

# 基于 Jackson 网络的地面分队防御作战兵力部署方法

褚凯轩<sup>1</sup>, 常天庆<sup>1†</sup>, 孔德鹏<sup>2</sup>

(1. 陆军装甲兵学院 兵器与控制系, 北京 100072; 2. 92942 部队, 北京 100161)

**摘要:** 针对地面分队防御作战兵力部署信息利用不足、未能宏观统筹部署的问题, 提出一种基于 Jackson 网络的地面分队防御作战兵力部署方法, 实现对地面分队防御作战兵力部署的建模和优化. 首先, 分析地面分队防御敌突击武器的规律和特点, 用 Jackson 排队网络对该过程建模, 并针对第 1 道防线节点顾客到达率大于服务率的问题, 提出 Jackson 排队网络的等效转化方法; 其次, 依据各节点的吞吐量和顾客到达率, 研究节点间的顾客转移量、外来顾客到达率, 并最终求得敌目标对防线的突破率; 最后, 基于博弈思想, 建立双层迭代寻优结构, 计算敌最优进攻策略下我方的最优决策. 仿真实验验证了所提出的改进人工蜂群算法在求解所研究模型收敛性和稳定性方面的优势以及所提出双层迭代寻优结构的合理性, 同时讨论了毁伤概率对兵力部署的影响.

**关键词:** 地面分队防御作战; Jackson 排队网络; 兵力部署; 双层迭代寻优; 人工蜂群算法

中图分类号: E920.8

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.0612

引用格式: 褚凯轩, 常天庆, 孔德鹏. 基于 Jackson 网络的地面分队防御作战兵力部署方法 [J]. 控制与决策, 2023, 38(1): 219-227.

## Forces deployment method of ground unit defensive combat based on Jackson network

CHU Kai-xuan<sup>1</sup>, CHANG Tian-qing<sup>1†</sup>, KONG De-peng<sup>2</sup>

(1. Department of Weaponry and Control, Army Academy of Armored Forces, Beijing 100072, China; 2. Unit 92942, Beijing 100161, China)

**Abstract:** In order to solve the problems of insufficient information utilization and macro overall deployment of ground unit's defensive combat forces, a method of ground unit's defensive combat forces deployment based on the Jackson network is proposed to realize the modeling and optimization of ground unit's defensive combat forces deployment. Firstly, the law and characteristics of the ground detachment's defense against enemy assault weapons are analyzed, and the Jackson queuing network is used to model it. Aiming at the problem that the customer arrival rate is greater than the service rate, the equivalent transformation method of the Jackson queuing network is proposed. Then, according to the throughput and customer arrival rate of each node, the volume of customer transfer and external customer arrival rate between nodes are studied, and the penetration rate of the enemy target to the defense line is obtained. Finally, based on the game theory, a two-level iterative optimization structure is established to calculate the optimal decision of our party under the optimal attack strategy of the enemy. Simulation experiments verify the advantages of the improved artificial bee colony algorithm in solving the convergence and stability of the proposed model and the rationality of the two-layer iterative optimization structure. Meanwhile, the influence of damage probability on the decision of military deployment is discussed.

**Keywords:** ground unit defense operations; Jackson queuing network; forces deployment; two-level iterative optimization; artificial bee colony algorithm

## 0 引言

兵力部署优化问题是作战筹划的重要研究内容之一<sup>[1]</sup>. 地面分队防御作战兵力部署指的是, 指挥员根据战场环境、敌我双方所处态势、作战任务等因素, 合理确定分队的阵型和兵力数量, 以期最大程度

发挥分队的作战效能, 有效抵御敌攻势<sup>[2]</sup>. 在防御兵力规模、作战环境及敌兵力一定的条件下, 部署是影响作战效果的决定性因素<sup>[3]</sup>. 目前, 地面分队兵力部署的研究还十分匮乏, 主要依靠指挥员的经验进行决策, 主观性较大, 因此, 研究一种理性的兵力部署方法,

收稿日期: 2021-04-11; 录用日期: 2021-09-22.

<sup>†</sup>通讯作者. E-mail: changtianqing@263.net.

克服经验决策法的缺点具有重要意义。

提高兵力部署科学性的关键是要建立合理的数学模型,继而求解模型,得到最优方案.文献[4]将排队论应用于多防线异类武器防御系统的突防过程分析和突防概率计算,从而指导兵力的优化部署,该方法只能用于评价给定的兵力部署方案,无法生成新的方案.文献[5]提出一种基于排队论的防空兵力部署优化模型,将武器系统综合防御效能作为优化目标,建立多型防空武器扇形部署优化模型,该模型依据经验预测敌情,采取重点方向重点部署的策略;而实际战场上,防御方难以准确预测敌情,基于经验预测敌情的方法不符合实战情况.文献[6]提出一种基于Memetic算法的兵力机动部署优化方法,通过对兵力效能评估寻优兵力部署方案,但是该方法未能考虑兵力机动的成本.文献[7]基于爱尔兰排队系统相关理论构建了多级防御和协同防御模式下的作战效能预测模型,并分析和比较了两种模型的作战效能.以上主要是基于防空作战效能预测的兵力部署方法,地面分队兵力部署研究较少.文献[2]提出通过预测作战效能来选择坦克分队火力部署方案的方法,建立了一种基于生灭过程作战效能预测模型,但是该方法考虑因素较少,仅提供一种思路而没有形成系统的兵力部署方法,仅能评估已知方案而不能生成方案.

将敌突击目标的来袭视为顾客,将我作战分队视为服务台,将作战分队对目标的阻击视为服务,则防御作战可视为一个典型的排队系统.本文采用Jackson排队网络建立地面分队防御作战模型,为地面分队防御作战的兵力部署问题提供科学决策.

## 1 预备知识

### 1.1 排队论基本概念

排队论是在对各种排队系统概率规律性总结的基础上,解决排队系统最优设计和最优控制问题的方法.一般的排队过程可以描述为:顾客由顾客源出发,到达服务机构,按排队规则排队等待服务,服务机构按服务规则给顾客服务,顾客接受完服务后离开.尽管排队系统多种多样,但是,一个基本的排队系统必由输入过程、排队规则、服务机构及服务规则3个基本部分组成.输入过程描述顾客来源以及顾客到达排队系统的规律,例如随机分布、确定型分布、泊松分布.排队规则描述顾客排队等待的队列和接受服务的次序,一般包括损失制、等待制和混合制3种.服务规则指服务机构的服务时间,例如负指数分布.

对于某服务节点,如果顾客到达服从参数为 $\lambda$ 的泊松分布,服务时间服从参数为 $\mu$ 的负指数分布,服

务台个数为1,顾客源数量无限,系统容量无限,顾客到达间隔时间与服务时间之间相互独立,则称该服务节点为一个标准 $M/M/1$ 模型. $M/M/1$ 排队系统的状态平衡方程为

$$\begin{cases} \lambda P_0 = \mu P_1, \\ \lambda P_{n-1} + \mu P_{n+1} = \lambda P_n + \mu P_n. \end{cases} \quad (1)$$

其中: $n \geq 1$ , $P_n$ 表示系统状态为 $n$ 的概率,且满足

$$\sum_{n=0}^{+\infty} P_n = 1. \quad (2)$$

联立方程(1)和(2),得

$$P_n = (1 - \rho)\rho^n. \quad (3)$$

其中 $\rho = \lambda/\mu$ 反映了服务机构的忙碌程度,要使系统稳定,必须满足 $\rho < 1$ .

### 1.2 Jackson开网模型

多节点排队模型相互串并联,即构成了排队网络模型.Jackson开网排队系统中,顾客可以来自系统之外,也可以来自系统内部.一个顾客接受某节点服务后可以离开系统,也可以以一定的概率转移到另一个节点继续接受服务.

**定理1** 若 $N_1(t)$ 和 $N_2(t)$ 分别是参数为 $\lambda_1$ 和 $\lambda_2$ 的泊松流,且 $N_1(t)$ 与 $N_2(t)$ 相互独立,则合成流 $N_1(t) + N_2(t)$ 是参数为 $\lambda_1 + \lambda_2$ 的泊松流.

**定理2** 某事件流 $N(t)$ 是参数为 $\lambda$ 的泊松流,若顾客以概率为 $p$ ( $0 < p < 1$ )到达系统,则顾客到达系统的事件流是参数为 $\lambda p$ 的泊松流.

定理1和定理2是排队网络模型在数学上成立的前提,也是后文中用Jackson开网模型解决实际问题的基础.

一个具有 $M$ 节点的Jackson开网模型具有如下性质:

1) 有唯一外部顾客源节点.每位新到顾客以概率 $r_{out,i}$ 到达节点 $i$ ,满足 $\sum_{i=1}^M r_{out,i} = 1$ .由定理2可知,从系统外到系统内的顾客满足泊松流.

2) 有唯一外部终点.系统内某位顾客接受完服务后,以概率 $r_{i,out}$ 离开系统.由定理2可知,从系统内转移到系统外的顾客满足泊松流.

3) 顾客在系统内部随机转移.在节点 $i$ 接受完服务后以概率 $r_{i,j}$ 转移到节点 $j$ ,且满足 $r_{i,out} + \sum_{j=1}^M r_{i,j} = 1$ .由定理2可知,节点内部转移的顾客满足泊松流;由定理1可知,多泊松流汇合进入某一节点仍然满足泊松流.

基于本文的应用背景,设定每个节点均为一个

$M/M/1$ 排队系统,则有:

4) 外部顾客源节点到达率服从参数为 $\lambda$ 的泊松分布. 每个内部服务机构,服务时间服从参数为 $\mu_i$ 的负指数分布,且不同节点服务率相互独立,与顾客到达相互独立.

模型状态可表示为

$$\mathbf{n} = (n_1, n_2, \dots, n_m), \quad (4)$$

其中 $n_i$ 表示节点 $i$ 中顾客个数.

定义 $M$ 维向量 $\mathbf{I}_i = (0, 0, \dots, 1, 0, \dots, 0)$ , 向量 $\mathbf{I}_i$ 满足第 $i$ 个元素为1,其他元素为0.

平衡情况下,系统状态的流入与流出是相等的,满足下式:

$$\begin{aligned} \lambda p(\mathbf{n}) + \sum_{i=1}^M \mu_i p(\mathbf{n}) = \\ \sum_{i=1}^M \lambda r_{\text{out},i} p(\mathbf{n} - \mathbf{I}_i) + \sum_{i=1}^M \mu_i r_{i,\text{out}} p(\mathbf{n} + \mathbf{I}_i) + \\ \sum_{i=1}^M \sum_{j=1}^M \mu_i r_{ij} p(\mathbf{n} + \mathbf{I}_i - \mathbf{I}_j). \end{aligned} \quad (5)$$

等号左侧表示状态流出的原因,包括系统外部顾客进入系统、系统内部顾客离开系统或系统内顾客转移;等号右侧表示状态流入的途径,包括状态从 $\mathbf{n} - \mathbf{I}_i$ 变为 $\mathbf{n}$ 、从 $\mathbf{n} + \mathbf{I}_i$ 变为 $\mathbf{n}$ 和从 $\mathbf{n} + \mathbf{I}_i - \mathbf{I}_j$ 变为 $\mathbf{n}$ . 为了求解平衡方程,定义节点 $i$ 的平均吞吐量为

$$\theta_i = r_{\text{out},i} \lambda + \sum_{j=1}^M r_{ji} \theta_j. \quad (6)$$

其中: $r_{\text{out},i} \lambda$ 为系统外顾客进入节点 $i$ 的到达率, $\sum_{j=1}^M r_{ji} \theta_j$ 为从其他节点转入节点 $i$ 的顾客到达率, $\theta_i$ 相当于 $M/M/1$ 排队系统中的顾客到达率. 于是,忙碌程度 $\rho_i = \theta_i / \mu_i$ ,则有

$$p_i(n_i) = (1 - \rho_i) \rho_i^{n_i}, \quad (7)$$

其中 $n_i$ 表示向量 $\mathbf{n}$ 中,第 $i$ 个元素为 $n_i$ ,其他元素任意. 文献[8]证明了Jackson开网中各个节点的运行是相互独立的,因此,整个网络的稳态概率解可以表示为各节点稳态解的乘积,即

$$p(\mathbf{n}) = \prod_{i=1}^M p_i(n_i). \quad (8)$$

## 2 地面分队防御作战模型

地面分队防御作战中,需要在多个关键节点部署兵力,每个节点部署的兵力对敌目标的阻击可视为服务台对顾客提供服务:当防御兵力消灭敌军时,视为顾客接受完服务后离开排队网络系统;当某节点的防御兵力未能消灭敌军时,视为敌军通过该节点进

入另一节点. 地面分队防御作战与Jackson开网排队模型有诸多契合之处,便于建立排队模型,但是也存在以下问题:1)排队网络模型中,各个节点必须满足 $\rho_i < 1$ ,否则系统无法稳定,而防御作战中,敌军多数从第1层防线的节点进入系统,造成 $\rho_i > 1$ ,不满足稳定条件,因此必须对网络进行调整;2)排队网络中,顾客离开系统表示服务完成,而防御作战中,顾客离开系统,除了有被我方毁伤的部分,还有可能是敌军突破我防线而进入我部纵深,这两种情况需区别对待.

### 2.1 模型建立

设我地面分队共设有 $W$ 层防线,每层防线有 $M_k$ 个防御节点, $k$ 表示层数,每个节点可以部署一定数量的兵力,节点 $i$ 的服务率为 $\mu_i$ , $\mu_i$ 正比于兵力数量. 顾客在系统内的转移,只能从前一层防线的节点转移到后一层防线的节点,从第 $k-1$ 层节点到第 $k$ 层节点的转移概率 $R_{k-1,k} = [r_{ij}]_{M_{k-1} \times M_k}$ ,满足 $\sum_{j=1}^{M_k} r_{ij} = 1$ .  $i$ 表示第 $k-1$ 层的某一节点编号,满足 $1 \leq i \leq M_{k-1}$ ;  $j$ 为第 $k$ 层的某一节点编号,满足 $1 \leq j \leq M_k$ . 节点之间的转移概率可以依据地形地势等战场环境信息进行预测,也可以采用最坏假定的思想,认定敌方采用最不利于我方的节点转移比例. 顾客从系统内节点 $i$ 转移到系统外的概率为 $r_{i,\text{out}}$ , $r_{i,\text{out}}$ 即为对目标的毁伤概率.

### 2.2 网络模型等效转化

本文模型假设敌方突破某防御节点后会继续向我方纵深进军,不会向同层节点或前一层节点转移,也不会重复访问本节点,即本文的Jackson开网模型为非回馈网络,确保了后续各节点独立,且客户到达过程仍为泊松过程. 考虑到当突击武器面对薄弱防守时,会直接突破该节点进入对方纵深,相当于在该节点未接受服务,因此,可将超出当前节点负荷的顾客直接注入下一层的节点. 设定节点 $i$ 的最高负荷为 $0.9\mu_i$ ,具体操作如下.

#### 2.2.1 首层节点

首先,计算首层节点的系统外顾客到达率,即

$$\lambda_{\text{out},i} = \lambda r_{\text{out},i}. \quad (9)$$

其中: $i = 1, 2, \dots, M_1$ ,  $M_1$ 为首层节点个数; $\lambda$ 为顾客进入排队系统的总到达率.

然后,判断各节点是否超过最大负荷,并由此决定节点的等效输入量和吞吐量,即

$$\lambda_i = \begin{cases} 0.9\mu_i, & \lambda_{\text{out},i} > 0.9\mu_i; \\ \lambda_{\text{out},i}, & \lambda_{\text{out},i} \leq 0.9\mu_i. \end{cases} \quad (10)$$

$$\theta_i = \lambda_i. \quad (11)$$

首层节点的等效输入量为 $\lambda_i$ , 吞吐量为 $\theta_i$ , 二者相等.

### 2.2.2 第2层节点

首先, 计算该层节点的等效系统外顾客到达率, 即

$$\lambda_{\text{out},j} = \sum_{i=1}^{M_1} (\lambda_{\text{out},i} - \lambda_i) r_{ij}. \quad (12)$$

其中:  $j = 1, 2, \dots, M_2$ ,  $M_2$  为第2层节点个数. 即超出上一层节点最大服务负荷的部分, 按转移比例直接转嫁到本层节点.

然后, 计算上一层节点转移来的顾客量

$$\varphi_j = \sum_{i=1}^{M_1} \theta_i (1 - r_{i,\text{out}}) r_{ij}. \quad (13)$$

被节点 $i$ 服务过的顾客以 $r_{i,\text{out}}$ 的比例直接离开排队系统, 以 $1 - r_{i,\text{out}}$ 的比例转移到下一层节点.

最后, 计算本层节点的等效输入量

$$\lambda_j = \begin{cases} 0.9\mu_j, & \varphi_j > 0.9\mu_j; \\ \lambda_{\text{out},j}, & \varphi_j < 0.9\mu_j - \lambda_{\text{out},j}; \\ 0.9\mu_j - \varphi_j, & \text{otherwise.} \end{cases} \quad (14)$$

以及本层节点的吞吐量

$$\theta_j = \begin{cases} 0.9\mu_j, & \varphi_j > 0.9\mu_j; \\ \lambda_{\text{out},j} + \varphi_j, & \varphi_j < 0.9\mu_j - \lambda_{\text{out},j}; \\ 0.9\mu_j, & \text{otherwise.} \end{cases} \quad (15)$$

### 2.2.3 第 $k$ 层节点

第 $k$  ( $k > 2$ ) 层节点的情况更复杂, 除了要考虑直接转移来的系统外顾客到达率和上一层节点转移来的顾客量, 还要分情况讨论 $k-1$ 层节点的服务率是否能够消化从 $k-2$ 层节点转移来的顾客量. 当 $k-1$ 层节点能够服务从 $k-2$ 层节点转移来的顾客量时, 本层节点接受系统内上层节点转移的顾客量为

$$\varphi_j = \sum_{i=1}^{M_{k-1}} \theta_i (1 - r_{i,\text{out}}) r_{ij}. \quad (16)$$

当 $k-1$ 层节点不足以服务从 $k-2$ 层节点转移来的顾客量时, 则 $k-2$ 层节点和 $k-1$ 层节点都会向本层节点转移顾客, 有

$$\varphi_j = \sum_{i=1}^{M_{k-1}} \theta_i (1 - r_{i,\text{out}}) r_{ij} + \sum_{i=1}^{M_{k-1}} (\varphi_i - \theta_i) r_{ij}. \quad (17)$$

用阶跃函数可将式(16)和(17)合并为

$$\varphi_j = \sum_{i=1}^{M_{k-1}} \theta_i (1 - r_{i,\text{out}}) r_{ij} +$$

$$\sum_{i=1}^{M_{k-1}} \varepsilon(\varphi_i - \theta_i) (\varphi_i - \theta_i) r_{ij}, \quad (18)$$

其中阶跃函数 $\varepsilon(\cdot)$ 满足

$$\varepsilon(\varphi_i - \theta_i) = \begin{cases} 1, & \varphi_i - \theta_i \geq 0; \\ 0, & \varphi_i - \theta_i < 0. \end{cases}$$

最后, 计算本层节点的等效输入量 $\lambda_j$ 和吞吐量 $\theta_j$ , 计算方法同2.2.2节中的式(14)和(15).

### 2.3 优化函数

作为防御一方, 优化目标是使突破我防线的敌军最少, 目标函数表示为

$$\min J = \lambda_{\text{unserved}} + \varphi_{\text{out}}. \quad (19)$$

其中:  $\lambda_{\text{unserved}}$  为未被服务的顾客, 即

$$\lambda_{\text{unserved}} = \sum_{i=1}^{M_W} (\lambda_{\text{out},i} - \lambda_i); \quad (20)$$

$\varphi_{\text{out}}$  表示接受服务但是未被消灭的顾客, 而

$$\varphi = \sum_{i=1}^{M_W} \theta_i (1 - r_{i,\text{out}}) r_{ij} + \sum_{i=1}^{M_W} \varepsilon(\varphi_i - \theta_i) (\varphi_i - \theta_i) r_{ij}. \quad (21)$$

优化函数值受到诸多因素的影响. 防御方决定各节点的兵力分配, 即各节点的服务率 $\mu_i$ ; 进攻方决定系统内的顾客转移比例 $r_{i,j}$ 和系统外进入系统首层节点的比例 $r_{\text{out},i}$ ; 武器装备性能和客观战场条件决定毁伤概率 $r_{i,\text{out}}$ . 由此可见, 影响优化函数值的因素并非都由我方决定, 只有将双方的博弈对抗特性加入到决策优化过程中, 才能获得具有说服力的决策方案<sup>[9]</sup>. 优化函数可表示为

$$\begin{aligned} \min & J(\mu_i, r_{i,j}, r_{\text{out},i}); \\ \text{s.t.} & \begin{cases} \max J(r_{i,j}, r_{\text{out},i} | \mu_i), \\ L_{\text{out},i} \leq r_{\text{out},i} \leq U_{\text{out},i}, \\ L_{i,j} \leq r_{i,j} \leq U_{i,j}. \end{cases} \end{aligned} \quad (22)$$

约束条件中的 $\max J(r_{i,j}, r_{\text{out},i} | \mu_i)$ 为底层迭代的优化函数, 表示在防御方阵地部署 $\mu_i$ 明确的情况下, 进攻方采取最优的兵力转移比例 $r_{\text{out},i}$ 和 $r_{i,j}$ , 从而获取最大的防线突破率 $\text{Max } J$ . 而 $\min J(\mu_i, r_{i,j}, r_{\text{out},i})$ 表示上层迭代的优化函数, 即在防御方采取任何兵力部署方式下进攻方一定会采取最优的进攻方式的假定下, 防御方选择最优的兵力部署方式, 使进攻方突破率最低. 上层寻优的每一次迭代, 都要调用下层优化函数寻优的结果, 而下层优化函数的求解需要下层的寻优迭代. 采取双层迭代模型, 可以寻找双方对抗博弈下的纳什均衡解, 对于防守而言是一种

稳妥的决策思路,符合作战实际.

双层迭代优化模型如图1所示.

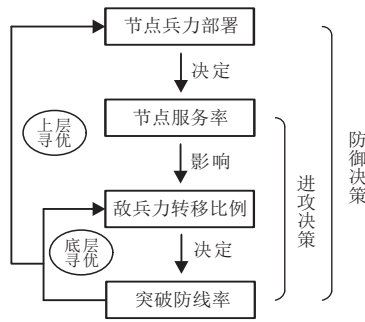


图1 双层迭代优化模型示意图

### 3 模型求解

双层迭代优化模型中:底层寻优为进攻方在防守方兵力部署已知的情况下寻优最优兵力转移比例,问题的维数为  $M_1 + \sum_{k=1}^{W-1} M_k M_{k+1}$ ;上层寻优为防守方寻优最优兵力部署方案,问题维数等于节点个数. 以后文仿真实验中4层防线战例为例,4层防线各层防线节点个数为4、3、3、2的情况下,底层寻优问题的维数为31,上层寻优问题的维数为12. 由此可见,问题维数较大,尤其是底层问题,如果进一步增加防线层数或节点数,则问题的维数会大幅度增加,使得求解本文模型的计算压力较大.

本文采用人工蜂群算法 (artificial bee colony, ABC) 求解模型. ABC算法是一种模拟蜜蜂采蜜行为的群智能优化方法<sup>[10]</sup>,算法具有寻优效果好、控制参数少、实现简单等特点<sup>[11]</sup>,是目前最为有效的优化算法之一. ABC算法已经在功率流优化<sup>[12]</sup>、路径规划问题<sup>[13]</sup>、武器目标分配问题<sup>[14]</sup>等多个方面广泛应用. 但ABC算法仍存在食物源开发与探索之间的矛盾,以及收敛速度较慢、局部求精能力较弱的问题<sup>[15]</sup>. ABC算法参数较少,蜜蜂分工明确,特征显著,便于针对具体问题进行具体的设计和应用,通过对ABC算法进行适当的改进和应用设计,可以显著提高算法的性能.

#### 3.1 标准ABC算法<sup>[16]</sup>

人工蜂群由雇佣蜂、观察蜂和侦查蜂组成,每个蜜蜂所在的位置即为一个可行解,所在位置对应的蜜源质量为解的质量.

##### 3.1.1 初始化

对于一个  $m$  维问题,蜜源位置向量  $x_i = [x_{i1}, x_{i2}, \dots, x_{im}]$ ,  $i = 1, 2, \dots, SN$ ,  $SN$  表示种群个数. 蜜源初始位置随机产生,解空间上限  $ub = [ub_1, ub_2, \dots, ub_m]$ , 下限  $lb = [lb_1, lb_2, \dots, lb_m]$ , 初始解为

$$x_{id} = lb_d + (ub_d - lb_d) \cdot \text{rand}(0, 1). \quad (23)$$

其中:  $i = 1, 2, \dots, SN$ ;  $d = 1, 2, \dots, m$ ;  $x_{id}$  是向量  $x_i$  的第  $d$  维变量;  $\text{rand}(0, 1)$  是  $[0, 1]$  上的随机数.

##### 3.1.2 雇佣蜂

每一个食物源对应一个雇佣蜂  $x_i$ , 雇佣蜂  $x_i$  随机选择另一只蜜蜂  $x_k$  进行邻域搜索并更新位置, 获得新的蜜源  $v_i$ , 有

$$v_{id} = x_{id} + (x_{id} - x_{kd}) \cdot \text{rand}(-1, 1). \quad (24)$$

其中:  $v_{id}$  为向量  $v_i$  的第  $d$  维变量;  $\text{rand}(-1, 1)$  为  $[-1, 1]$  上均匀分布的随机数;  $k \in \{1, 2, \dots, SN\}$ , 且满足  $k \neq i$ , 获得新的蜜源后, 按照贪婪选择的方式更新食物源. 如果新的食物源  $v_i$  的质量高于原食物源  $x_i$  的质量, 则  $v_i$  取代  $x_i$ ; 否则  $x_i$  保持不变.

##### 3.1.3 观察蜂

雇佣蜂更新一轮后, 将蜜源信息分享给观察蜂, 观察蜂根据食物源的质量进行概率选择. 第  $i$  个食物源被观察蜂选择的概率为

$$p_i = \text{fit}_i / \sum_{j=1}^{SN} \text{fit}_j. \quad (25)$$

求极小值问题时, 适应度值  $\text{fit}_i$  按照下式计算:

$$\text{fit}_i = \begin{cases} 1/(1 + f_i), & f_i \geq 0; \\ 1 + |f_i|, & f_i < 0. \end{cases} \quad (26)$$

其中  $f_i$  是第  $i$  个蜜源的评价值, 由所求解问题的目标函数计算得到.

观察蜂依据概率选择食物源, 与雇佣蜂相同.

##### 3.1.4 侦查蜂

当雇佣蜂和观察蜂多次搜索而未发生更新时, 放弃该蜜源, 变为侦查蜂, 侦查蜂按照式(23)随机初始化一个新的蜜源代替.

### 3.2 求解本文模型的ABC算法

针对本文提出的基于 Jackson 网络的地面分队防御作战兵力部署模型的特点, 对 ABC 算法进行有针对性地设计和改进.

#### 3.2.1 搜索策略

标准 ABC 算法中, 雇佣蜂和观察蜂都采用式(24)进行搜索: 雇佣蜂是盲目搜索, 没有任何方向性, 只在进行贪婪选择时确保优质解被保留; 观察蜂仅在选择邻居时, 倾向于选择适应度高的个体作为邻居, 收敛性不强. 本文对搜索策略进行如下改进.

1) 雇佣蜂邻域最优解引导策略<sup>[17]</sup>.

雇佣蜂阶段应侧重于探索能力, 提高发现优良区域的概率. 在雇佣蜂之间建立信息交互机制, 每只雇

佣蜂拥有各自的局部视野,能够发现局部视野内的最优解,并飞向局部最优解.通过这种方式,在雇佣蜂之间建立起协作机制,使雇佣蜂进化策略有了方向性.

定义雇佣蜂 $x_i$ 与 $x_j$ 的距离公式为

$$d_{i,j} = \sum_{t=1}^m \text{bool}(x_{it} \neq x_{jt}). \quad (27)$$

定义雇佣蜂视野范围

$$\text{Vis}_i = \sum_{j=1}^{\text{SN}} d_{i,j} / (\text{SN} - 1). \quad (28)$$

当 $d_{i,j} \leq \text{Vis}_i$ 时,表示蜜蜂 $j$ 位于蜜蜂 $i$ 的视野范围内,蜜蜂 $i$ 视场内最大适应度的蜜蜂所在蜜源位置 $x_{\text{nbest}}^i$ 即为其邻域最优解.

2) 观察蜂全局精英引导策略.

观察蜂拥有全局视野,能够发现全局的精英解,并飞向精英解所在位置.选择种群中适应度值最高的一部分群体组成精英群体,观察蜂随机选择精英解并接受引导.精英的数量 $T = \text{ceil}(\rho \cdot \text{SN})$ , $\rho$ 表示精英解比例, $\text{ceil}(\cdot)$ 为向上取整函数.

### 3.2.2 更新方式

雇佣蜂和观察蜂搜索到局部最优解或全局精英解后,需要接受其引导.本文模型中,无论是上层的防御兵力部署问题,还是下层兵力转移问题,都需避免超出可行域,因此,在更新方程中引入修正算子.

对于雇佣蜂,采用下式进行更新:

$$v_{i,d} = \begin{cases} y_{i,d} + \alpha, & y_{\text{nbest},d}^i > y_{i,d}; \\ y_{i,d}, & y_{\text{nbest},d}^i = y_{i,d}; \\ y_{i,d} - \alpha, & y_{\text{nbest},d}^i < y_{i,d}. \end{cases} \quad (29)$$

为避免新解 $v_i$ 超出可行域,直接引入修正算子

$$v_{i,\text{rand}} = \begin{cases} y_{i,\text{rand}} - \alpha, & y_{\text{nbest},d}^i > y_{i,d}; \\ y_{i,\text{rand}} + \alpha, & y_{\text{nbest},d}^i < y_{i,d}. \end{cases} \quad (30)$$

其中: $y_{\text{nbest}}^i$ 为蜜蜂 $i$ 视场范围内的局部最优解, $y_{i,\text{rand}}$ 为蜜蜂 $i$ 所对应解向量的随机一维与 $d$ 不同的元素, $\alpha$ 为更新速率.底层循环中 $\alpha$ 为0.1,上层循环中 $\alpha$ 取1.

观察蜂随机选择全局精英解,并接受其引导,有

$$v_{i,d} = \begin{cases} y_{i,d} + \alpha, & y_{\text{glite},d} > y_{i,d}; \\ y_{i,d}, & y_{\text{glite},d} = y_{i,d}; \\ y_{i,d} - \alpha, & y_{\text{glite},d} < y_{i,d}. \end{cases} \quad (31)$$

修正算子

$$v_{i,\text{rand}} = \begin{cases} y_{i,\text{rand}} - \alpha, & y_{\text{glite},d} > y_{i,d}; \\ y_{i,\text{rand}} + \alpha, & y_{\text{glite},d} < y_{i,d}. \end{cases} \quad (32)$$

其中 $y_{\text{glite}}$ 为随机选择的一个精英解.

### 3.2.3 选择策略

采用贪婪算法对雇佣蜂和观察蜂搜索之后的新旧解进行选择,该策略会造成过收敛,陷入局部最优.本文提出 $\varepsilon$ -贪心策略,更新过程中当新解适应度值高于旧解时,以 $\varepsilon$ 的概率选择新解,以 $1 - \varepsilon$ 的概率维持旧解.

$\varepsilon$ -贪心策略公式为

$$x_{\text{new}} = \begin{cases} x_i, & \text{fit}(v_i) < \text{fit}(x_i); \\ v_i, & \text{fit}(v_i) > \text{fit}(x_i), \text{rand} < \varepsilon; \\ x_i, & \text{fit}(v_i) > \text{fit}(x_i), \text{rand} > \varepsilon. \end{cases} \quad (33)$$

## 4 仿真实验

本节共设置3组实验.第1组实验为底层寻优实验,即在我方防御兵力部署已知的条件下,寻优敌方最优进攻策略,为了验证本文改进ABC算法的进步性,引入对比算法进行比较.第2组实验为一个完整的地面分队防御作战Jackson开网模型仿真算例,说明本文模型的实用意义,并验证本文所提出的双层迭代模型在攻防博弈中的有效性.第3组实验分析我方兵力对敌军毁伤概率变化而对最优兵力部署的影响,说明防御纵深与火力密度的辩证关系.

### 4.1 底层寻优实验

底层寻优为敌军进攻提供最优决策,是在我方兵力部署确定情况下,敌军寻求最优兵力转移比例,使得突破防线率最大.共设置3组不同规模战例,每种规模下设置随机生成3种不同的防御兵力部署方案.为了确保实验的随机性并便于读者复现数据,采用轮盘赌的方式生成兵力部署方案,并采用Matlab的 $\text{rand}('state', 0)$ 、 $\text{rand}('state', 2)$ 、 $\text{rand}('state', 4)$ 函数生成数据.我方对进攻目标的毁伤概率 $r_{i,\text{out}} = 0.4$ , $i$ 表示节点编号.具体战例数据见表1.

表1 战例数据

防线数	兵力	节点数	敌到达率	防御兵力部署方案
2	10	4 3	5	1, 1, 0, 3 1, 2, 2
				1, 4, 1, 1 1, 0, 2
				1, 0, 1, 2 2, 3, 1
3	15	4 3 3	8	1, 1, 1, 0 3, 0, 2 3, 2, 2
				1, 2, 3, 2 0, 2, 2 0, 3, 0
				1, 2, 1, 0 3, 1, 2 0, 4, 1
4	20	4 3 3 2	10	1, 0, 2, 0 2, 3, 0 2, 1, 3 2, 4
				1, 1, 5, 2 2, 0, 2 1, 1, 0 3, 2
				1, 2, 1, 1 1, 4, 3 2, 0, 2 2, 1

系统外敌兵力向系统内第1层防线节点转移比例的可行区间为

$$R_{\text{out},\text{line1}} \in \{[0.2, 0.4] [0.2, 0.4] [0.1, 0.3] [0.1, 0.3]\}.$$

系统内节点转移比例的可行区间为

$$R_{line1,line2} \in \left\{ \begin{matrix} [0.3, 0.5] & [0.2, 0.4] & [0.2, 0.3] \\ [0.2, 0.4] & [0.2, 0.4] & [0.2, 0.4] \\ [0.1, 0.3] & [0.4, 0.6] & [0.2, 0.4] \\ [0.1, 0.3] & [0.2, 0.3] & [0.4, 0.6] \end{matrix} \right\},$$

$$R_{line2,line3} \in \left\{ \begin{matrix} [0.3, 0.5] & [0.2, 0.4] & [0.1, 0.3] \\ [0.2, 0.3] & [0.2, 0.5] & [0.2, 0.3] \\ [0.1, 0.3] & [0.2, 0.4] & [0.3, 0.4] \end{matrix} \right\},$$

$$R_{line3,line4} \in \left\{ \begin{matrix} [0.5, 0.6] & [0.3, 0.5] \\ [0.3, 0.5] & [0.4, 0.6] \\ [0.4, 0.6] & [0.5, 0.8] \end{matrix} \right\}.$$

采用本文算法及GRABC<sup>[18]</sup>、IGPSO<sup>[19]</sup>、RNADE<sup>[20]</sup>算法求解最佳兵力转移比例. 参数设置NP = 50, limit = 5, ρ = 0.1, ε = 0.2(本文算法), p = 0.1 (GRABC), c<sub>1</sub> = 2, c<sub>2</sub> = 2, ω<sub>max</sub> = 0.9, ω<sub>min</sub> = 0.4, δ = 0.01, τ = 0.75 (IGPSO). 3种规模战例的最大评价次数分别为200、500、1000. 由于本文算法引入了修正算子, 确保每次搜索都不会超出可行域, 为公平起见, 当对比算法搜索解超出可行域时直接重新搜索, 且不计入优化函数评价次数.

各算法独立运行100次, 底层寻优实验测试报告见表2. 其中: mean表示算法求得的目标函数值与最优值J<sub>max</sub>的差值的均值, mean越接近0, 算法收敛精度越高; Std表示各独立实验结果的标准差, Std越小算法稳定性越好; SR(100%)表示函数值在最大函数评价次数内收敛到J<sub>max</sub>的概率, SR(99%)表示函数值在最大函数评价次数内收敛到0.99J<sub>max</sub>的概率. 最优值J<sub>max</sub>设定为所有算法对战例求得的最大目标函数值.

从表2中数据可以看出, 采用本文算法解决底层寻优问题的收敛精度和稳定性都最好. 对于战例1~战例3和战例4~战例6的双层防线、3层防线的战例, 本文方法以接近100%的概率找到最优解; 而对于4层防线的战例, 本文方法虽然不能保证每次都寻优到最优解, 但是能够以接近100%的概率找到0.99J<sub>max</sub>. 考虑到战场环境瞬息万变, 战机稍纵即逝, 对于较大规模的战例, 能够在有限的时间内找到次优解也是有意义的. 本文算法在底层寻优表现出的良好性能为完整的双层迭代模型求解打下了基础.

#### 4.2 双层寻优实验

底层寻优是在我方兵力部署确定的情况下, 寻求敌军最优兵力转移比例, 使得突破防线率最大. 上层寻优为我方的防御兵力部署提供最优方案, 使得敌军

表2 战例数据

战例	指标	GRABC	IGPSO	RNADE	本文算法
战例1	mean	1.671e-03	6.012e-04	0	0
	Std	1.217e-02	3.178e-03	0	0
	SR(100%)	93%	96%	100%	100%
	SR(99%)	99%	100%	100%	100%
战例2	mean	1.942e-04	1.121e-04	1.322e-04	0
	Std	1.904e-03	1.245e-03	1.415e-03	0
	SR(100%)	98%	99%	99%	100%
	SR(99%)	100%	100%	100%	100%
战例3	mean	1.723e-04	0	0	0
	Std	1.813e-03	0	0	0
	SR(100%)	99%	100%	100%	100%
	SR(99%)	100%	100%	100%	100%
战例4	mean	0	9.570e-04	1.341e-03	3.321e-04
	Std	0	5.451e-03	6.104e-03	3.317e-03
	SR(100%)	100%	97%	95%	99%
	SR(99%)	100%	100%	100%	100%
战例5	mean	1.262e-03	6.422e-04	2.977e-03	0
	Std	7.155e-03	4.513e-03	1.387e-02	0
	SR(100%)	97%	98%	95%	100%
	SR(99%)	100%	100%	97%	100%
战例6	mean	1.106e-03	0	6.301e-03	0
	Std	6.407e-03	0	6.275e-02	0
	SR(100%)	97%	100%	99%	100%
	SR(99%)	99%	100%	99%	100%
战例7	mean	1.114e-03	6.712e-03	4.465e-03	6.792e-04
	Std	5.148e-03	2.615e-02	1.867e-02	3.672e-03
	SR(100%)	95%	92%	90%	96%
	SR(99%)	100%	94%	97%	100%
战例8	mean	1.533e-03	1.486e-03	1.971e-03	7.221e-04
	Std	8.717e-03	1.023e-02	1.057e-02	5.469e-03
	SR(100%)	97%	97%	95%	98%
	SR(99%)	97%	99%	99%	99%
战例9	mean	6.106e-03	3.345e-03	4.949e-03	1.031e-04
	Std	2.895e-02	1.900e-02	2.472e-02	1.025e-03
	SR(100%)	93%	96%	95%	99%
	SR(99%)	96%	97%	96%	100%

突破我防线率最低, 上层迭代直接调用底层寻优的结果. 本节实验采用4.1节的最大规模战例数据, 我方共设定4层防线, 各层防线分别设定4、3、3、2个防御节点, 共计有20个兵力单位, 预测敌军到达率为10.

本文算法参数NP = 50, limit = 5, ρ = 0.2, ε = 0.2, 设定优化函数最大评价次数为2000. 将文献[21]的层次风驱动优化(WDO)算法作为对比算法, 以最终寻优结果作为比较对象. 本文算法和层次WDO算法求解双层迭代模型的结果对比如图2所示. 可以看出, 本文算法比层次WDO算法收敛精度更优, 说明了本文采用双层迭代寻优结构的科学性. 层次WDO算法效果不理想的原因是, 该方法采用“一定一动”的交互迭代方式, 无法保证底层迭代的充分寻优, 而本

文双层迭代可以有效地克服这一问题. 底层寻优的高收敛精度为上层寻优打下了良好的基础.

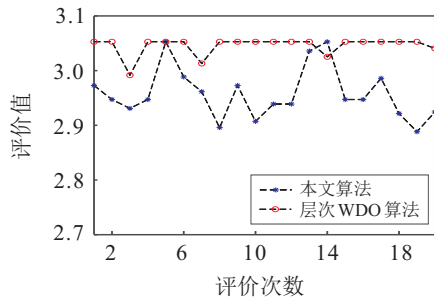


图2 本文算法和层次WDO算法对比

根据本文算法求得的最优解,我方采取的兵力分配方案  $\mu = 2, 2, 1, 1|2, 2, 2|1, 2, 1|2, 2$ . 由2.2节的排队网络等效转化方法,可求得各节点的等效外界注入率为  $\lambda = [1.8, 1.8, 0.9, 0.9, 0.52, 0.504, 0.504, 0.1498, 0.6557, 0, 0.8555, 0.5183]$ , 各节点的吞吐量  $\theta = [1.8, 1.8, 0.9, 0.9, 1.168, 1.8, 1.8, 0.9, 1.8, 1.8]$ . 稳定状态下排队网络如图3所示. 敌兵力对我防线的突破率为3.053, 其中有0.8928的敌兵力未接受服务而直接突破防线, 有2.16的敌兵力接受服务但未被消灭而突破防线.

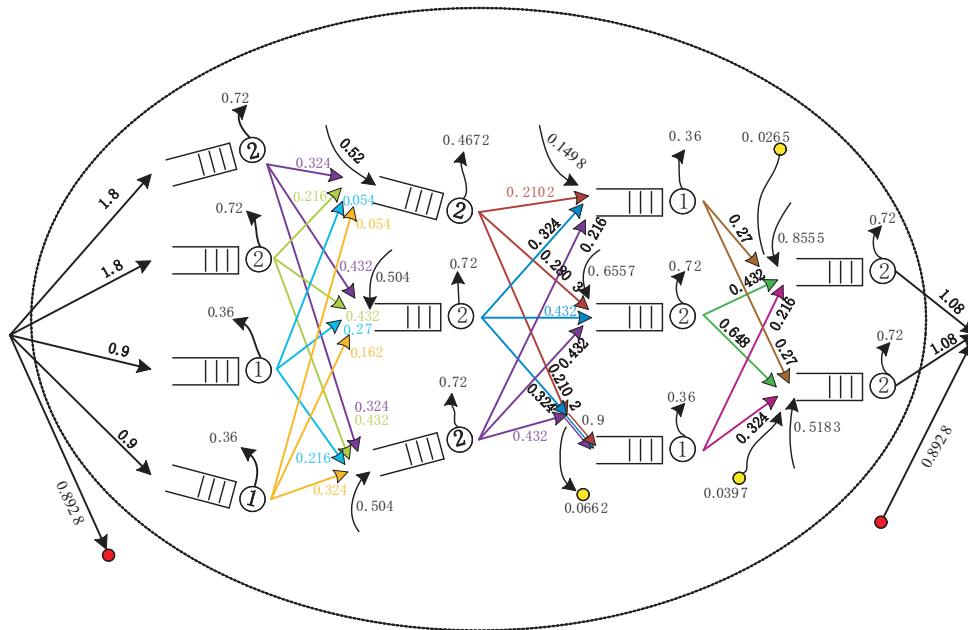


图3 稳定状态下排队网络

### 4.3 毁伤概率实验

对敌目标的毁伤概率取决于敌我装备性能和地理环境等客观因素,在交战之前可以依据经验给出估计值,毁伤概率会直接影响兵力部署方案. 本节实验比较不同毁伤概率下的最佳兵力部署方案. 分别设定毁伤概率为0.2、0.4、0.6、0.8进行双层迭代寻优,毁伤概率实验结果见表3.

表3 毁伤概率实验结果

毁伤概率	最优兵力部署方案	各层兵力数
0.2	2, 2, 1, 1 1, 2, 2 1, 2, 1 2, 3	6, 5, 4, 5
0.4	2, 2, 1, 1 2, 2, 2 1, 2, 1 2, 2	6 6 4 4
0.6	2, 2, 2, 1 1, 2, 2 2, 2, 2 1, 1	7 5 6 2
0.8	3, 2, 2, 2 2, 2, 2 1, 1, 1 1, 1	9 6 3 2

当毁伤概率为0.2和0.4时,在各层防线上部署的兵力相对比较均衡,强调防御纵深;而当毁伤概率为0.8时,在第1道防线部署近一半的兵力,力争在首层防线即消灭大部分敌军. 可以看出,随着毁伤概率增

大,最优兵力部署方案从侧重于防御纵深向侧重于火力密度转变. 对于杀伤概率高的武器系统,采用增大火力密度,力争在第1层防线歼灭来袭之敌;而对于杀伤概率一般的武器系统,则应采取加大火力纵深,逐层阻击的策略. 本实验以毁伤概率为例来说明防御纵深与火力密度的辩证关系,事实上,兵力对比、敌进攻到达率、防御节点的单位服务率等因素与之类似.

### 5 结论

本文将Jackson网络应用于地面分队防御作战兵力部署,通过建立排队网络模型、网络等效转化、双层迭代寻优解算模型,求得最优兵力分配方案,为解决地面防御作战兵力部署提供了一种量化的决策方法. 本文模型原理简单、指标定义明确且相对独立,便于引入更多因素,例如阻击敌军过程中我方装备损毁,地形造成的敌军到达率变化,动态兵力部署问题等. 后续研究拟进一步增加网络模型考虑的因素.

## 参考文献(References)

- [1] 褚凯轩, 常天庆, 孔德鹏, 等. 基于蜂群算法的坦克阵地部署与火力分配模型[J]. 系统工程与电子技术, 2022, 44(2): 546-556.  
(Chu K X, Chang T Q, Kong D P, et al. Bee colony algorithm based model of tank troop deployment and firepower allocation[J]. Systems Engineering and Electronics, 2022, 44(2): 546-556.)
- [2] Chen J W, Chang T Q, Ma D Z, et al. Firepower deployment method of tank unit based on operational effectiveness calculation Retraction[J]. Journal of Computer Applications, 2016, 36(1): 275-281.
- [3] 邢清华, 刘付显. 区域防空部署优化系统建模[J]. 系统工程与电子技术, 2006, 28(5): 712-715.  
(Xing Q H, Liu F X. Modeling on area air defense optimization deployment system[J]. Systems Engineering and Electronics, 2006, 28(5): 712-715.)
- [4] Nian S L, Yan J G, Liu Z C. Research on the model simulation of air defence military strength deployment optimization[J]. Microcomputer Information, 2012, 28(4): 15-16.
- [5] 赵鹏蛟, 李建国. 基于排队论的防空兵力部署优化模型[J]. 火力与指挥控制, 2017, 42(11): 38-42.  
(Zhao P J, Li J G. An optimization model for air defense troops deployment based on queuing theory[J]. Fire Control & Command Control, 2017, 42(11): 38-42.)
- [6] 赵鹏蛟, 李建国, 李红霞, 等. 基于Memetic算法的要地防空兵力机动部署优化方法[J]. 火力与指挥控制, 2018, 43(9): 25-29.  
(Zhao P J, Li J G, Li H X, et al. Optimization of troops maneuver deployment for key-point air defense based on memetic algorithm[J]. Fire Control & Command Control, 2018, 43(9): 25-29.)
- [7] Cao L, Dong Q, Peng W, et al. Effectiveness evaluation of ballistic missile defense system based on queuing theory[J]. Journal of Nanjing University of Science and Technology, 2011, 35(4): 470-474.
- [8] Jackson J R. Networks of waiting lines[J]. Operations Research, 1957, 5(4): 518-521.
- [9] Yao Z X, Li M, Chen Z J, et al. Mission decision-making method of multi-aircraft cooperatively attacking multi-target based on game theoretic framework[J]. Chinese Journal of Aeronautics, 2016, 29(6): 1685-1694.
- [10] Rajasekhar A, Lynn N, Das S, et al. Computing with the collective intelligence of honey bees — A survey[J]. Swarm and Evolutionary Computation, 2017, 32: 25-48.
- [11] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm[J]. Journal of Global Optimization, 2007, 39(3): 459-471.
- [12] Bai W L, Eke I, Lee K Y. An improved artificial bee colony optimization algorithm based on orthogonal learning for optimal power flow problem[J]. Control Engineering Practice, 2017, 61: 163-172.
- [13] 张占义, 朱金达. 蜂群优化算法的机器人路径规划[J]. 兵器装备工程学报, 2020, 41(7): 152-157.  
(Zhang Z Y, Zhu J D. Robot path planning based on improved ABC optimization algorithm[J]. Journal of Ordnance Equipment Engineering, 2020, 41(7): 152-157.)
- [14] Chang T Q, Kong D P, Hao N, et al. Solving the dynamic weapon target assignment problem by an improved artificial bee colony algorithm with heuristic factor initialization[J]. Applied Soft Computing, 2018, 70: 845-863.
- [15] 孔德鹏, 常天庆, 戴文君, 等. 基于排序选择和精英引导的改进人工蜂群算法[J]. 控制与决策, 2019, 34(4): 781-786.  
(Kong D P, Chang T Q, Dai W J, et al. An improved artificial bee colony algorithm based on the ranking selection and the elite guidance[J]. Control and Decision, 2019, 34(4): 781-786.)
- [16] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm[J]. Journal of Global Optimization, 2007, 39(3): 459-471.
- [17] Lin Q Z, Zhu M M, Li G H, et al. A novel artificial bee colony algorithm with local and global information interaction[J]. Applied Soft Computing, 2018, 62: 702-735.
- [18] Li G H, Cui L Z, Fu X H, et al. Artificial bee colony algorithm with gene recombination for numerical function optimization[J]. Applied Soft Computing, 2017, 52: 146-159.
- [19] 王皓, 欧阳海滨, 高立群. 一种改进的全局粒子群优化算法[J]. 控制与决策, 2016, 31(7): 1161-1168.  
(Wang H, Ouyang H B, Gao L Q. An improved global particle swarm optimization[J]. Control and Decision, 2016, 31(7): 1161-1168.)
- [20] 吴文海, 郭晓峰, 周思羽, 等. 改进差分进化算法求解武器目标分配问题[J]. 系统工程与电子技术, 2021, 43(4): 1012-1021.  
(Wu W H, Guo X F, Zhou S Y, et al. Improved differential evolution algorithm for solving weapon-target assignment problem[J]. Systems Engineering and Electronics, 2021, 43(4): 1012-1021.)
- [21] 徐兰, 苏翔. 求解双层规划优化问题的层次风驱动优化算法[J]. 控制与决策, 2016, 31(10): 1894-1898.  
(Xu L, Su X. Hierarchical wind driven optimization method for solving bi-level programming problem[J]. Control and Decision, 2016, 31(10): 1894-1898.)

## 作者简介

褚凯轩(1993—), 男, 博士生, 从事坦克火力规划、智能决策的研究, E-mail: 1016938865@qq.com;

常天庆(1963—), 男, 教授, 博士生导师, 从事火控系统智能化、信息感知与控制、智能决策等研究, E-mail: changtianqing@263.net;

孔德鹏(1990—), 男, 工程师, 博士, 从事火力规划、智能决策等研究, E-mail: kongdp55@163.com.

(责任编辑: 李君玲)