

# 基于可适应匈牙利算法的武器-目标分配问题

张进<sup>1</sup>, 郭浩<sup>1</sup>, 陈统<sup>2</sup>

(1. 江苏自动化研究所, 江苏 连云港 222006; 2. 91431 部队, 海南 海口 570100)

**摘要:** 当前各类智能优化算法求解武器-目标分配问题时, 存在耗时长、优化结果不唯一等缺陷, 而匈牙利算法具有耗时短、求解结果稳定的优势, 但其适应性较差, 目前尚未见二者的对比分析文献。针对此现象, 对比分析传统匈牙利算法与智能优化算法的耗时性与稳定性, 展现了匈牙利算法的优势; 提出统一效率矩阵, 创建可适用于所有类型目标分配问题的可适应匈牙利算法; 通过实例应用验证了可适应匈牙利算法的正确性。

**关键词:** 武器-目标分配; 匈牙利算法; 整数规划; 智能优化算法

中图分类号: E911 文献标志码: A 文章编号: 1000-1093(2021)06-1339-06

DOI: 10.3969/j.issn.1000-1093.2021.06.025

## Weapon-target Assignment Based on Adaptable Hungarian Algorithm

ZHANG Jin<sup>1</sup>, GUO Hao<sup>1</sup>, CHEN Tong<sup>2</sup>

(1. Jiangsu Institute of Automation, Lianyungang 222006, Jiangsu, China;

2. Unit 91431 of PLA, Haikou 570100, Hainan, China)

**Abstract:** When various intelligent optimization algorithms are used to solve the weapon-target assignment problem, they have the disadvantages of long time-consuming and non-unique optimization results. Hungary algorithm has the advantages of short time-consuming and stable optimization results, but its adaptability is poor. Currently, the comparison and analysis of intelligent optimization algorithms and Hungarian algorithm has not been reported. For this phenomenon, the time-consuming and stability of traditional Hungarian algorithm and intelligent optimization algorithms are compared, which shows the advantages of Hungarian algorithm. An adaptable Hungarian algorithm that can be applied to all types of weapon-target assignment problems is established by proposing a unified efficiency matrix. And then some examples are used to verify the correctness of the adaptable Hungarian algorithm.

**Keywords:** weapon-target assignment; Hungarian algorithm; integer programming; intelligent optimization algorithm

## 0 引言

武器-目标分配是指根据作战目的、战场态势和武器性能等因素, 按照一定的最优分配原则将多种武器分配给多个来袭目标, 从而取得最佳打

击效果的方法<sup>[1]</sup>, 其常见的数学模型包括最大毁伤模型以及最大价值模型<sup>[2]</sup>, 如(1)式~(3)式所示:

$$\max f_{MD} = \sum_{j=1}^n r_j \left[ 1 - \prod_{i=1}^m (1 - p_{ij})^{a_{ij}} \right], \quad (1)$$

收稿日期: 2020-07-21

基金项目: 国防预先研究项目(1020403020101)

作者简介: 张进(1991—), 男, 工程师, 硕士。E-mail: jinzhang405@126.com

$$\max f_{MV} = \sum_{j=1}^n \sum_{i=1}^m p_{ij} a_{ij}, \quad (2)$$

$$\sum_{j=1}^n a_{ij} \leq 1, \sum_{i=1}^m a_{ij} \leq m, \quad (3)$$

式中:  $n$  表示目标数量;  $r_j$  表示第  $j$  个目标的威胁程度;  $m$  表示武器数量;  $p_{ij}$  为第  $i$  个火力单元对第  $j$  个目标的命中概率;  $a_{ij} = 0$  或  $1$   $0$  表示不选中,  $1$  表示选中。

武器-目标分配问题作为一种最优化问题,近年来被许多学者采用各种智能算法求解。杨飞等<sup>[3]</sup>采用整数域粒子群优化(PSO)算法研究多平台武器目标分配问题;董朝阳等<sup>[4]</sup>利用改进的遗传算法(GA)求解航空兵编队对地攻击武器目标分配模型;刘家义等<sup>[5]</sup>基于改进加速梯度下降算法研究了目标分配问题等。然而,武器目标分配问题本质上属于非线性 0-1 整数规划问题,是一类特殊的最优化问题,智能算法虽然能够用于求解但是却无法充分发挥其优势,存在求解耗时长、优化结果不唯一等缺陷,实际作战中这是致命的且不被允许的,必须在保证有解的基础上,提高求解的质量和速度<sup>[6]</sup>。

指派问题作为运筹学中的经典问题,与武器-目标分配问题有着众多相似之处,指派问题的定义为:由  $L$  个人完成  $K$  项工作,且每个人完成每项工作的效率不同,确定任务指派方案使得完成任务总的效率最高<sup>[7]</sup>。标准指派问题的数学模型可以表示为(4)式~(5)式:

$$\max f = \sum_{k=1}^K \sum_{l=1}^L C_{lk} b_{lk}, \quad (4)$$

$$\sum_{k=1}^K b_{lk} \leq 1, \sum_{l=1}^L b_{lk} \leq 1, \quad (5)$$

式中:  $C_{lk}$  表示第  $l$  个人做第  $k$  项工作的效率;  $b_{lk} = 0$  或  $1$   $0$  表示不选择,  $1$  表示选择。对比(1)式~(3)式与(4)式~(5)式可知,武器-目标分配问题与指派问题的差别在于约束条件的不同,当约束条件变为某一目标只能被一种武器攻击时,武器-目标分配问题就转换为标准的指派问题。一直以来,匈牙利算法(HA)被公认为是指派问题的标准解法,其针对标准指派模型的特殊形式(方形矩阵),基于匈牙利数学家 Konig 提出的独立零元素定理,只采用矩阵变换等操作就能求出模型最优解,但研发初期,其解算流程基本是基于人工操作,随着计算机语言对矩阵变换等操作的编译,使得 HA 拥有计算速度快、解算结果稳定等优势<sup>[8-9]</sup>。2002 年,

柳毅等<sup>[10]</sup>利用 HA 研究了攻击机与目标机的分配问题,并提出一对多不平衡分配问题的解法,但并未提出多对一不平衡分配问题的解法。2013 年,谷稳<sup>[11]</sup>基于进化 HA 研究了无人机目标分配问题,并考虑了不平衡目标分配情况,但并未建立统一的效率矩阵。2015 年,李元左等<sup>[12]</sup>利用 HA 解算了炮兵火力分配模型,但并未解决一对多或是多对一的不平衡分配问题。以上基于 HA 研究目标分配问题的文献都未曾将 HA 与智能优化算法进行对比分析,无法体现二者的优劣势。

本文在充分对比分析 HA 与常用智能优化算法的基础上,提出适用于所有目标分配问题的可适应 HA,并利用该算法求解了复杂约束条件下的武器-目标分配问题。

## 1 传统 HA 与智能优化算法的比较

### 1.1 传统 HA

传统 HA 是 Kuhn 通过引用匈牙利数学家 Konig 关于矩阵中独立 0 元素定理:在效率矩阵(也称代价矩阵)的任意行或列加上或者减去一个常数不会改变最优分配方案的基础上,提出用于解决指派问题的优化方法<sup>[11-12]</sup>。HA 解决标准指派问题的步骤<sup>[13-14]</sup>如下:

步骤 1 构建  $d \times d$  的效率矩阵。

步骤 2 用效率矩阵的每一行元素减去该行中的最小元素,再从每列元素中减去该列的最小元素。

步骤 3 在矩阵中找到某个 0,如果其行或列中没有加星号的 0,则将该 0 标记上星号(简称星标 0),直至找遍所有的 0。

步骤 4 如果该列中有带星号的 0,则划线该列,如果所有的列都被划线,则最优值已找到,否则转步骤 5。

步骤 5 找到一个未被划线的 0 并标注它(简称标注 0)。如果包含该标注 0 的行中没有星号 0,则转步骤 6;否则,划线此行,并找出包含带星号 0 的列。以这种方式继续,直到没有未被划线的 0,并保存最小的未划线值,转步骤 7。

步骤 6 按照以下过程,构造一系列交替的标注 0 和星号 0,首先令  $Z_0$  代表步骤 5 中未被划线的标注 0,再令  $Z_0$  表示  $Z_0$  列中星号 0(如果有),最后令  $Z_2$  表示  $Z_1$  行中的标注 0(始终为 1 个),继续直到标记 0( $Z_0$ )的列中没有星号 0( $Z_1$ )。对标注 0 标记星标,变为星号 0。返回步骤 4。

步骤 7 将最小未划线值加到每个划线行的

每个元素,并从每个未划线列的每个元素中减去它。返回步骤5。

## 1.2 对比与分析

(6)式表示的是 $m$ 个武器与 $n$ 个目标的效率矩阵,一般而言,效率矩阵规模越大,则武器-目标分配问题就越复杂。为比较分析不同规模分配问题下HA与智能优化算法的耗时性与稳定性,本文分别选取 $10 \times 10$ 、 $20 \times 20$ 、 $50 \times 50$ 、 $100 \times 100$ 规模的武器-目标分配问题,其中 $p_{ij}$ 的数值由计算机随机产生,以求取总效率最低为优化目标函数。目前,GA、粒子群优化(PSO)算法及基于以上两种算法改进的优化算法被频繁用于求解武器-目标分配问题,因此,本文选取改进的GA<sup>[15]</sup>及PSO算法<sup>[16]</sup>与HA进行对比,运行环境为Windows 7系统、i5-2450M处理器、4GB运行内存的笔记本。

$$\begin{matrix} & A_1 & A_2 & \cdots & A_n \\ \begin{matrix} W_1 \\ W_2 \\ \vdots \\ W_m \end{matrix} & \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mn} \end{bmatrix} \end{matrix}, \quad (6)$$

式中: $W_m$ 表示第 $m$ 个武器; $A_n$ 表示第 $n$ 个目标; $p_{ij}$ 可以为 $W_i$ 武器对 $A_j$ 目标的毁伤概率,也可以取毁伤概率与该目标威胁程度的乘积,视目标函数而定。

### 1.2.1 耗时性对比

对4种不同规模的分配问题分别采用3种算法求解,每种规模下各算法都被运行10次,取其运行时间平均值作为最终耗时,由于PSO算法和GA运行时间与迭代次数相关,本文不采用以迭代次数为终止迭代条件,采用最优值不再更新作为终止迭代条件,运行结果如图1所示。

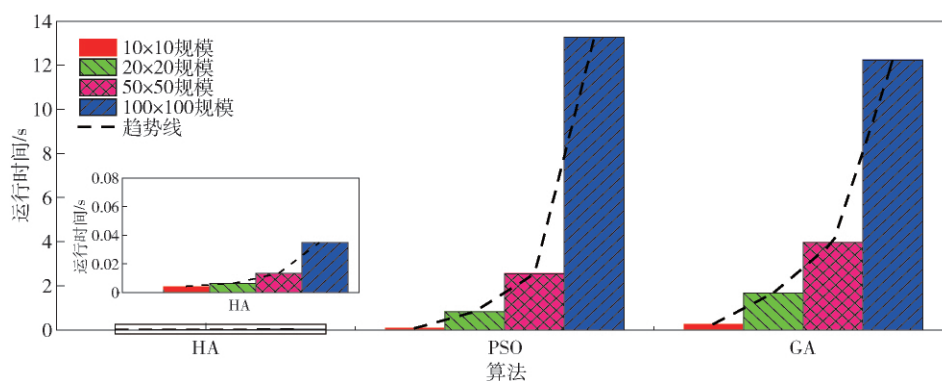


图1 HA、PSO算法、GA运行时间对比

Fig. 1 Time consuming comparison of Hungarian algorithm, particle swarm optimization algorithm and genetic algorithm

从图1中可以看出:1)在不同规模分配问题中,HA始终是3种算法中耗时最短的算法,以 $20 \times 20$ 规模的分配问题为例,HA平均耗时为0.006 s, PSO算法平均耗时为0.830 s, GA耗时1.674 s,相比于PSO算法和GA,HA的平均耗时要小两个数量级,在实际作战使用时,可有效缩短最优方案生成时间;2)随着问题规模的上升,PSO算法和GA的平均耗时都在呈指数式增长,例如,相比于 $50 \times 50$ 规模的分配问题,求解 $100 \times 100$ 分配问题时,PSO算法平均耗时增加了5.2倍,GA平均耗时增加了3.1倍,而HA只增加了2.5倍,即使耗时增加了2.5倍,HA求解 $100 \times 100$ 规模分配问题也只耗时0.348 s,仍然处于可接受范围内。

### 1.2.2 稳定性对比

将不同规模武器-目标分配问题下3种算法运行10次求解的最优值进行对比,为直观分析结果,

本文绘制了箱型图,如图2所示。图2(a)表示 $10 \times 10$ 规模分配问题的最优值,图2(b)表示 $20 \times 20$ 规模分配问题的最优值,图2(c)表示 $50 \times 50$ 规模分配问题的最优值,图2(d)表示 $100 \times 100$ 规模分配问题的最优值。

从图2(a)~图2(d)中可以看出:1)在不同规模分配问题中,HA始终是3种算法中求解最稳定的算法,其最优值始终收敛于某一个值,这是因为HA是基于矩阵的操作,对某一确定的效率矩阵,其求解结果也是确定的,而对于PSO算法及GA,均为群体智能优化算法,存在一定的随机性,因此求解结果并不都收敛于某一个值<sup>[6]</sup>;2)从图2(a)中可以看出,对于小规模分配问题,PSO算法及GA求解结果大概率都能获取与HA同样的结果,但对于大规模分配问题(见图2(c)及图2(d)),PSO算法及GA求解的最优值都远小于HA。

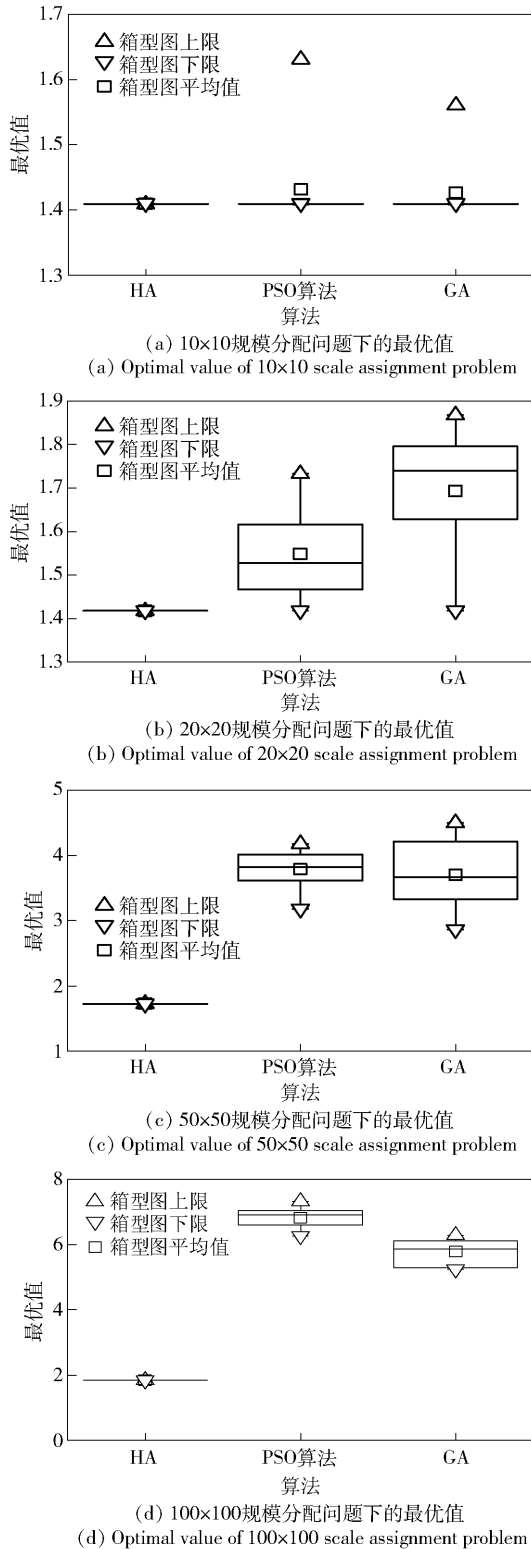


图 2 不同规模分配问题下的最优值比较

Fig. 2 Comparison of optimal values for different scale weapon-target assignment problems

综上所述,HA 的耗时性及稳定性均优于 PSO 算法及 GA. 但传统的 HA 不能适用于所有类型的分配问题,因此有必要对传统 HA 进行适应性改进.

## 2 可适应 HA

传统 HA 只能用于求解一对一的分配问题,后续学者通过研究 HA 的求解过程,提出利用“加边补零法”使效率矩阵满足 HA 的求解条件(效率矩阵必须为方形矩阵),从而用于求解武器数多于目标数或武器数小于目标数的不平衡目标分配问题,还提出利用“虚拟目标数量法”解决 1 个武器攻击多个目标的分配问题<sup>[9,11]</sup>. 但截止目前,很少有学者研究多个武器同时攻击 1 个目标的分配问题,也未建立适用于所有类型分配问题的统一效率矩阵.

本文在前人研究的基础上,提出  $m$  个武器与  $n$  个目标,其中每个武器至多可以攻击  $i$  个目标,每个目标至多可以被  $j$  个武器攻击的统一效率矩阵,矩阵形式如图 3 所示.

		$j=1$				$j=2$				$j=x$			
		$A_1$	$A_2$	$\dots$	$A_n$	$A_1$	$A_2$	$\dots$	$A_n$	$A_1$	$A_2$	$\dots$	$A_n$
$i=1$	$W_1$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$
	$W_2$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	$W_m$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$
$i=2$	$W_1$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$
	$W_2$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	$W_m$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$
$\dots$		$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$i=y$	$W_1$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$	$p_{11}$	$p_{12}$	$\dots$	$p_{1n}$
	$W_2$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$	$p_{21}$	$p_{22}$	$\dots$	$p_{2n}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	$W_m$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$	$p_{m1}$	$p_{m2}$	$\dots$	$p_{mn}$

图 3 可适应 HA 的统一效率矩阵

Fig. 3 Uniform efficiency matrix of adaptable Hungarian algorithm

图 3 中黑色实线矩形框中的矩阵即为统一效率矩阵,当约束条件为每个武器至多可以攻击 2 个目标(即  $y=2$ ) ,则  $i=1, i=2$  矩阵块(见图 3 中的蓝色虚线矩形框)纳入统一效率矩阵,当约束条件为每个目标至多可以被 2 个武器攻击(即  $x=2$ ) 时,则  $j=1, j=2$  矩阵块(见图 3 中红色虚线矩形框)纳入统一效率矩阵.最后采用“加边补 0 法”使效率矩阵变为方形矩阵.虽然处理复杂约束条件会导致统一效率矩阵维度变大,但从 1.2.1 节中的分析可知,HA 处理大规模分配问题时耗时依然很小.

## 3 实例应用

为验证可适应 HA 的正确性以及处理复杂约束条件的可行性,本文选取文献[4]中的航空兵编队

对地攻击仿真实例数据(见表1),实例中共设置目标12批, $W_1$ 武器4个、 $W_2$ 武器4个、 $W_3$ 武器2个,共10个武器。本文实例分以下两种情形:1)每个目标至多可以被1个武器攻击;2)每个目标至多可以被2个武器攻击,分别构建了统一效率矩阵并采用可适应HA进行求解。

表1 文献[4]中的毁伤概率及目标威胁度数据

Tab.1 Damage probability and target threat parameters in Ref. [4]

目标	威胁度	$W_1$	$W_2$	$W_3$
$T_1$	0.23	0.786 2	0.908 5	0.779 1
$T_2$	0.05	0.959 0	0.988 1	0.937 5
$T_3$	0.12	0.878 8	0.937 5	0.863 1
$T_4$	0.04	0.968 9	0.985 0	0.932 4
$T_5$	0.11	0.908 5	0.951 2	0.877 5
$T_6$	0.08	0.937 5	0.979 1	0.910 0
$T_7$	0.01	0.981 3	0.995 4	0.980 4
$T_8$	0.18	0.842 5	0.926 9	0.840 0
$T_9$	0.03	0.968 9	0.990 8	0.951 6
$T_{10}$	0.07	0.951 2	0.983 2	0.915 9
$T_{11}$	0.02	0.974 4	0.991 9	0.963 9
$T_{12}$	0.06	0.955 2	0.985 0	0.927 1

当每个目标至多可以被1个武器攻击时,构建的统一效率矩阵如图4所示。可适应HA求解结果在图4中被红框标记,其中 $W_1$ 武器攻击 $T_4$ 、 $T_6$ 、 $T_{10}$ 、 $T_{12}$ , $W_2$ 武器攻击 $T_1$ 、 $T_3$ 、 $T_5$ 、 $T_8$ , $W_3$ 武器攻击 $T_2$ 、 $T_9$ ,与文献[4]中的求解结果相一致,验证了可适应HA的正确性。

为进一步比较可适应HA、GA及PSO算法的性能,同时采用3种算法求解以上两种实例10次,结果如下:

	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_7$	$A_8$	$A_9$	$A_{10}$	$A_{11}$	$A_{12}$		$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_7$	$A_8$	$A_9$	$A_{10}$	$A_{11}$	$A_{12}$
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573	$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573	$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573	$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573	$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591	$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591	$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591	$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591	$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_3$	0.1792	0.0469	0.1036	0.0373	0.0965	0.0728	0.0098	0.1512	0.0285	0.0641	0.0193	0.0556	$W_3$	0.1792	0.0469	0.1036	0.0373	0.0965	0.0728	0.0098	0.1512	0.0285	0.0641	0.0193	0.0556
$W_3$	0.1792	0.0469	0.1036	0.0373	0.0965	0.0728	0.0098	0.1512	0.0285	0.0641	0.0193	0.0556	$W_3$	0.1792	0.0469	0.1036	0.0373	0.0965	0.0728	0.0098	0.1512	0.0285	0.0641	0.0193	0.0556
补													补												

图5 统一效率矩阵及求解结果(实例2)

Fig.5 Uniform efficiency matrix and solution results (Case 2)

	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_7$	$A_8$	$A_9$	$A_{10}$	$A_{11}$	$A_{12}$
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_1$	0.1808	0.0480	0.1055	0.0388	0.0999	0.0750	0.0098	0.1517	0.0291	0.0666	0.0195	0.0573
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_2$	0.2090	0.0494	0.1125	0.0394	0.1046	0.0783	0.0100	0.1668	0.0297	0.0688	0.0198	0.0591
$W_3$	0.1792	0.0469	0.1036	0.0373	0.0965	0.0728	0.0098	0.1512	0.0285	0.0641	0.0193	0.0556
$W_3$	0.1792	0.0469	0.1036	0.0373	0.0965	0.0728	0.0098	0.1512	0.0285	0.0641	0.0193	0.0556
补												

图4 统一效率矩阵及求解结果(实例1)

Fig.4 Uniform efficiency matrix and solution results (Case 1)

实例1:可适应HA平均求解时间为 $5.343 \times 10^{-3}$  s,最优解求解次数10/10;GA平均求解时间为 $3.031 \times 10^{-1}$  s,最优解求解次数10/10;PSO算法平均求解时间为 $1.673 \times 10^{-1}$  s,最优解求解次数10/10.从以上结果可看出,3种算法都能稳定地求出最优解,但可适应HA的求解时间显著低于另外两种算法。当每个目标至多可以被2个武器攻击时,构建的统一效率矩阵如图5所示,可适应HA求解结果在图5中被红框标记,其中 $T_1$ 目标被2个 $W_2$ 武器同时攻击, $T_3$ 目标被2个 $W_3$ 武器同时攻击, $T_5$ 目标被2个 $W_1$ 武器同时攻击, $T_6$ 目标被2个 $W_1$ 武器同时攻击, $T_8$ 目标被2个 $W_2$ 武器同时攻击,效率矩阵最优值为1.3086。

实例2:可适应HA平均求解时间为 $8.461 \times 10^{-3}$  s,最优解求解次数10/10;GA平均求解时间为 $6.309 \times 10^{-1}$  s,最优解求解次数9/10;PSO算法平均求解时间为 $3.732 \times 10^{-1}$  s,最优解求解次数9/10.从以上结果可看出,可适应HA的求解时间和稳定性都要优于另外两种算法。

以上仿真实例应用结果表明,可适应HA可用于解决复杂约束条件下的武器-目标分配问题。



## 4 结论

本文针对传统 HA 适用性差的缺陷,提出了可求解所有类型目标分配问题的可适应 HA,并通过实例应用得到了以下结论:

- 1) 相比于常见的智能优化算法,可适应 HA 算法具有求解耗时短、结果稳定的优势;
- 2) 可适应 HA 可用于求解复杂约束条件下的武器-目标分配问题。

## 参考文献(References)

- [1] 陈曼,周凤星.改进粒子群算法的舰载武器目标分配[J].火力与指挥控制,2018,43(11):72-76.  
CHEN M, ZHOU F X. Shipborne weapon-target assignment based on improved particle swarm optimization[J]. Fire Control & Command Control, 2018, 43(11): 72-76. (in Chinese)
- [2] 黄大山,徐克虎,王天召.求解 WTA 问题的智能算法评价准则[J].火力与指挥控制,2013,38(8):43-46.  
HUANG D S, XU K H, WANG T Z. An evaluation criteria for intelligent algorithm in solving weapon target assignment problem[J]. Fire Control & Command Control, 2013, 38(8): 43-46. (in Chinese)
- [3] 杨飞,王青,侯砚泽.基于整数域改进粒子群优化算法的多平台武器目标分配[J].兵工学报,2011,32(7):906-912.  
YANG F, WANG Q, HOU Y Z. Weapon-target assignment in multi-launcher system based on improved integer field particle swarm optimization algorithm[J]. Acta Armamentarii, 2011, 32(7): 906-912. (in Chinese)
- [4] 董朝阳,路遥,王青.改进的遗传算法求解火力分配优化问题[J].兵工学报,2016,37(1):97-102.  
DONG C Y, LU Y, WANG Q. Improved genetic algorithm for solving firepower distribution[J]. Acta Armamentarii, 2016, 37(1): 97-102. (in Chinese)
- [5] 刘家义,王刚,张杰,等.基于改进 AGD-分布式多智能体系统的目标优化分配模型[J].系统工程与电子技术,2020,42(4):863-870.  
LIU J Y, WANG G, ZHANG J, et al. Target optimal assignment model based on improved AGD-distributed multi-agent system[J]. Systems Engineering and Electronics, 2020, 42(4): 863-870. (in Chinese)
- [6] 杨进帅,李进,王毅.武器-目标分配问题研究[J].火力与指挥控制,2019,44(5):6-11.  
YANG J S, LI J, WANG Y. Study of weapon target assignment problem[J]. Fire Control & Command Control, 2019, 44(5): 6-11. (in Chinese)
- [7] 李廷鹏,钱彦岭,李岳.基于改进匈牙利算法的多技能人员调度方法[J].国防科技大学学报,2016,38(2):144-149.  
LI T P, QIAN Y L, LI Y. Multi-skilled labor allocating method based on improved Hungary algorithm[J]. Journal of National University of Defense Technology, 2016, 38(2): 144-149. (in Chinese)
- [8] ZHU H B, LIU D N, ZHANG S Q, et al. Solving the many to many assignment problem by improving the Kuhn-Munkres algorithm with backtracking[J]. Theoretical Computer Science, 2016, 618: 30-41.
- [9] CHOPRA S, NOTARSTEFANO G, RICE M, et al. A distributed version of the Hungarian method for multirobot assignment[J]. IEEE Transactions on Robotics, 2017, 33(4): 932-947.
- [10] 柳毅,佟明安.匈牙利算法在多目标分配中的应用[J].火力与指挥控制,2002,27(4):34-37.  
LIU Y, TONG M A. An application of Hungarian algorithm to the multi-target assignment[J]. Fire Control & Command Control, 2002, 27(4): 34-37. (in Chinese)
- [11] 谷稳.基于进化匈牙利算法的目标分配问题研究及应用[D].西安:西安电子科技大学,2013.  
GU W. Study and application of target allocation based on evolution Hungarian algorithm[J]. Xi'an: Xi'an University of Electronic Science and Technology, 2013. (in Chinese)
- [12] 李元左,杨晓段.炮兵火力分配建模及优化方法[J].指挥控制与仿真,2015,37(2):48-52.  
LI Y Z, YANG X D. Optimal assignment model of artillery firepower[J]. Command Control & Simulation, 2015, 37(2): 48-52. (in Chinese)
- [13] KUHN H W. The Hungarian method for the assignment problem[J]. Naval Research Logistics Quarterly, 1955, 2(1): 83-97.
- [14] BOURGEOIS F, LASSALLE J. An extension of the Munkres algorithm for the assignment problem to rectangular matrices[J]. Communications of the ACM, 1971, 14(12): 802-804.
- [15] 陈彬,韩林,余旺盛,等.基于改进遗传算法的装备调配[J].系统工程,2010,28(9):98-102.  
CHEN B, HAN L, YU W S, et al. Equipment distribution based on modified genetic algorithm[J]. Systems Engineering, 2010, 28(9): 98-102. (in Chinese)
- [16] 孙晓雅,林焰.一种新的离散粒子群算法在指派问题中的应用[J].计算机应用研究,2009,26(11):4091-4093,4097.  
SUN X Y, LIN Y. Using new DPSO algorithm to solve assignment problem[J]. Application Research of Computers, 2019, 26(11): 4091-4093, 4097. (in Chinese)