

基于二分图的群体追逃博弈任务分配算法

游志航, 于洋, 苏昂, 党志清

青岛大学, 山东 青岛

收稿日期: 2022年9月4日; 录用日期: 2022年10月3日; 发布日期: 2022年10月10日

摘要

本文考察群体追逃博弈中追逐者的任务分配问题。通过构建二分图, 设计全新算法对追逐者进行任务分配, 将群体追逃博弈转化成多个多追一博弈。在多追一情形下给出了追逐者及逃避者的最优策略。运用本文所建立的算法, 在考虑扰动因素的情形下, 原本冗余的追逐者有可能转化为积极的追逐者。数值算例验证了任务分配的可行性及稳定性。

关键词

群体追逃博弈, 二分图, 任务分配, 扰动

Task Allocation Algorithm for Group Pursuit-Evasion Game Based on Bipartite Graph

Zhihang You, Yang Yu, Ang Su, Zhiqing Dang

Qingdao University, Qingdao Shandong

Received: Sep. 4th, 2022; accepted: Oct. 3rd, 2022; published: Oct. 10th, 2022

Abstract

This paper examines the task assignment of chasers in a group pursuit-evasion game. By constructing a bipartite graph, a new algorithm is designed to assign tasks to the chasers, and the group pursuit-evasion game is transformed into a game of multiple chases. In the case of chasing more than one, the optimal strategies of chasers and evaders are given. Using the algorithm established

in this paper, under the condition of considering disturbance factors, the original redundant chasers may be transformed into active chasers. Numerical examples verify the feasibility and stability of task assignment.

Keywords

Group Pursuit-Evasion Game, Bipartite Graph, Task Assignment, Disturbance

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

追逃博弈(Pursuit-Evasion Games)是一类特殊的零和博弈,其中追逐者(Pursuer)致力于在最短的时间内将状态引导到一个给定的目标,而他的对手逃避者(Evader)则希望尽力延迟被捕获。文献[1]总结追逃博弈的近期研究成果,包括一追一博弈、多追一博弈和多追多博弈等。涉及多个追逐者和逃避者的群体追逃问题(MPNE)是目前应用最广泛的领域之一,军事上无人机集群的作战、自然界动物间的捕食猎杀都可作为群体追逃博弈的例子。文献[2]研究追逐者与逃避者最大速度相等情形下的多追一博弈,分别给出追逐者保证捕获和逃避者保证逃跑的条件。文献[3] [4]研究逃避者具有速度优势的多追一博弈,文献[3]考察追逐者与逃避者的最优策略,文献[4]关注捕获发生的条件。文献[5]运用几何方法研究时间最优的多追一博弈,并给出求解追逃双方最优策略的算法。文献[6]给出时间最优的多追一博弈的全局 Stackelberg 解(Global Stackelberg Solution),并证明解的鲁棒性。

近年来,群体追逃博弈的主要解决方法是把复杂的多追多博弈转化成多个一追一(1P1E)或多追一(MP1E)博弈,这必然涉及到追逐者的任务分配问题。文献[7] [8] [9] [10]使用 Voronoi 图解决任务分配问题。文献[7]提出一种在整个空间上构建追逐者的 Voronoi 图的算法,把多追一博弈转化为地区分配问题加以解决。文献[8]使用动态 Voronoi 图研究接力追逃博弈,给出追逐者的任务分配方案。文献[9]以[8]为基础研究动态流场环境下的接力追逃博弈,给出追逐者的任务分配,同时保证任务分配随时间动态更新。文献[10]研究三维动态流场环境下的群体追逃博弈,并基于 Voronoi 图设计算法对追逐者进行任务分配。文献[11]通过构建追逐者与逃避者之间的通讯图(Communication Graphs)对追逐者进行任务分配,求解零和情形下的最大最小策略,并给出非合作情形下群体追逃博弈的纳什均衡。文献[12]运用 Apollonius 圆作为工具求解具有边界限制的追逃博弈,给出一追一博弈追逃双方的最优策略,并把结论推广到多追一、多追多博弈。文献[13]研究追逐者策略集受限制情形下的任务分配问题,并提出将“积极(Active)”与“冗余(Redundant)”的概念应用到分配问题中。

文献[12]中的最优策略在多追一博弈中被证明是有效的,但把结论扩展到多追多博弈中时,空间的维度和局中人的数目增加导致计算难度增大。此时一种可行的方法是将群体追逃博弈转化为多个一追一或多追一子博弈,在每个时刻为每个追逐者分配一个逃避者作为目标。即使这种方法只会得到局部最优解(Sub-optimal Solution),在追逐者和逃避者规模较大时这是可接受的。本文的主要目的是针对实际应用场景,考虑到追逐者可能发生机械故障等意外情况,为最小化这种情况产生的影响,提出一种任务分配算法,并在考察扰动因素的情形下验证冗余的追逐者在使用该算法的情形下有可能会转化为积极的追逐者。

本文的其余部分安排如下:第2节对多追多博弈进行描述;第3节定义预计捕获时间,给出多追一博弈中追逐者与逃避者的最优策略;第4节中以预计捕获时间为指标,利用 Kuhn-Munkres 算法为每个追

逐者分配一个逃避者作为目标, 将群体追逃博弈划分为多个一追一或多追一子博弈; 第 5 节使用数值算例验证了任务分配的可行性, 并在有扰动因素的情形下验证了冗余的追逐者有可能会转化为积极的追逐者; 第 6 节对全文进行总结并提出后续研究方向。

2. 问题描述

本节考察 M 个追逐者和 N 个逃避者参与的追逃微分博弈, 记为 $\Gamma(M, N)$, 博弈发生在 2 维欧式空间上且无边界。本文假设追逐者知道逃避者的最优策略, 所考察的模型事实上成为 Stackelberg 博弈。追逃双方均具有简单运动的性质, 即能够无约束地瞬时改变前进方向。其中追逐者的目的是捕获所有逃避者, 捕获的方式为点捕获, 即某个追逐者与逃避者的位置重合; 与之相反, 逃避者的目的是尽可能防止被捕获。记第 i 个追逐者为 P_i , 第 j 个逃避者为 E_j 。

追逐者与逃避者的状态方程如下:

$$\begin{aligned} \dot{x}_i^p(t) &= v_p \cos \phi_i(t), & x_i^p(t_0) &= x_{i0}^p, \\ \dot{y}_i^p(t) &= v_p \sin \phi_i(t), & y_i^p(t_0) &= y_{i0}^p, & i &= 1, \dots, M, \\ \dot{x}_j^e(t) &= v_e \cos \psi_j(t), & x_j^e(t_0) &= x_{j0}^e, \\ \dot{y}_j^e(t) &= v_e \sin \psi_j(t), & y_j^e(t_0) &= y_{j0}^e, & j &= 1, \dots, N, \end{aligned} \quad (1)$$

其中 $\mathbf{p}_i(t) = (x_i^p(t), y_i^p(t)) \in \mathbb{R}^2$, $\mathbf{e}_j(t) = (x_j^e(t), y_j^e(t)) \in \mathbb{R}^2$ 分别表示追逐者 P_i , 逃避者 E_j 在 t 时刻的位置, $\mathbf{p}_i(t_0) = (x_i^p(t_0), y_i^p(t_0)) \in \mathbb{R}^2$, $\mathbf{e}_j(t_0) = (x_j^e(t_0), y_j^e(t_0)) \in \mathbb{R}^2$ 分别为追逐者 P_i , 逃避者 E_j 在初始时刻 t_0 的位置。 $\phi_i(t)$, $\psi_j(t) \in [-\pi, \pi)$ 分别表示追逐者 P_i , 逃避者 E_j 的输入控制, 追逃双方通过在每个时刻选择策略控制前进方向。常量 v_p , v_e 是追逐者与逃避者的线速度, 本文假设 $v_p > v_e$, 以保证点捕获一定在有限时间内发生。记捕获逃避者 E_j 所用时间为

$$T_j = \min \{t \mid \exists i \in \{1, \dots, M\}, \rho(\mathbf{p}_i(t), \mathbf{e}_j(t)) = 0\}, \quad (2)$$

其中 $\rho(\mathbf{p}_i(t), \mathbf{e}_j(t)) = \sqrt{(x_j^e(t) - x_i^p(t))^2 + (y_j^e(t) - y_i^p(t))^2}$ 表示 P_i 与 E_j 在 t 时刻的距离。

所有逃避者均被捕获时博弈终止, 终止时刻

$$T = \max \{T_j\}, j = 1, \dots, N. \quad (3)$$

定义 $\Gamma(M, N)$ 的值函数

$$V = \min_{\phi(t)} \max_{\psi_j(t)} \sum_{j=1}^N T_j, i = 1, 2, \dots, M; j = 1, 2, \dots, N. \quad (4)$$

对于较大的 M, N , 直接求解(4)式得到追逃双方的最优策略在实际应用中是不可行的, 本文通过如下步骤求解追逃双方的局部最优策略(Sub-optimal Strategy):

步骤 1 在 $\Gamma(M, N)$ ($M \geq N > 1$) 中, 为每个追逐者分配一个逃避者作为目标, 将 $\Gamma(M, N)$ 划分为 N 个一追一或多追一子博弈。

步骤 2 在 $\Gamma(M, 1)$ ($M > 1$) 中, 考察追逐者知道逃避者的最优策略的情形, 给出追逐者和逃避者的最优策略。

3. 多追一博弈

3.1. 追逃双方的最优控制求解

本节考察由 M 个追逐者和 1 个逃避者参与的博弈 $\Gamma(M, 1)$ 。不失一般性, 记逃避者为 E_1 , 距离最近

的追逐者为 $P_I (I \in \{1, \dots, M\})$ ，追逐双方从初始时刻开始，每经过一次时间步长为 Δt 的迭代需要输入一次控制。由于 v_p ， v_e 是常量，求解最短捕获时间的控制等价于求解距离下降最快的控制。故逃避者 E_1 的目的是在已知状态 $\mathbf{p}_I = \mathbf{p}_I(t)$ ， $\mathbf{p}_i = \mathbf{p}_i(t) (i=1, \dots, M, i \neq I)$ ， $\mathbf{e}_1 = \mathbf{e}_1(t)$ 和步长 Δt 情形下选择策略 $\psi_1 = \psi_1(t)$ 使得自身尽可能远离最近的追逐者，即逃避者 E_1 的值函数可设为

$$V_{e_1}(\mathbf{p}_I, \mathbf{e}_1; \psi_1) = \max_{\psi_1} \rho(\mathbf{p}_I, \mathbf{e}_1(t + \Delta t)), \quad (5)$$

其中

$$\rho(\mathbf{p}_I, \mathbf{e}_1(t + \Delta t)) = \sqrt{(x_1^e + \Delta t v_e \cos \psi_1 - x_1^p)^2 + (y_1^e + \Delta t v_e \sin \psi_1 - y_1^p)^2}.$$

下面寻找使逃避者值函数最大化的策略，令 ρ 对 ψ_1 求偏导等于 0，得

$$\begin{aligned} \frac{\partial \rho}{\partial \psi_1} &= \frac{2(x_1^e + \Delta t v_e \cos \psi_1 - x_1^p) \times (-\Delta t v_e \sin \psi_1)}{2\rho(\mathbf{p}_I, \mathbf{e}_1(t + \Delta t))} \\ &+ \frac{2(y_1^e + \Delta t v_e \sin \psi_1 - y_1^p) \times \Delta t v_e \cos \psi_1}{2\rho(\mathbf{p}_I, \mathbf{e}_1(t + \Delta t))} = 0, \end{aligned} \quad (6)$$

即

$$\frac{\sin \psi_1^*}{\cos \psi_1^*} = \frac{y_1^e - y_1^p}{x_1^e - x_1^p}. \quad (7)$$

解上式可得

$$\psi_1^* = \arctan 2 \frac{y_1^e - y_1^p}{x_1^e - x_1^p}. \quad (8)$$

由(8)式可知，逃避者的最优策略跟自身位置 \mathbf{e}_1 及距其最近的追逐者的位置 \mathbf{p}_I 有关。

追逐者 $P_i (i=1, \dots, M)$ 的目的是在已知状态 $\mathbf{p}_I = \mathbf{p}_I(t)$ ， $\mathbf{p}_i = \mathbf{p}_i(t) (i=1, \dots, M, i \neq I)$ ， $\mathbf{e}_1(t)$ ，步长 Δt 和逃避者的最优策略 ψ_1^* 的情形下选择策略 ϕ_i 使自身尽可能接近逃避者 E_1 ，即追逐者 $P_i (i=1, \dots, M)$ 的值函数可设为

$$V_{p_i}(\mathbf{p}_i, \mathbf{e}_1; \phi_i) = \min_{\phi_i} \rho(\mathbf{p}_i(t + \Delta t), \mathbf{e}_1(t + \Delta t)), \quad (9)$$

其中

$$\rho(\mathbf{p}_i(t + \Delta t), \mathbf{e}_1(t + \Delta t)) = \sqrt{(x_1^e + \Delta t v_e \sin \psi_1^* - x_1^p - \Delta t v_p \cos \phi_i)^2 + (y_1^e + \Delta t v_e \cos \psi_1^* - y_1^p - \Delta t v_p \sin \phi_i)^2}.$$

寻找使追逐者值函数最小化的策略，令 ρ 对 ϕ_i 求偏导等于零，得

$$\begin{aligned} \frac{\partial \rho}{\partial \phi_i} &= \frac{2(x_1^e + \Delta t v_e \sin \psi_1^* - x_1^p - \Delta t v_p \cos \phi_i) \times \Delta t v_p \sin \phi_i}{2\rho(\mathbf{p}_i(t + \Delta t), \mathbf{e}_1(t + \Delta t))} \\ &+ \frac{2(y_1^e + \Delta t v_e \cos \psi_1^* - y_1^p - \Delta t v_p \sin \phi_i) \times (-\Delta t v_p \cos \phi_i)}{2\rho(\mathbf{p}_i(t + \Delta t), \mathbf{e}_1(t + \Delta t))} = 0, \end{aligned} \quad (10)$$

即

$$\frac{\sin \phi_i^*}{\cos \phi_i^*} = \frac{y_1^e + v_e \sin \psi_1^* - y_1^p}{x_1^e + v_e \cos \psi_1^* - x_1^p}. \quad (11)$$

解上式可得

$$\phi_i^* = \arctan 2 \frac{y_1^e + v_e \sin \psi_1^* - y_i^p}{x_1^e + v_e \cos \psi_1^* - x_i^p}. \quad (12)$$

由(12)式可知, 追逐者 P_i 的最优策略跟自身位置 \mathbf{p}_i 及逃避者的位置 \mathbf{e}_1 和逃避者的最优策略 ψ_1^* 有关。

3.2. 最优控制的几何解释

本节使用 Apollonius 圆[14]来描述(12)式。Apollonius 圆是一组到两定点距离满足给定比值的点集。设追逐者 P_i 和逃避者 E_j 的位置为 $\mathbf{p}_i(t) = (x_i^p(t), y_i^p(t))$, $\mathbf{e}_j(t) = (x_j^e(t), y_j^e(t))$, 记追逐者 P_i 和逃避者 E_j 对应的 Apollonius 圆为 $A_{ij}(t)$, 圆心位于 $\mathbf{O}_{ij}(t) = \left(\frac{x_j^e - \alpha^2 x_i^p}{1 - \alpha^2}, \frac{y_j^e - \alpha^2 y_i^p}{1 - \alpha^2} \right)$, 半径为 $r_{ij}(t) = \frac{\alpha}{1 - \alpha^2} \rho(\mathbf{p}_i, \mathbf{e}_j)$ 。 $\mathbf{u}(t)$ 是 Apollonius 圆 $A_{ij}(t)$ 上的某一点, 常量 $\alpha = \frac{\rho(\mathbf{u}, \mathbf{e}_j)}{\rho(\mathbf{u}, \mathbf{p}_i)} = \frac{v_e}{v_p} < 1$ 。假设 E_j 在 t 时刻选择沿方向 $\vec{w}_1(t)$ 逃跑, P_i 都有一个对应的方向 $\vec{\phi}_i(\vec{w}_1(t))$ 保证其在 Apollonius 圆 $A_{ij}(t)$ 上的某一点 $\mathbf{p}_i(T_{ji})$ 处捕获 E_j 。

定义预计捕获时间 T_{ji} 为追逐者 P_i 预计捕获逃避者 E_j 的时刻, 则有 $T_j = \min\{T_{ji}\}, i=1, \dots, M$ 。其中 T_{ji} 满足

$$\rho(\mathbf{p}_i(T_{ji}), \mathbf{e}_j(T_{ji})) = 0, \frac{\rho(\mathbf{e}_j(T_{ji}), \mathbf{e}_j)}{\rho(\mathbf{p}_i(T_{ji}), \mathbf{p}_i)} = \frac{v_e}{v_p}, \quad (13)$$

可得

$$T_{ji} = \frac{\rho(\mathbf{p}_i(T_{ji}), \mathbf{p}_i)}{v_p}. \quad (14)$$

对于上式, 可以通过联立 Apollonius 圆的方程与逃避者的速度方向所在直线方程, 求得两个交点, 并选取使 T_{ji} 值较大的点作为 $\mathbf{p}_i(T_{ji})$, 进而可以求得 T_{ji} 的值。因此可以将追逐者 P_i 的最优策略改写为

$$\phi_i^* = \arctan \frac{x_i^p(T_{ji}) - x_i^p}{y_i^p(T_{ji}) - y_i^p}, i=1, \dots, M. \quad (15)$$

至此本节完成 $\Gamma(M, 1)$ 追逃双方最优策略的求解: 考察由 M 个追逐者和 1 个逃避者参与的博弈 $\Gamma(M, 1)$, 给定任意 $t \in [t_0, T_1]$ 时刻所有局中人的位置, 假设逃避者 E_1 和追逐者 P_i 的值函数分别为(5)和(9)式, 则逃避者 E_1 在 t 时刻的最优策略为(8)式, 追逐者 P_i 在 t 时刻的最优策略为(15)式。

图 1 直观地描述了博弈 $\Gamma(2, 1)$, 可以看出 $\mathbf{e}_1(t_0)$, $\mathbf{p}_1(T_{11})$ 和 $\mathbf{p}_2(T_{12})$ 三点共线, 由于 $\mathbf{p}_2(T_{12})$ 距离 $\mathbf{e}_1(t_0)$ 更近, 假设所有人均不改变速度方向, 追逐者 P_2 将在 $\mathbf{p}_2(T_{12})$ 处捕获逃避者, 捕获时间为 T_{12} 。

注 1 需要特别说明的是, 双方的最优策略是具有反馈性质的, 当某个时刻距离最小的追逐者与上个时刻不同时, 追逃双方通过重新计算(8)和(15)式动态更新最优策略。

注 2 当某个 t 时刻有多个追逐者与逃避者 E_1 的距离同为最小值时, 值函数 V_{e_1} 不满足连续可微, 故 E_1 在 t 时刻选择随机方向前进。追逐者 $P_i (i=1, \dots, M)$ 在 t 时刻无法知道 E_1 的最优策略, 故 $P_i (i=1, \dots, M)$ 选择

$$\phi_i^* = \arctan 2 \frac{y_1^e - y_i^p}{x_1^e - x_i^p}. \quad (16)$$

作为(15)式的替代。在 t 时刻双方都不是最优策略，但是当步长 $\Delta t \rightarrow 0$ 时，双方的策略对捕获时间 T_i 的影响可以忽略。

注 3 在逃避者 E_i 选择最优策略的情形下，每个时刻只有距离最近的追逐者 P_j 和预计捕获时间 T_{ji} 最短的追逐者 P_i 会对捕获时间 T_i 产生影响。

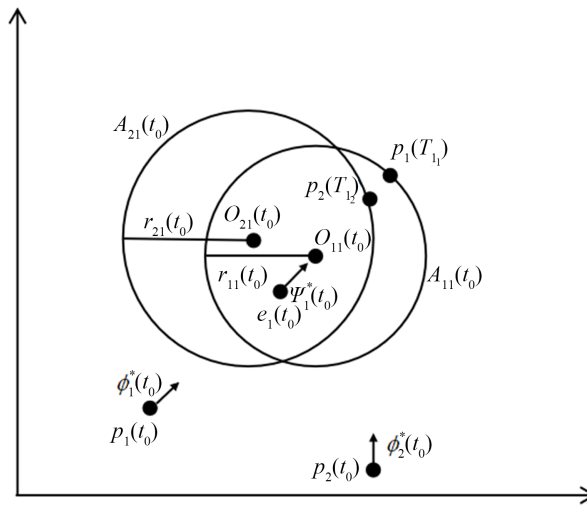


Figure 1. The optimal strategy of chasing and fleeing in game $\Gamma(2,1)$

图 1. 博弈 $\Gamma(2,1)$ 中追逃双方的最优策略

4. 群体追逃博弈任务分配

本节考察由 M 个追逐者和 N 个逃避者参与的博弈 $\Gamma(M, N)$ ($M \geq N > 1$)，主要研究追逐者的任务分配问题。

4.1. 二分图的构建

构建 $\Gamma(M, N)$ 对应的二分图[15] $G = (V, E, A)$ ，其中顶点集 $V = \{V_{p1}, \dots, V_{pM}, V_{e1}, \dots, V_{eN}\}$ 表示 M 个追逐者和 N 个逃避者的顶点集合，它可以分为两个子集 $V_1 = \{V_{p1}, \dots, V_{pM}\}$ 和 $V_2 = \{V_{e1}, \dots, V_{eN}\}$ ，记顶点 V_{pi} 与 V_{ej} 之间有边 E_{ij} ，故边集 $E = \{E_{ij} \mid i=1, \dots, M; j=1, \dots, N\}$ ，同时边 E_{ij} 上有权值 a_{ij} ，权矩阵 $A = \{a_{ij} \mid i=1, \dots, M; j=1, \dots, N\}$ 。

构建 $\Gamma(M, N)$ 对应的任务分配矩阵 $\mu^{M \times N} = \{\mu_{ij} \mid i=1, \dots, M; j=1, \dots, N\}$ ，如果将逃避者 E_j 分配给追逐者 P_i ，则 $\mu_{ij} = 1$ ，否则 $\mu_{ij} = 0$ 。这里规定一个追逐者只能选择一个逃避者作为目标，即

$$\sum_{j=1}^N \mu_{ij} \leq 1, i=1, 2, \dots, M.$$

4.2. 基于扰动因素的冗余追逐者的任务分配策略

由注 3 得到，在逃避者选择最优策略的情形下， $\Gamma(M, 1)$ ($M > 1$) 中至多有 $M - 1$ 个追逐者可以认为是

冗余的，他们对捕获时间没有影响。

下面考察具有扰动因素的群体追逃博弈 $\tilde{\Gamma}(M, N)$ ($M \geq N$)。假设每个追逐者在博弈开始后均有概率发生故障，导致其无法继续参与博弈。

任务分配 $K = (k_1, k_2, \dots, k_N)$ 是一个 N 维向量，其中 k_j 表示分配给逃避者 E_j 的追逐者的个数， $j = 1, \dots, N$ 。分配给逃避者 E_j 的追逐者发生故障的个数可以视为离散随机变量 X_j 。假设 X_j 服从参数为 k_j 和 ε 的二项分布，记为 $X_j \sim B(k_j, \varepsilon)$ ，其中 ε 为每个追逐者发生故障的概率。

定义事件 Z ：每个逃避者在博弈开始后至少分配到 1 个追逐者进行追逐，则

$$P(Z) = \prod_{j=1}^N P(X_j \leq k_j - 1) = \prod_{j=1}^N (1 - \varepsilon^{k_j}). \quad (17)$$

下面寻找使事件 Z 发生概率最大的分配 K ，即求解非线性规划问题

$$\begin{aligned} & \max_{k_j} P(Z), \\ & \text{s.t. } k_1 + k_2 + \dots + k_N = M, \\ & \quad k_j \geq 1, j = 1, \dots, N. \end{aligned} \quad (18)$$

由(17)式得

$$\ln P(Z) = \sum_{j=1}^N \ln(1 - \varepsilon^{k_j}). \quad (19)$$

设 $f(x) = 1 - \varepsilon^x$ ，由 $f''(x) < 0$ 得 $f(x)$ 是上凸函数，可得 $\ln f(x)$ 也是上凸函数，那么有

$$\frac{\sum_{j=1}^N \ln f(k_j)}{N} \leq \ln f\left(\frac{\sum_{j=1}^N k_j}{N}\right) = \ln f\left(\frac{M}{N}\right) \quad (20)$$

当且仅当 $k_1 = k_2 = \dots = k_N = \frac{M}{N}$ 时等号成立，此时 $P(Z)$ 取最大值，因此在进行任务分配时应尽可能平均分配追逐者。

4.3. 基于二分图的任务分配

设追逐者 P_i 预计捕获逃避者 E_j 的时间为 T_{ji} ，给出任意时刻所有追逐者和逃避者的位置，求解所有 T_{ji} ，并作为权值记录到二分图 G 中。不断调用 Kuhn-Munkres 算法[15]，从未分配的追逐者中选择预计捕获时间总和最小的 N 个追逐者分别分配给对应的逃避者，将已分配的追逐者和与之相关的边从二分图 G 中删去，直到任务分配结束为止。

本文建立 BGBTA (Bipartite Graph Based Target Assignment)，尝试刻画整个追逃博弈及其相应的动态任务分配过程。

BGBTA

Input 追逐者与逃避者的数目 M, N ，初始状态 $p_i(t_0), e_j(t_0), i = 1, \dots, M; j = 1, \dots, N$ ，迭代步长 Δt

Step 1 为每个逃避者 E_j 计算最近的追逐者 P_i^j 。

Step 2 根据初始状态计算所有预计捕获时间 T_{ji} 。

Step 3 构建二分图 G 。

Step 4 调用 Kuhn-Munkres 算法, 从未分配追逐者集合中选出最多 N 个追逐者, 为每个逃避者分配一个追逐者, 将已分配的追逐者和与之相关的边从图上删去, 更新二分图 G , 直到所有追逐者均已分配。

Step 5 每个逃避者选择(8)式作为策略, 每个追逐者选择(15)式作为策略。直到某个时刻存在某个逃避者的最近的追逐者与上个时刻不同或博弈终止。

Step 6 如果博弈终止, 则算法终止; 否则转 Step 1。

事实上, 只需要将追逐者与逃避者的数目 M, N , 初始状态 $p_i(t_0), e_j(t_0), i=1, \dots, M; j=1, \dots, N$, 迭代步长 Δt 输入 BGBTA, 它能够将 $\Gamma(M, N)$ 按照逃避者划分为 N 个多追一子博弈, 并使事件 Z 发生的概率最大。每个子博弈中追逐双方使用第 3 节给出的最优策略, 进而得到一个 MPNE 问题的局部最优解, 且最优策略在迭代过程中实现动态更新。

5. 数值算例

本节给出追逐者与逃避者数目分别为 $M = N$ 、 $N = 2N$ 、 $M = 5N$ 的数值算例, 并考虑扰动因素。求解所有预计捕获时间 T_{j_i} , 作为权值记录到二分图 G 。依次给出任务分配的二分图、任务分配矩阵、追逐双方的运动轨迹, 计算全部逃避者被捕获的时间总和 T 。设追逐者速度为 $\alpha = 1 \text{ m/s}$, 逃避者速度为 $\beta = 0.6 \text{ m/s}$, 迭代步长 $\Delta t = 0.1 \text{ s}$ 。

5.1. $M = N$ 情形

追逐者数量 $M = 3$, 逃避者数量 $N = 3$ 。追逐者与逃避者的初始状态分别为

$$\begin{aligned} p_1(t_0) &= (4, 28), & p_2(t_0) &= (19, 3), & p_3(t_0) &= (25, 5), \\ e_1(t_0) &= (22, 1), & e_2(t_0) &= (9, 2), & e_3(t_0) &= (3, 25). \end{aligned}$$

由 BGBTA 得到 t_0 时刻追逐者任务分配的二分图及其相应的任务分配矩阵如图 2 所示, 追逐者和逃避者的运动轨迹如图 3 所示。

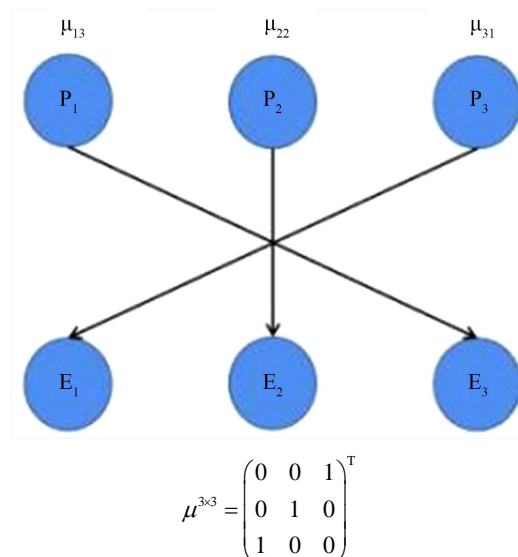


Figure 2. Bipartite graph of task assignments of chasers in $\Gamma(3,3)$ and their task assignment matrix

图 2. $\Gamma(3,3)$ 中追逐者任务分配的二分图及其任务分配矩阵

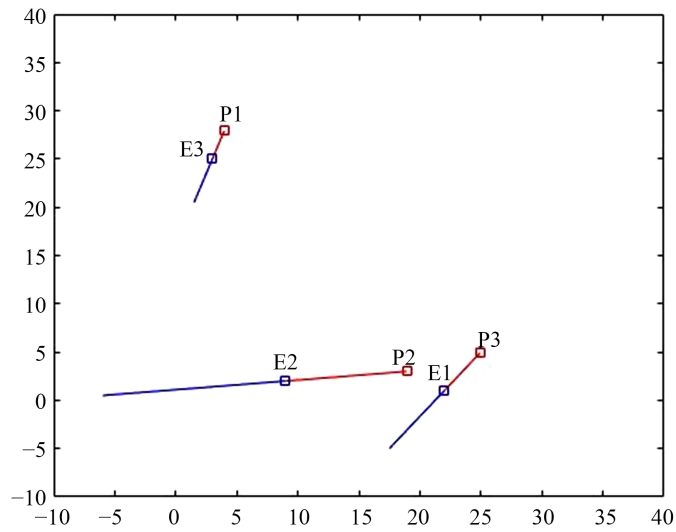


Figure 3. The trajectory of the chasing and fleeing parties in $\Gamma(3,3)$

图 3. $\Gamma(3,3)$ 中追逃双方的运动轨迹

5.2. $M = 2N$ 情形

追逐者数量 $M = 6$ ，逃避者数量 $N = 3$ 。追逐者与逃避者的初始状态分别为

$$\begin{aligned} p_1(t_0) &= (4, 28), & p_2(t_0) &= (19, 3), & p_3(t_0) &= (25, 5), \\ p_4(t_0) &= (9, 17), & p_5(t_0) &= (5, 30), & p_6(t_0) &= (20, 6), \\ e_1(t_0) &= (22, 1), & e_2(t_0) &= (9, 2), & e_3(t_0) &= (3, 25). \end{aligned}$$

由 BGBTA 得到 t_0 时刻追逐者任务分配的二分图及其相应的任务分配矩阵如图 4 所示，追逐者和逃避者的运动轨迹如图 5 所示。

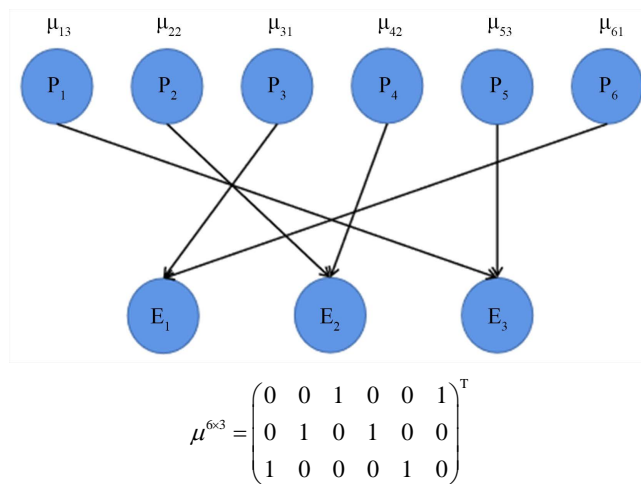


Figure 4. Bipartite graph of task assignments of chasers in $\Gamma(6,3)$ and their task assignment matrix

图 4. $\Gamma(6,3)$ 中追逐者任务分配的二分图及其任务分配矩阵

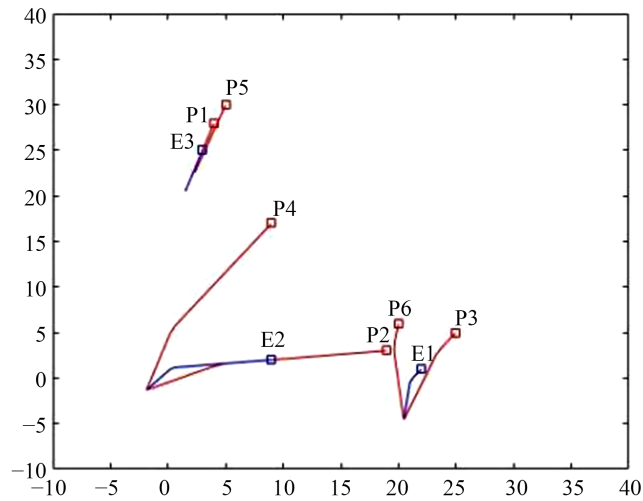


Figure 5. The trajectory of the chasing and fleeing parties in $\Gamma(6,3)$

图 5. $\Gamma(6,3)$ 中追逃双方的运动轨迹

5.3. $M = 5N$ 情形

追逐者数量 $M = 15$ ，逃避者数量 $N = 3$ 。追逐者与逃避者的初始状态分别为

$$\begin{aligned}
 p_1(t_0) &= (25, 28), & p_2(t_0) &= (4, 28), & p_3(t_0) &= (19, 3), \\
 p_4(t_0) &= (9, 17), & p_5(t_0) &= (29, 29), & p_6(t_0) &= (5, 30), \\
 p_7(t_0) &= (29, 15), & p_8(t_0) &= (25, 5), & p_9(t_0) &= (13, 28), \\
 p_{10}(t_0) &= (24, 29), & p_{11}(t_0) &= (30, 3), & p_{12}(t_0) &= (12, 29), \\
 p_{13}(t_0) &= (21, 23), & p_{14}(t_0) &= (23, 12), & p_{15}(t_0) &= (20, 6), \\
 e_1(t_0) &= (22, 1), & e_2(t_0) &= (9, 2), & e_3(t_0) &= (3, 25).
 \end{aligned}$$

由 BGBTA 得到 t_0 时刻追逐者任务分配的二分图及其相应的任务分配矩阵如图 6 所示，追逐者和逃避者的运动轨迹如图 7 所示。

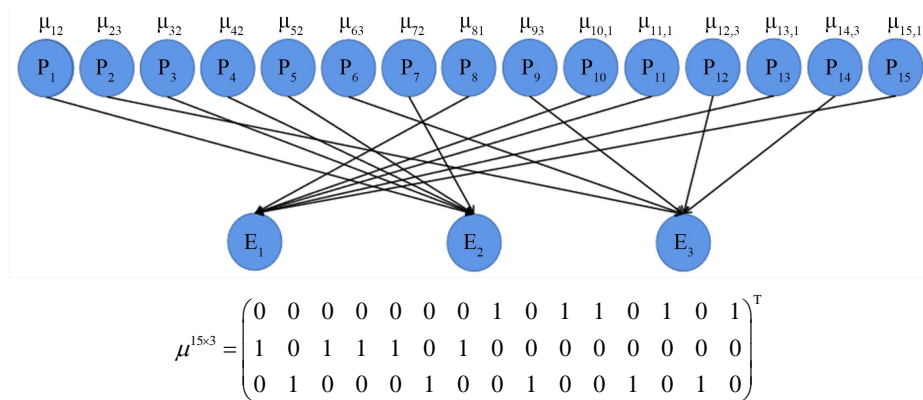


Figure 6. The bipartite graph of the task assignment of the chaser in $\Gamma(15,3)$ and its task assignment matrix

图 6. $\Gamma(15,3)$ 中追逐者任务分配的二分图及其任务分配矩阵

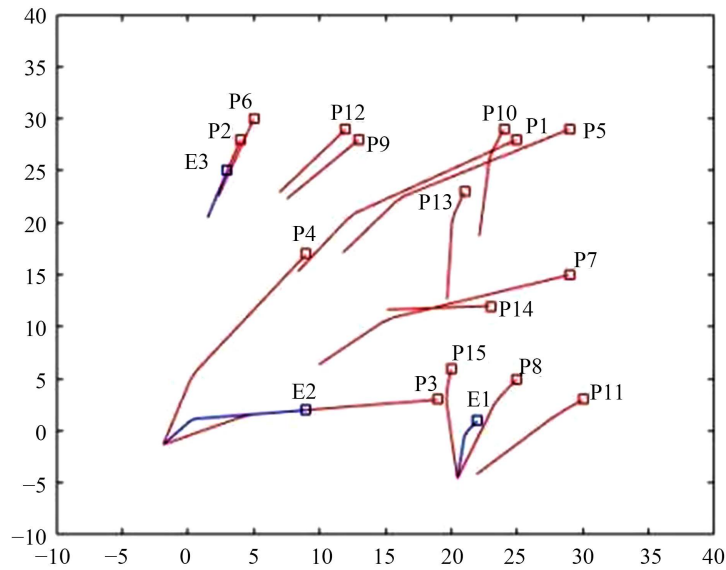


Figure 7. The trajectory of the pursuit and escape in $\Gamma(15,3)$

图 7. $\Gamma(15,3)$ 中追逃双方的运动轨迹

5.4. 加入扰动因素情形

下面考察具有扰动因素的群体追逃博弈，假设追逐者 P_8 在 3 s 时发生故障，无法继续参与博弈。由 BGBTA 得到 $\tilde{\Gamma}(6,3)$ 与 $\tilde{\Gamma}(15,3)$ 追逐者和逃避者的运动轨迹分别如图 8、图 9 所示。其中追逐者任务分配的二分图及其任务分配矩阵与不加入扰动因素时相同。

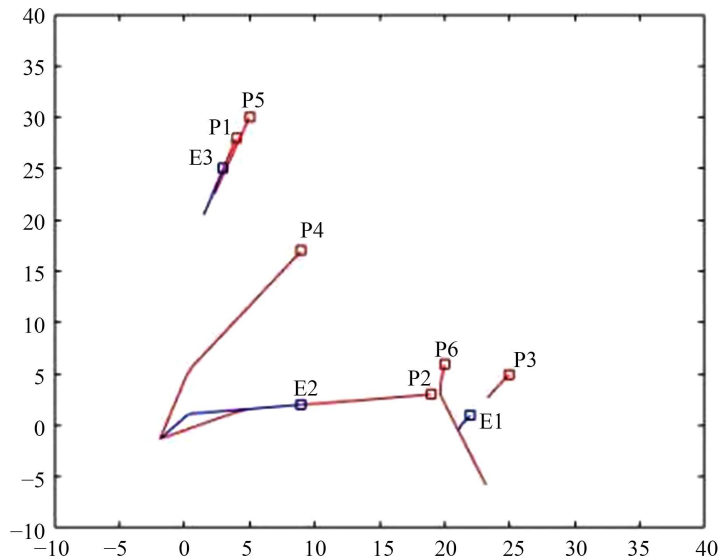


Figure 8. The trajectories of the chasing and fleeing parties in Game $\tilde{\Gamma}(6,3)$ with disturbance factors

图 8. 具有扰动因素的博弈 $\tilde{\Gamma}(6,3)$ 中追逃双方运动轨迹

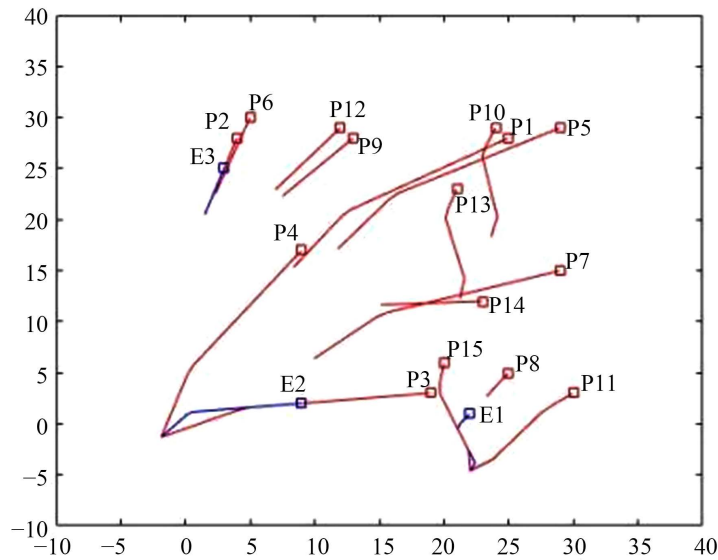


Figure 9. The trajectories of chasing and fleeing parties in Game $\tilde{\Gamma}(15,3)$ with disturbance factors

图 9. 具有扰动因素的博弈 $\tilde{\Gamma}(15,3)$ 中追逃双方运动轨迹

5.5. 结果分析

表 1 给出五种情形下的捕获时间, 表 2 给出五种情形下的任务分配。由表 2, $\Gamma(15,3)$ 使用 BGBTA 时最先分配的 3 个追逐者与 $\Gamma(3,3)$ 相同, $\Gamma(15,3)$ 使用 BGBTA 时最先分配的 6 个追逐者与 $\Gamma(6,3)$ 相同。根据表 1 和表 2, 分配 2 个追逐者捕获 E_1 的时间小于分配 1 个追逐者捕获 E_1 的时间, 而分配 5 个追逐者捕获 E_1 的时间和分配 2 个追逐者相同, 这说明 P_8 和 P_{15} 是积极的, P_{11} , P_{13} 和 P_{10} 是冗余的。

类似地, 对于 E_2 , 分配 2 个追逐者捕获 E_1 的时间小于分配 1 个追逐者捕获 E_1 的时间, 而分配 5 个追逐者捕获 E_1 的时间和分配 2 个追逐者相同, 这说明 P_3 , P_4 是积极的, P_5 , P_7 和 P_1 是冗余的。对于 E_3 , 分配 1 个追逐者和 5 个追逐者捕获时间没有差别, 这同样说明 P_2 是积极的, P_6 , P_9 , P_{12} 和 P_{14} 是冗余的。综上, 在 $\Gamma(15,3)$ 中 15 个追逐者只有 P_2 , P_3 , P_4 , P_8 和 P_{15} 是积极的, 其余 10 个追逐者是冗余的。

针对具有扰动因素的情形, $\tilde{\Gamma}(6,3)$ 中 3s 后由 P_8 , P_{15} 等 2 个追逐者追逐 E_1 变为 P_{15} 独自追逐, 最后捕获 E_1 的时间大于 $\Gamma(6,3)$, 但小于 $\Gamma(3,3)$, 这说明 P_8 和 P_{15} 是积极的。但是, 在 $\tilde{\Gamma}(15,3)$ 中 3s 后由 P_8 , P_{15} 等 5 个追逐者追逐 E_1 变为 P_{15} 等 4 个追逐者追逐, 最后捕获 E_1 的时间大于 $\Gamma(6,3)$, 但小于 $\tilde{\Gamma}(6,3)$, 这说明在 $\Gamma(15,3)$ 中冗余的 P_{11} 在 $\tilde{\Gamma}(15,3)$ 中转化为了积极的追逐者。

Table 1. Capture time of the group pursuit-evasion game in five scenarios

表 1. 五种情形下的群体追逃博弈的捕获时间

	T_1	T_2	T_3	T
$\Gamma(3,3)$	12.6	25.2	8.0	45.8
$\Gamma(6,3)$	11.0	22.0	8.0	41.0
$\Gamma(15,3)$	11.0	22.0	8.0	41.0
$\tilde{\Gamma}(6,3)$	12.6	22.0	8.0	42.6
$\tilde{\Gamma}(15,3)$	11.4	22.0	8.0	41.4

Table 2. Task assignment of the group pursuit-evasion game in five scenarios
表 2. 五种情形下的群体追逃博弈的任务分配

	E_1	E_2	E_3
$\Gamma(3,3)$	P_8	P_3	P_2
$\Gamma(6,3)$	P_8, P_{15}	P_3, P_4	P_2, P_6
$\Gamma(15,3)$	$P_8, P_{15}, P_{11}, P_{13}, P_{10}$	P_3, P_4, P_5, P_7, P_1	$P_2, P_6, P_9, P_{14}, P_{12}$
$\tilde{\Gamma}(6,3)$	P_8, P_{15}	P_3, P_4	P_2, P_6
$\tilde{\Gamma}(15,3)$	$P_8, P_{15}, P_{11}, P_{13}, P_{10}$	P_3, P_4, P_5, P_7, P_1	$P_2, P_6, P_9, P_{14}, P_{12}$

6. 结论

本文的主要贡献是针对追逐者已知逃避者最优策略情形下的群体追逃博弈，在构建二分图的基础上，设计算法对追逐者进行任务分配，进而将群体追逃博弈转化成多个一追一或多追一子博弈。在多追一子博弈中借助 Apollonius 圆给出追逐者及逃避者在任意时刻的最优策略，进而得到群体追逃博弈的一个局部最优解。由数值算例分析可得，在群体追逃博弈 $\Gamma(M, N)$ 中，若博弈双方均采用最优策略，最多能有 $M - N$ 个追逐者可以视为冗余的。针对实际应用场景，考虑到追逐者可能发生机械故障等意外情况，为最小化这种情况产生的影响，本文为所有追逐者都分配了目标。本文所建立算法的优越性在于，在考虑扰动因素的情形下，原本冗余的追逐者有可能转化为积极的追逐者。

后续的研究工作包括以下几个方向：一是研究空间中具有障碍情形下的群体追逃博弈；二是研究追逐者和/或逃避者速度不一致情形下的群体追逃博弈；三是在高维空间中研究考察上述情形。

参考文献

- [1] Weintraub, I.E., Pachter, M. and Garcia, E. (2020) An Introduction to Pursuit-Evasion Differential Games. 2020 *IEEE American Control Conference (ACC)*, Denver, 1-3 July 2020, 1049-1066. <https://doi.org/10.23919/ACC45564.2020.9147205>
- [2] Pshenichnyi, B.N. (1976) Simple Pursuit by Several Objects. *Cybernetics*, **12**, 484-485. <https://doi.org/10.1007/BF01070036>
- [3] Jin, S. and Qu, Z. (2010) Pursuit-Evasion Games with Multi-Pursuer vs. One Fast Evader. 2010 *8th World Congress on Intelligent Control and Automation IEEE*, Jinan, 6-9 July 2010, 3184-3189.
- [4] Chen, J., Zha, W., Peng, Z. and Gu, D. (2016) Multi-Player Pursuit-Evasion Games with One Superior Evader. *Automatica*, **71**, 24-32. <https://doi.org/10.1016/j.automatica.2016.04.012>
- [5] Von Moll, A., Casbeer, D.W., Garcia, E. and Milutinović, D. (2018) Pursuit-Evasion of an Evader by Multiple Pursuers. 2018 *International Conference on Unmanned Aircraft Systems (ICUAS) IEEE*, Dallas, 12-15 June 2018, 133-142. <https://doi.org/10.1109/ICUAS.2018.8453470>
- [6] Von Moll, A., Pachter, M., Garcia, E., Casbeer, D.W. and Milutinović, D. (2020) Robust Policies for a Multiple-Pursuer Single-Evader Differential Game. *Dynamic Games and Applications*, **10**, 202-221. <https://doi.org/10.1007/s13235-019-00313-3>
- [7] Bakolas, E. and Tsiotras, P. (2010) The Zermelo-Voronoi Diagram: A Dynamic Partition Problem. *Automatica*, **46**, 2059-2067. <https://doi.org/10.1016/j.automatica.2010.09.003>
- [8] Bakolas, E. and Tsiotras, P. (2012) Relay Pursuit of a Maneuvering Target Using Dynamic Voronoi Diagrams. *Automatica*, **48**, 2213-2220. <https://doi.org/10.1016/j.automatica.2012.06.003>
- [9] Sun, W. and Tsiotras, P. (2017) Sequential Pursuit of Multiple Targets under External Disturbances via Zermelo-Voronoi Diagrams. *Automatica*, **81**, 253-260. <https://doi.org/10.1016/j.automatica.2017.03.015>
- [10] Sun, W., Tsiotras, P. and Yezzi, A.J. (2019) Multiplayer Pursuit-Evasion Games in Three-Dimensional Flow Fields. *Dynamic Games and Applications*, **9**, 1188-1207. <https://doi.org/10.1007/s13235-019-00304-4>
- [11] Lopez, V.G., Lewis, F.L., Wan, Y., Sanchez, E.N. and Fan, L. (2019) Solutions for Multiagent Pursuit-Evasion Games on Communication Graphs: Finite-Time Capture and Asymptotic Behaviors. *IEEE Transactions on Automatic Control*, **65**, 1911-1923. <https://doi.org/10.1109/TAC.2019.2926554>

-
- [12] Garcia, E., Casbeer, D.W., Von Moll, A. and Pachter, M. (2020) Multiple Pursuer Multiple Evader Differential Games. *IEEE Transactions on Automatic Control*, **66**, 2345-2350. <https://doi.org/10.1109/TAC.2020.3003840>
- [13] Makkapati, V.R. and Tsiotras, P. (2019) Optimal Evading Strategies and Task Allocation in Multi-Player Pursuit-Evasion Problems. *Dynamic Games and Applications*, **9**, 1168-1187. <https://doi.org/10.1007/s13235-019-00319-x>
- [14] Isaacs, R. (1999) *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*. Courier Corporation, North Chelmsford.
- [15] Bondy, J.A. and Murty, U.S.R. (1976) *Graph Theory with Applications*. Macmillan, New York. <https://doi.org/10.1007/978-1-349-03521-2>