# 到达-规避场景中的协调防御分配与高效在线优化

Junwei Liu、Zikai Ouyang、Jiahui Yang、Hua Chen、Haibo Lu 和 Wei Zhang

arXiv:2304.00234v2 [cs.RO] 3 Jun 2023

**摘要——本文提出了一种双层在线优化策略，用于在一般凸环境中的多人到达-规避博弈中操控防御机器人。我们的目标是在事先不了解攻击者策略的情况下，拦截尽可能多的攻击机器人。为兼顾最优性与效率，我们的方法交替执行针对个体攻击者的防御者联盟协调，以及基于预测的单攻击协调结果将联盟分配给攻击者。我们开发了一种用于单攻击防御协同的在线凸规划技术，该技术不仅能够适应联合状态，还能确定可保证成功拦截攻击的初始联合状态最大区域。我们的防御分配算法采用分层迭代方法，在带有单调性约束的情况下近似求解整数线性规划，从而降低计算负担，同时确保防御性能随时间提升。在二维和三维环境中开展的大量仿真验证了我们方法的有效性，并展示了其在轮式移动机器人和四旋翼飞行器中的适用性。**

索引词—**多机器人系统、协同、任务分配、优化与最优控制。**

## I. 引言

许多多机器人系统要有效执行任务，既需要团队成员之间相互协作，也需要与对抗性智能体展开竞争。多人对抗博弈以目标冲突为特征，是研究协作与竞争之间微妙相互作用的理想平台。基于多人对抗博弈的技术已应用于众多现实场景，包括搜索与救援 [1]，自主车辆 [2]，安全防护 [3]，等。然而，由于解空间维数高、需要快速决策，且未知对手策略会带来不确定性，要以最优且高效的方式求解多人对抗博弈仍是一项艰巨挑战。尽管深度多智能体强化学习 [4]–[6]提供了强有力的工具，但这类方法目前难以在训练过程中应对非平稳性，限制了其在现实对抗场景中的应用。

Junwei Liu 和 Hua Chen 任职于南方科技大学系统设计与智能制造学院，中国深圳 518055 {liujw,chenh6}@sustech.edu.cnZikai Ouyang 和 Wei Zhang 任职于南方科技大学系统设计与智能制造学院，中国深圳 518055，同时任职于鹏城实验室，中国深圳 518000ouyang2022@mail.sustech.edu.cn, zhangw3@sustech.edu.cnJiahui Yang 任职于南方科技大学机械与能源工程系，中国深圳 51805511911213@mail.sustech.edu.cnHaibo Lu 任职于鹏城实验室，中国深圳 518000luhb@pcl.ac.cn

本文深入研究一类称为多人可达-规避博弈的多人对抗博弈。在这类博弈中，两支团队在不了解对手策略的情况下，力求攻击或防御一个目标集。可达-规避博弈与其他类型的对抗博弈密切相关，例如追逃博弈 [7]–[11]，夺旗 [12]–[14]，以及周界防御 [15]–[17]，等。在典型的多人可达-规避博弈中，攻击方团队试图在避免被捕获的同时，尽可能派遣更多机器人抵达目标集；防御方团队则试图通过在目标集之外捕获攻击者，阻止其入侵。本文聚焦于凸环境下多人可达-规避博弈中的防御视角，提出一种有效策略，使防御方团队无论攻击方团队如何决策，都能最大限度地减少成功进入目标集的攻击者数量。

为多机器人系统制定防御策略的一种传统方法是采用可达-规避微分博弈框架。该框架将问题表述为一个由两名对立参与者进行的零和博弈，旨在确定能否引导联合状态到达目标集，同时不进入规避集 [18]，[19]。不同于传统的 Hamilton-Jacobi可达性问题 [20]，，可达-规避微分博弈还因规避集施加的状态约束而面临额外挑战。在可达-规避微分博弈框架下，可以通过离线求解 Hamilton-Jacobi-Isaacs（HJI）方程或其相关变体来生成防御策略 [21]。然而，HJI方法及其数值近似 [22]，[23] 难以消除固有的计算复杂性，导致其在大规模多机器人系统中的可扩展性较差。

为克服这些局限性，本文聚焦于分解方法，该方法将问题划分为两个彼此独立但相互关联的子问题：单攻击防御协调和防御分配。前者涉及为防御者联盟制定协调策略，以阻止单个攻击者到达目标集。如果成功，我们称防御者联盟战胜了攻击者。后者则通过评估所选单攻击协调场景的预测结果，力求将防御者联盟最优地分配给攻击者。通过交替求解这两个子问题，可以获得防御策略。与HJI方法相比，分解方法利用问题结构并提高计算效率，从而提供了更易处理的解。我们为这两个子问题引入在线优化技术，确保我们的方法能够适应动态交互，并具备实时适应性和可扩展性。

作为我们框架的基础，我们提出一种基于在线凸规划的防御协同算法，用于

任意防御者联盟对抗单个攻击者。该算法能够适应防御者联盟与攻击者的每一种联合状态，并识别出初始联合状态的最大防御获胜区域，而不受攻击者采用的任何可容许策略的影响。我们的方法以安全可达集这一概念为基础来设计目标函数。安全可达集表示在当前联合状态下，攻击者能够到达且不会被任何防御者捕获的位置集合。随后，我们将单攻击防御协调问题转化为在线鲁棒最大化问题，其中的鲁棒性考虑了攻击者可能采取的最坏情况。求解这一在线鲁棒最大化问题，即可得到相应的防御协同策略。

基于单攻击防御协同的研究结果，我们开发了一种分配算法，可确保防御性能单调提升。该算法利用单攻击防御协同场景的预测结果，将问题构建为依赖联合状态的整数线性规划（ILP）。我们采用启发式方法求解这些整数线性规划，并加入单调性约束。该约束确保随着博弈推进，预计无法到达目标集的攻击者数量单调不减，从而推动防御性能持续提升。我们的算法核心是一种分层迭代方法，用于近似求解这些整数线性规划；各层次依据主动防御集和不可约性的概念定义，以平衡最优性与效率。通过汇总各层次得到的次优解，我们得出一个解，在保留一定最优性的同时，大幅减少整数线性规划中选定的防御者联盟与攻击者配对数量。

### *A.* 相关工作

本小节概述单攻击防御协同与防御分配领域的当前前沿进展，并指出各领域普遍存在的问题和尚未解决的挑战。

一种单攻击防御协同方法是通过解析方式构造HJI解 [24], [25]。Garcia等人 [24]推导了在三维场景中，两个防御者如何针对单个攻击者实施最优防御策略；该场景的目标集为点目标集，且捕获半径为零。另一种方法采用混合追捕防御策略，将部分防御者指定为追捕者，专注于直接捕获攻击者 [26], [27]。文献 [27],表明，对于一组初始联合状态，在有界二维凸域中，攻击者在圆形目标集之外必定会被捕获。该方法指定一名防御者执行防御，其余防御者则采用基于攻击者Voronoi单元的区域最小化策略，该策略由文献 [28], [29]提出。

近年来，Yan等人提出了一系列基于几何的方法，用于推导防御协同策略 [30]–[33]。这些方法的核心是刻画能够通过可行策略安全到达的攻击者位置的集合。该概念最初在 [34] 中被称为“安全区域”，随后在 [35] 中针对协同追捕被重新表述为“安全可达集”。我们采用“安全可达集”这一术语，以强调安全性与可达性的结合。在捕获半径为零且防御者的最大速度均大于攻击者速度的情况下，安全可达集可退化为阿波罗尼斯圆/球。基于阿波罗尼斯圆（分别为球），文献 [30]（分别为 [32]）针对凸域中涉及两个防御者的初始联合状态集，给出了一种解析防御策略，其中目标集的形状为线段（分别为平面），从而确保攻击者无法进入目标集。这种防御协同技术已扩展到特定二维和三维场景中的任意数量防御者 [31], [33]。值得注意的是， [33] 中提出的方法能够处理捕获半径非零的情况。

尽管单攻击防御协同策略取得了进展，现有方法主要集中于具有特定目标集的具体情况，或捕获半径为零的场景。因此，如何在凸环境中针对一般凸目标集和非零捕获半径给出全面的解，仍是一项尚待解决的挑战。此外，迄今提出的防御策略仅适用于联合空间的一部分，因而可能无法充分应对攻击者偏离最优策略的现实情况。这些局限凸显了开发更具适应性的防御协同方法的必要性，使其能够应用于更广泛的场景。

在防御分配问题中，该问题可以改写为联合状态相关整数线性规划。与经典的多机器人分配问题不同， [36], [37],这里的目标函数取决于对所有防御者共同对抗单个攻击者时结果的估计，直接求解通常会带来较大的计算负担。在 [38],中，作者依靠HJI方法，仅考察防御者与攻击者之间的两两交互。尽管这种方法能够通过Ford-Fulkerson算法 [39] 或其改进算法Hopcroft-Karp算法 [40], 在多项式时间内求解，但它简化过度，降低了防御者之间的协作程度。 [41] 中的研究着重于在假设所有攻击者均可被捕获的前提下进行性能提升分配，这限制了该方法的适用性。根据基于几何的方法所得到的单攻击防御协同结果， [31] 中的整数线性规划仅在初始时刻求解，随后在整个过程中保持所得分配不变。这种方法能够保证捕获一定数量的攻击者，但对于智能体之间的动态交互缺乏鲁棒性。为克服这一局限， [31] 中提出的方法又在[33] 中得到改进，采用多项式时间方法近似求解整数线性规划。然而，这种解需要枚举单攻击防御协同的所有可能组合，在实际应用中可能耗时较长。

根据前文讨论，显然，现有的防御分配方法未能解决确定选定的防御者联盟与个体攻击者对的估计结果时所面临的耗时

问题，尤其是在无法轻易离线提供估计结果的情况下。此外，当由于效率约束而无法获得整数线性规划的最优解时，现有方法也无法为提升协同防御性能提供指导。因此，亟需创新的防御分配方案，在最优性与计算效率之间取得平衡。

### *B.* 贡献

我们的贡献概括为四点 d:

1) 我们的防御策略将以往方法的适用范围扩展至包含一般凸目标集的凸域，适用于二维和三维空间，并可应对防御者具有非零捕获半径的场景。

2) 我们的单攻击防御协调算法通过在线凸规划实现，旨在获得最大防御获胜区域。该算法适用于所有联合状态，而不像现有策略那样仅局限于联合空间的一小部分。因此，在面对非最优攻击策略时，该算法能够提高获胜概率。3) 我们提出了一种防御分配算法，可有效降低确定所选防御者联盟—攻击者对预测结果所需的计算负载。这一效率上的重大进步尤为重要，因为现有方法往往难以应对耗时的计算，从而限制了其实用性。此外，我们的算法旨在逐步提升防御性能，确保其持续改善。当效率约束导致无法实现最优分配时，这一特性尤为宝贵。

4) 我们通过二维和三维场景中的仿真以及真实物理引擎验证，对算法进行了全面评估。严格的测试结果表明，与现有最先进方法相比，我们的算法在最优性和效率方面均有所提升，展现出明显优势。

### *C.* 组织结构与符号说明

本文其余部分的组织结构如下。第二节对多人可达-规避博弈中的协同防御问题进行表述，并将其分解为两个子问题：单攻击防御协调和防御分配。第三节提出联盟针对单个攻击者的防御策略。第四节将防御分配问题重新表述为整数线性规划，并提出一种用于近似求解的启发式方法。最后，第五节给出仿真结果，以评估我们方法的有效性。

本文中我们使用以下符号。令$\mathbb{R}^n$ 为 $n$维欧几里得空间，$||\cdot||$ 为欧几里得范数。给定向量 $v$，归一化函数 $\mathcal{N}(v)$定义为

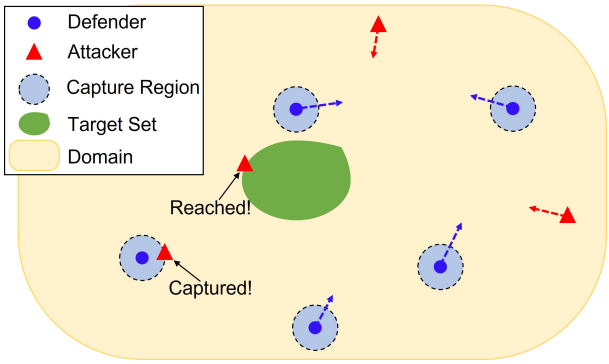



图 1. 多人可达-规避博弈的一个二维实例。

$\mathcal{N}(v) = \frac{v}{||v||}$，若 $v \neq 0$ 成立；否则为 $\mathbf{0}$。在标量情形下，$\mathcal{N}$等价于符号函数。集合 $S$ 的指示函数记为 $\mathbf{1}_S(\cdot)$，定义为 $\mathbf{1}_S(x) = 1$，若 $x \in S$ 成立；否则为 $\mathbf{0}$。集合 $S$ 的基数记为 $|S|$。给定一组向量 $v_1, \ldots, v_n$，用$(v_1, \ldots, v_n)$表示由 $v_i$ 中 $i = 1, \ldots, n$ 依次排列的元素构成的列向量。集合 $S$ 的$N$ 重笛卡尔积记为 $S^N$。

## II. 问题表述

考虑一场多人可达-规避博弈，其中一支防御机器人团队（称为防御者）与一支攻击机器人团队（称为 攻击者）在包含指定目标集的有限域内展开竞争。攻击方团队的主要目标是派出尽可能多的机器人，在避免被任何防御者捕获的同时到达目标集；而防御团队则力求阻止攻击者进入目标集。图 1 展示了该博弈的示意图。本文主要从防御视角出发，旨在制定一种防御团队策略，在不了解攻击者策略的情况下，尽可能减少成功到达目标集的攻击者总数。

我们研究域和目标集分别记为 $\mathcal{D}$ 和 $\mathcal{G}$ 的情形，并假设它们有界、闭且凸。具体而言，我们假设 $\mathcal{D}$ 和 $\mathcal{G}$ 分别由光滑凸函数 $d_k$ 和 $g_l$ 的零次水平集定义，其中 $k$ 和 $j$ 分别属于指标集 $\mathcal{I}_D$ 和 $\mathcal{I}_G$。因此，这些集合可表示如下：

$$\mathcal{D} = \{q \in \mathbb{R}^n \mid \max_{k \in \mathcal{I}_D} d_k(q) \leq 0\} \tag{1}$$

$$\mathcal{G} = \{q \in \mathbb{R}^n \mid \max_{l \in \mathcal{I}_G} g_l(q) \leq 0\}. \tag{2}$$

在此背景下，我们将机器人建模为质点，其中防御者的指标为 $\mathcal{N} = \{1, \ldots, N\}$，攻击者的指标为 $\mathcal{M} = \{1, \ldots, M\}$。此外，假设机器人在域 $\mathcal{D}$ 内以单积分器动力学运动：

$$\dot{p}_i^d = u_i^d, \ \forall i \in \mathcal{N} \tag{3}$$

$$\dot{p}_j^a = u_j^a, \ \forall j \in \mathcal{M} \tag{4}$$

其中，$p_i^d$和 $u_i^d$（分别为 $p_j^a$ 和 $u_j^a$）表示防御者 $i$（分别为攻击者 $j$）的位置和速度。

我们将 $x=(p_1^d、\ldots\ldots、p_N^d、p_1^a、\ldots\ldots、p_M^a)$定义为系统（3）-（4）的联合状态。因此，联合状态满足以下约束：

$$x \in \mathcal{D}^{N+M}. \tag{5}$$

此外，我们将防御者 $i$ 和攻击者 $j$ 的控制输入分别定义为 $u_i^d$ 和 $u_j^a$。这些控制输入需满足以下约束：

$$u_i^d \in \mathcal{U}_i^d = \{v_i^d \in \mathbb{R}^n \mid ||v_i^d|| \le v_{i,\max}^d\},\ \forall i \in \mathcal{N} \tag{6}$$

$$u_j^a \in \mathcal{U}_j^a = \{v_j^a \in \mathbb{R}^n \mid ||v_j^a|| \le v_{j,\max}^a\},\ \forall j \in \mathcal{M}. \tag{7}$$

其中，$v_{i,\max}^d$ 和 $v_{j,\max}^a$ 分别是防御者 $i$ 和攻击者 $j$ 的最大速度。为防止攻击者轻易甩开防御者，我们对最大速度比施加约束，要求防御者具备不低于攻击者的速度能力：

$$\gamma_{ij} = \frac{v_{i,\max}^d}{v_{j,\max}^a} \ge 1,\ \forall (i,j) \in \mathcal{N} \times \mathcal{M} \tag{8}$$

其中，$\gamma_{ij}$ 表示防御者 $i$ 的最大速度与攻击者 $j$ 的最大速度之比。

在多人到达-规避博弈中，防御者与攻击者之间的交互由其相对位置决定。具体而言，如果攻击者 $j$ 与防御者 $i$ 之间的位置距离小于正常数 $r_i$，则称攻击者 $j$ 被防御者 $i$捕获：

$$||p_j^a - p_i^d|| < r_i \tag{9}$$

其中，$r_i$ 表示第 $i$ 个捕获半径。由此，防御者 $i$ 的捕获区域定义为 $\mathcal{R}_i = \{q \in \mathcal{D} \mid ||q - p_i^d|| < r_i\}$。

为简化分析，我们假设已被捕获或已到达目标集的攻击者将保持静止，而防御者在捕获攻击者后可以继续执行其任务。该假设使我们能够专注于活跃攻击者，即既未被捕获也未到达目标集的攻击者。联合状态 $x$ 下的活跃攻击者由 $\mathcal{M}_a(x) = \{j \in \mathcal{M} \mid \rho_j(x_j) = 1\}$ 编号，其中 $x_j = (p_1^d, \ldots, p_N^d, p_j^a)$ 和 $\rho_j$ 定义如下：

$$\rho_j(x_j) = \begin{cases} 1, & \text{attacker } j \text{ is active} \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

考虑到这一点，我们可以将攻击者 $j$ 的输入约束修改如下：

$$u_j^a \in \tilde{\mathcal{U}}_j^a(x_j) = \begin{cases} \{0\}, & \rho_j(x_j) = 0 \\ \mathcal{U}_j^a, & \rho_j(x_j) = 1, \end{cases} \quad \forall j \in \mathcal{M}. \tag{11}$$

在该博弈中，我们采用状态反馈信息模式，其中每个团队根据当前联合状态做出决策，而无法获取对方团队的控制输入。在此信息模式下，防御者 $i$（相应地，攻击者 $j$）的可容许 策略被定义为从联合状态 $x$ 到输入约束集 $\mathcal{U}_i^d$（相应地，$\tilde{\mathcal{U}}_j^a(x_j)$）的映射，使得 $u_i^d = \pi_i^d(x)$（相应地，$u_j^a = \pi_j^a(x)$）。我们将可容许防御策略定义为所有防御者的可容许策略的集合，记为 $\pi^d = (\pi_1^d, \ldots, \pi_N^d)$。类似地，可容许攻击策略记为 $\pi^a = (\pi_1^a, \ldots, \pi_M^a)$，定义为所有攻击者的可容许策略的集合。

多人到达-规避博弈的结果通过收益函数进行评估，该函数量化截至终端时间 $t_f$ 到达目标集的攻击者总数。我们将终端时间 $t_f$ 视为自由的，这意味着博弈将持续进行，直到所有攻击者均已被捕获或到达目标集。给定初始联合状态 $x_0$以及一对可容许防御策略和攻击策略 $\pi^d$ 和 $\pi^a$，收益函数正式定义如下：

$$J(x_0, \pi^d, \pi^a) = \sum_{j \in \mathcal{M}} \mathbf{1}_{\mathcal{G}}(p_j^a(t_f)). \tag{12}$$

防御方团队的目标是通过选择适当的防御策略，在任何可容许攻击策略下，最小化收益函数 $J$ 的值。

**问题 1（多人可达-规避博弈中的协作防御）：** 给定由（1）和（2）定义的域 $\mathcal{D}$ 与目标集 $\mathcal{G}$，以及系统（3）-（4）的初始联合状态 $x_0$，我们旨在寻找一个可容许防御策略 $\pi^d$，使其针对任意可容许攻击策略 $\pi^a$ 最小化收益函数 $J$。这对应于求解以下极小极大最优控制问题：

$$\min_{\pi^d} \max_{\pi^a}\ J(x_0, \pi^d, \pi^a)$$
$$\text{s.t.}\ \ x(0) = x_0,\ (3),\ (4),\ (5),\ (6),\ (8),\ (11).$$

由于多种因素，求解问题 1 面临重大挑战，例如联合状态的高维数、收益函数的离散性，以及智能体之间同时存在协作与竞争交互。一种广泛用于应对这些挑战的方法是采用微分博弈表述，其中涉及求解 HJI方程或其不变集。然而，随着联合状态维数的增加，该方法的计算复杂度变得难以处理，尤其是在多智能体场景中。

为应对上述挑战，我们将问题 1 分解为两个相互关联的子问题：

- **单攻击者防御协调问题：** 给定一个防守者联盟和一个单个攻击者，设计一种联盟的可容许策略，使攻击者无论采用何种可容许策略，都无法到达目标集。在此背景下，联盟是指一组协同合作以防御特定攻击者的防御者。

- **多攻击防御分配问题：** 基于单攻击防御协调问题的解，设计一种分配算法，将联盟分配给活跃攻击者，以最小化收益函数（12）。请注意，活跃攻击者是指尚未被捕获或尚未到达目标集的攻击者。

单攻击者防御协调问题以连续时间方式求解，其中多人到达-规避博弈的结果是二元的：攻击者要么成功到达目标集，要么在目标集之外被捕获。我们称联盟获胜，当且仅当攻击者永远无法进入目标集；否则，联盟被视为战败。因此，单攻击防御

协调问题旨在为联盟找到一种对抗攻击者的获胜策略。相反，多攻击防御分配问题采用离散时间框架处理，其中每个时刻对应一次分配的实施。我们将这些时刻称为分配时刻。在每个分配时刻，分配算法会根据当前联合状态重新评估活跃攻击者到各联盟的分配。

综合这两个子问题的解，即可得到问题1的整体解。给定当前联合状态，我们首先检查截至上一时刻防御者与攻击者之间的相对距离，以确定活跃攻击者集合 $\mathcal{M}_a$。随后，多攻击者防御分配问题的解决定将活跃攻击者分配给各个联盟。对于每个指定的联盟—攻击者对，我们再应用单攻击者防御协调问题的解，计算相应联盟中每个防御者的控制输入。这种双层方案持续实时更新，将高维、复杂的极小极大最优控制问题有效分解为规模更小、更易处理的问题。

## III. 针对单个攻击者的防御协调

本节针对涉及单个攻击者的多人到达-规避博弈，提出一种适用于联盟的防御策略。我们首先形式化定义安全可达集，并推导其子水平集表示。利用安全可达集与目标集之间的最小平方距离，我们构造一个目标函数，将单攻击防御协调问题转化为在线鲁棒最大化问题。该鲁棒最大化问题的解导出一种基于在线凸规划的防御策略，可确保联盟在一类特定初始联合状态区域内获胜。为提高联盟对抗非最优攻击策略时的获胜概率，我们进一步提出一种适用于所有初始联合状态的双模式切换防御策略。

### *A.* 安全可达集

在单个攻击者的多人到达-规避博弈中，初始联合状态的二元结果由可达-规避集 [18]决定。该集合包含攻击者能够到达目标集且不被任何防御者捕获的联合状态。通过数值求解 HJI方程，可将可达-规避集表示为 HJI解 [42]的次水平集。然而，与该方法相关的反向计算会遭遇维数灾难，阻碍其在多智能体系统 [23]中的应用。相比之下，安全可达集旨在收集给定当前联合状态 [35]下攻击者能够到达且仍可避免被捕获的位置。该集合可通过前向计算求得，此计算只涉及攻击者的位置状态，无需求解目标集可达性，因此计算效率更高。我们将证明，在我们的问题设定下，初始联合状态的二元结果也可以通过安全可达集来表征。

下面给出联盟—攻击者对的安全可达集的形式化定义。考虑一个联盟 $\mathcal{C}$ 和一个攻击者 $j$，其动力学分别由 (3) 和 (4) 描述，并将它们的联合状态记为 $x_j^{\mathcal{C}} = (p_i^d,\ i \in \mathcal{C};\ p_j^a)$。根据 (9)，攻击者 $j$ 的无捕获条件可表示如下：

$$\max_{i\in\mathcal{C}} s_i(p_i^d, p_j^a) \leq 0 \tag{13}$$

其中 $s_i(p_i^d,\ p_j^a) = r_i - ||p_i^d - p_j^a||$。为考虑智能体轨迹对其初始状态和策略的依赖性，我们引入记号 $\chi_j^a(\cdot;\ p_j^a(0)\ \pi_j^a)$ 和 $\chi_i^d(\cdot;\ p_i^d(0)\ \pi_i^d)$，分别表示攻击者 $j$ 从 $p_j^a(0)$ 出发并采用攻击策略 $\pi_j^a$时随时间演化的轨迹，以及防御者 $i$ 从 $p_i^d(0)$ 出发并采用防御策略 $\pi_i^d$时随时间演化的轨迹。

**定义1（安全可达集）：** 给定 $(\mathcal{C},\ j)$ 的联合状态 $x_j^{\mathcal{C}}$，攻击者 $j$在联盟 $\mathcal{C}$ 作用下诱导的安全可达集 (SRS) $\Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})$由满足以下性质的所有位置 $q \in \mathcal{D}$组成：

1)（可达性）存在一个时刻 $\tau$ 以及区间 $[0, \tau]$ 上的可容许攻击策略 $\pi_j^a$，使得攻击者 $j$ 能够在时刻 $\tau$到达位置 $q$：

$$q = \chi_j^a(\tau; p_j^a, \pi_j^a);$$

2)（安全性）对于防御者 $i \in \mathcal{C}$采用的任意可容许防御策略 $\pi_i^d$，捕获条件（13）对所有 $t \in [0, \tau]$均成立：

$$\max_{i\in\mathcal{C}} s_i(\chi_i^d(t; p_i^d, \pi_i^d), \chi_j^a(t; p_j^a, \pi_j^a)) \leq 0,\ \forall t \in [0, \tau].$$

SRS通过纳入双方对手的动态约束和输入约束，为评估捕获条件（13）提供了一种预测性方法。当SRS非空时，这意味着攻击者当前位于SRS内，并且根据定义1，当前联合状态满足无捕获条件。此外，SRS的大小可用于预测攻击者在不损害安全性的情况下自由移动的能力。较大的SRS表明，无论采取何种防御策略，攻击者都有更多安全移动的选择。

与可达-规避集类似，SRS 也具有次水平集表示。考虑 SRS 内的一点 $q$，攻击者$j$可在时间 $\tau$到达该点。同时，防御者 $i$可通过常值策略 $v_{i,\ \max}^d \frac{q-p_i^d}{||q-p_i^d||}$到达点 $q_i^d = p_i^d + v_{i,\ \max}^d \frac{q-p_i^d}{||q-p_i^d||}\tau$。根据定义1，$q$满足无捕获条件 $\max_{i\in\mathcal{C}} s_i\ q_i^d\ q = \max_{i\in\mathcal{C}} r_i - \left|||q - p_i^d|| - v_{i\max}^d \tau\right| \leq 0$。另一方面，由于攻击者能够安全地到达 $q$，时间 $\tau$ 必须满足 $\tau \in [\tau_{\min}, \frac{||q-p_i^d||}{v_{i,\max}^d})$，其中 $\tau_{\min} = \frac{||q-p_j^a||}{v_{j\max}^a}$. 表示攻击者 $j$ 到达 $q$ 的最短时间。这意味着 $\max_{i\in\mathcal{C}} r_i - \left|||q - p_i^d|| - v_{i\max}^d \tau_{\min}\right| = \max_{i\in\mathcal{C}} r_i + \gamma_{ij}||q - p_j^a|| - ||q - p_i^d|| \leq 0$，可改写为 $\max_{i\in\mathcal{C}} c_{ij}(q,\ x_{ij}) \leq 0$，其中

$$c_{ij}(q, x_{ij}) = (\gamma_{ij}||q - p_j^a|| + r_i)^2 - ||q - p_i^d||^2 \tag{14}$$

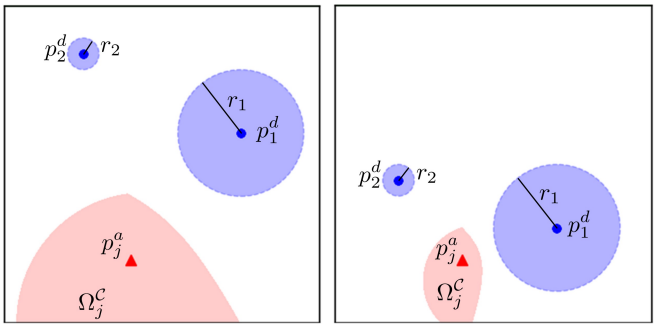



图2. 二维域中不同联合状态下SRS的可视化。蓝色圆盘表示防御者的捕获区域，红色区域展示了特定攻击者的SRS。参数选取为 $\gamma_{1j}=1$ $\gamma_{2j}=2$ $r_1=2$ $r_2=0.5$ 、和。由左至右可见，SRS随联合状态变化；在考虑防御者捕获半径的同时，随着攻击者与防御者之间的距离减小，其大小也随之减小。

以及 $x_{ij}=(p_i^d,\ p_j^a)$。此外，对每个 $i\in\mathcal{C}$ 满足不等式 (14)，足以确保 $q$ 位于 SRS 内。

**命题1（次水平集表示）：** 定义1中定义的SRS是闭且凸的，其形式为

$$\Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})=\bigcap_{i\in\mathcal{C}}\left\{q\in\mathcal{D}\mid c_{ij}(q,x_{ij})\le 0\right\} \tag{15}$$

其中 $c_{ij}$ 由公式（14）给出。

证明：我们已经证明，对于每个$q\in\Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})$，$\max_{i\in\mathcal{C}} c_{ij}(q,x_{ij})\le 0$，其余内容详见附录A。■

SRS通过容纳非零捕获半径，成为Voronoi图和阿波罗尼斯图的推广。如公式（15）所示，当所有捕获半径均为零时，若 $\gamma_{ij}=1$，SRS简化为Voronoi单元；若 $\gamma_{ij}>1$，则简化为阿波罗尼斯圆/球。不同于Voronoi图和阿波罗尼斯图，SRS根据攻击者在不被捕获的情况下自由移动的能力对空间进行划分，同时将防御者的捕获半径纳入考量，如图2所示。SRS的这一独特特性能够更真实地表示攻击者与防御者之间的交互，并通过应对不可预测的攻击者运动来增强防御协同。

### *B.* 单次攻击目标函数

基于 SRS 概念，接下来我们为单攻击者防御协调问题构建一个目标函数。鉴于 SRS 涉及攻击者的前向可达性，因此，自然可以根据 SRS 与目标集之间的最小距离，定义一个将二者联系起来的目标函数。基于这一考虑，我们提出将 SRS 与目标集之间的最小平方距离作为目标函数，并将其称为单次攻击目标函数：

$$\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}})=\min_{\substack{q\in\Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})\\ \tilde{q}\in\mathcal{G}}}||q-\tilde{q}||^2. \tag{16}$$

由于 SRS 和目标集均为凸集且具有次水平集表示，根据命题1和关于目标集的假设，可以通过求解以下参数化凸规划来确定 $\Phi_j^{\mathcal{C}}$ 的值：

$$\begin{aligned}\min_{(q,\tilde{q})\in\mathbb{R}^{2n}}\quad & ||q-\tilde{q}||^2\\ \text{s.t.}\quad & c_{ij}(q,x_{ij})\le 0,\ \forall i\in\mathcal{C}\\ & d_k(q)\le 0,\ \forall k\in\mathcal{I}_D\\ & g_l(\tilde{q})\le 0,\ \forall l\in\mathcal{I}_G\end{aligned} \tag{17}$$

其中，联合状态 $x_j^{\mathcal{C}}$ 作为参数向量。

单次攻击目标函数为单攻击者多防御者可达-规避博弈的二元结果提供了定量度量。需要特别注意的是，对于非空 SRS，$\Phi_j^{\mathcal{C}}$ 的值始终为非负；当 SRS 与目标集的交集为空时，该值为正。这表明，保持 $\Phi_j^{\mathcal{C}}$ 的正值对于防止攻击者到达目标集至关重要。因此，可以根据单次攻击目标函数推导出联盟的获胜保证条件。

**引理 1（防御获胜条件）：** 对于初始联合状态下给定的联盟-攻击者对（$\mathcal{C}$，$j$），当且仅当存在一种可容许防御策略，使得对于任意可容许攻击策略，单攻击目标函数在整个持续期间始终为正：

$$\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}(t))>0,\ \forall t\in[0,t_f).$$

证明：对于充分性，若 $\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}(t))$ 在所有$t\in[0,t_f)$内始终为正，则 SRS 在此时间区间内不会与目标集相交。由于攻击者位置始终包含在非空的 SRS 中，攻击者永远无法进入目标集。这意味着联盟必定赢得博弈。对于必要性，反设联盟获胜，但对于某个 $T\in[0,t_f)$，$\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}(T))=0$。根据定义1，存在一种从时间 $T$ 开始的可容许攻击策略，能在有限时间内将攻击者 $j$ 送入目标集。然而，这与联盟获胜的假设矛盾。■

**注1（在线鲁棒最大化）：** 引理 1 确认，若在可容许防御策略下，$\Phi_j^{\mathcal{C}}$ 的严格零超水平集是鲁棒正不变 [43]，则联盟能够保证战胜任何可容许攻击策略。鲁棒性意味着，联盟必须针对最坏情况攻击策略最大化单攻击目标函数 $\Phi_j^{\mathcal{C}}$。因此，最优防御协调问题可以转化为联盟在每个联合状态下对单攻击目标函数进行在线鲁棒最大化。

### *C.* 防御获胜策略

当单攻击目标函数 $\Phi_j^{\mathcal{C}}$ 取正值时，我们通过对其进行在线鲁棒最大化，继续为联盟 $\mathcal{C}$ 推导防御获胜策略。根据微分博弈的Pontryagin最大值原理（参见 [44]中的定理 8.2），对联盟 $\mathcal{C}$ 的$\Phi_j^{\mathcal{C}}$进行鲁棒最大化，等价于针对输入 $u_i^d$、$i\in\mathcal{C}$对其时间导数 $\dot{\Phi}_j^{\mathcal{C}}$ 进行鲁棒最大化。注意，

$\Phi_j^{\mathcal{C}}$ 沿联合状态轨迹的时间导数为

$$\dot{\Phi}_j^{\mathcal{C}} = \sum_{i\in\mathcal{C}} \frac{\partial \Phi_j^{\mathcal{C}}}{\partial p_i^d} u_i^d + \frac{\partial \Phi_j^{\mathcal{C}}}{\partial p_j^a} u_j^a. \tag{18}$$

为了衡量每个智能体输入对 $\dot{\Phi}_j^{\mathcal{C}}$ 的影响，我们需要确定 $\Phi_j^{\mathcal{C}}$ 关于所有智能体位置的梯度。幸运的是，通过调用 Karush–Kuhn–Tucker（KKT）条件（参见 [44] 中的定理 12.1），可以在几乎处处建立 $\Phi_j^{\mathcal{C}}$ 的梯度与式（14）中给出的 $c_{ij}$、$i \in \mathcal{C}$ 的梯度之间的线性对应关系。

**命题 2（梯度对应关系）：** 假设对于区域 $\mathcal{D}' \subset \mathcal{D}$ 内的每个联合状态 $x_j^{\mathcal{C}}$，凸规划（17）都存在唯一解。那么，对于几乎处处的 $x_j^{\mathcal{C}} \in \mathcal{D}'$，存在非负常数 $\lambda_{ij}^*$、$i \in \mathcal{C}$，使得

$$\begin{aligned} \frac{\partial \Phi_j^{\mathcal{C}}}{\partial p_i^d} &= \lambda_{ij}^* \frac{\partial c_{ij}}{\partial p_i^d}(\xi_j^{\mathcal{C}}, x_{ij}), \quad \forall i \in \mathcal{C} \\ \frac{\partial \Phi_j^{\mathcal{C}}}{\partial p_j^a} &= \sum_{i\in\mathcal{C}} \lambda_{ij}^* \frac{\partial c_{ij}}{\partial p_j^a}(\xi_j^{\mathcal{C}}, x_{ij}) \end{aligned} \tag{19}$$

其中，$\xi_j^{\mathcal{C}}$ 是凸规划（17）唯一解的第一个 $n$ 维分量。

证明：见附录 B。 ■

命题 2 提供了一种可处理的过程，用于生成防御策略，在假设式（20）几乎处处成立的情况下，近似稳健地最大化 $\Phi_j^{\mathcal{C}}$。该过程包括针对每个 $q \in \Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})$ 和 $q \neq p_j^a$ 推导函数 $c_{ij}$、$i \in \mathcal{C}$ 的梯度，如下所示：

$$\begin{aligned} \frac{\partial c_{ij}}{\partial p_i^d}(q, x_{ij}) &= 2(q - p_i^d)^T, \ i \in \mathcal{C} \\ \frac{\partial c_{ij}}{\partial p_j^a}(q, x_{ij}) &= -2\zeta_{ij}(q, p_j^a)(q - p_j^a)^T \end{aligned} \tag{20}$$

其中 $\zeta_{ij}(\xi_j^{\mathcal{C}}, q = 19$ ——— $\lambda_{ij}^*$，$p_j^a) = \gamma_{ij}^2 + \frac{r_i\gamma_{ij}}{||q-p_j^a||}$。将 (20) 代入 ( )，并注意到该参数为非负数，可以得到联盟 $\mathcal{C}$ 中防御者 $i$ 的近乎鲁棒最优防御策略：

$$\pi_i^d(x_j^{\mathcal{C}}) = v_{i,\max}^d \mathcal{N}(\xi_j^{\mathcal{C}} - p_i^d), \ i \in \mathcal{C} \tag{21}$$

对于使凸规划 (17) 具有唯一解的 $x_j^{\mathcal{C}} \in \mathcal{D}'$。此外，可以证明，只要 $\Phi_j^{\mathcal{C}}$ 保持为正，(17) 的解的唯一性就能得到保证。

**命题 3（解的唯一性）：** 对于区域内的每个联合状态$x_j^{\mathcal{C}}$，该区域定义为

$$\mathcal{D}_j^{\mathcal{C}} = \left\{ x_j^{\mathcal{C}} \in \mathcal{D}^{|\mathcal{C}|+1} \mid \Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}) > 0 \right\}$$

凸规划 (17) 具有唯一解。证明：见附录 C。 ■

现在我们可以给出本节的主要结果，即当联合状态 $x_j^{\mathcal{C}}$ 位于区域 $\mathcal{D}_j^{\mathcal{C}}$ 内时，由 (21) 给出的近乎鲁棒最优防御策略能够确保联盟获胜。

**定理 1（制胜防御策略）：** 对于区域 $\mathcal{D}_j^{\mathcal{C}}$ 内的任意初始联合状态，无论攻击者 $j$ 采用何种可容许的攻击策略，联盟 $\mathcal{C}$ 在防御策略 (21) 下都能确保获胜。

证明：根据命题 2，将 (19) 连同 (20) 和 (21) 代入 (18) 可知，对于几乎所有的 $x_j^{\mathcal{C}} \in \mathcal{D}_j^{\mathcal{C}}$，

$$\begin{aligned} \dot{\Phi}_j^{\mathcal{C}} =& \sum_{i\in\mathcal{C}} 2\lambda_{ij}^* v_{i,\max}^d (\xi_j^{\mathcal{C}} - p_i^d)^T \mathcal{N}(\xi_j^{\mathcal{C}} - p_i^d) \\ & - \sum_{i\in\mathcal{C}} 2\lambda_{ij}^* \zeta_{ij}(\xi_j^{\mathcal{C}}, p_j^a)(\xi_j^{\mathcal{C}} - p_j^a)^T u_j^a \\ \geq & \sum_{i\in\mathcal{C}} 2v_{i,\max}^d \lambda_{ij}^* (||\xi_j^{\mathcal{C}} - p_i^d|| - \gamma_{ij}||\xi_j^{\mathcal{C}} - p_j^a|| - r_i) \\ \geq & 0 \end{aligned}$$

其中，第一个和第二个不等式分别由 $||u_j^a|| \leq v_{j\max}^a$ $\xi_j^{\mathcal{C}} \in$ $_j^{\mathcal{C}}$ $x_j^{\mathcal{C}}$ $\Phi_j^{\mathcal{C}}$ $x_j^{\mathcal{C}}$ $t$ 、以及 Ω ( ) 成立。

因此，( ( )) 在时间区间 $[0, t_f)$ 上是非递减的。这意味着，对于所有 $t \in [0, t_f)$，$\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}(t)) \geq \Phi_j^{\mathcal{C}}\ x_j^{\mathcal{C}}(0)) > 0$，进而完成证明，正如引理 1 所确立的那样。■

**注 2（最大防御获胜区域）：** 防御获胜区域 $\mathcal{D}_j^{\mathcal{C}}$ 在以下意义下是最大的：$\mathcal{D}_j^{\mathcal{C}}$ 包含所有联盟 $\mathcal{C}$ 能够确保战胜攻击者 $j$ 的初始联合状态。这是因为，如果初始联合状态 $x_j^{\mathcal{C}}$ (0) 不在 $\mathcal{D}_j^{\mathcal{C}}$ 中，则根据 $\mathcal{D}_j^{\mathcal{C}}$ 的定义，它将导致 $\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}$ (0)) $= 0$。这意味着攻击者 $j$ 可以安全地到达目标集，正如引理 1 所述。因此，通过检查当前联合状态是否位于防御获胜区域 $\mathcal{D}_j^{\mathcal{C}}$ 内，即可完全确定单攻击者多防御者到达-规避博弈的结果。

### *D.* 双模式切换防御协调

我们确定了一个联合状态区域 $\mathcal{D}_j^{\mathcal{C}}$，在该区域内，近乎鲁棒最优防御策略（21）能够确保联盟获胜。该防御策略依赖于凸规划（17）的解，而对于位于 $\mathcal{D}_j^{\mathcal{C}}$ 之外的联合状态，该凸规划通常无法保证解的唯一性。在现实场景中，由于攻击策略未知，防御策略若能处理任意联合状态，而不仅限于区域 $\mathcal{D}_j^{\mathcal{C}}$ 内的联合状态，将更为有利。这样可以提高联盟针对非最优攻击策略获胜的可能性。在本节剩余部分，我们将重点利用 SRS 的概念，将防御策略扩展到覆盖整个联合状态空间。

我们首先针对区域 $\mathcal{D}_j^{\mathcal{C}}$之外的联合状态，提出一种专门设计的新目标函数。单攻击目标函数 $\Phi_j^{\mathcal{C}}$不适用于此目的，因为它无法区分不在 $\mathcal{D}_j^{\mathcal{C}}$中的联合状态，导致所有此类联合状态的函数值均为零。鉴于智能攻击者倾向于安全、快速地到达目标集，且对于 $x_j^{\mathcal{C}} \notin \mathcal{D}_j^{\mathcal{C}}$，SRS 与目标集的交集非空，因此将目标函数定义为攻击者位置与 SRS 和目标集交集之间的最小平方距离：

$$\bar{\Phi}_j^{\mathcal{C}}(x_j^{\mathcal{C}}) = \min_{q\in\Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})\cap\mathcal{G}} ||q - p_j^a||^2. \tag{22}$$

**算法1：** 双模式切换防御协调

**数据：** 防御者 $p_i^d, i \in \mathcal{C}$ 和攻击者 $p_j^a$ 的位置
**结果：** 防御者输入 $u_i^d, i \in \mathcal{C}$
**对于每个防御者** $i \in \mathcal{C}$ **执行**
  $\Phi_j^{\mathcal{C}}$, $\xi_j^{\mathcal{C}} \leftarrow$ 式（17）的最优值和解 $(\mathcal{C}, j)$
  **若** $\Phi_j^{\mathcal{C}} > 0$ **则**
    $u_i^d \leftarrow v_{i,\text{最大值}}^d \mathcal{N}(\xi_j^{\mathcal{C}} - p_i^d)$
  **else**
    (23) 的解，其中 $(\mathcal{C}, j)$
    $u_i^d \leftarrow v_{i,\text{最大值}}^d \mathcal{N}(\bar{\xi}_j^{\mathcal{C}} - p_i^d)$
  **end**
**end**

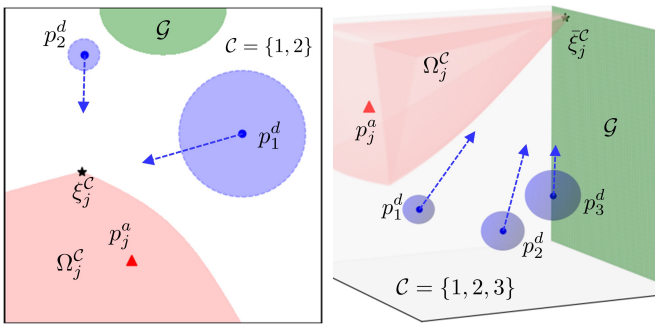



图 3. 二维和三维情形下的防御协同。左图中，近似最优航路点 $\xi_j^{\mathcal{C}}$ 表示 SRS $\Omega_j^{\mathcal{C}}$ 内距离目标集 $\mathcal{G}$ 最近的点。右图中，近似最优航路点 $\bar{\xi}_j^{\mathcal{C}}$ 表示 SRS $\Omega_j^{\mathcal{C}}$ 与目标集 $\mathcal{G}$ 的交集内距离攻击者最近的点。在各种情形下，这两个近似最优航路点都为联盟拦截攻击者提供了直观的参考点。

SRS 的凸性使得能够通过求解以下参数化凸规划来确定 $\bar{\Phi}_j^{\mathcal{C}}(x_j^{\mathcal{C}})$:

$$\begin{aligned} \min_{q \in \mathbb{R}^n} \quad & ||q - p_j^a||^2 \\ \text{s.t.} \quad & c_{ij}(q, x_{ij}) \le 0, \ \forall i \in \mathcal{C} \\ & g_l(q) \le 0, \ \forall l \in \mathcal{I}_G. \end{aligned} \tag{23}$$

由于将 $p_j^a$ 投影到凸闭集 $\Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}}) \cap \mathcal{G}$ 上的结果是唯一的 [45]，因此方程 (23) 的解是唯一的。

随后采用目标函数 $\bar{\Phi}_j^{\mathcal{C}}$ 推导防御策略，使得 $x_j^{\mathcal{C}} \notin \mathcal{D}_j^{\mathcal{C}}$。根据 $\bar{\Phi}_j^{\mathcal{C}}$ 的定义，联盟的目标是随时间鲁棒地最大化 $\bar{\Phi}_j^{\mathcal{C}}$，这可以通过鲁棒地最大化 $\bar{\Phi}_j^{\mathcal{C}}$ 的时间导数来实现。为此，我们需要确定 $\bar{\Phi}_j^{\mathcal{C}}$ 相对于联盟中每个防御者位置的梯度。与命题 2 类似，可以证明，对于几乎所有的 $x_j^{\mathcal{C}} \notin \mathcal{D}_j^{\mathcal{C}}$，都存在一组非负常数 $\mu_{ij}^*$、$i \in \mathcal{N}$，使得

$$\begin{aligned} \frac{\partial \bar{\Phi}_j^{\mathcal{C}}}{\partial p_i^d} &= \mu_{ij}^* \frac{\partial c_{ij}}{\partial p_i^d}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}), \ \forall i \in \mathcal{C} \\ \frac{\partial \bar{\Phi}_j^{\mathcal{C}}}{\partial p_j^a} &= \sum_{i \in \mathcal{C}} \mu_{ij}^* \frac{\partial c_{ij}}{\partial p_j^a}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}) - 2(\bar{\xi}_j^{\mathcal{C}} - p_j^a)^T \end{aligned} \tag{24}$$

其中，$\bar{\xi}_j^{\mathcal{C}}$ 是凸规划 (23) 的唯一解。由于 (24) 的证明与命题 2 的证明相似，其证明可参见附录 B。考虑到 (24) 几乎处处成立，将 (20) 中的 $q = \bar{\xi}_j^{\mathcal{C}}$代入 (24)，可得针对 $x_j^{\mathcal{C}} \notin \mathcal{D}_j^{\mathcal{C}}$的近乎鲁棒最优防御策略：

$$\pi_i^d(x_j^{\mathcal{C}}) = v_{i,\max}^d \mathcal{N}(\bar{\xi}_j^{\mathcal{C}} - p_i^d), \ \forall i \in \mathcal{C}. \tag{25}$$

该防御策略与针对 $x_j^{\mathcal{C}} \in \mathcal{D}_j^{\mathcal{C}}$ 的防御策略 (21) 一起，构成了适用于整个联合状态空间的双模式切换防御策略。协调针对单个攻击者的防御的算法如算法1所示。

**注释3（防御协调的近似最优航路点）：** 算法1表明，$n$维向量定义为

$$\tilde{\xi}_j^{\mathcal{C}} = \begin{cases} \xi_j^{\mathcal{C}}, & x_j^{\mathcal{C}} \in \mathcal{D}_j^{\mathcal{C}} \\ \bar{\xi}_j^{\mathcal{C}}, & \text{otherwise} \end{cases} \tag{26}$$

通过求解凸规划 (17) 或 (23) 得到的该点是联盟的近似最优航路点。根据联合状态是否位于区域 $\mathcal{D}_j^{\mathcal{C}}$ 内，如图 3 所示，该航路点为联盟中的防御者提供了协调行动的参考点。遵循这一近似最优航路点，联盟便能有效调整其防御策略，以抵御单个攻击者。

**注 4（个体攻击策略）：** 作为我们分析的副产物，我们还可以推导出单个攻击者的策略。与防御者的目标相反，攻击者寻求对 $x_j^{\mathcal{C}} \in \mathcal{D}_j^{\mathcal{C}}$ 鲁棒地最小化 $\Phi_j^{\mathcal{C}}$，并对 $x_j^{\mathcal{C}} \notin \mathcal{D}_j^{\mathcal{C}}$ 鲁棒地最小化 $\bar{\Phi}_j^{\mathcal{C}}$。采用与推导防御策略类似的方法，我们得到以下攻击策略：

$$\pi_j^a(x_j^{\mathcal{C}}) = v_{j,\max}^a \mathcal{N}(\tilde{\xi}_j^{\mathcal{C}} - p_j^a) \tag{27}$$

其中，定义如（26）所示的 $\tilde{\xi}_j^{\mathcal{C}}$ 也可作为攻击者的近似最优航路点。本文省略了详细推导，因为我们的主要关注点在于防御视角。

## IV. 多攻击者防御分配

本节深入探讨将联盟分配给多个攻击者的动态问题。在此前关于单攻击场景下防御协同的研究结果基础上，我们将防御分配问题重新表述为联合状态依赖整数线性规划（ILP）。然而，随着智能体规模增大，实时求解该 ILP 可能会变得计算上不可行。为解决这一问题，我们提出一种替代方案，结合两个关键组成部分：分层整数线性规划（HILP）方法和单调防御增强分配（MDEA）算法。HILP 方法通过迭代构建层次结构

该层次结构基于主动防御集和不可约性的概念，从而得到一种计算高效的次优解。随后，MDEA 算法将该次优解纳入其中，同时施加单调性约束，确保未能到达目标集的攻击者预期数量随时间推移保持非递减。

### *A.* 整数线性规划公式

根据 Gerkey 等人提出的经典多机器人任务分配分类体系，防御分配问题可归类为单任务机器人、多机器人任务和瞬时分配问题。在当前联合状态下，每个防御者一次最多可防御一个攻击者，每个攻击者可同时由多个防御者进行防守，并且分配仅涉及当前时刻。为表示该分配，使用一个称为分配矩阵的二进制矩阵 $\mathcal{A} = (a_{ij})_{N\times M}$，其中当攻击者 $j$ 被分配给防御者 $i$时，元素 $a_{ij} = 1$，否则为 0。此外，由于所有防御者均为单任务机器人，分配矩阵必须满足无冲突 约束，即分配矩阵的行和不应超过 1。

为更好地建模每个联盟-攻击者对对防御性能的影响，我们引入一种替代矩阵表示~。设 $\mathcal{C}_k$、 $k = 1$、.....、 $N$ 为所有不同联盟的列表，其中 $\tilde{N} = 2^N - 1$。我们将联盟分配矩阵 定义为 $\Theta = (\theta_{kj})_{\tilde{N}\times M}$，使得

$$\theta_{kj} \in \{0, 1\},\ \forall k = 1, \ldots, \tilde{N},\ j = 1, \ldots, M \tag{28}$$

并且 $\theta_{kj} = 1$如果攻击者 $j$ 被分配至联盟 $\mathcal{C}_k$。联盟分配矩阵 $\Theta$ 通过以下线性变换自然地诱导出分配矩阵 $\mathcal{A}$：

$$\mathcal{A} = \mathcal{B}\Theta \tag{29}$$

其中， $\mathcal{B} = (b_{ik})_{N\times\tilde{N}}$ 表示<style id='4'>关联矩阵，用于表示防御者与联盟之间的二元关系。具体而言，当 $b_{ik} = 1$ 时为 $i \in \mathcal{C}_k$，否则为 0。为保持单任务属性，通过诱导出的分配矩阵定义无冲突约束如下：

$$\sum_{j=1}^{M}\sum_{k=1}^{\tilde{N}} b_{ik}\theta_{kj} \le 1,\ \forall i = 1, \ldots, N. \tag{30}$$

此外，还对 Θ施加<style id='1'>无冗余 约束：

$$\sum_{k=1}^{\tilde{N}} \theta_{kj} \le 1,\ \forall j = 1, \ldots, M \tag{31}$$

确保每个攻击者至多被分配至一个联盟。

现在，我们提出一个目标函数，其中决策变量是联盟分配矩阵的元素。回顾定理 1，若在联合状态 $x_j^{\mathcal{C}_k}$ 下存在 $\mathcal{C}_k$ 的可容许防御策略，可阻止攻击者 $j$ 到达目标集。鉴于此，若 $\Phi_j^{\mathcal{C}_k}(x_j^{\mathcal{C}_k}) > 0$，则我们将 $(\mathcal{C}_k,\ j)$ 在 $x_j^{\mathcal{C}_k}$ 处指定为一个可行的联盟-攻击者对。根据式 (12) 中给出的收益函数，将攻击者 $j$ 分配给联盟 $\mathcal{C}_k$ 的奖励记为 $w_{kj}$；若 $(\mathcal{C}_k,\ j)$ 可行且攻击者 $j$ 处于活动状态，则设为 1，否则设为 0，即

$$w_{kj}(x_j) = \rho_j(x_j)\mathcal{N}(\Phi_j^{\mathcal{C}_k}(x_j^{\mathcal{C}_k}))$$

其中 $\rho_j$ 已在式（10）中给出。随后，在给定联合状态 $x$ 下，根据联盟分配矩阵 Θ的选择，将目标函数定义为所有已分配联盟—攻击者对的奖励总和：

$$\Gamma(\Theta, x) = \sum_{k=1}^{\tilde{N}}\sum_{j=1}^{M} w_{kj}(x_j)\theta_{kj}. \tag{32}$$

我们将 $\Gamma$ 称为<style id='2'>多重攻击目标函数，其表示从当前联合状态无法访问目标集的攻击者的期望数量。采用该目标函数的原因在于，它能够评估多重攻击场景下的防御性能。

**引理 2（防御性能评估）：** 令 $\pi^d$ 为一种可容许防御策略，并与时变联盟分配矩阵 $\Theta(\cdot)$ 相关联，且这些矩阵始终满足约束（28）、（30）和（31）。假设存在一个时刻 $t_0 \ge 0$，使得对于所有可容许攻击策略，

$$\Gamma(\Theta(t), x(t)) + N_c(t) \ge M_c \tag{33}$$

在时间区间 $[t_0, t_f)$ 上成立，其中 $N_c(t)$ 表示截至时刻 $t$ 被捕获的攻击者数量，且 $M_c$ 为正整数。则防御策略 $\pi^d$ 下的收益函数 $J$ 有界于 $M - M_c$，即

$$J(x(0), \pi^d, \pi^a) \le M - M_c$$

对于任意可容许攻击策略 $\pi^a$。

证明：令 $N_e(t)$ 表示截至时刻 $t$ 已进入目标集的攻击者数量。在时刻 $t$，每个攻击者必然处于被捕获状态、活跃状态，或已经进入目标集。因此，我们有 $M = N_e(t) + M_a(t) + N_c(t)$。由于多重攻击目标函数 $\Gamma$ 仅涉及活跃攻击者，因此可得 $\Gamma(\Theta(t)\ x(t)) \le M_a(t)$。结合不等式（33），对于所有 $t \in [t_0, t_f)$，有 $M_c \le \Gamma(\Theta(t)\ x(t)) + N_c(t) \le M_a(t) + N_c(t)$，这意味着对于所有 $t \in [t_0, t_f)$，$N_e(t) \le M - M_c$。因此，根据收益函数的定义，我们得到 $J(x(0)\ \pi^d, \pi^a) = N_e(t_f) \le M - M_c$。 ■

**备注 5（在线最大化与单调性约束）：** 引理 2 表明，针对任何可行的攻击策略，最小化收益函数 $J$ 的一种方法是在每个联合状态 $x$ 下，最大化关于联盟分配矩阵 $\Theta$ 的多重攻击目标函数 $\Gamma$。由此可得沿联合状态轨迹求解多攻击防御分配问题的以下整数线性规划（ILP）：

$$\begin{aligned} &\max_{\Theta}\ \Gamma(\Theta, x) \\ &\text{s.t.}\ \ (28), (30), (31). \end{aligned} \tag{34}$$

此外，为确保整体防御性能，引理 2 强调了维持函数 $\Gamma + N_c$ 在级别$M_c > 0$下的鲁棒正不变超水平集的重要性。这里，值 $\Gamma(\Theta(t)\ x(t)) + N_c(t)$ 表示未能到达目标集的攻击者的预期数量，

时间 $t_o$。保持超水平集性质的一个充分条件是施加单调性约束，具体如下：

$$\begin{aligned}&\Gamma(\Theta(t_l),x(t_l))+N_c(t_l)\\&\geq\Gamma(\Theta(t_{l-1}),x(t_{l-1}))+N_c(t_{l-1}),\ \forall l=1,\ldots,L\end{aligned}\tag{35}$$

其中，$t_l$、$l=0$、.....、$L$ 为分配时刻，且 $t_0=0$。值得注意的是，整数线性规划 (34) 的最优解自动满足单调性约束 (35)。

尽管求解整数线性规划 (34) 可得到防御分配问题的最优解，但这会带来巨大的计算负担，尤其是在智能体数量增加时。其主要原因是可能的联盟-攻击者对数量庞大，这不仅导致整数线性规划规模巨大，更重要的是，确定多重攻击目标函数需要耗费大量计算开销。对于 $N$ 名防御者和 $M_a$ 名活跃攻击者，共有 $NM_a$ 个可能的联盟-攻击者对，这意味着需要计算同等数量的形式为 (17) 的凸规划，以确定相应的奖励。为减轻计算负担，我们在第 IV-C 节提出一种用于整数线性规划 (34) 的启发式算法，以生成次优解；随后在第 IV-D 节将该解输入分配算法，以确保满足单调性约束 (35)。具体而言，在详细介绍该启发式算法之前，我们将在下一节推导两个关键组成部分，它们共同构成一个层次结构，有助于生成整数线性规划 (34) 的次优解。

### *B.* 主动防御集与不可约性

整数线性规划（ILP）(34) 的最优解倾向于选择由较小联盟构成的可行联盟-攻击者对。一方面，多重攻击目标函数仅受可行联盟-攻击者对影响，因为不可行对的奖励为零。另一方面，较小的联盟不太可能违反无冲突约束 (30)，因此更有可能对多重攻击目标函数产生正向贡献。因此，在求解整数线性规划（ILP）(34) 时，兼顾最优性与效率的关键在于：以较低的计算成本，尽可能多地找出防御者数量较少的联盟-攻击者对的可行子对。这里，联盟-攻击者对的子对 是指联盟的一个子集，且该子集与同一攻击者配对。为此，后续分析将从最优性和效率两个方面着手，寻找联盟-攻击者对的可行子对。

作为出发点，我们重新考察满足可行联盟-攻击者对 $(\mathcal{C},j)$ 的凸规划 (17)。如定理 1 所示，从 (17) 的解中提取的向量 $\xi_j^{\mathcal{C}}$ 为联盟 $\mathcal{C}$ 确定了一个近似最优航路点，该点必须位于 SRS $\Omega_j^{\mathcal{C}}$ 的边界上。此外，由于 SRS 的凸性，在 $q=\xi_j^{\mathcal{C}}$ 处必有某些不等式约束 $c_{ij}(q,x_{ij})\leq 0,\ i\in\mathcal{C}$ 处于活动状态。因此，我们可以通过凸规划 (17) 中包含该近似最优航路点 $\xi_j^{\mathcal{C}}$ 的活动集来定义一个防御者索引集。

**定义 2（主动防御集）：** 假设联盟-攻击者对 $(\mathcal{C},\ j)$ 在联合状态 $x_j^{\mathcal{C}}$ 下可行。$(\mathcal{C},\ j)$ 的主动防御集（ADS） $\Lambda_j^{\mathcal{C}}(x_j^{\mathcal{C}})$ 是以下内容的活动集

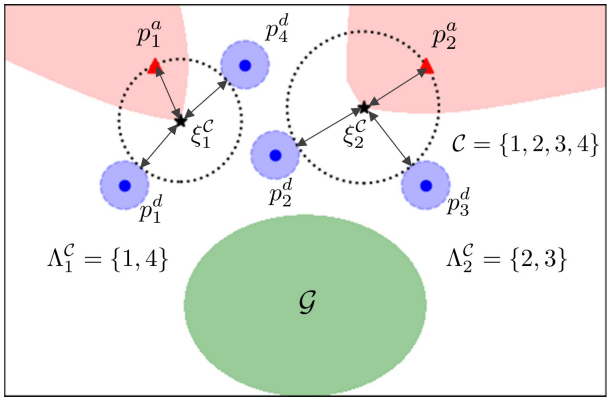



图4. 在 $\gamma_{ij}=1$ 情形下，二维域中 ADS 的几何解释。在此情形下，ADS $\Lambda_j^{\mathcal{C}}$ 是防御者索引的集合，其中每个防御者 $i\in\Lambda_j^{\mathcal{C}}$ 的捕获区域（以蓝色圆盘表示）均与以近似最优航路点 $\xi_j^{\mathcal{C}}$ 为圆心、半径为 $||\xi_j^{\mathcal{C}}-p_j^a||$ 的虚线圆相切。

在近似最优航路点 $\xi_j^{\mathcal{C}}$ 处，式 (17) 中的不等式约束 $c_{ij}(q,x_{ij})\leq 0,\ i\in\mathcal{C}$，即

$$\Lambda_j^{\mathcal{C}}(x_j^{\mathcal{C}})=\left\{i\in\mathcal{C}\mid c_{ij}(\xi_j^{\mathcal{C}},x_{ij})=0\right\}.\tag{36}$$

ADS 的构造不需要计算任何额外的凸规划。这是因为联盟-攻击者对的可行性已经通过形式为（17）的凸规划得到验证，而近似最优航路点也可以作为该计算的副产品确定。值得注意的是，ADS 概念确实提供了一种基于邻近性的方法，用于获取一个特定的子对；如图4所示，该子对同时与攻击者的位置和目标集的位置相关联。此外，可以证明，这样的子对继承了原联盟-攻击者对的可行性。

**命题 4（可行性）：** 如果联盟-攻击者对（$\mathcal{C},\ j$）在联合状态 $x_j^{\mathcal{C}}$ 下是可行的，那么子对（$\Lambda_j^{\mathcal{C}},\ j$）在相应的联合状态下也是可行的。证明：参见附录D。■

下面讨论如何生成规模足够小的可行子对。尽管 ADS 能够在不增加计算开销的情况下生成可行子对，但所得子对的规模相对于可行性而言可能并非最小。换言之，子对中可能包含一些冗余防御者，在不影响子对可行性的情况下可以将其移除。为准确量化这种冗余性，我们引入以下定义。

**定义 3（不可约性）：** 若在联合状态 $x_j^{\mathcal{C}}$ 下，不存在 $\mathcal{C}$ 的真子集 $\mathcal{C}'$ 使得 $(\mathcal{C}',j)$ 可行，则称可行联盟-攻击者对 $(\mathcal{C},\ j)$ 在该状态下是不可约的。

给定一个可行联盟-攻击者对（$\mathcal{C},j$），我们可以设计一个迭代过程，生成所有不可约子对。该过程首先评估 $\mathcal{C}$的大小为1的子集。如果 $|\mathcal{C}|=1$，则（$\mathcal{C},j$）自然被判定为不可约，过程随即终止。否则，我们通过确认是否满足 $\Phi_j^{\mathcal{C}'}>0$，来检查每个大小为1的子集 $\mathcal{C}'$ 的不可约性。随后，对于使（$\mathcal{C}',j$）不可约的 $\mathcal{C}'$，我们从 $\mathcal{C}$ 中移除其元素，形成新集合 $\mathcal{C}_r^{(1)}$。如果 $|\mathcal{C}_r^{(1)}|\leq 1$，则所有

**算法 2：** $(\mathcal{C},\ j)$ 的不可约子对

**输入：** 位置 $p_i^d \in \mathbb{R}^{\ n}$，$i \in \mathcal{C}$ 和 $p_j^a \in \mathbb{R}^n$
**结果：** 由不可约子对 $\mathcal{P}$组成的集合
**初始化：** $\mathcal{P} \leftarrow \emptyset$，$\mathcal{C}_{\ r} \leftarrow \mathcal{C}$
**对** $i = 1$ **至** $n$ **执行**
  **如果** $|\mathcal{C}_r| < i$，**则**
    **跳出**
  **否则，如果** $|\mathcal{C}| = i$，**则**
    将 $(\mathcal{C},\ j)$ 放入 $\mathcal{P}$
    **跳出**
  **else**
    **对所有满足** $\mathcal{C}' \subset \mathcal{C}_r$ **的** $|\mathcal{C}'| = i$，**执行**
      $\Phi_j^{\mathcal{C}'} \leftarrow$ （17）的最优值，满足（$\mathcal{C}', j$）
      **如果** $\Phi_j^{\mathcal{C}'} > 0$ **那么**
        将（$\mathcal{C}', j$）放入 $\mathcal{P}$
        从 $\mathcal{C}_r$ 中移除 $\mathcal{C}'$ 的元素
      **end**
    **end**
  **end**
**end**

不可约子对均已被识别，过程结束。如果需要进一步迭代，则对于 $l = 2, \ldots, |\mathcal{C}|$，我们对 $\mathcal{C}_r^{(l-1)}$的所有大小为$l$的子集重复上述过程，其中，$\mathcal{C}_r^{(l)}$同样通过从 $\mathcal{C}_r^{(l-1)}$中移除元素来定义。该过程持续进行，直至在第 $l$次迭代时满足 $|\mathcal{C}_r^{(l)}| \leq l$或 $|\mathcal{C}| = l$。此外，注意到任意不可约子对的联盟基数均以域的维数为上界，因此可以减少迭代次数。

**命题 5（上界基数）：** 在一个 $n$ 维域中，如果联盟-攻击者对（$\mathcal{C}, j$）是不可约的，那么 $\mathcal{C}$ 的基数不大于 $n$。

证明： 请参见附录 E。 ■

我们在算法 2 中总结了从可行联盟-攻击者对中识别不可约子对的结果。需要强调的是，在求解整数线性规划 (34) 时，我们可以专注于不可约联盟 - 攻击者对，而不是所有可能的联盟-攻击者对。这是因为，将一个联盟-攻击者对替换为其某个不可约子对，不会影响多重攻击目标函数的最优值。一般而言，算法 2 可以减少确定多重攻击目标函数所需的凸规划数量。然而，直接使用不可约性的概念可能会导致较大的计算负载。例如，在最坏情况场景下，计算一个联盟-攻击者对的所有不可约子对，最多需要计算 $N$ 个凸规划（$\mathcal{N}, j$）。为了提高计算效率，更有效的方法是将不可约性的概念与 ADS 的概念相结合。

### C. 分层整数线性规划

现在，我们展示如何整合 ADS 和不可约性概念，以建立 ILP (34) 的第一层层次结构。我们首先检查形如 ($\mathcal{N}$、 $j$) 的联盟-攻击者对，并且只关注其中可行的联盟-攻击者对。随后，我们将基于邻近性的子对 ($\Lambda_j^{\mathcal{N}}$、 $j$) 纳入考虑，其中 $\Lambda_j^{\mathcal{N}}$ 是可行联盟 - 攻击者对 ($\mathcal{N}$ 、 $j$) 的 ADS。最后，我们选择 ($\Lambda_j^{\mathcal{N}}$、 $j$) 的不可约子对，以细化多重攻击目标函数 $\Gamma$。因此，我们将最高优先级分配给不可约且满足 $\mathcal{C}$ 为 $\Lambda_j^{\mathcal{N}}$ 子集的联盟-攻击者对 ($\mathcal{C}$、 $j$)。与直接优先考虑 ($\mathcal{N}$、 $j$) 的所有不可约子对相比，该方法减少了潜在的计算难题，同时提供了相当数量的可行子对。

在确定第一个层次结构级别后，我们可以推导出 ILP (34) 的一个次优解。令 $\Xi_1$ 表示包含所有最高优先级联盟-攻击者对的第一个优先级集合。当 $\Xi_1$ 为空时，不存在可行联盟-攻击者对，此时 ILP 解仅对应于零联盟分配矩阵。为了将联盟-攻击者对的选择严格限制在 $\Xi_1$ 内，我们对决策变量施加以下约束:

$$\theta_{kj} = 0,\ \forall (\mathcal{C}_k, j) \notin \Xi_1. \tag{37}$$

通过在 ILP (34) 中加入约束 (37) 并重新求解，我们得到一个记为 $\Theta_1$ 的次优解，该解为多攻击者目标函数的最优值提供了一个下界。为了评估 $\Theta_1$，我们定义两个指标集 $\mathcal{N}_r^{(1)}$ 和 $\mathcal{M}_r^{(1)}$，其中 $\mathcal{N}_{(1)r}$ 包含根据 $\Theta_1$ 仍未被分配给任何攻击者的防御者的指标，而 $j \in \mathcal{M}_r^{(1)}$ 则在 $\Theta_1$ 中省略攻击者 $j$ 且 $(\mathcal{N}, j)$ 可行时成立。如果 $\mathcal{N}_r^{(1)}$ 或 $\mathcal{M}_r^{(1)}$ 中任一个为空，则不存在更多可用的联盟-攻击者对，无需继续后续步骤。

上述过程可在条件允许时迭代应用，以改进次优解。在第 $l \geq 2$ 次迭代中，通过识别所有不可约联盟-攻击者对（$\mathcal{C}, j$），构建第 $l$ 个优先级集合 $\Xi_l$；对于这些联盟-攻击者对，$\mathcal{C}$ 包含在某个可行的（$\mathcal{N}_r^{(l-1)}, j$）及 $j \in \mathcal{M}_r^{(l-1)}$ 的 ADS 中。随后，通过以下二元约束将决策变量限制为 $\Xi_l$：

$$\theta_{kj} = 0,\ \forall (\mathcal{C}_k, j) \notin \Xi_l \tag{38}$$

从而得到原整数线性规划（34）的一个子问题:

$$\begin{aligned} &\max\ \Gamma(\Theta, x) \\ &\text{s.t. (28), (30), (31), (38).} \end{aligned} \tag{39}$$

对于子问题（39）的一个解，记为 $\Theta_l$，通过将$\Theta_l$ 添加到先前的次优解中，更新整数线性规划（34）的次优解，得到

$$\Theta_{\text{HILP}} = \sum_{k=1}^{l} \Theta_k.$$

当 $\Xi_l$为空，或指标集 $\mathcal{N}_r^{(l)}$和 $\mathcal{M}_r^{(l)}$中有一个为空时，该过程在第 $l$ 次迭代终止，其中，$\mathcal{N}_{(l)r}$和 $\mathcal{M}_r^{(l)}$分别通过从 $\mathcal{N}_r^{(l-1)}$和 $\mathcal{M}_r^{(l-1)}$中排除与 $\tilde{\Theta}_l$ 相关的指标而获得。

总之，我们求解整数线性规划 (34) 的方法概述于算法 3 中，该算法在最优性和效率之间取得了平衡，后续命题将对此加以证明。

**命题 6（最优性—效率权衡）：** 算法 3 具有两个关键性质:

**算法 3：** 分层整数线性规-划

**数据：** 位置 $p_i^d$、 $i \in \mathcal{N}$和 $p_j^a$、 $j \in \mathcal{M}_a$
**结果：** 联盟分配矩阵 $\Theta_{\text{HILP}}$
初始化 $\Theta_{\text{HILP}} \leftarrow 0_{\tilde{N} \times M}$, $\mathcal{N}_r \leftarrow \mathcal{N}$ $\mathcal{M}_r \leftarrow \mathcal{M}_a$
**当** $\mathcal{N}_r \neq \emptyset$ 且 $\mathcal{M}_r \neq \emptyset$ **时**
  $\Xi \leftarrow \emptyset$
  **对所有** $j \in \mathcal{M}_r$ **执行**
    $\Phi_j^{\mathcal{N}_r} \leftarrow$ (17) 的最优值，满足 $(\mathcal{N}_r, j)$
    **if** $\Phi_j^{\mathcal{N}_r} > 0$ **则**
      $\Lambda_j^{\mathcal{N}_r} \leftarrow (\mathcal{N}_r, j)$ 的 ADS
      $\mathcal{P} \leftarrow$ 满足 $(\Lambda_j^{\mathcal{N}_r}, j)$ 时算法 2 的结果
      将 $\mathcal{P}$ 的所有元素放入 $\Xi$
    **else**
      从 $\mathcal{M}_r$中移除 $j$
    **end**
  **end**
  **如果** $\Xi = \emptyset$ **则**
    **中断**
  **end**
  $\Theta \leftarrow$ 满足 (38) 的整数线性规划 (39) 的解，其中 $\Xi$
  $\Theta_{\text{HILP}} \leftarrow \Theta_{\text{HILP}} + \Theta$
  从 $\mathcal{N}_r$中移除所有满足 $(\mathcal{B}\Theta)(i, :) \neq 0_{1\times M}$ 的 $i$
  移除所有满足 $\Theta(:, j) \neq 0$ 的 $j$ $_{\tilde{N}\times 1}$ 来自 $\mathcal{M}_r$
**end**

1)（最优性）如果第一次迭代后 $\mathcal{M}_r$ 为空，则次优解 $\Theta_{\text{HILP}}$ 是整数线性规划 (34) 的最优解。

2)（效率）如果每次迭代中每个 $(\mathcal{N}_r, j)$ 的 ADS 基数至多为 $n$，那么形式为 (17) 的凸规划的计算总次数小于 $2^{n-1}M_a(1 + M_a)$。

证明：参见附录 F。 ■

算法 3 的命题 6 假设在许多实际场景中均成立。例如，如果联盟-攻击者对 $(\mathcal{N}, j)$ 对应的 ADS $\Lambda_j^{\mathcal{N}_r}$ $j \in \mathcal{M}_a$两两不相交，那么集合 $\mathcal{M}_r$ 在~第一次迭代后便为空，次优解 $\Theta$ 就是整数线性规划 (34) 的最优解。当攻击者相对于防御者稀疏分布时，这种场景很常见。此外，实证结果表明，ADS 的基数通常不超过防御者的数量。与直接求解整数线性规划相比，后者形式为 (17) 的凸规划的计算次数会随着防御者数量呈指数级增长，而我们的算法的计算时间在实证中不受防御者数量影响，仅与活跃攻击者的数量呈二次关系。因此，在防御者众多的大规模场景中，尤其是在传统方法计算上不可行时，算法 3 为解决整数线性规划提供了一种可行的替代方案。

### *D.* 单调防御增强分配

一旦获得算法 3 得出的次优解，我们就通过纳入单调性约束 (35) 对其进行调整。具体而言，在每个时间步，如果满足单调性约束 (35)，我们就将联盟分配矩阵更新为算法 3 得出的解；否则，保留更新后的先前联盟分配矩阵。请注意，活跃攻击者的数量可能会在连续两个时间步之间发生变化。因此，在执行更新之前，我们需要将先前联盟分配矩阵中对应于当前时间步被捕获攻击者的列置零。这种方法的优点是无需额外计算整数线性规划，即可保持不等式 (34) 的有效性，从而降低计算开销。

虽然上述方法有效地纳入了单调性约束 (35)，但可能会导致防御分配出现不期望的振荡行为。当防御者的分配在两个不同的攻击者之间反复切换时，就会出现这种振荡，并可能损害系统的性能。这些振荡源于算法 3 可能产生非唯一解，从而导致 (35) 中的等式成立。为解决这一问题，我们采用更严格的单调性约束 (35)：只有在严格满足不等式约束 (35) 时，才更新联盟分配矩阵。这样一来，当 (35) 中的等式成立时，我们就保留先前的联盟分配矩阵，从而避免振荡，同时仍能强制执行单调性约束 (35)。

我们针对多目标防御分配问题提出的方法总结于算法 4中，该算法由两个部分组成。第一部分将单调性约束 (35) 应用于算法 3 得到的解。~对于初始步骤， $\Theta^p$ 和 $\mathcal{M}_a^p$ 分别设为 $\Theta_{\text{HILP}}$和 $\mathcal{M}_a^p = \mathcal{M}$。第二部分采用贪心算法，将每个尚未分配的防御者分配给最近的尚未分配的活跃攻击者。这一方法受到如下观察的启发：防御者与攻击者之间的距离越近，SRS 越低。为此，我们首先根据线性方程 (29)，将联盟分配矩阵转换为相应的分配矩阵。随后使用贪心算法更新分配矩阵，再通过关联矩阵的伪逆将其转换回联盟分配矩阵。贪心算法计算效率高，是算法4第一部分的有益补充。

现在，我们能够将多攻击防御分配问题的解编码为多人到达-规避博弈中的可容许防御策略。给定算法4的输出 $\Theta_{\text{MDEA}} = (\theta_{kj})$，我们将分配规则定义如下：

$$\sigma(i) = \begin{cases} j, & i \in \mathcal{C}_k \ \& \ \theta_{kj} = 1 \\ \text{null}, & \text{otherwise} \end{cases} \tag{40}$$

其中，null表示防御者未被分配的情况。对于每个活跃攻击者 $j \in \mathcal{M}_a$，由 $\sigma$ 诱导的针对攻击者 $j$的联盟定义为：

$$\tilde{\mathcal{C}}_j = \{i \in \mathcal{N} \mid \sigma(i) = j\}.$$

将算法1与 $\sigma$ 中所述的分配规则相结合，我们得到一种可行的防御策略，其形式为：

**算法4：** 单调防御增强分配

**数据：** 位置 $p_i^d, i \in \mathcal{N}$ 和 $p_j^a, j \in \mathcal{M}_a$，解算法 3 $\Theta_{\text{HILP}}$ 的先前联盟分配矩阵 $\tilde{\Theta}^p$，先前索引集活跃攻击者 $\mathcal{M}^p{}_a$

**结果：** 联盟分配矩阵 $\Theta_{\text{MDEA}}$

**if** $\mathcal{M}_a = \emptyset$ **然后**

  **返回** $\Theta_{\text{MDEA}} \leftarrow 0_{\tilde{N} \times M}$

**end**

**第一部分：单调性约束执行**

$\Gamma \leftarrow$ $\Theta_{\text{HILP}}$ 中非零列向量的数量

$\Gamma^p \leftarrow$ 非零列向量的数字 $\tilde{\Theta}^p$

**如果** $\Gamma > \Gamma^p - |\mathcal{M}_a^p| + |\mathcal{M}_a|$，**则**

  $\Theta_{\text{MDEA}} \leftarrow \Theta_{\text{HILP}}$

**else**

  **如果** $|\mathcal{M}_a^p| \neq |\mathcal{M}_a|$，**则**

    **对于所有** $j \in \mathcal{M}_a^p \setminus \mathcal{M}_a$，**执行**

      $\tilde{\Theta}^p(:, j) \leftarrow 0_{\tilde{N} \times 1}$

    **end**

  **end**

  $\Theta_{\text{MDEA}} \leftarrow \tilde{\Theta}^p$

**end**

$\tilde{\Theta}^p \leftarrow \Theta_{\text{MDEA}}$，$\mathcal{M}_a^p \leftarrow \mathcal{M}_a$

**第二部分：未分配防守者的贪心分配**

$\mathcal{A} \leftarrow \mathcal{B}\Theta_{\text{MDEA}}$

将所有满足 $i \in \mathcal{N}$ with $\mathcal{A}(i,:) = 0_{1 \times M}$ 的对象放入 $\mathcal{N}_r$

将所有满足 $j \in \mathcal{M}_a$ with $\mathcal{A}(:, j) = 0_{N \times 1}$ 的对象放入 $\mathcal{M}_r$

**如果** $\mathcal{M}_r \neq \emptyset$ 并且 $\mathcal{N}_r \neq \emptyset$，**则**

  **对于所有** $i \in \mathcal{N}_r$，**执行**

    $j_{\min} \leftarrow \arg\min_{j \in \mathcal{M}_r} ||p_i^d - p_j^a||$

    $(i, j_{\min})$ $\mathcal{A} \leftarrow 1$的第$j_{\min}$个元素

  **end**

**end**

$\Theta_{\text{MDEA}} \leftarrow \mathcal{B}^\dagger \mathcal{A}$ 其中，$\mathcal{B}^\dagger$ 是 $\mathcal{B}$的伪逆

其形式为：

$$\tilde{\pi}_i^d(x) = v_{i,\max}^d \mathcal{N}(\xi_{\sigma(i)} - p_i^d),\ i \in \mathcal{N} \tag{41}$$

其中，$\xi_j = \tilde{\xi}_j^{\mathcal{C}_j}$ 是式（26）中定义的近似最优航路点。如果某个防御者未被分配任务，则将其控制输入设为零，以实现节能。

**定理2（性能保证防御策略）：** 对于任意初始联合状态 $x_0$，在由式（41）给出的防御策略 $\tilde{\pi}^d$ 下的收益函数 $J$，以及任意可容许攻击策略，都满足以下不等式：

$$J(x_0, \tilde{\pi}^d, \pi^a) \leq M - \Gamma(\Theta_{\text{MDEA}}^0, x_0) - N_c(0) \tag{42}$$

其中，$\Gamma$ 是式（32）中定义的多重攻击目标函数，$\Theta_{\text{MDEA}}^0$ 是算法4的初始解，$N_c(0)$是在初始联合状态下捕获的攻击者数量。此外，函数 $\Gamma + N_c$ 沿系统轨迹单调不减。

证明：算法4中强制执行单调性约束(35)，确保序列 $\Gamma(\Theta_{\text{MDEA}}(t_l)\ x(t_l)) + N_c(t_l)\}_{l=0}^L$ 单调不减。此外，注意到，由于防御者对活跃攻击者的分配仅在时刻 $t_l$ 更新，$\Gamma(\Theta_{\text{MDEA}}(t)\ x(t)) + N_c(t)$ 在每个时间区间 $[t_{l-1}, t_l)$ 内保持不变。因此，$\Gamma + N_c$ 单调不减。这表明不等式(33)在 $M_c = \Gamma(\Theta_{\text{MDEA}}^0, x_0) + N_c(0)$ 时成立；进而，根据引理 2，收益函数满足不等式(42)。■

定理2充分保证了我们的防御策略（41）能够实现不低于式（42）中所规定收益函数界限的性能水平，并且这一结论适用于任意初始联合状态。定理2还表明，函数 $\Gamma + N_c$表示未能到达目标集的攻击者的预期数量，并且该函数随时间单调不减。这一性质表明，随着博弈的进行，式（42）中收益函数的上界会逐渐收紧，从而使所提出策略的性能持续提升。在下一节中，我们将展示实证结果，以说明其在实际场景中的性能提升。

## V. 仿真

本节通过一系列仿真实验，评估我们提出的单攻击防御协调与分配算法的性能和效率。所有算法均使用 Python 编写，凸规划和整数线性规划则通过 CVXPY求解器 [46] 实现。实验在搭载 Ubuntu 20.04 操作系统的计算机上进行，该计算机配备 i9-10900 5.2 GHz 中央处理器和 32 GB 内存，具有 10 个物理核心和 20 个线程。为展示我们方法的有效性，我们在二维和三维场景中将算法与当前最先进的算法进行对比。首先，我们考察双模切换防御协同（DMSDC）算法（算法1），并以此为基础开展后续的单调防御增强分配（MDEA）算法（算法4）实验。此外，为展示算法在实际环境中的适用性，我们还将评估扩展至更真实的仿真环境，并使用 Gazebo 和机器人操作系统（ROS） [47]。

### A. 单攻击防御协调

我们首先针对单攻击者防御协同问题使用 DMSDC。我们将本方法与两种现有方法进行比较： [27] 中针对二维场景提出的多智能体追捕-防御策略，以及 [24]中针对三维场景提出的基于解析 HJI 的策略。我们在 $[-5, 5]^2$ 的二维矩形区域中进行仿真，并以单位圆作为目标集；同时在 $[-5, 5]^3$ 的三维盒状区域中进行仿真，并以原点作为目标集。为检验本方法在不同初始联合状态下保证联盟获胜的有效性，我们针对二维和三维场景，使用不同数量的防御者和不同捕获半径进行了 2000 次随机仿真。为确保公平比较，我们在所有方法中采用公式 (27) 给出的攻击策略，并将最大速度比设为 1。根据联盟是否赢得博弈，将仿真结果分为以下四类：

- 真正例：两种方法均成功捕获攻击者的案例所占百分比。
- 假负例：我们的方法成功捕获攻击者，而对比方法未能捕获攻击者的案例所占的百分比。
- 假正例：我们的方法未能捕获攻击者，而对比方法成功捕获攻击者的案例所占的百分比。
- 真负例：两种方法均未能捕获攻击者的案例所占的百分比。

表 I 展示了二维仿真结果，凸显了我们所提出的算法相较于多智能体追捕-防御策略的优越性能。值得注意的是，在捕获攻击者的案例中，我们的算法比多智能体追捕-防御策略多 7.1% 至 22.4%，且不存在我们的算法失败而多智能体追捕-防御策略成功的情况。有趣的是，随着捕获半径增大，我们的算法记录到的假负例百分比也随之升高。这一现象可归因于多智能体策略中采用的 Voronoi图并未内在地考虑非零捕获半径，而我们的算法利用攻击者的安全可达集来设计更有效的防御协同策略。此外，我们注意到，当防御者数量从 2 增加到 3 时，在两种捕获半径情形下，假负例百分比均有所升高。这是因为多智能体追捕-防御策略仅使用一名防御者进行防御，其余防御者则充当追捕者。相比之下，我们的算法充分发挥多个防御者的集体优势，实现整体协同。

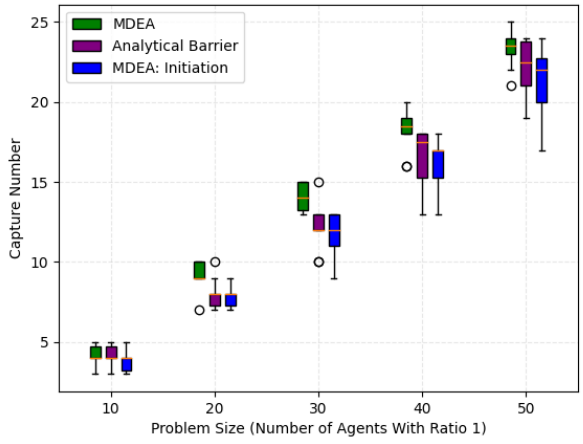
表 I 二维多防御者单攻击者可达-规避博弈的仿真结果。

| 防御者数字 | 捕获半径 | True 正 | 否 负 | 否 正 | True 负 |
|---|---|---|---|---|---|
| 2 | 0.5 | 25.6% | **7.1%** | **0%** | 67.3% |
| 3 | 0.5 | 33.0% | **10.0%** | **0%** | 57.0% |
| 2 | 3.0 | 33.9% | **18.8%** | **0%** | 47.3% |
| 3 | 3.0 | 41.6% | **22.4%** | **0%** | 36.0% |

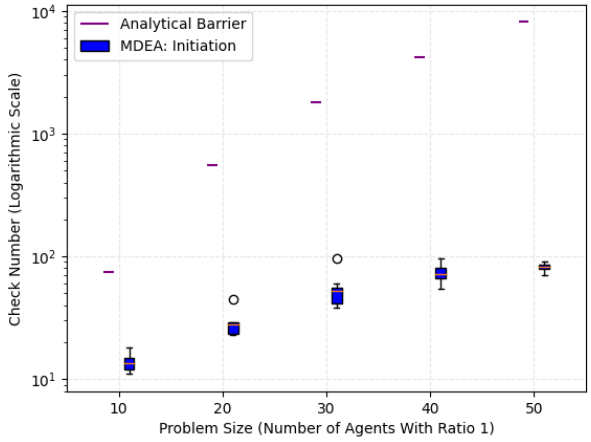
另一方面，表 II 中展示的三维仿真结果表明，我们提出的算法优于基于解析 HJI 的策略。具体而言，当捕获半径为 2 时，与基于解析 HJI 的策略相比，我们的算法最多可在多 13.6% 的情况下捕获攻击者，并且不存在我们的算法失败而基于解析 HJI 的策略成功的情况。需要强调的是，基于解析 HJI 的策略针对捕获半径为零的情况设计；因此，当捕获半径接近零时，我们的算法性能与基于解析 HJI 的策略高度一致，这进一步证明了我们方法的最优性。此外，与我们的二维比较类似，随着防御者数量增加，我们的算法表现出更好的性能。这种改进是由于基于解析 HJI 的策略仅关注包含一个或两个防御者的场景，而我们的方法能够适应多个防御者。

三维多防御者单攻击者可达-规避博弈的仿真结果。

| 防御者数字 | 捕获半径 | True 正 | 否 负 | 否 正 | True 负 |
|---|---|---|---|---|---|
| 2 | 0.1 | 66.5% | **0.9%** | **0%** | 32.6% |
| 3 | 0.1 | 72.2% | **1.6%** | **0%** | 26.2% |
| 2 | 2.0 | 84.3% | **7.3%** | **0%** | 8.4% |
| 3 | 2.0 | 81.6% | **13.6%** | **0%** | 5.2% |




(a)

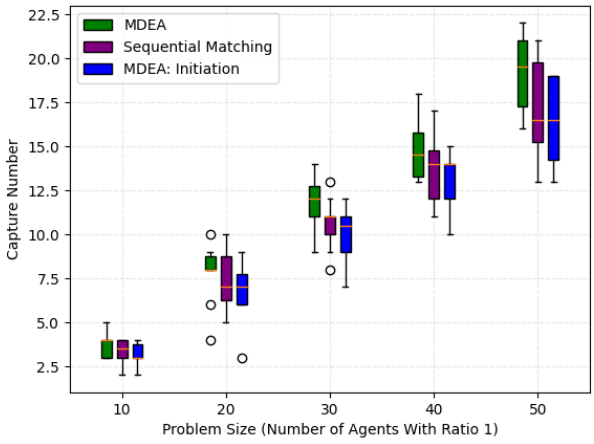



(b)

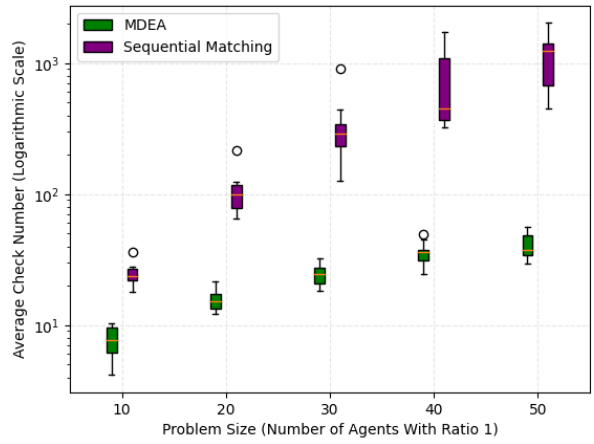
图5. 我们的MDEA算法与二维场景下解析屏障方法的性能比较。(a)：捕获数量比较。(b)：检查数量比较。

### *B.* 多攻击防御分配

接下来，我们考察用于多攻击防御分配问题的 MDEA 算法。我们将该方法与两种现有算法进行比较：针对二维场景的解析屏障（见 [31]），以及针对三维场景的序贯匹配（见 [33]）。为了开展实验，我们使用五组智能体，并使防御者与攻击者的比例保持为 1:1。每组中的智能体数量从 10 到 50 不等，并针对每组采用 10 个不同的初始




(a)




(b)

图6. 在三维场景中，我们的MDEA算法与顺序匹配方法之间的性能比较。（a）：捕获数量比较。（b）：平均检查次数比较。

联合状态进行仿真。我们采用与第 V-A 节所述相同的域，并针对不同场景调整目标集。具体而言，二维场景的矩形区域为 $[-5,5]^2$，目标集为线段 $y=5$；三维场景的盒状区域为$[-5,5]^3$，目标集为平面 $z=-5$。

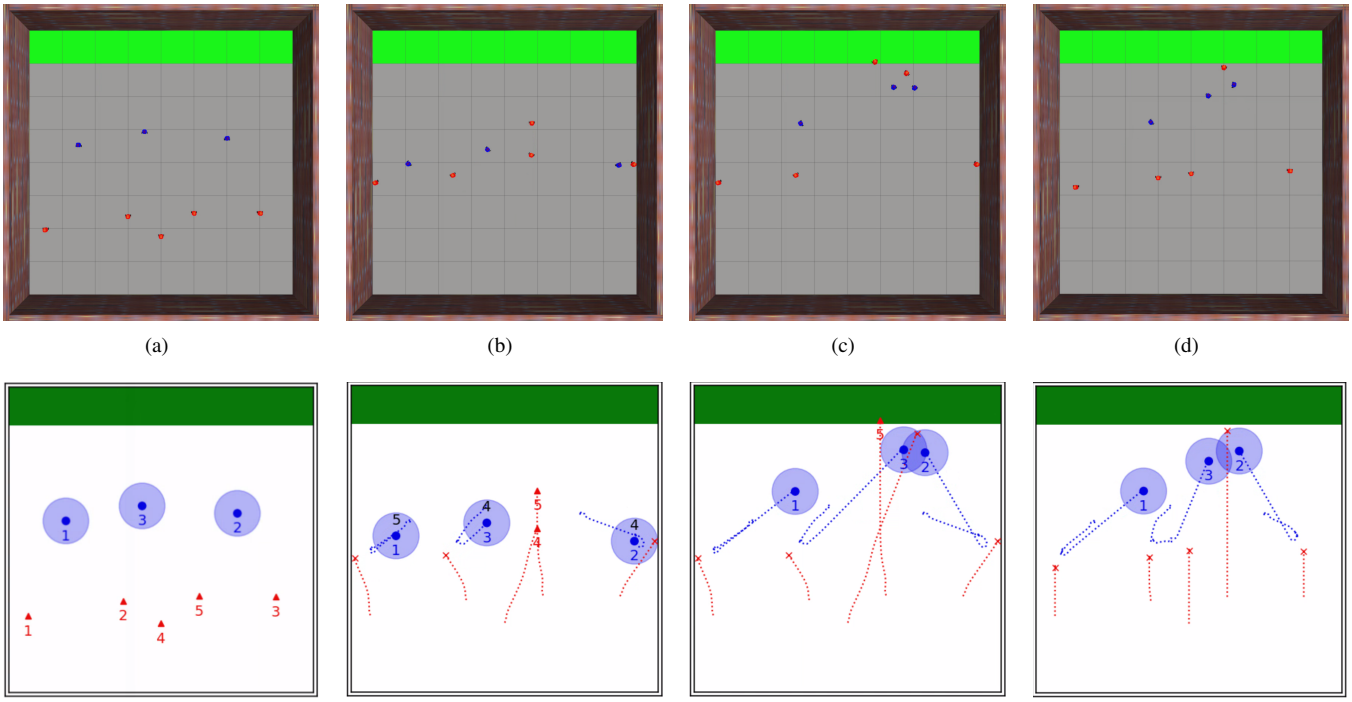
在仿真过程中，我们跟踪两个指标，以评估所提出算法的有效性和计算效率。第一个指标是捕获数量，表示整个博弈过程中被捕获的攻击者总数，该指标与收益函数互为补充。第二个指标是检查次数，表示在（32）所定义的多重攻击目标函数中计算的形如（17）的凸规划数量。该指标对计算复杂度起着至关重要的作用，尤其是在智能体规模增加时，我们的实证结果也证明了这一点。在这些情况下，可以看出，当捕获半径趋近于零时，我们的单次攻击防御协调策略（21）是 [31] 中所提出策略的极限情形，并且包含其中所提出的 [33] 作为一种特殊情形。为确保与其他算法进行公平比较，对于二维场景，我们采用较小的 $r_i=0.05$ 米捕获半径，对于三维场景，则采用 $r_i=0.2$ 米捕获半径。同样，我们采用（27）中给出的攻击策略，并将最大速度比设为 1，如第五-A节所述。

我们在图5中使用箱线图展示二维仿真的结果，表明我们的MDEA算法优于解析屏障方法。三种颜色分别表示三种情况：1）在每个分配时刻应用MDEA；2）解析屏障，仅在初始联合状态下进行分配；3）仅在初始联合状态下实施MDEA，与解析屏障类似。如图5（a）所示，MDEA相比解析屏障实现了显著的性能提升，其中位数始终高于解析屏障。此外，MDEA所实现的捕获数量超过了在初始联合状态下获得的捕获数量，证明了我们方法的防御性能提升，这一点已在定理2中得到证明。在图5（b）中可以观察到，解析屏障在每种情况下都表现出恒定的检查次数，该次数等于整数线性规划中决策变量的维数（34）。相比之下，MDEA在初始联合状态下的检查次数显著低于解析屏障，凸显了我们方法更高的计算效率。

在三维仿真中，我们的 MDEA算法也优于序贯匹配方法，如图 6 所示。与二维仿真一样，我们用相同的三种颜色表示三种情况，只是将解析屏障替换为序贯匹配。如图 6(a) 所示，MDEA 的捕获数量中位数更高，表现优于序贯匹配。此外，MDEA 达到的最终捕获数量超过了算法最初得到的预期捕获数量，证明我们的方法能够有效提升防御性能。另外，图 6(b) 显示，MDEA 达到的平均检查次数显著低于序贯匹配；平均检查次数是对每一步的检查次数取平均得到的。这体现了我们方法的计算效率。

### *C.* 使用 *ROS* 的 *Gazebo* 仿真器

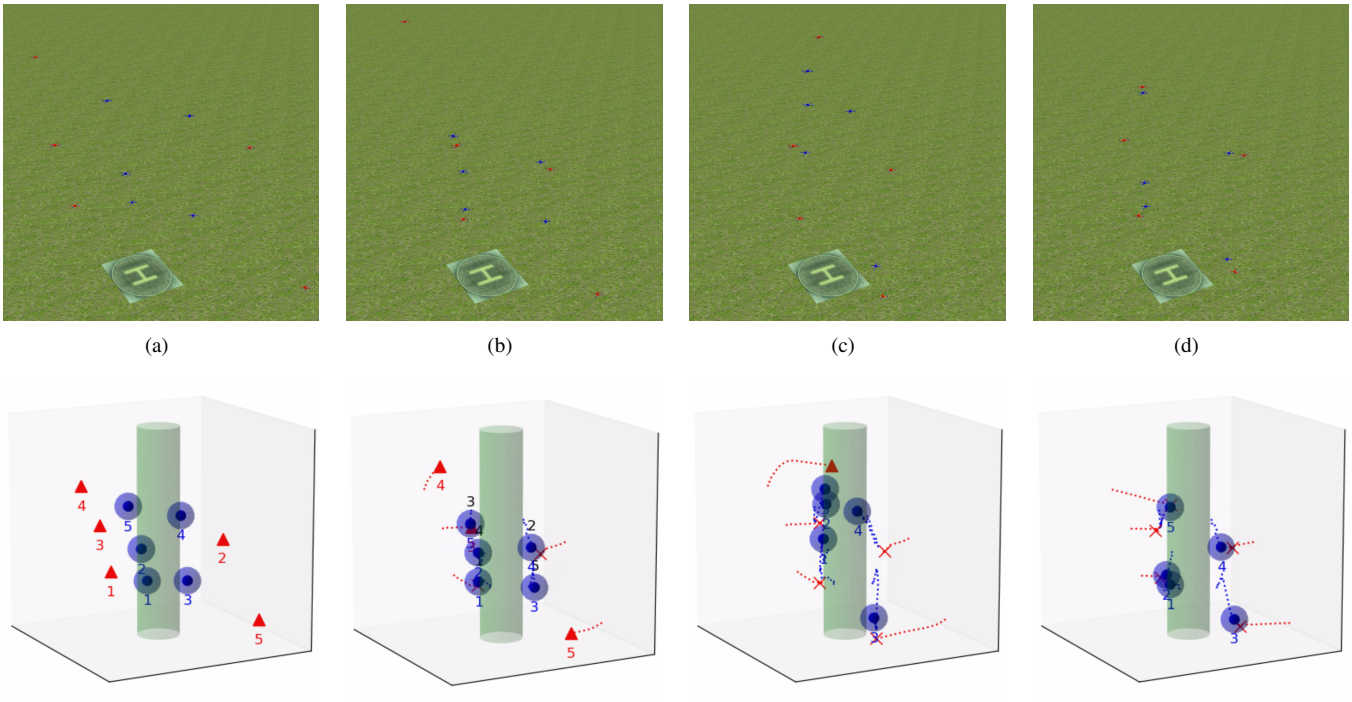
最后，我们使用结合 ROS 的 Gazebo 仿真器评估所提出算法的实际适用性。为应对真实世界场景，我们采用了 Wang 等人提出的一种基于控制屏障函数的多智能体避碰方法 [48]。此外，我们使用 [49] 中提供的映射，将算法生成的速度向量转换为相应的线速度和角速度指令。我们在两种不同环境中开展实验：二维室内场景和三维室外场景。室内环境设计为一个四周有墙的正方形空间，其中一个墙边设有矩形区域作为目标集。在此场景中，我们部署了 TurtleBot3 Burger 轮式移动机器人作为智能体，其最大线速度和角速度分别为 0.22m/s 和 2.84rad/s。另一方面，室外环境由一个高于地面的盒状区域构成，其中有一个圆柱形



(a) (b) (c) (d)

(e) (f) (g) (h)

图7. 室内场景中三名防御者（蓝色）与五名攻击者（红色）的二维 Gazebo 仿真及可视化结果。绿色区域表示目标集，蓝色圆点上方的黑色数字表示瞬时分配索引。(a) 和 (e) 初始配置；(b) 和 (f) 13 s 时的配置（最优攻击策略情形）；(c) 和 (g) 最终配置（最优攻击策略情形）；(d) 和 (h) 最终配置（直线攻击策略情形）。



(a) (b) (c) (d)

(e) (f) (g) (h)

图8. 室外场景中五名防御者（蓝色）与五名攻击者（红色）的三维 Gazebo 仿真及可视化结果。如 (e)–(h) 中的绿色圆柱所示，目标集由圆柱形禁飞区表示。蓝色圆点上方的黑色数字表示瞬时分配索引。(a) 和 (e) 初始配置；(b) 和 (f) 5 s 时的配置（最优攻击策略情形）；(c) 和 (g) 最终配置（最优攻击策略情形）；(d) 和 (h) 最终配置（直线攻击策略情形）。

将该区域指定为目标集，以表示现实场景中的禁飞区。在此，采用 Iris 四旋翼机器人作为智能体。计算得到的速度向量被用作参考点，以调节机器人的速度和航向。线速度和角速度参考值分别设为 0.8m/s 和 0rad/s，使 Iris 四旋翼机器人能够快速移动至参考点，同时保持安全。TurtleBot3 Burger 和 Iris 机器人的安全半径分别为 0.1m 和 0.5m，捕获半径分别设为 0.6m 和 1.5m。两个机器人平台均以 50Hz 的频率运行，防御分配频率为 10Hz。

为测试所提出算法的有效性，我们采用两种不同的攻击策略，并考察防御方团队对这些变化的适应能力。第一种策略是式（27）给出的最优攻击策略，其中攻击者 $j$ 朝着式（26）所指定的近似最优航路点 $\tilde{\xi}_j^{\mathcal{N}}$ 移动。第二种策略称为直线接近策略，即每个攻击者都朝目标集上的最近点前进，旨在尽可能缩短到达目标集所需的时间，而不考虑被防御者捕获的潜在风险。仿真过程见随附视频，图7和图8展示了示例快照。在由五名攻击者和三名防御者组成的二维室内场景中，我们提出的算法表现出很高的有效性：采用最优攻击策略时，最终捕获数量为四；采用直线攻击策略时，最终捕获数量为五。同样，在由五名攻击者和五名防御者组成的三维室外场景中，该算法展现出稳健性：采用最优攻击策略时，最终捕获数量为四；采用直线攻击策略时，最终捕获数量为五。这些结果证实了我们提出的算法在不同环境中的有效性。

## VI. 结论

本文提出了一种面向多人可达-规避博弈中防御者的双层在线优化策略，旨在面对未知攻击策略时，最小化在凸域内进入凸目标集的成功攻击者数量。我们设计了一种双模式切换防御算法，使防御者联盟能够通过在线凸规划应对个体攻击者。该算法为初始联合状态导出最大防御获胜区域，在提高获胜概率的同时实现了可扩展性。此外，我们提出了一种单调防御增强分配算法，通过结合单次攻击协调的预测结果，确定整数线性规划（ILPs）中的联盟分配矩阵。该算法采用分层迭代方法近似求解整数线性规划，并引入单调性约束，在保持预期失败攻击者数量随时间非递减的同时，减少所选的联盟-攻击者对数量。我们在多种多机器人系统和环境中评估了该方法的性能，结果表明其在最优性和效率方面具有优越性。此外，Gazebo 仿真证实了该方法在现实世界中的适用性以及实际机器人应用的潜力。未来，我们将把该方法扩展到非凸环境，并集成更复杂的智能体模型，从而进一步提升所提出框架的性能和适用性。此外，考虑具有不同优先级的攻击者，或许能更好地反映现实世界场景，并进一步完善该方法所采用的防御策略。

## 附录

### A. 命题1的证明

证明：为证明 SRS 的凸性，我们首先说明，对于每个 $i \in \mathcal{C}$，$c_{ij}(q,\ x_{ij})$ 关于 $q$ 是凸的。注意到，$c_{ij}$ 可以分解为三个项之和：$c_{ij}(q,\ x_{ij}) = (\gamma_{ij}^2 - 1)\ ||q||^2 + 2r_i\gamma_{ij}||q - p_j^a|| + [2\ p_i^d - \gamma_{ij}^2 p_j^a\ ^T q + \gamma_{ij}^2||p_j^a||^2 - (\ )\ (\ )$。$||p_i^d||^2 + r_i^2]$ 8

由于最大速度比约束 ( ) 以及 $r_i$ 的正性，前两项的系数均为非负。因此，这两项是凸的，因为（平方）欧几里得范数是凸函数。随后，由于最后一项是线性的，且凸函数之和仍为凸函数，$c_{ij}(q,\ x_{ij})$ 的凸性即可得出。

还需证明：若 $q \in \mathcal{D}$ 满足不等式 $\max_{i\in\mathcal{C}} c_{ij}(q, x_{ij}) \leq 0$，则 $q$ 属于 SRS。对于任意这样的 $q$，考虑时间区间 $[0, \tau]$ 上的恒定攻击策略 $\pi_j^a = v_{j,\max}^a \mathcal{N}(q - p_j^a)$，其中 $\tau = \frac{||q-p_j^a||}{v_{j\max}^a}$，。注意到攻击者 $j$ 可以使用 $\pi_j^a$，沿连接 $p_j^a$ 和 $q$ 的线段 $L$ 从 $p_j^a$ 到达 $q$，因此 SRS 的可达性性质成立。另一方面，$L$ 上的任意一点都可表示为 $q_\lambda = (1-\lambda)p_j^a + \lambda q$，其中参数为 $\lambda \in [0,1]$；攻击者 $j$ 使用 $\pi_j^a$ 可在时刻 $\lambda\tau$ 到达该点。在同一时刻，防御者 $i$ 能够从 $p_i^d$ 到达且距离 $q_\lambda$ 最近的点，记为 $q_i^{d*},\lambda$，必定位于连接 $p_i^d$ 和 $q_\lambda$ 的直线上，因为防御者 $i$ 可以向任意方向移动。因此可得 $q_i^{d*},\lambda = p_i^d + v_{i,\max}^d \frac{q_\lambda - p_i^d}{||q_\lambda - p_i^d||}\lambda\tau$。此外，$c_{ij}(q, x_{ij})$ 的凸性意味着 $c_{ij}(q_\lambda, x_{ij}) \leq (1-\lambda)c_{ij}(p_j^a, x_{ij}) + \lambda c_{ij}(q, x_{ij}) \leq 0$，进而表明 $s_i\ q_i^{d*}{}_\lambda\ q_\lambda = r_i - \big|\big|q_\lambda - p_i^d|| - \gamma_{ij}||q_\lambda - p_j^a||\big|\big| = \gamma_{ij}\overline{||q_\lambda - (\ ,\ )}$。$p_j^a|| + r_i - ||q_\lambda - p_i^d|| \leq 0$

因此，对于任意可容许防御策略 $\pi_i^d$，对于所有 $\lambda \in [0,1]$，均有 $s_i(\chi_i^d(\lambda\tau;\ p_i^d, \pi_i^d)\ q_\lambda) \leq s_i(q_i^{d*},\lambda, q_\lambda) \leq 0$，由此得到 SRS 的安全性性质。 ■

### B. 命题 2 和 (24) 的证明

证明：考虑 (17) 的拉格朗日函数：

$$\mathcal{L}^d(\hat{q}, \hat{\lambda}) = ||q - \tilde{q}||^2 + \sum_{i\in\mathcal{C}} \lambda_{ij} c_{ij}(q, x_{ij}) + \sum_{k\in\mathcal{I}_D} \bar{\lambda}_k d_k(q) + \sum_{l\in\mathcal{I}_G} \tilde{\lambda}_l g_l(\tilde{q})$$

其中，$\hat{q} = (q,\ \tilde{q})$ 和 $\lambda = (\lambda_{ij},\ i \in \mathcal{C}; \lambda_k,\ k \in \mathcal{I}_D; \lambda_l,\ l \in \mathcal{I}_G)$ 是拉格朗日乘子。KKT 条件表明，对于最优解 $\hat{q}^* = (\xi_j^{\mathcal{C}},\ \tilde{\xi}_j^{\mathcal{C}})$，存在一个

ˆ 拉格朗日乘子 $\lambda^*$，其分量均非负，使得 $\frac{\partial \mathcal{L}^d}{\partial \hat{q}}(\hat{q}^*, \hat{\lambda}^*) = 0$，即，

$$2(\xi_j^{\mathcal{C}} - \tilde{\xi}_j^{\mathcal{C}})^T + \sum_{i\in\mathcal{C}} \lambda_{ij}^* \frac{\partial c_{ij}}{\partial q}(\xi_j^{\mathcal{C}}, x_{ij}) + \sum_{k\in\mathcal{I}_D} \bar{\lambda}_k^* \frac{\partial d_k}{\partial q}(\xi_j^{\mathcal{C}}) = 0 \tag{43}$$

$$2(\xi_j^{\mathcal{C}} - \tilde{\xi}_j^{\mathcal{C}})^T - \sum_{l\in\mathcal{I}_G} \tilde{\lambda}_l^* \frac{\partial g_l}{\partial \tilde{q}}(\tilde{\xi}_j^{\mathcal{C}}) = 0. \tag{44}$$

此外，对于任意内点 $x_j^{\mathcal{C}} \in \mathcal{D}'$ 和任意 $i \in \mathcal{C}$，以下三种情况之一成立：i) $c_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) < 0$；ii) $c_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) = 0$ 且 $\dot{c}_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) = 0$；iii) $c_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) = 0$ 且 $\dot{c}_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) \neq 0$。特别地，由于 $c_{ij}$ 是连续可微的，根据实分析的结果，满足最后一种情况的内点集合的测度为零[50]。因此，互补条件 $\lambda_{ij}^* c_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) = 0$ 意味着对于几乎处处的 $x_j^{\mathcal{C}} \in \mathcal{D}'$，$\lambda_{ij}^* \dot{c}_{ij}(\xi_j^{\mathcal{C}},\ x_{ij}) = 0$，即，

$$\lambda_{ij}^* \left( \frac{\partial c_{ij}}{\partial q}(\xi_j^{\mathcal{C}}, x_{ij})\dot{\xi}_j^{\mathcal{C}} + \frac{\partial c_{ij}}{\partial p_i^d}(\xi_j^{\mathcal{C}}, x_{ij}) u_i^d + \frac{\partial c_{ij}}{\partial p_j^a}(\xi_j^{\mathcal{C}}, x_{ij}) u_j^a \right) = 0,\ \forall i \in \mathcal{C}. \tag{45}$$

可分别对互补条件 $\bar{\lambda}_k^* d_k(\xi_j^{\mathcal{C}}) = 0,\ k \in \mathcal{I}_D$ 和 $\tilde{\lambda}_l^* g_l(\tilde{\xi}_j^{\mathcal{C}}) = 0,\ l \in \mathcal{I}_G$ 进行类似推导，从而得到，对于几乎处处的 $x_j^{\mathcal{C}} \in \mathcal{D}'$，

$$\bar{\lambda}_k^* \frac{\partial d_k}{\partial q}(\xi_j^{\mathcal{C}})\dot{\xi}_j^{\mathcal{C}} = 0,\ \forall k \in \mathcal{I}_D \tag{46}$$

., 且对于所有 $l \in \mathcal{I}_G$，$\tilde{\lambda}_l^* \frac{\partial g_l}{\partial \tilde{q}}(\tilde{\xi}_j^{\mathcal{C}})\dot{\tilde{\xi}}_j^{\mathcal{C}} = 0$ 成立；后一个等式与 (44) 一起可得

$$2(\xi_j^{\mathcal{C}} - \tilde{\xi}_j^{\mathcal{C}})^T \dot{\tilde{\xi}}_j^{\mathcal{C}} = \sum_{l\in\mathcal{I}_G} \tilde{\lambda}_l^* \frac{\partial g_l}{\partial \tilde{q}}(\tilde{\xi}_j^{\mathcal{C}})\dot{\tilde{\xi}}_j^{\mathcal{C}} = 0. \tag{47}$$

因此，

$$\begin{aligned}
\dot{\Phi}_j^{\mathcal{C}} &\overset{\text{def}}{=} 2(\xi_j^{\mathcal{C}} - \tilde{\xi}_j^{\mathcal{C}})^T(\dot{\xi}_j^{\mathcal{C}} - \dot{\tilde{\xi}}_j^{\mathcal{C}}) \\
&\overset{(47)}{=} 2(\xi_j^{\mathcal{C}} - \tilde{\xi}_j^{\mathcal{C}})^T \dot{\xi}_j^{\mathcal{C}} \\
&\overset{(43)}{=} -\sum_{i\in\mathcal{C}} \lambda_{ij}^* \frac{\partial c_{ij}}{\partial q}(\xi_j^{\mathcal{C}}, x_{ij})\dot{\xi}_j^{\mathcal{C}} - \sum_{k\in\mathcal{I}_D} \bar{\lambda}_k^* \frac{\partial d_k}{\partial q}(\xi_j^{\mathcal{C}})\dot{\xi}_j^{\mathcal{C}} \\
&\overset{(45)+(46)}{=} \sum_{i\in\mathcal{C}} \lambda_{ij}^* \frac{\partial c_{ij}}{\partial p_i^d}(\xi_j^{\mathcal{C}}, x_{ij}) u_i^d + \sum_{i\in\mathcal{C}} \lambda_{ij}^* \frac{\partial c_{ij}}{\partial p_j^a}(\xi_j^{\mathcal{C}}, x_{ij}) u_j^a
\end{aligned}$$

对于几乎处处的 $x_j^{\mathcal{C}} \in \mathcal{D}'$，该式成立。通过考察 (18)，命题 2 的证明即告完成。

为验证等式 (24)，我们考虑 (23) 的拉格朗日函数：

$$\mathcal{L}^a(q, \hat{\mu}) = ||q - p_j^a||^2 + \sum_{i\in\mathcal{C}} \mu_{ij} c_{ij}(q, x_{ij}) + \sum_{l\in\mathcal{I}_G} \tilde{\mu}_l g_l(q)$$

其中拉格朗日乘子 $\hat{\mu} = (\mu_{ij},\ i \in \mathcal{C},\ \tilde{\mu}_l,\ l \in \mathcal{I}_G)$。与命题 2 的证明类似，KKT 条件意味着存在一个各分量均为非负的拉格朗日乘子 $\hat{\mu}^*$，使得

$$2(\bar{\xi}_j^{\mathcal{C}} - p_j^a)^T + \sum_{i\in\mathcal{C}} \mu_{ij}^* \frac{\partial c_{ij}}{\partial q}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}) + \sum_{l\in\mathcal{I}_G} \tilde{\mu}_l^* \frac{\partial g_l}{\partial q}(\bar{\xi}_j^{\mathcal{C}}) = 0 \tag{48}$$

并且对于几乎处处的 $x_j^{\mathcal{C}} \notin \mathcal{D}_j^{\mathcal{C}}$，

$$\mu_{ij}^* \left( \frac{\partial c_{ij}}{\partial q}(\bar{\xi}_j^{\mathcal{C}}, x_{ij})\dot{\bar{\xi}}_j^{\mathcal{C}} + \frac{\partial c_{ij}}{\partial p_i^d}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}) u_i^d + \frac{\partial c_{ij}}{\partial p_j^a}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}) u_j^a \right) = 0,\ \forall i \in \mathcal{C} \tag{49}$$

and

$$\tilde{\mu}_l^* \frac{\partial g_l}{\partial q}(\bar{\xi}_j^{\mathcal{C}})\dot{\bar{\xi}}_j^{\mathcal{C}} = 0,\ \forall l \in \mathcal{I}_G. \tag{50}$$

因此，只需注意到以下事实即可完成证明：

$$\begin{aligned}
&\dot{\bar{\Phi}}_j^{\mathcal{C}} + 2(\bar{\xi}_j^{\mathcal{C}} - p_j^a)^T u_j^a \\
&\overset{\text{def}}{=} 2(\bar{\xi}_j^{\mathcal{C}} - p_j^a)^T \dot{\bar{\xi}}_j^{\mathcal{C}} \\
&\overset{(48)}{=} -\sum_{i\in\mathcal{C}} \mu_{ij}^* \frac{\partial c_{ij}}{\partial q}(\bar{\xi}_j^{\mathcal{C}}, x_{ij})\dot{\bar{\xi}}_j^{\mathcal{C}} - \sum_{l\in\mathcal{I}_G} \tilde{\mu}_l^* \frac{\partial g_l}{\partial q}(\bar{\xi}_j^{\mathcal{C}})\dot{\bar{\xi}}_j^{\mathcal{C}} \\
&\overset{(49)+(50)}{=} \sum_{i\in\mathcal{C}} \mu_{ij}^* \frac{\partial c_{ij}}{\partial p_i^d}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}) u_i^d + \sum_{i\in\mathcal{C}} \mu_{ij}^* \frac{\partial c_{ij}}{\partial p_j^a}(\bar{\xi}_j^{\mathcal{C}}, x_{ij}) u_j^a.
\end{aligned}$$

■

### C. 命题 3 的证明

证明：采用反证法进行证明。假设存在 $q_1, q_2 \in \Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})$，使得 $\Psi(q_1) = \Psi(q_2)$，其中

$$\Psi(q) = \min_{\tilde{q}\in\mathcal{G}} ||q - \tilde{q}||^2 \tag{51}$$

是从 $q$ 到目标集的最小平方距离。我们声称，连接 $q_1$ 和 $q_2$ 的线段，记作 $q_\lambda = \lambda q_1 + (1-\lambda)q_2$，其参数为 $\lambda \in [0,1]$，必定位于由 $c_{kj}(q, x_{kj}) = 0$ 定义的零水平集上，其中某个 $k \in \mathcal{N}$。首先，注意到由于 SRS 的凸性，线段 $q_\lambda$ 必须位于 SRS 内。其次，根据 $\Phi_j^{\mathcal{C}}$ 的定义，我们知道对于所有 $q \in \Omega_j^{\mathcal{C}}(x_j^{\mathcal{C}})$，$\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}) \leq \Psi(q)$，因此对于所有 $\lambda \in [0,1]$，$\Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}}) \leq \Psi(q_\lambda)$。此外，由于 $\Psi$ 是凸的，我们有 $\Psi(q_\lambda) \leq \lambda\Psi(q_1) + (1-\lambda)\Psi(q_2) = \Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}})$。因此，对于所有 $\lambda \in [0,1]$，$\Psi(q_\lambda) = \Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}})$，这表明线段 $q_\lambda$ 位于 SRS 的边界上。最后，我们通过如下方式验证这一主张：注意到如果对于某个 $\lambda_0 \in [0,1]$，$\max_{i\in\mathcal{N}} c_{ij}(q_{\lambda_0}, x_{ij}) < 0$，那么点 $q_{\lambda_0}$ 严格位于 SRS 内部，这将与 $q_{\lambda_0}$ 位于 SRS 边界上的结论相矛盾。然而，这一主张与由 $c_{kj}(q, x_{kj}) = 0$ 定义的零水平集不包含线段这一事实相矛盾。这是因为 $(q_1 - q_2)^T \nabla_q c_{kj} = 0$ 不可能有无穷多个解，其中

$$\nabla_q c_{kj} = 2\ \gamma_{kj}^2 - 1\ q + (\ )(\ )$$

$2r_k\gamma_{kj}\ \frac{q-p_j^a}{||q-p_j^a||} + 2\ p_k^d - \gamma_{kj}^2 p_j^a$

是 $c_{kj}(q, x_{kj}) = 0$ 的零水平集的法向量。■

### D. 命题 4 的证明

证明：只需用反证法证明 $\Phi_j^{\mathcal{C}'}(x_j^{\mathcal{C}'}) = \Phi_j^{\mathcal{C}}\ x_j^{\mathcal{C}}$ $\mathcal{C}' = \Lambda_j^{\mathcal{C}}$ $\Psi\ \xi_j^{\mathcal{C}'} = \Psi\ \xi_j^{\mathcal{C}}$ ( )，其中，或等价地，证明 ( ) ( )，其中 $\Psi$ 在 (51) 中定义。假设 $\Psi(\xi_j^{\mathcal{C}'}) \neq \Psi(\xi_j^{\mathcal{C}})$。在这种情况下，由于 $\xi_j^{\mathcal{C}} \in \Omega_j^{\mathcal{C}'}$，我们有 $\Psi(\xi_j^{\mathcal{C}'}) < \Psi(\xi_j^{\mathcal{C}})$。为导出矛盾，我们考虑连接 $\xi_j^{\mathcal{C}'}$ 和 $\xi_j^{\mathcal{C}}$ 的线段，并用 $\mu \in [0,1]$ 将其参数化为 $\xi_\mu = \mu\xi_j^{\mathcal{C}'} + (1-\mu)\xi_j^{\mathcal{C}}$。由于 $\mathcal{C}'$ 是 $(\mathcal{C}, j)$ 的 ADS，因此对于所有 $i \in \mathcal{C} \setminus \mathcal{C}'$，$c_{ij}(\xi_j^{\mathcal{C}}, x_{ij}) < 0$。由 $c_{ij}(\xi_\mu, x_{ij})$ 关于 $\mu$ 的连续性可知，对于每个$i \in \mathcal{C} \setminus \mathcal{C}'$，存在 $\mu_i \in (0,1)$，使得对于所有 $\mu \in [0, \mu_i,]$，$c_{ij}(\xi_\mu, x_{ij}) \leq 0$。令 $\bar{\mu}$ 为所有 $\mu_i$ $c_{ij}\ \xi_{\bar{\mu}}\ x_{ij} \leq 0$ $i \in \mathcal{C} \setminus$ 中的最小 $\mathcal{C}'$ 值。于是，对于所有 , ( , )。此外，由于线段 $\xi_\mu$ 位于 $\Omega_j^{\mathcal{C}'}(x_j^{\mathcal{C}'})$ 内，对于所有 $i \in \mathcal{C}'$，我们有 $c_{ij}(\xi_{\bar{\mu}}, x_{ij}) \leq 0$。因此，$\xi_{\bar{\mu}} \in \Omega_j^{\mathcal{C}}$。然而，注意到 $\Psi\ \xi_{\bar{\mu}} \leq \bar{\mu}\Psi\ \xi_j^{\mathcal{C}} + 1 - \bar{\mu}\ \Psi\ \xi_j^{\mathcal{C}'} <$ ( ) ( ) ( ) ( ) $\Psi$ $(\xi_j^{\mathcal{C}})$，这与 $\Psi$ 的最小值在 $\xi_j^{\mathcal{C}}$ 处取得这一事实相矛盾。■

### E. 命题 5 的证明

证明：根据命题 4，只需证明：对于任意可行联盟-攻击者对 $(\mathcal{C}, j)$（其中$|\Lambda_j^{\mathcal{C}}| > n$），存在 $\Lambda_j^{\mathcal{C}}$ 的一个子集 $\mathcal{C}'$，使得 $|\mathcal{C}'| \leq n$，且 $\Phi_j^{\mathcal{C}'}(x_j^{\mathcal{C}'}) = \Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}})$。对于每个 $i \in \Lambda_j^{\mathcal{C}}$，近似最优航路点 $\xi_j^{\mathcal{C}}$ 位于零水平集 $c_{ij}(\xi_j^{\mathcal{C}}, x_{ij}) = 0$ 的切线上，因此满足线性方程 $(q - \xi_j^{\mathcal{C}})^T \nabla_q c_{ij}(\xi_j^{\mathcal{C}}, x_{ij}) = 0$；其中，$\nabla_q c_{ij}(\xi_j^{\mathcal{C}}, x_{ij})$ 是 $c_{ij}(q, x_{ij}) = 0$ 在 $q = \xi_j^{\mathcal{C}}$ 处的法向量。由于一个 $n$ 维的解至多可由 $n$ 个独立线性方程唯一确定，因此存在 $\Lambda_j^{\mathcal{C}}$ 的一个子集 $\mathcal{C}'$，使得 $|\mathcal{C}'| \leq n$，且对于所有 $i \in \mathcal{C}'$，$\xi_j^{\mathcal{C}}$ 是线性方程 $(q - \xi_j^{\mathcal{C}})^T \nabla_q c_{ij}(\xi_j^{\mathcal{C}}, x_{ij}) = 0$ 的唯一解。此外，根据 $\Omega_j^{\mathcal{C}}$ 的次水平集表示，可以选择 $\mathcal{C}'$，使得对于 $\xi_j^{\mathcal{C}}$ 的某个邻域 $U$，$U \cap \Omega_j^{\mathcal{C}} = U \cap \Omega_j^{\mathcal{C}'}$。因此，由 SRS 的凸性可得 $\Phi_j^{\mathcal{C}'}(x_j^{\mathcal{C}'}) = \Phi_j^{\mathcal{C}}(x_j^{\mathcal{C}})$。■

### F. 命题 6 的证明

证明：最优性：注意，多重攻击目标函数 $\Gamma$的最大可能值等于满足联盟-攻击者对 $(\mathcal{N}, j)$ 可行的活跃攻击者 $j$ 的数量。根据算法 3，只有当 $(\mathcal{N}, j)$ 不可行，或攻击者 $j$ 包含在次优解$\Theta_1$中时，索引 $j$ 才会在第一次迭代后从 $\mathcal{M}_r$ 中移除。因此，第一次迭代˜后 $\mathcal{M}_r$ 为空，表明 $\Theta$ 达到了 $\Gamma$的最大可能值，因而是整数线性规划 (34) 的最优解。效率：在算法 3 的每次迭代中，有两个步骤需要计算形式为 (17) 的凸规划：判断联盟-攻击者对 $(\mathcal{N}_r, j)$ 的可行性，以及识别 $(\Lambda_j^{\mathcal{N}_r}, j)$ 的所有不可约子对。前一步只需一次计算；后一步使用算法 2 执行，在 ADS 的基数不超过 $n$的条件下，最多需要 $\sum_{k=1}^{n-1}\binom{n}{k} = 2^n - 2$ 次计算。因此，在该次迭代中，针对一个攻击者所需的计算次数最多为 $2^n - 1$。此外，迭代次数受活跃攻击者 $M_a$的数量限制，因为每次迭代后至少有一个攻击者会从集合 $\mathcal{M}_r$ 中移除。因此，所有迭代的计算次数总计最多为$\sum_{k=1}^{M_a}(2^n - 1)k$，小于 $2^{n-1}M_a(1 + M_a)$。■

参考文献


[1] R. Vidal、O. Shakernia、H. J. Kim、D. H. Shim 和 S. Sastry，“概率追逃博弈：理论、实现与实验评估”，IEEE 机器人学与自动化汇刊，第 18 卷，第 5 期，第 662–669 页，2002 年。[2] S. Bansal、M. Chen、S. Herbert 和 C. J. Tomlin，“Hamilton-Jacobi 可达性：简要概述与最新进展”，载于《2017 IEEE 第 56 届决策与控制年度会议（CDC）》，2017 年，第 2242–2253 页。[3] C. Robin 和 S. Lacroix，“多机器人目标检测与跟踪：分类体系与综述”，《自主机器人》，第 40 卷，第 729–760 页，2016 年。[4] R. Lowe、Y. I. Wu、A. Tamar、J. Harb、O. Pieter Abbeel 和 I. Mordatch，“混合协作—竞争环境下的多智能体行动者—评论家”，《神经信息处理系统进展》，第 30 卷，2017 年。[5] T. Rashid、M. Samvelyan、C. S. De Witt、G. Farquhar、J. Foerster 和 S. Whiteson，“深度多智能体强化学习的单调价值函数分解”，《机器学习研究杂志》，第 21 卷，第 1 期，第 7234–7284 页，2020 年。[6] K. Zhang、Z. Yang 和 T. Bas¸ar，“多智能体强化学习：理论与算法的选择性概述”，《强化学习与控制手册》，第 321–384 页，2021 年。[7] T. H. Chung、G. A. Hollinger 和 V. Isler，“移动机器人中的搜索与追逃博弈”，《自主机器人》，第 31 卷，第 316 页，2011 年，第 4 期，第 299 页。[8] E. Bakolas 和 P. Tsiotras，“使用动态 Voronoi 图追捕机动目标”，Automatica，第 48 卷，第 2220 页，2012 年，第 9 期，第 2213 页。[9] A. Alexopoulos、T. Schmidt 和 E. Badreddin，“使用无人机进行追逃博弈中的协同追捕”，载于《2015 IEEE/RSJ 国际智能机器人与系统会议（IROS）》。IEEE，2015 年，第 4538–4543 页。[10] C. De Souza、R. Newbury、A. Cosgun、P. Castillo、B. Vidolov 和 D. Kuli´c，“基于深度强化学习的去中心化多智能体追捕”，IEEE 机器人学与自动化快报，第 6 卷，第 3 期，第 4552–4559 页，2021 年。[11] R. Zhang、Q. Zong、X. Zhang、L. Dou 和 B. Tian，“无人机之战：基于深度强化学习、采用在线运动规划的多无人机追逃博弈”，《IEEE 神经网络与学习系统汇刊》，2022 年。[12] H. Huang、J. Ding、W. Zhang 和 C. J. Tomlin，“对抗场景中规划的微分博弈方法：夺旗案例研究”，载于《2011 IEEE 机器人与自动化国际会议》。IEEE，2011 年，第 1451–1456 页。[13] E. Garcia、D. W. Casbeer 和 M. Pachter，“夺旗微分博弈”，载于《2018 IEEE 决策与控制会议（CDC）》。IEEE，2018 年，第 4167–4172 页。[14] M. Jaderberg、W. M. Czarnecki、I. Dunning、L. Marris、G. Lever、A. G. Castaneda、C. Beattie、N. C. Rabinowitz、A. S. Morcos、A. Ruderman 等，“基于群体的强化学习在三维多人游戏中的人类水平表现”，《科学》，第 364 卷，第 6443 期，第 859–865 页，2019 年。[15] D. Shishika 和 V. Kumar，“多人周界防御问题的局部博弈分解”，载于《2018 IEEE 决策与控制会议（CDC）》。IEEE，2018 年，第 2093–2100 页。[16] L. Guerrero-Bonilla 和 D. V. Dimarogonas，“基于集合不变性的周界监视”，IEEE 机器人学与自动化快报，第 6 卷，第 1 期，第 9–16 页，2020 年。[17] S. Velhal、S. Sundaram 和 N. Sundararajan，“面向周界防御的去中心化多机器人时空多任务分配方法”，IEEE 机器人学汇刊，2022 年。[18] K. Margellos 和 J. Lygeros，“到达—规避微分博弈的 Hamilton–Jacobi 表述”，IEEE 自动控制学报，第 56 卷，第 8 期，第 1849–1861 页，2011 年。[19] J. F. Fisac、M. Chen、C. J. Tomlin 和 S. S. Sastry，“具有时变动力学、目标和约束的到达—规避问题”，载于《第 18 届混合系统国际会议：计算与控制》

[20] I. M. Mitchell、A. M. Bayen 和 C. J. Tomlin，《连续动态博弈中可达集的时变 Hamilton-Jacobi 表述》，IEEE 自动控制学报，第50卷第7期，第947–957页，2005年。[21] T. Bas¸ar 和 G. J. Olsder，《动态非合作博弈论》。SIAM，1998年。[22] B. Landry、M. Chen、S. Hemley 和 M. Pavone，《通过平方和优化与动态规划求解到达-规避问题》，载于2018年 IEEE/RSJ 国际智能机器人与系统会议（IROS）。IEEE，2018年，第4325–4332页。[23] Z. Zhou、J. Ding、H. Huang、R. Takei 和 C. Tomlin，《到达-规避问题中的高效路径规划算法》，Automatica，第89卷，第28–36页，2018年。[24] E. Garcia、D. W. Casbeer 和 M. Pachter，《三维空间中一类多人到达-规避微分博弈的最优策略》，IEEE 机器人学与自动化快报，第5卷第3期，第4257–4264页，2020年。[25] H. Fu 和 H. H.-T. Liu，《利用 HJI 方程论证采用速度更快的防御者和凸目标区域的目标防御博弈几何解的合理性》，Automatica，第149卷，文章编号110811，2023年。[26] S. Pan、H. Huang、J. Ding、W. Zhang、C. J. Tomlin 等，《平面中的追捕、逃逸与防御》，载于2012年美国控制会议（ACC）。IEEE，2012年，第4167–4173页。[27] Z. Deng 和 Z. Kong，《针对单个攻击者的多智能体协同追捕-防御策略》，IEEE 机器人学与自动化快报，第5卷第4期，第5772–5778页，2020年。[28] H. Huang、W. Zhang、J. Ding、D. M. Stipanovi´c 和 C. J. Tomlin，《多个追捕者在平面中的有保证分散式追逃》，载于2011年第50届 IEEE 决策与控制会议及欧洲控制会议。IEEE，2011年，第4835–4840页。[29] A. Pierson、Z. Wang 和 M. Sc hwager，《拦截失控机器人：一种利用多个追捕者捕获多个逃逸者的算法》，IEEE 机器人学与自动化快报，第2卷第2期，第530–537页，2016年。[30] R. Yan、Z. Shi 和 Y. Zhong，《两个防御者和一个攻击者参与的到达-规避博弈：一种解析方法》，IEEE 控制论汇刊，第49卷第3期，第1035–1046页，2018年。[31] ——，《通过解析屏障在凸域中进行多人到达-规避博弈的任务分配》，IEEE 机器人学汇刊，第36卷第1期，第107–124页，2019年。[32] ——，《两个防御者和一个攻击者在高维空间中防守子空间》，IEEE 控制论汇刊，第52卷第5期，第3998–4011页，2020年。[33] R. Yan、X. Duan、Z. Shi、Y. Zhong 和 F. Bullo，《基于匹配的三维异构多人到达-规避微分博弈捕获策略》，Automatica，第140卷，文章编号110207，2022年。[34] R. Isaacs，《微分博弈：在战争与追捕、控制和优化中的应用数学理论》。Courier Corporation，1999年。[35] Z. Zhou、W. Zhang、J. Ding、H. Huang、D. M. Stipanovi´c 和 C. J. Tomlin，《采用 Voronoi 划分的协同追捕》，Automatica，第72卷，第64–72页，2016年。[36] B. P. Gerkey 和 M. J. Matari´c，《多机器人系统中任务分配的形式化分析与分类体系》，《国际机器人研究杂志》，第23卷第9期，第939–954页，2004年。[37] G. A. Korsah、A. Stentz 和 M. B. Dias，《多机器人任务分配的综合分类体系》，《国际机器人研究杂志》，第32卷第12期，第1495–1512页，2013年。[38] M. Chen、Z. Zhou 和 C. J. Tomlin，《基于成对结果的多人到达-规避博弈》，IEEE 自动控制学报，第62卷第3期，第1451–1457页，2017年。[39] L. R. Ford 和 D. R. Fulkerson，《网络中的最大流》，《加拿大数学杂志》，第8卷，第399–404页，1956年。[40] J. E. Hopcroft 和 R. M. Karp，《二分图最大匹配的一个 n^5/2 算法》，SIAM 计算杂志，第2卷第4期，第225–231页，1973年。[41] E. Garcia、D. W. Casbeer、A. Von Moll 和 M. Pachter，《多追捕者多逃逸者微分博弈》，IEEE 自动控制学报，第66卷第5期，第2345–2350页，2020年。[42] I. M. Mitchell 等，《水平集方法工具箱》，英属哥伦比亚大学计算机科学系技术报告 TR-2007-11，第31页，2007年。[43] F. Blanchini，《控制中的集合不变性》，Automatica，第35卷第11期，第1747–1767页，1999年。[44] J. Nocedal 和 S. Wright，《数值优化》。Springer Science & Business Media，2006年。[45] S. Boyd、S. P. Boyd 和 L. Vandenberghe，《凸优化》。剑桥大学出版社，2004年。[46] S. Diamond 和 S. Boyd，“Cvxpy：一种嵌入 Python 的凸优化建模语言”，《机器学习研究杂志》，第 17 卷第 1 期，第 2909–2913 页，2016 年。[47] N. Koenig 和 A. Howard，“Gazebo（一种开源多机器人仿真器）的设计与使用范式”，载于*2004 IEEE/RSJ* 国际智能机器人与系统会议（*IROS*）（*IEEE* 编号：*04CH37566*），第 3 卷。IEEE，2004 年，第 2149–2154 页。[48] L. Wang、A. D. Ames 和 M. Egerstedt，“面向无碰撞多机器人系统的安全屏障证书”，*IEEE* 机器人学汇刊，第 33 卷第 3 期，第 661–674 页，2017 年。[49] S. G. Lee、Y. Diaz-Mercado 和 M. Egerstedt，“使用时变密度函数的多机器人控制”，*IEEE* 机器人学汇刊，第 31 卷第 2 期，第 489–493 页，2015 年。[50] H. L. Royden 和 P. Fitzpatrick，实分析。麦克米伦纽约，1988 年，第 32 卷。