathrm{o}}$，$\mathcal{F}_{\mathrm{u}}$仅需要邻域信息，因此该策略可以以去中心化方式执行。2）平移不变性：该策略使用相对信息，并且对 $\mathbb{R}^2$中的平移保持不变。3）置换等变性：函数$\mathcal{F}_{\mathrm{m},\,\mathrm{a}}$，$\mathcal{F}_{\mathrm{m},\,\mathrm{o}}$，$\mathcal{F}_{\mathrm{u}}$ 是齐次的，因此该策略对置换（即智能体重排序）具有等变性。

五、实验

本节对我们的方法进行评估。首先，我们使用四个智能体和四个障碍物进行概念验证。然后，展示我们的方法如何解决固定控制障碍函数无法处理的导航场景。接着，展示我们的方法在障碍物更多的场景中的泛化能力。最后，报告真实世界实验的结果。

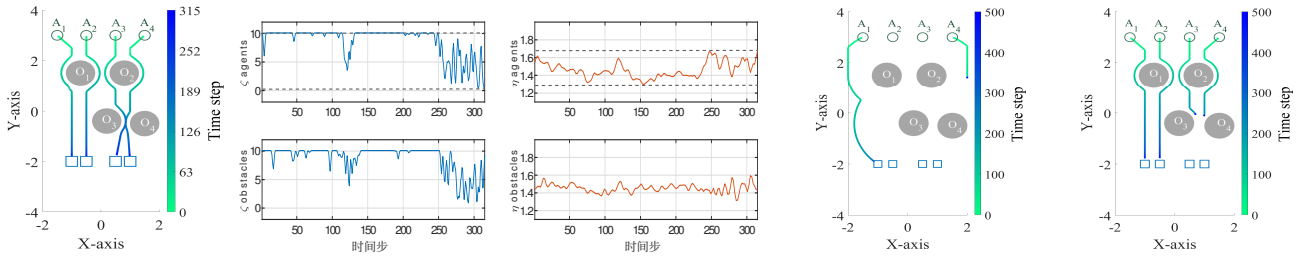

图 2. (a) 采用基于图神经网络的策略生成在线控制屏障函数时的智能体轨迹。绿色圆圈表示初始位置，蓝色方块表示目标位置，灰色圆圈表示障碍物。绿色到蓝色的线表示智能体轨迹，颜色条表示时间尺度。**(b)-(c)** 智能体 A_3 相对于其他智能体和障碍物的时变控制屏障函数参数 $\zeta_{3,\,a}$, $\zeta_{3,\,o}$ 和 $\eta_{3,\,a}$, $\eta_{3,\,o}$。(b)-(c) 顶部图中的垂直线表示时变控制屏障函数参数的最大值和最小值。**(d)** 采用最小固定控制屏障函数参数（即最保守情形）时的智能体轨迹。智能体 A_2、 A_3 和 A_4 的控制屏障函数过于保守，其控制器没有可行解。**(e)** 采用最大固定控制屏障函数参数（即最激进情形）时的智能体轨迹。智能体 A_3 和 A_4 的控制屏障函数过于激进，在 O_3 和 O_4之间的狭窄通道前陷入停滞，此处的控制器没有可行解。

A. 概念验证

我们考虑图 2a 所示的环境。智能体的半径为 0.15m，并在工作空间的上方区域随机初始化，任务是前往下方区域的目标位置。障碍物的半径为 0.5m，分布在智能体的初始位置与目标位置之间。

实现细节。 智能体由位置 $\{\mathbf{p}_i\}_{i=1}^N$ 和速度 $\{\mathbf{v}_i\}_{i=1}^N$ 表示，障碍物由位置 $\{\mathbf{p}_{\ell,\mathrm{o}}\}_{\ell=1}^M$ 表示。在每个时间步，每个智能体根据邻域信息，利用其局部策略生成期望的控制屏障函数参数，并将其输入二次规划控制器，以生成朝向目的地的可行速度。如果所有智能体都到达目的地，或回合超时，则回合结束。感知范围（即通信半径）为 2 米；每个方向上的最大速度为每个时间步 0.5 米；最大时间步数为 500；时间间隔为 0.05 秒。在时间 t，奖励定义为

$$r_i^{(t)} = \left(\frac{\mathbf{p}_i^{(t)} - \mathbf{d}_i}{\|\mathbf{p}_i^{(t)} - \mathbf{d}_i\|_2} \cdot \frac{\mathbf{v}_i^{(t)}}{\|\mathbf{v}_i^{(t)}\|_2}\right)\|\mathbf{v}_i^{(t)}\|_2 + r_{QP}^{(t)}, \qquad (21)$$

其中，第一项用于奖励朝目的地快速移动，第二项表示二次规划控制器的不可行性惩罚。图神经网络的消息聚合和特征更新函数由多层感知机（MLP）构成，训练采用近端策略优化 [39]。**性能**。 图 2a 展示了使用在线控制屏障函数时的智能体轨迹。智能体从初始位置平稳地移动至目的地，且没有发生碰撞。图 2b–2c 分别展示了一个示例智能体的控制屏障函数参数 $\boldsymbol{\zeta}$, $\boldsymbol{\eta}$ 相对于其他智能体和障碍物的变化情况。我们可以看到*(i)* $\boldsymbol{\zeta}$ 的值在其大部分轨迹中都保持最大，这可以理解为放宽控制屏障函数约束，以实现朝目的地的快速移动；*(ii)*，在时间步 100 至 150 之间， $\boldsymbol{\zeta}$ 的值下降，而 $\boldsymbol{\eta}$ 的值上升，使控制屏障函数约束趋于保守，从而避免智能体间拥堵。也就是说， A_3 在准备通过 O_3 和 O_4之间的狭窄通道时，会为 A_4 减速；*(iii)*，在时间步 250 之后， $\boldsymbol{\zeta}$ 和 $\boldsymbol{\eta}$ 的值趋于随机，因为 A_3 已接近目的地，此时控制屏障函数参数的作用微乎其微。

为了展示控制障碍函数中保守与激进行为之间固有的权衡，我们从图中的时变控制屏障函数参数中选取最小值和最大值。图 2b-2c（虚线）对应最保守和最激进的控制障碍函数，我们选取这些参数，使用固定控制障碍函数进行导航 [21], [25] 。对于图 2d 中的最小值， A_2, A_3 停留不动， A_1, A_4 沿环境边界移动，轨迹过于保守，且只有 A_1 到达目的地。对于图 2e 中的最大值，尽管所有智能体都激进地朝目的地移动， A_3 和 A_4 却在 O_3 与 O_4之间的狭窄通道前陷入停滞。这是因为智能体彼此之间以及与障碍物距离过近，导致其控制器没有可行解。这凸显了我们方法的重要性：该方法能够提供在线控制障碍函数。

B. 可行性

我们在三种不同的导航场景中进行测试：狭窄通道、交叉和奇异点。对于固定控制屏障函数，我们采用穷举网格搜索，从 $[0.1, 10] \times [1.0, 2.0]$ 中搜索 100 种组合的最优参数（ζ， η）。

图 3 展示了我们的方法和三种场景下最优固定控制屏障函数的性能。总体而言，结果表明，尽管我们穷举遍历了参数空间，仍然存在固定控制屏障函数无解的场景。相比之下，在线控制障碍函数能够解决这些不可行场景（即所有智能体都能成功到达目的地）。对于图 3a–3b 中的狭窄通道 场景以及图 3c–3d 中的交叉 场景，我们的方法通过对智能体进行优先级排序，并使其采取不同程度的保守／激进行为，从而避免智能体之间发生冲突。对于图 3e–3f 中的奇异点场景，智能体与其目的地在中间的障碍物处对齐。基于固定控制屏障函数的控制器会在可行控制集的边缘或顶点生成控制，因此智能体会陷入局部极小值 [40],; 而我们的方法通过在线控制障碍函数调参，帮助智能体摆脱这种困境。

C. 泛化

我们通过在此前未见过的环境中测试训练好的策略，展示了我们方法的泛化能力。我们考虑两个基线：*(i)* 通过穷举网格搜索得到的最优固定控制障碍函数，以及*(ii)* 参数随机变化的控制障碍函数。第一种方法穷举搜索参数空间，并为固定控制障碍函数选择最优值。由于现有研究主要集中于确定最优固定控制障碍函数

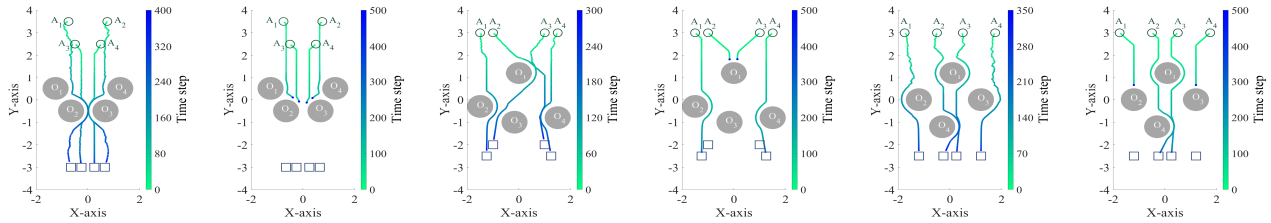

图 3. 不同不可行场景中，由基于图神经网络的调优策略生成的在线控制障碍函数以及通过穷举网格搜索选出的最优固定控制障碍函数对应的智能体轨迹。**(a)** 狭窄通道场景中的在线控制障碍函数。**(b)** 狭窄通道场景中的最优固定控制障碍函数。**(c)** 交叉场景中的在线控制障碍函数。**(d)** 交叉场景中的最优固定控制障碍函数。**(e)** 奇异点场景中的在线控制障碍函数。**(f)** 奇异点场景中的最优固定控制障碍函数。

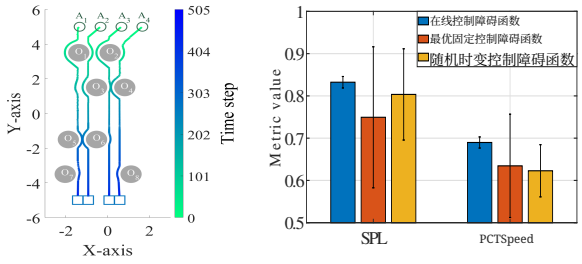

图 4. (a) 用于训练在线控制障碍函数的基于图神经网络的策略，并用于对固定控制屏障函数进行网格搜索的环境。智能体轨迹由训练后的基于图神经网络的策略所生成的在线控制障碍函数产生。**(b)** 比较基于图神经网络的策略生成的在线控制障碍函数、通过穷举网格搜索得到的最优固定控制障碍函数，以及采用随机参数的时变控制障碍函数的性能。

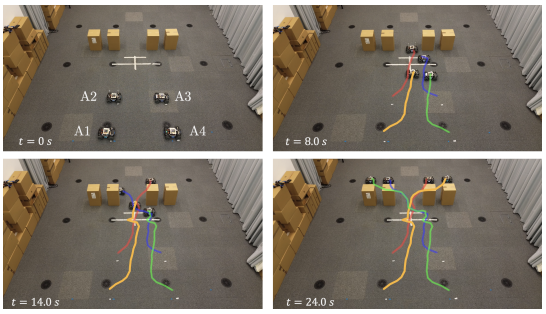

图 5. 使用 DJI Robomasters 进行的真实世界实验。机器人需要穿过狭窄通道并抵达预先设定的目标位置。在线控制屏障函数能够消解机器人之间的冲突。

对于特定任务， [11], [41], [42], 我们考虑这一基线，以近似采用固定控制障碍函数的控制器的最优性能。请注意，这种方法效率低下，此处仅作为参考。第二种方法每 10 个时间步随机选择控制障碍函数参数，其效率与我们的方法相当。

我们考虑包含 8 个障碍物的更大环境，其中最大时间步长为 750。我们在图 4a 所示的环境中训练在线控制屏障函数方法，并通过网格搜索优化固定控制屏障函数，然后通过随机偏移初始位置、目标位置和障碍物位置对它们进行测试。我们采用两个指标衡量性能：*(i)* 路径长度加权成功率（SPL） [43] 和*(ii)* 最大速度百分比（PCTSpeed）。前者是一项严格的指标，综合考量成功率和路径长度；后者表示平均速度与最大速度之比。

图 4b 展示了对 20 次随机初始化结果取平均后的结果。我们的方法在两项指标上均优于基线，具有更高的期望值和更低的标准差。这与理论发现相一致：*(i)* 在线控制障碍函数根据瞬时环境状态协调智能体的保守/激进行为，从而实现无拥堵的平滑导航，获得更高的预期性能；*(ii)*在线控制障碍函数通过冲突消解来解决不可行场景，从而以更低的标准差提升鲁棒性。随机控制障碍函数的 SPL 高于固定控制障碍函数，这是因为它们通过随机确定智能体的优先级，消解了部分不可行场景，并具有更高的成功率。然而，随机控制障碍函数的 PCT-Speed 较低，因为随机协调智能体会导致性能不佳，尽管导航任务能够成功完成。

D. 真实世界实验

我们开展了真实世界实验，以验证我们的方法。我们考虑了一个需要进行冲突消解的狭窄通道场景，用于多机器人导航。我们使用了四台定制的 DJIRobomasters，配备树莓派。每个机器人都处于具有部分可观测空间的环境中，感知范围为 2m，并使用外部遥测系统（OptiTrack）进行定位。我们使用 ROS2 作为通信中间件。在每个时刻，机器人都会接收其当前状态和邻居的状态，并部署分散式控制器进行导航。

图 5 展示了我们的方法如何引导所有机器人无碰撞地抵达各自的目的地，这是因为在线控制屏障函数通过在保守与激进的约束值之间进行调整，消解了机器人之间的冲突。在使用（优化后的）固定控制屏障函数进行相同实验时，机器人无法穿过狭窄通道，因为这种方法无法实现在线冲突消解。这一发现印证了我们的理论分析和数值仿真 [44,视频链接]。

VI. 结论

本文提出了用于分散式多智能体导航的在线 控制障碍函数。我们将多智能体导航问题表述为一个二次规划，其中采用控制李雅普诺夫函数实现状态收敛，并采用控制障碍函数施加安全约束；同时提出了一种在线控制障碍函数优化方法，可根据动态环境中的瞬时状态信息调整控制障碍函数。我们利用强化学习和图神经网络参数化来解决这一问题。前者支持无模型训练，后者则提供分布式智能体控制器。通过仿真和真实世界实验，我们表明，该方法能够协调智能体行为、避免其轨迹冲突并提升整体导航性能，同时确保安全性。未来，我们将把这项工作拓展至高维非线性系统。

参考文献

[1] J. Ota，“多智能体机器人系统作为分布式自主系统”，Advanced Engineering Informatics，第20卷，第1期，第59–70页，2006年。[2] Y. Wang、E. Garcia、D. Casbeer和F. Zhang，“多智能体系统的协同控制：理论与应用”，2017年。[3] A. Oroojlooy和D. Hajinezhad，“协同多智能体深度强化学习综述”，Applied Intelligence，第1–46页，2022年。[4] S. M. LaValle、J. J. Kuffner、B. Donald等，“快速探索随机树：进展与展望”，《算法与计算机器人学：新方向》，第5卷，第293–308页，2001年。[5] G. Wagner和H. Choset，“M*：一种具有性能界的完备多机器人路径规划算法”，载于IEEE/RSJ智能机器人与系统国际会议（IROS），2011年。[6] D. Foead、A. Ghifari、M. B. Kusuma、N. Hanafiah和E. Gunawan，“A*寻路系统性文献综述”，Procedia Computer Science，第179卷，第507–514页，2021年。[7] A. D. Ames、J. W. Grizzle和P. Tabuada，“基于控制障碍函数的二次规划及其在自适应巡航控制中的应用”，载于IEEE决策与控制会议（CDC），2014年。[8] Q. Nguyen和K. Sreenath，“用于强制满足高相对阶安全关键约束的指数控制障碍函数”，载于IEEE美国控制会议（ACC），2016年。[9] S.-C. Hsu、X. Xu和A. D. Ames，“基于控制障碍函数的二次规划及其在双足机器人行走中的应用”，载于IEEE美国控制会议（ACC），2015年。[10] U. Borrmann、L. Wang、A. D. Ames和M. Egerstedt，“用于安全群体行为的控制障碍证书”，IFAC-PapersOnLine，第48卷，第27期，第68–73页，2015年。[11] W. Xiao、C. A. Belta和C. G. Cassandras，“面向约束最优控制问题的可行性引导学习”，载于IEEE决策与控制会议（CDC），2020年。[12] J. Zeng、Z. Li和K. Sreenath，“利用离散时间控制障碍函数提升非线性模型预测控制的可行性与安全性”，载于IEEE决策与控制会议（CDC），2021年。[13] R. Cheng、M. J. Khojasteh、A. D. Ames和J. W. Burdick，“利用学习得到的不确定性和鲁棒控制障碍函数实现安全的多智能体交互”，载于IEEE决策与控制会议（CDC），2020年。[14] Z. Gao和A. Prorok，“面向多智能体导航的环境优化”，arXiv预印本 arXiv:2209.11279，2022年。[15] Q. Li、F. Gama、A. Ribeiro和A. Prorok，“用于去中心化多机器人路径规划的图神经网络”，载于IEEE/RSJ智能机器人与系统国际会议（IROS），2020年。[16] E. Tolstaya、F. Gama、J. Paulos、G. Pappas、V. Kumar和A. Ribeiro，“利用图神经网络学习机器人群体的分散式控制器”，载于机器人学习会议（CoRL），2020年。[17] Z. Gao、F. Gama和A. Ribeiro，“采用分布式在线学习的宽深图神经网络”，IEEE信号处理汇刊，第70卷，第3862–3877页，2022年。[18] L. Lindemann和D. V. Dimarogonas，“冲突的局部信号时序逻辑任务下多智能体系统的控制障碍函数”，IEEE控制系统快报，第3卷，第3期，第757–762页，2019年。[19] X. Tan和D. V. Dimarogonas，“控制障碍函数在多智能体系统中的分布式实现”，IEEE控制系统快报，第6卷，第1879–1884页，2021年。[20] O. ¨Ozkahraman和P. Ogren，“结合控制障碍函数与行为树的多智能体水下覆盖任务”，载于IEEE决策与控制会议（CDC），2020年。[21] M. Srinivasan、S. Coogan和M. Egerstedt，“利用有限时间控制障碍证书和时序逻辑控制多智能体系统”，载于IEEE决策与控制会议（CDC），2018年。[22] Z. Qin、K. Zhang、Y. Chen、J. Chen和C. Fan，“利用去中心化神经障碍证书学习安全多智能体控制”，arXiv预印本 arXiv:2101.05436，2021年。[23] M. Ahmadi、A. Singletary、J. W. Burdick和A. D. A mes，“通过离散时间障碍函数在多智能体部分可观测马尔可夫决策过程中合成安全策略”，载于IEEE决策与控制会议（CDC），2019年。[24] C. Yu、H. Yu和S. Gao，“利用图神经网络学习面向多智能体导航的控制可容许性模型”，arXiv预印本 arXiv:2210.09378，2022年。[25] C. Dawson、Z. Qin、S. Gao和C. Fan，“利用鲁棒神经李雅普诺夫-障碍函数实现安全非线性控制”，载于机器人学习会议（CoRL），2022年。[26] A. Robey、H. Hu、L. Lindemann、H. Zhang、D. V. Dimarogonas、S. Tu和N. Matni，“从专家示范中学习控制障碍函数”，载于IEEE决策与控制会议（CDC），2020年。

[27] P. Pauli、A. Koch、J. Berberich、P. Kohler 和 F. Allg¨ower，《利用 Lipschitz 界训练鲁棒神经网络》，*IEEE*控制系统快报，第 6 卷，第 121–126 页，2021 年。[28] C. Folkestad、Y. Chen、A. D. Ames 和 J. W. Burdick，《数据驱动的安全关键控制：利用 Koopman 算子合成控制障碍函数》，*IEEE*控制系统快报，第 5 卷，第 6 期，第 2012–2017 页，2021 年。[29] K. Xu、W. Xiao 和 C. G. Cassandras，《利用控制障碍函数实现可行性有保证的交通汇流控制》，载于*IEEE*美国控制会议（*ACC*），2022 年。[30] J. Zeng、B. Zhang 和 K. Sreenath，《采用离散时间控制障碍函数的安全关键模型预测控制》，载于*IEEE*美国控制会议（*ACC*），2021 年。[31] J. Breeden、K. Garg 和 D. Panagou，《采样数据系统中的控制障碍函数》，*IEEE*控制系统快报，第 6 卷，第 367–372 页，2021 年。[32] A. D. Ames、K. Galloway、K. Sreenath 和 J. W. Grizzle，《快速指数稳定控制李雅普诺夫函数与混合零动态》，*IEEE Transactions on Automatic Control*，第 59 卷，第 4 期，第 876–891 页，2014 年。[33] H. K. Khalil，《非线性系统（第三版）》，*Patience Hall*，第 115 卷，2002 年。[34] X. Xu、P. Tabuada、J. W. Grizzle 和 A. D. Ames，《安全关键控制中控制障碍函数的鲁棒性》，*IFAC-PapersOnLine*，第 48 卷，第 27 期，第 54–61 页，2015 年。[35] W. Xiao 和 C. Belta，《高相对阶系统的控制障碍函数》，载于*IEEE*决策与控制会议（*CDC*），2019 年。[36] F. Scarselli、M. Gori、A. C. Tsoi、M. Hagenbuchner 和 G. Monfardini，《图神经网络模型》，*IEEE*神经网络汇刊，第 20 卷，第 1 期，第 61–80 页，2009 年。[37] P. Veli˘ckovi´c、G. Cucurull、A. Casanova、A. Romero、P. Li`o 和 Y. Bengio，《图注意力网络》，载于国际学习表征会议（*ICML*），2018 年。[38] Z. Gao、E. Isufi 和 A. Ribeiro，《随机图神经网络》，*IEEE*信号处理汇刊，第 69 卷，4443，2021 年，第 4428 页。[39] J. Schulman、F. Wolski、P. Dhariwal、A. Radford 和 O. Klimov，《近端策略优化算法》，*arXiv* 预印本 *arXiv:1707.06347*，2017 年。[40] L. Wang，《利用障碍证书实现多机器人协调与安全学习》，2018 年。[41] X. Xiao、J. Dufek 和 R. Murphy，《显式运动风险表征》，载于*IEEE*安全、防护与救援机器人国际研讨会（*SSRR*），2019 年。[42] J. Usevitch 和 D. Panagou，《利用控制障碍函数方法提高采样数据系统的对抗韧性》，载于*IEEE*美国控制会议（*ACC*），2021 年。[43] P. Anderson、A. Chang、D. S. Chaplot、A. Dosovitskiy、S. Gupta、V. Koltun、J. Kosecka、J. Malik、R. Mottaghi、M. Savva 等，《具身导航智能体的评估》，*arXiv* 预印本 *arXiv:1807.06757*，2018 年。[44] Z. Gao、G. Yang 和 A. Prorok，《真实世界实验视频》，https://youtu.be/SVVWLnRh1KY，2023 年。