

蜂群作战的启发式目标分配方法*

魏喜庆¹, 杜厦²

(1. 上海电机学院, 上海 201306; 2. 西北工业大学 宁波研究院, 浙江 宁波 315103)

摘要: 由于低成本和智能化技术的发展, 蜂群作战正在成为现在的研究热点。随着蜂群规模的增加, 高效的目标分配算法成为其中的一个重要研究方向。为了提高目标分配的计算效率, 利用目标集群的剩余价值作为优化目标, 使用贪婪算法分步选取局部最优解, 最终获取目标分配的近似全局最优解。通过对典型目标分配问题的仿真, 验证了提出算法的有效性。

关键词: 蜂群; 无人系统; 目标分配; 启发式; 贪婪算法

DOI:10.3969/j.issn.1009-086x.2024.06.007

中图分类号: TP181; V279.2; TJ0 文献标志码: A 文章编号: 1009-086X(2024)-06-0047-05

引用格式: 魏喜庆, 杜厦. 蜂群作战的启发式目标分配方法[J]. 现代防御技术, 2024, 52(6):47-51.

Reference format: WEI Xiqing, DU Sha. Heuristic Target Assignment Method for Swarm Operations[J]. Modern Defence Technology, 2024, 52(6):47-51.

Heuristic Target Assignment Method for Swarm Operations

WEI Xiqing¹, DU Sha²

(1. Shanghai Dianji University, Shanghai 201306, China;

2. Ningbo Institute of Northwestern Polytechnical University, Ningbo 315103, China)

Abstract: With the development of low-cost and intelligent technologies, swarm operations are becoming a hot research topic. As the size of the swarm increases, efficient target assignment algorithms become an important research direction. In order to improve the computational efficiency of target assignment, this paper takes the surplus value of target clusters as the optimization goal, uses the greedy algorithm to select local optimal solutions step by step, and ultimately obtains an approximate global optimal solution for target assignment. Simulation of typical target assignment problems verifies the effectiveness of the proposed algorithm.

Keywords: swarm; unmanned systems; target assignment; heuristic; greedy algorithm

0 引言

随着机器人技术、人工智能和通信技术的发展, 多智能体的自主编队技术已经成为现实。卫星编队、无人机编队、临近空间飞行器编队、无人舰艇、无人车辆以及混合编队的研究和应用不断涌

现^[1-5]。更大规模的蜂群编队已经出现, 并在智能化无人系统中取得了发展和应用。2016年, 美国国防部战略能力办公室首次进行空射无人机蜂群演示验证, 使用3架F/A-18战斗机一次性释放出103架“山鹑”微型无人机, 无人机蜂群在空中自主编队飞行, 展示了良好的自主决策、轨迹修正和自适应编

* 收稿日期: 2023-08-21; 修回日期: 2023-11-15

第一作者简介: 魏喜庆(1982-), 男, 黑龙江鹤岗人。讲师, 博士, 研究方向为制导控制与人工智能。

队的能力^[6]。无人系统的编队作战已经具备了基本的作战能力,而蜂群数量的增加彻底改变了战场作战样式,预计可自主执行攻防任务的高达数万架规模的“超级蜂群”将在不久的将来出现。

面对高复杂性和强对抗性的战场环境,需要充分利用蜂群自主态势感知、信息融合和自主决策能力,其中的目标分配是自主决策能力的一项核心能力。事实上,武器目标分配(weapon target assignment, WTA)问题,较早地被用于解决导弹防御问题,来最大化地实现对来袭导弹的拦截。

李梦杰等^[7]对武器目标分配问题的建模方法、求解算法和实验验证等方面近年来取得的成果做了全面的综述。MANNE^[8]第1个提出了武器目标分配问题的数学模型,在武器对任何一个目标都具有相同杀伤概率的假设条件下,给出了该问题的最优解。随着武器目标分配问题的技术发展,研究者考虑的问题规模和模型假设简化方面都有所扩展。文献[9]开发了一种非线性分枝定界算法优化武器分配问题,减少目标函数的线性近似所导致求解误差,并使用分枝定界算法来求解非线性问题中较大规模的目标分配问题。文献[10]通过分段线性凸函数逼近WTA问题的非线性项,引入额外变量和约束作为代价,提出了一种称为分支和调整的精确求解方法。该算法建立在现有的分支切割或分支定界算法之上,并且可以使用现有的混合整数线性规划工具来实现。

武器目标分配问题已经被证明是一种N-P完全问题,随着“蜂群”规模的增加,导致目标分配计算量随之指数增加。针对大规模的武器目标分配问题,许多启发式算法被用于快速寻找次优解。文献[11-12]提出了一种超大规模邻域(very large-scale neighborhood, VLSN)搜索算法,使用隐式枚举算法来改进邻域搜索,可以较好地解决大规模实例问题。其他包括神经网络^[13-14]、遗传算法^[15]、蚁群算法^[16]和粒子群算法^[17]在内的多种智能算法,也被应用于武器目标分配问题的启发式求解。文献[9]还创建和测试了一种新的能够实时进行高质量求解的启发式算法。这种启发式方法,在0.01 s内找到大型实例(近百个目标)的解,但通常不能保证最优解。验证发现启发式算法对于较小的实例(10个目标)的解,与原问题的全局最优解相比具有较小

的平均误差。针对空间飞行器集群对抗中的多目标分配问题,对可达性、燃料消耗和拦截概率多级指标综合优化,文献[18]提出了采用目标分配指导表构建集群对抗目标分配的决策树,进行遍历搜索得出综合收益最优的目标分配方案。但对于规模更大的实例,其算法的实时性仍需要提升。

本文综合考虑了对目标打击的有效性和提高效率,以武器打击后的目标集群剩余价值作为优化指标,通过贪婪算法构建一种启发式多约束目标分配算法。有效实现对近似最优解搜索的同时,保证在大规模实例应用时良好的实时性。

1 目标分配模型

1.1 目标约束条件

武器目标分配问题可以分为静态目标分配问题和动态目标分配问题。静态目标分配问题指的是单回合作战的情况下,一次性地实现武器目标分配,而动态目标分配问题涉及多回合作战情况下目标分配。因为大多数的实际问题都属于静态目标分配问题,多数动态目标分配可以分解为多个静态目标分配问题,所以本文重点考虑静态武器目标分配问题,具体将其描述为找到一组不同类型武器(拥有不同战斗部的同一种武器可以认为是不同类型的武器)对一组目标的最佳分配,最大限度地提高对目标总的预期杀伤效应。对于 m 种类型的武器, n 个目标分配问题需要考虑下面约束:

(1) 分配数量约束:定义 x_{ij} 为分配给第 j 个目标的 i 类型的武器数量,每种类型武器可分配的武器数量要小于可用武器数量:

$$\sum_{j \in J} x_{ij} \leq w_i \quad i = 1, 2, \dots, m. \quad (1)$$

(2) 杀伤概率约束: p_{ij} 表示 i 种类型的武器对第 j 个目标的杀伤概率,杀伤概率是先验知识或根据战场情况实时计算值,其最大为1:

$$0 \leq p_{ij} \leq 1. \quad (2)$$

1.2 目标优化模型

武器类型集合为 $I = \{1, 2, \dots, m\}$,对应的武器数量集合为 $W = \{w_1, w_2, \dots, w_m\}$,给定目标集合为 $J = \{1, 2, \dots, n\}$,目标价值的集合为 $\{V_1, V_2, \dots, V_n\}$,目标价值可以是目标的实际价值或者是目标威胁等级归一化后的值。每种类型武器对不同目标具有不

同的杀伤概率,定义杀伤概率矩阵为

$$P = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & & \vdots \\ p_{m1} & \cdots & p_{mn} \end{bmatrix}. \quad (3)$$

式中: P 为 $m \times n$ 维矩阵, $p_{ij} \in P, i \in I$ 表示第 i 种武器类型, $j \in J$ 表示第 j 个目标。定义受到 i 种类型武器攻击后目标 j 的生存概率为 $q_{ij} = 1 - p_{ij}$,显然受到 x_{ij} 次攻击后,目标 j 的生存概率为 $q_{ij}^{x_{ij}}$,目标 j 的剩余价值为 $V_j q_{ij}^{x_{ij}}$ 。将目标集群的剩余价值 $V = \sum_{j \in J} V_j \prod_{i \in I} q_{ij}^{x_{ij}}$ 作为优化的指标函数,武器目标分配问题转化为下面的非线性整数规划问题:

$$\begin{aligned} & \min \sum_{j \in J} V_j \prod_{i \in I} q_{ij}^{x_{ij}}, \\ & \text{s.t.} \sum_{j \in J} x_{ij} \leq w_i, \quad \forall i \in I, \\ & x_{ij} \in \mathbf{N}, \quad \forall i \in I, j \in J, \end{aligned}$$

式中: \mathbf{N} 为自然数集。通过对上面问题的优化求解,可以得到武器目标分配问题的最优解,即每种类型的武器分配给不同目标的武器数量 x_{ij} ,构成解矩阵

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix}. \quad (4)$$

需要注意的是,武器类型集合为 I 中 $m = 1$ 时,意味着只有一种武器类型,其数目为 w_1 ,所有的武器具有相同的杀伤概率;当 $w_1 = w_2 = \cdots = w_m = 1$ 时,意味不同类型编队每种武器只有一个,对相同的目标也具有不同的杀伤概率。

2 目标分配算法

目标分配算法采用基于贪婪选择的启发式算法,在每一轮首先计算每一种类型武器对所有目标造成的杀伤损失,然后对杀伤损失进行排序,选择造成损失最大的武器进行目标分配,依次从武器中选择对目标集群造成最大杀伤的武器。在初始时刻,未对武器分配目标,目标剩余总价值为 $Val = \sum_{j \in J} V_j$,由于未进行目标分配,因此分配给目标的武器数量 $x_{ij} = 0$ 。

在每一轮进行目标分配之前,根据之前的分配结果可以计算目标的剩余总价值为 $Val = \sum_{j \in J} V_j \prod_{i \in I} q_{ij}^{x_{ij}}$ 。当给第 j 个目标分配了一个第 i 种

类型的武器进行打击后,计算新的目标集群剩余总价值可得 $Val = \sum_{j \in J} V_j q_{ij} \prod_{i \in I} q_{ij}^{x_{ij}}$ 。因此,减少的目标剩余价值就是对目标造成的损失(即为武器收益):

$$\Delta Val = V_j (1 - q_{ij}) \prod_{i \in I} q_{ij}^{x_{ij}} = V_j p_{ij} \prod_{i \in I} q_{ij}^{x_{ij}}. \quad (5)$$

遍历所有的武器类型和目标对,选取造成目标损伤最大 ΔVal 对应的索引,第 j 个目标分配的第 i 种类型武器的数目 $x_{ij}+1$,对应的第 i 种类型的可用武器数目 w_i-1 。如果 w_i 减少到0,则意味着第 i 种类型的可用武器数量已经耗尽,后续无法选择第 i 种类型的武器。为了计算简便,算法中令概率矩阵的第 i 行 $p_{i(j)}$ 置0,则在遍历中对目标的剩余价值没有影响。启发式目标分配算法流程如下:

开始:

计算初始目标价值 $Val = \sum_{j \in J} V_j, x_{ij} = 0$

Loop while $\sum_{i \in I} \sum_{j \in J} x_{ij} < \sum_{i \in I} w_i$

& $\sum_{j \in J} V_j \prod_{i \in I} q_{ij}^{x_{ij}} > V_{\text{threshold}}$:

Loop for $i \in I, j \in J$:

计算武器 i 对目标 j 打击造成的损失: $\Delta Val =$

$$V_j p_{ij} \prod_{i \in I} q_{ij}^{x_{ij}}$$

End Loop for

$i, j := \operatorname{argmax}_{i \in I, j \in J} \Delta Val$

$x_{ij} := x_{ij} + 1$

$w_i := w_i - 1$

If $w_i = 0$:

$$P_{i(j)} = 0$$

End if

End Loop while

结束

算法的终止条件,包括所有武器耗尽 $\sum_{i \in I} \sum_{j \in J} x_{ij} < \sum_{i \in I} w_i$,或者预计目标集群的剩余价值低于设置的阈值 $V_{\text{threshold}}$, $\sum_{j \in J} V_j \prod_{i \in I} q_{ij}^{x_{ij}} \leq V_{\text{threshold}}$ 。最终,得到

每种类型的武器分配给不同目标的武器数量 x_{ij} 。在计算结果中如果 $x_{ij}=0$,含义是对第 j 个目标未分配第 i 种类型武器, $x_{ij} = 1$ 意味着单发打击, $x_{ij} \geq 2$ 意味着多发齐射。

3 计算复杂度

对于启发式分配算法,在每一次迭代计算时,需要遍历计算每种类型武器对目标的收益值。不考虑武器耗尽的情况,在最严苛的情况要计算 m 种类型武器对 n 个目标的收益值,执行 $O(mn)$ 次计算。而对比经典的算法VLSN搜索启发式算法,当对目标集中的 k 个目标进行交换寻优,过程中采用动态规划,每次迭代需要进行 $O(m^2 \times 2^k)$ 计算^[19]。经过对比,在 $m \times 2^k > n$ 的情况下本算法的计算量将小于VLSN算法。

4 仿真实验

下面以对飞行器集群目标打击分配问题为例,分配方案的优化指标为目标剩余价值最低,采用文中提出的目标分配算法进行验证。

例1:假定初始时刻拦截飞行器为3种(其中第1和第3种数目为2),目标飞行器的数目为3。表1中给出仿真算例1的参数,其中包含武器类型、数量、目标价值和武器杀伤概率。

表1 目标和武器信息

Table 1 Target and weapon information

武器类型(数量)	$V_1=70$	$V_2=60$	$V_3=25$
1($w_1=2$)	0.8	0.9	0.5
2($w_2=1$)	0.6	0.2	0.4
3($w_3=2$)	0.4	0.4	0.8

例1的目标分配过程,在表2中说明了启发式目标分配算法的武器收益计算和寻优的步骤。

表2 前3轮计算的拦截弹收益值

Table 2 Missile gains of the first 3 iterations

武器类型 <i>i</i> (数量)	目标 <i>j</i>			
	1	2	3	
ΔVal (第1轮)	1	56	54	12.5
	2	42	12	10
	3	28	24	20
ΔVal (第2轮)	1	11.2	54	12.5
	2	8.4	12	10
	3	5.6	24	20
ΔVal (第3轮)	1	0	0	0
	2	8.4	1.2	10
	3	5.6	2.4	20

在第1轮分配时,计算的拦截弹收益值 ΔVal 中,第1种武器对第1个目标的收益值最大为56,

因此将第1种武器的第1个分配给第1个目标;在第2轮分配时,计算的拦截弹收益值 ΔVal 中,第1种武器对第1个目标的打击收益为双发相比单发的收益增量为11.2,而第1种武器对第2个目标的收益值最大为54,因此将第1种武器的第2个分配给第2个目标;在第3轮分配时,计算的拦截弹收益值 ΔVal 中,由于第1种武器已经耗尽所以对所有目标的打击收益计算为0,第3种武器对第3个目标的收益值最大为20,因此将第3种武器的第1个分配给第3个目标。后面依次选取目标,直到满足迭代终止条件,返回选取的分配值,如表3所示。

表3 武器目标分配结果

Table 3 Results of target assignment

目标	1	2	3
武器	1, 3	2	4, 5

例2:下面考虑蜂群作战情况,对大规模实例情况进行仿真。定义 V_{\min} 为遍历搜索得到的全局最优解, V_{VLSN} 为VLSN算法的求解, V_{HRST} 为本文启发式算法(heuristic method, HRST)的求解,进而算法精度分别为 $(V_{\text{VLSN}} - V_{\min})/V_{\text{VLSN}}$ 和 $(V_{\text{HRST}} - V_{\min})/V_{\text{HRST}}$ 。将本文提出的算法与已有的VLSN优化算法^[12]进行求解精度和时间的对比,仿真结果如表4所示。当武器和目标数目都是80的情况,VLSN算法的精度是0.15%,而本文算法精度是0.17%,两者精度相当。寻优过程的耗时,VLSN是0.16s,而本文的算法是0.12s,计算时间略优于VLSN。当武器和目标数目都是200的情况,VLSN算法的精度是0.36%,而本文算法精度是0.45%,两者精度相当。寻优过程的耗时,VLSN是0.83s,而本文的算法是0.45s,计算时间明显优于VLSN。

表4 蜂群目标武器分配精度和时间

Table 4 Precision and time of swarm target assignment

武器	目标	精度/%		时间/s	
		VLSN	HRST	VLSN	HRST
80	80	0.15	0.17	0.16	0.12
200	200	0.36	0.45	0.83	0.45

5 结束语

本文针对蜂群飞行器集群对抗目标分配问题,利用目标集群的剩余价值作为优化目标,使用贪婪算法分步选取局部最优解,最终获取目标分配的近似全局

最优解。通过仿真算例详细说明了启发式目标分配算法武器收益迭代计算和寻优的步骤,验证了蜂群作战情况下所提出方法解决大规模实例的有效性。

参考文献:

- [1] SHOUMAN M, BANDO M, HOKAMOTO S. Output Regulation Control for Satellite Formation Flying Using Differential Drag [J]. *Journal of Guidance, Control, and Dynamics*, 2019, 42(10): 2220–2232.
- [2] WANG Ximan, BALDI S, FENG Xuewei, et al. A Fixed-Wing UAV Formation Algorithm Based on Vector Field Guidance [J]. *IEEE Transactions on Automation Science and Engineering*, 2023, 20(1): 179–192.
- [3] ZHANG Yao, WANG Xiao, TANG Shengjing. A Globally Fixed-Time Solution of Distributed Formation Control for Multiple Hypersonic Gliding Vehicles [J]. *Aerospace Science and Technology*, 2020, 98: 105643.
- [4] HADI B, KHOSRAVI A, SARHADI P. A Review of the Path Planning and Formation Control for Multiple Autonomous Underwater Vehicles [J]. *Journal of Intelligent & Robotic Systems*, 2021, 101(4): 67.
- [5] ISSA B A, RASHID A T. A Survey of Multi-Mobile Robot Formation Control [J]. *International Journal of Computer Applications*, 2019, 181(48): 12–16.
- [6] XU Ximeng, XU Jihui, TIAN Wenjie, et al. Review on Research and Development of Unmanned Aerial Vehicle Swarm Cooperative Operation [C]//*Proceedings of the 2022 5th International Conference on Robot Systems and Applications*. New York: Association for Computing Machinery, 2022: 54–58.
- [7] 李梦杰, 常雪凝, 石建迈, 等. 武器目标分配问题研究进展:模型、算法与应用 [J]. *系统工程与电子技术*, 2023, 45(4): 1049–1071.
- LI Mengjie, CHANG Xuening, SHI Jianmai, et al. Developments of Weapon Target Assignment: Models, Algorithms, and Applications [J]. *Systems Engineering and Electronics*, 2023, 45(4): 1049–1071.
- [8] MANNE A S. A Target-Assignment Problem [J]. *Operations Research*, 1958, 6(3): 346–351.
- [9] KLINE A G, AHNER D K, LUNDAY B J. Real-Time Heuristic Algorithms for the Static Weapon Target Assignment Problem [J]. *Journal of Heuristics*, 2019, 25(3): 377–397.
- [10] ANDERSEN A C, PAVLIKOV K, TOFFOLO T A M. Weapon-Target Assignment Problem: Exact and Approximate Solution Algorithms [J]. *Annals of Operations Research*, 2022, 312(2): 581–606.
- [11] AHUJA R K, ORLIN J B, SHARMA D. A Composite Very Large-Scale Neighborhood Structure for the Capacitated Minimum Spanning Tree Problem [J]. *Operations Research Letters*, 2003, 31(3): 185–194.
- [12] AHUJA R K, KUMAR A, JHA K C, et al. Exact and Heuristic Algorithms for the Weapon-Target Assignment Problem [J]. *Operations Research*, 2007, 55(6): 1136–1146.
- [13] WACHOLDER E. A Neural Network-Based Optimization Algorithm for the Static Weapon-Target Assignment Problem [J]. *ORSA Journal on Computing*, 1989, 1(4): 232–246.
- [14] 丁振林, 刘冠龙, 谢艺, 等. 基于强化学习与神经网络的动态目标分配算法 [J]. *电子设计工程*, 2020, 28(13): 54–60.
- DING Zhenlin, LIU Guanlong, XIE Yi, et al. Dynamic Targets Assignment with Reinforcement Learning and Neural Network [J]. *Electronic Design Engineering*, 2020, 28(13): 54–60.
- [15] SAHIN M A, LEBLEBICIOGLU K. A Genetic Algorithm for Weapon Target Assignment Problem [C]//*Proceedings of the Proceedings of the 2009 Summer Computer Simulation Conference*. Vista: Society for Modeling & Simulation International, 2009: 61–65.
- [16] LEE Z J, LEE C Y, SU Shunfeng. An Immunity-Based Ant Colony Optimization Algorithm for Solving Weapon-Target Assignment Problem [J]. *Applied Soft Computing*, 2002, 2(1): 39–47.
- [17] ZENG Xiangping, ZHU Yunlong, NAN Lin, et al. Solving Weapon-Target Assignment Problem Using Discrete Particle Swarm Optimization [C]//*Proceedings of the 2006 6th World Congress on Intelligent Control and Automation*. Piscataway: IEEE, 2006: 3562–3565.
- [18] 邹子缘, 陈琪锋. 基于决策树搜索的空间飞行器集群对抗目标分配方法 [J]. *航空学报*, 2022, 43(增1): 78–88.
- ZOU Ziyuan, CHEN Qifeng. Decision Tree-Based Target Assignment for Confrontation of Multiple Space Vehicles [J]. *Acta Aeronautica et Astronautica Sinica*, 2022, 43(S1): 78–88.
- [19] KLINE A, AHNER D, HILL R. The Weapon-Target Assignment Problem [J]. *Computers & Operations Research*, 2019, 105: 226–236.