

文章编号: 2095-2163(2022)08-0157-04

中图分类号: TP391

文献标志码: A

# 基于改进人工蜂群算法的机器人任务最优指派

张翔, 朱范炳

(信阳学院 大数据与人工智能学院, 河南 信阳 464000)

**摘要:** 为了快速准确地求解机器人任务最优指派问题, 本文提出了一种改进的人工蜂群算法。考虑到指派问题解决方案离散性的特点, 在应用蜂群算法求解问题的过程中对可行解空间进行离散编码; 在采蜜蜂和观察蜂更新解时, 采用列状态转移的方法生成候选解, 保证解的可行性和多样性。实验结果表明, 改进的蜂群算法在求解指派问题模型时, 具有良好的收敛速度和求解精度。

**关键词:** 最优指派; 改进的人工蜂群算法; 离散编码; 列状态转移

## Solution of robot task assignment based on improved artificial bee colony algorithm

ZHANG Xiang, ZHU Fanbing

(School of Big Data and Artificial Intelligence, Xinyang College, Xinyang Henan 464000, China)

**[Abstract]** In order to solve the optimal assignment problem of robot tasks quickly and accurately, this paper proposes an artificial bee colony optimization algorithm. Considering the discrete characteristics of the solution of the assignment problem, the feasible solution space is discretely encoded in the process of applying the bee colony algorithm to solve the problem. So the feasibility and diversity of solutions could be guaranteed. The simulation results show that the improved bee colony algorithm has good convergence speed and accuracy in solving the assignment problem model.

**[Key words]** optimal assignment; improved artificial bee colony algorithm; dispersion codes; column state transition

## 0 引言

任务分配问题也称为指派问题, 是一类典型的 0-1 型规划问题, 属于组合优化问题中的 NP-Complete 问题, 并在诸多领域中有很强的适用性。生产和生活中的很多实际问题, 如工作分配、车辆调度、航班安排、车间设备分布和生产安排等都属于指派问题的范畴<sup>[1-2]</sup>。移动救援机器人的任务分配是以取得最大时效为目标, 也可运用指派问题模型来进行求解。求解指派问题最有效的标准计算方法是库恩提出的匈牙利算法, 但是匈牙利算法的适用条件比较严格, 一些场景下的任务分配研究可能会导致算法不收敛; 且不利于用计算机来做编程处理<sup>[1-2]</sup>。因此, 研究者提出了求解指派问题的改进算法。近年来, 自从仿生计算和群体智能问世以来, 许多研究者用智能计算成功求解了指派问题。文献[3]提出改进粒子群优化算法求解任务指派问题。文献[4-5]提出用蚁群算法和改进的蚁群算法求解指派问题。文献[6]提出最优指派问题的 DNA 算法。文献[7-8]在用蚁群和蜂群算法求解离散解问

题上做了一些指导性工作。

人工蜂群算法(Artificial Bee Colony, ABC)由土耳其学者 Karaboga 于 2005 年提出, 是一种模拟自然界蜜蜂采蜜、寻找优良蜜源行为的元启发式算法, 能够有效求解连续数值优化问题<sup>[9]</sup>。Karaboga 等人的研究指出, 相比遗传算法、差分进化算法及粒子群算法, ABC 算法在数值函数寻优中有更出色的表现。也有研究表明, ABC 算法具有设置参数少、操作简单、工程通用性强的优点, 随着对算法讨论的深入, 现在已将 ABC 算法运用到组合优化问题中。

考虑到 ABC 算法的优点, 本文采用离散编码方法对可行解进行编码, 提出一种改进的人工蜂群算法(Improved Artificial Bee Colony, IABC), 用来求解移动机器人救援任务的指派问题。通过对指派问题的建模, 运用 IABC 算法进行求解, 并与其它方法加以比较, 最后得出结论。

## 1 指派问题描述与建模

典型的指派问题是把  $k$  项任务分配给  $n$  个执行者来完成。由于任务性质和每个执行者的实操能力

**作者简介:** 张翔(1991-), 男, 硕士, 助教, 主要研究方向: 智能信息服务系统; 朱范炳(1990-), 男, 硕士, 助教, 主要研究方向: 智能计算与复杂网络理论。

收稿日期: 2022-03-21

哈尔滨工业大学主办 ◆ 专题设计与应用

各有不同,面对不同任务所花费的成本和取得的收益  $c_{ij}$  也有所不同,一般要求分配方案实现最小的成本和最大的收益。本文将  $n$  个搜救任务区域指派给  $n$  个移动机器人,要求一个机器人只搜索一个区域,一个搜救区域只由一个机器人搜索,这是一类标准的指派问题。

对救援问题,最大的时效性是首要考虑的目标,  $f_{cost}$  表示指派问题的成本函数,一般要求取得最小成本,因此可以建立静态单目标的救援机器人任务指派模型,推导得出的数学公式为<sup>[10]</sup>:

$$f_{cost} = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (1)$$

其中,  $x_{ij} = 1$  表示安排第  $i$  个机器人搜索第  $j$  个区域;  $x_{ij} = 0$  表示不安排第  $i$  个机器人搜索第  $j$  个区域;  $c_{ij}$  为指派问题的成本系数矩阵。

## 2 人工蜂群算法原理

ABC 算法的基本要素包括蜂群、蜜源和蜜源适应度,将蜂群分为采蜜蜂、观察蜂和侦察蜂三种。ABC 算法的关键问题是适应度函数设计和解搜索策略的选择,一般通过较大的适应度值引导算法向全局最优进化,适应度大的食物源对应解的质量好。对于最大值优化问题,可用待优化问题的目标函数  $f$  代表适应度函数  $fit$ ; 对于最小值优化问题,适应度函数的数学定义式可写为:

$$fit = \begin{cases} \frac{1}{1+f} & f \geq 0 \\ 1 + abs(f) & f < 0 \end{cases} \quad (2)$$

ABC 算法的初始化阶段,设置最大迭代次数蜂群中蜜蜂数量的二分之一和食物源数量相等,且所有蜜蜂都是侦察蜂模式。研究中,随机产生  $n$  个解并计算其适应度,将适应度按由大到小的顺序排列,前半作为采蜜蜂,后半作为观察蜂和侦察蜂。此处需要用到的公式为:

$$x_{id} = x_{idmin} + rand(0,1)(x_{idmax} - x_{idmin}) \quad (3)$$

对于任一解  $x_i$  的任一分量  $x_{id}(d = 1, 2, \dots, D)$  都进行初始化,  $x_{idmin}$  代表可行解空间分量的最小值,  $x_{idmax}$  代表可行解空间分量的最大值。

接下来,对各研究阶段拟展开阐释分述如下。

(1) 采蜜蜂搜索阶段:采蜜蜂在初始阶段的蜜源附近,通过式(4)搜索产生一个新解,作为候选蜜源进行开采:

$$v_{id} = x_{id} + rand(-1,1)(x_{id} - x_{jd}) \quad (4)$$

其中,  $j \in \{1, 2, \dots, N\}, j \neq i$  表示在  $N$  个蜜源

中随机选取一个不同于  $x_i$  的蜜源。计算新解的适应度  $fit_i$  并进行适应度大小评价,采用贪心算法在  $v_i$  和  $x_i$  中选择。

(2) 观察蜂跟随阶段:所有采蜜蜂完成搜索后,采蜜蜂会把解的信息及适应度分享给观察蜂。观察蜂通过选择概率  $P_i$  决定每只采蜜蜂被跟随的概率,对此可表示为:

$$P_i = \frac{fit_i}{\sum_{k=1}^N fit_k} \quad (5)$$

或者

$$P_i = \frac{fit_i}{\max(fit_k)} \quad (6)$$

若新解的适应度比之前的好,观察蜂会将其用新解更新;反之,观察蜂会将其保留,同时解的迭代搜索次数加 1。

(3) 侦察蜂阶段:如果某一食物源在被搜索可重复开采次数后仍未做更新,相应的采蜜蜂和观察蜂则会放弃该蜜源,转换为侦察蜂模式,按式(3)随机搜索,寻找一个新的蜜源代替被舍弃的蜜源。接下来将返回到采蜜蜂的搜索阶段,3 种蜜蜂依次工作,重复循环搜索,最终找到待优化问题的最优解。

## 3 求解指派问题的改进人工蜂群算法

本文提出的改进人工蜂群算法是在标准人工蜂群算法原理的基础上,在生成食物源时采用离散数据编码的形式,提出了一种列状态移动交换的方法,保证了生成的候选食物源对应解的可行性。

### 3.1 食物源位置的编码

ABC 算法中每一个食物源代表优化问题的一个可行解。本文针对移动救援机器人任务指派问题的特点进行编码,设计离散的 IABC 算法。设有  $n$  个待救援区域需要分配给  $n$  个移动机器人去搜索,任一食物源的位置  $x_s$  是一个  $n \times n$  的矩阵,代表一种指派方案,排序形式如式(7)所示:

$$eg: x[ :, :, s ] = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \leftrightarrow \begin{cases} x_{11} = 1 \\ x_{24} = 1 \\ x_{33} = 1 \\ x_{42} = 1 \\ x_{55} = 1 \end{cases} \leftrightarrow \{1 \ 4 \ 3 \ 2 \ 5\} \quad (7)$$

矩阵  $x_s$  的行标表示机器人编号,列标表示救援区域编号;  $x_s$  是一个经过初等变换的单位对角矩

阵,其特征是任意一行和任意一列只有一个元素“1”,其余位置均为“0”。 $x_{ij} = 1$  表示指派第  $i$  个机器人去搜索第  $j$  个区域。

### 3.2 食物源位置更新方式

ABC 算法在优化连续型数值函数时,采蜜蜂和观察蜂都是按照公式(4)来更新食物源位置。本文中 IABC 算法食物源  $x_s$  是一个特殊矩阵,为了满足指派问题解决方案的要求,矩阵元素只包含 0 和 1。为了保证更新解的离散性,以及在指派问题模型上的可行性,采用基于矩阵列交换的列状态转移方法。以 6 个救援区域搜索任务为例来说明生成候选解的列状态转移方法,如式(8)所示:

$$x_i: \{1 \ 3 \ 2 \ 5 \ 4\} \rightarrow v_i: \left( \begin{array}{l} \{3 \ 1 \ 2 \ 5 \ 4\} \\ \{2 \ 3 \ 1 \ 5 \ 4\} \\ \{5 \ 3 \ 2 \ 1 \ 4\} \\ \{4 \ 3 \ 2 \ 5 \ 1\} \\ \{1 \ 2 \ 3 \ 5 \ 4\} \\ \{1 \ 5 \ 2 \ 3 \ 4\} \\ \{1 \ 4 \ 2 \ 5 \ 3\} \\ \{1 \ 3 \ 5 \ 2 \ 4\} \\ \{1 \ 3 \ 4 \ 5 \ 2\} \\ \{1 \ 3 \ 2 \ 4 \ 5\} \end{array} \right) \quad (8)$$

其中,数字是任务编号的排序,即可行解矩阵列的排序,  $x_i$  表示当前食物源,  $v_i$  表示更新得到的候选解。随机选择一个位置的列,令其与不同位置的列进行转移。每一次的列状态转移都能得到新的可行解,二次转移是指在一次转移的基础上,按照一次转移的方式生成新的可行解,二次转移能够增加食物源的多样性。对于高维指派问题,列状态转移更新可行解时可以增加随机选择列位置的个数,每个位置的状态转移方法和单个列位置交换方法相同。

### 3.3 适应度函数设计

ABC 算法中用适应度值评价食物源质量,即对应解的优劣;一般根据较大适应度原则引导算法收敛。在救援机器人任务指派问题中,要求一个指派方案完成搜索任务的时效最高,即花费的时间成本目标函数  $f_{cost}$  值最小。因此,设计 IABC 求解救援机器人任务指派问题的适应度函数具体如下:

$$fit = \frac{1}{1 + f_{cost}} \quad (9)$$

## 4 实验及结果分析

本文采用 10 维和 22 维任务的成本矩阵  $C$  模拟

救援机器人搜索任务分配的时效成本矩阵,进行 IABC 算法验证。

### 4.1 实验一

实验一选用 10 维的指派问题,成本矩阵的数学表达式为:

$$C = \begin{pmatrix} 204 & 230 & 106 & 102 & 119 & 216 & 106 & 218 & 161 & 149 \\ 140 & 120 & 199 & 102 & 243 & 155 & 124 & 183 & 115 & 144 \\ 114 & 163 & 102 & 190 & 141 & 214 & 228 & 110 & 122 & 149 \\ 163 & 228 & 102 & 156 & 221 & 207 & 114 & 249 & 184 & 168 \\ 223 & 208 & 114 & 194 & 172 & 172 & 167 & 241 & 128 & 237 \\ 108 & 234 & 236 & 117 & 221 & 238 & 178 & 197 & 238 & 154 \\ 222 & 158 & 106 & 167 & 160 & 110 & 177 & 149 & 229 & 217 \\ 126 & 179 & 103 & 117 & 110 & 225 & 170 & 219 & 198 & 210 \\ 228 & 172 & 101 & 206 & 170 & 142 & 126 & 157 & 111 & 117 \\ 201 & 197 & 147 & 229 & 111 & 109 & 175 & 179 & 120 & 238 \end{pmatrix} \quad (10)$$

这是一个匈牙利算法收敛的矩阵。本实验中分别用匈牙利算法和 IABC 算法对 10 维指派问题求解,IABC 算法参数设置如下:蜂群数量 40,食物源数量 20,最大迭代次数 50。运行 IABC 算法 30 次求取平均值并记录时间,迭代寻优收敛过程如图 1 所示,数据结果见表 1。

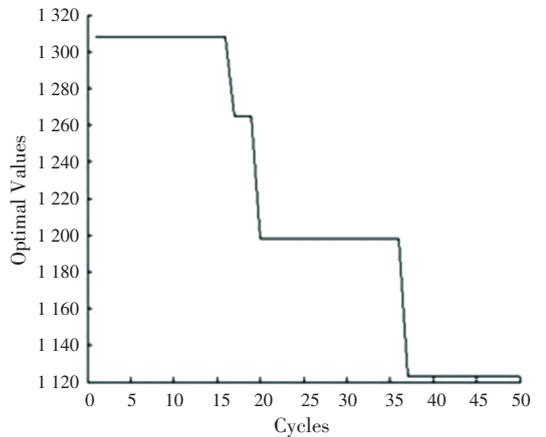


图 1 实验一中 IABC 算法迭代收敛过程

Fig. 1 IABC algorithm iterative convergence process in the first test

表 1 实验一中 3 种算法所得结果数据对比

Tab. 1 Comparison of the results obtained by the three algorithms in the first test

算法	最小值	最大值	用时/s
匈牙利算法	1 123	1 123	0.059 5
蚁群算法	1 123	1 123	0.895 3
IABC	1 123	1 123	0.081 0

### 4.2 实验二

实验二选用 22 维的指派问题,成本矩阵的数学表达式为:

$$C = \begin{pmatrix} 4 & 2 & 6 & 2 & 9 & 9 & 8 & 5 & 6 & 8 & 6 & 9 & 7 & 1 & 9 & 9 & 9 & 5 & 2 & 8 & 4 & 2 \\ 5 & 9 & 6 & 3 & 8 & 9 & 6 & 4 & 6 & 6 & 6 & 6 & 1 & 2 & 2 & 9 & 6 & 4 & 1 & 5 & 3 & 2 \\ 1 & 6 & 2 & 8 & 4 & 3 & 1 & 2 & 9 & 2 & 7 & 4 & 6 & 9 & 3 & 6 & 9 & 4 & 8 & 3 & 4 & 9 \\ 6 & 4 & 9 & 5 & 4 & 4 & 4 & 1 & 2 & 1 & 3 & 3 & 5 & 6 & 5 & 5 & 1 & 8 & 4 & 2 & 8 & 6 \\ 7 & 7 & 2 & 6 & 1 & 3 & 1 & 4 & 8 & 7 & 2 & 3 & 1 & 5 & 9 & 8 & 9 & 5 & 1 & 7 & 3 & 6 \\ 9 & 6 & 2 & 8 & 5 & 4 & 8 & 4 & 8 & 1 & 7 & 5 & 4 & 2 & 1 & 9 & 9 & 4 & 8 & 4 & 5 & 2 \\ 3 & 1 & 9 & 8 & 7 & 3 & 9 & 5 & 3 & 6 & 7 & 1 & 2 & 5 & 7 & 1 & 9 & 4 & 2 & 4 & 5 & 3 \\ 2 & 2 & 5 & 8 & 9 & 8 & 5 & 4 & 7 & 8 & 9 & 6 & 7 & 9 & 1 & 9 & 7 & 6 & 1 & 5 & 4 & 1 \\ 5 & 3 & 5 & 2 & 9 & 4 & 8 & 9 & 8 & 4 & 3 & 4 & 4 & 1 & 8 & 5 & 1 & 4 & 1 & 4 & 2 & 5 \\ 8 & 8 & 2 & 6 & 1 & 2 & 3 & 2 & 9 & 8 & 3 & 3 & 9 & 4 & 7 & 7 & 1 & 6 & 7 & 7 & 3 & 5 \\ 4 & 1 & 6 & 2 & 8 & 1 & 7 & 5 & 7 & 6 & 3 & 3 & 5 & 1 & 7 & 1 & 7 & 6 & 4 & 4 & 2 & 6 \\ 5 & 4 & 6 & 4 & 1 & 7 & 3 & 9 & 1 & 8 & 2 & 9 & 9 & 5 & 9 & 4 & 5 & 2 & 3 & 9 & 1 & 5 \\ 8 & 7 & 9 & 7 & 9 & 5 & 5 & 1 & 5 & 9 & 2 & 4 & 1 & 9 & 6 & 6 & 9 & 3 & 3 & 9 & 2 & 2 \\ 2 & 9 & 7 & 7 & 8 & 3 & 2 & 3 & 3 & 5 & 8 & 9 & 9 & 5 & 6 & 8 & 9 & 2 & 8 & 4 & 7 & 8 \\ 7 & 2 & 2 & 3 & 5 & 4 & 6 & 6 & 2 & 1 & 7 & 6 & 8 & 8 & 2 & 2 & 2 & 8 & 9 & 3 & 8 & 7 \\ 8 & 5 & 2 & 5 & 6 & 4 & 2 & 1 & 1 & 2 & 2 & 3 & 4 & 6 & 3 & 6 & 4 & 8 & 6 & 2 & 7 & 7 \\ 7 & 5 & 4 & 5 & 1 & 7 & 4 & 2 & 1 & 9 & 2 & 7 & 1 & 6 & 6 & 3 & 7 & 9 & 2 & 8 & 1 & 4 \\ 7 & 2 & 2 & 5 & 6 & 6 & 7 & 9 & 6 & 4 & 5 & 6 & 4 & 3 & 7 & 5 & 3 & 4 & 2 & 6 & 2 & 7 \\ 4 & 4 & 6 & 2 & 6 & 2 & 1 & 9 & 8 & 4 & 7 & 8 & 8 & 4 & 9 & 5 & 5 & 3 & 4 & 2 & 4 & 6 \\ 6 & 4 & 6 & 1 & 3 & 7 & 8 & 5 & 2 & 4 & 7 & 6 & 6 & 9 & 3 & 1 & 8 & 8 & 6 & 1 & 5 & 1 \\ 5 & 9 & 9 & 9 & 2 & 1 & 2 & 6 & 2 & 8 & 4 & 1 & 2 & 2 & 2 & 3 & 9 & 1 & 8 & 9 & 8 \\ 4 & 2 & 8 & 9 & 7 & 9 & 7 & 5 & 8 & 7 & 1 & 1 & 7 & 9 & 7 & 1 & 4 & 7 & 5 & 6 & 1 & 5 \end{pmatrix} \quad (11)$$

这个矩阵对匈牙利算法不收敛。本实验中用IABC算法对22维指派问题求解,IABC算法参数设置如下:蜂群数量60,食物源数量30,最大迭代次数100。运行IABC算法30次求取平均值并记录时间,迭代寻优收敛过程如图2所示,数据结果记录见表2。

算法对大规模指派问题不收敛;蚁群算法也是较早应用在任务指派问题中的群体智能算法,但该算法设置参数多,计算量大。本文以救援机器人搜索区域分配问题为背景,建立以时效性能为目标的指派数学模型;提出改进的人工蜂群算法(IABC)求解该模型;采用2个不同规模的指派算例模拟救援机器人任务指派问题。仿真结果表明:IABC算法在求解指派问题时,设置参数少,通用性强,收敛速度快,算法稳定性好,是一种优秀的算法。

参考文献

- [1] 顾大权,左莉,侯太平,等. “匈牙利法”存在的问题及改进方法[J]. 微机发展,2003,13(04):76-78.
- [2] 李元左,杨晓段. 指派问题的改进算法[J]. 价值工程,2015,7(02):303-304.
- [3] 张新明,王豆豆,陈海燕,等. 强化最优和最差狼的郊狼优化算法及其二次指派问题应用[J]. 计算机应用,2019,39(10):2985-2991.
- [4] 田倩南,李昆鹏,李文莉,等. 机场任务指派问题的优化方案研究[J]. 运筹与管理,2019,28(11):1-8.
- [5] 高原,李仁传,张合勇,等. 考虑目标差异的多目标指派问题研究[J]. 海军工程大学学报,2020,32(05):102-106,112.
- [6] 蒋洪迅,马仁义. 面向靠桥率及道口冲突率的航班-机位指派问题优化模型及其启发式算法研究[J]. 系统科学与数学,2021,41(01):75-98.
- [7] SHUNMUGAPRIYA P, KANMANI S. A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid) [J]. Swarm and Evolutionary Computation, 2017, 36: 27-36.
- [8] 沈奥,周树道,王敏,等. 旋翼无人机协同任务指派问题研究与算法改进[J]. 计算机测量与控制,2020,28(09):182-186.
- [9] 张长泽,李引珍,尹胜男,等. 对具有二部图特点的二次指派问题的研究[J]. 制造业自动化,2021,43(02):63-70.

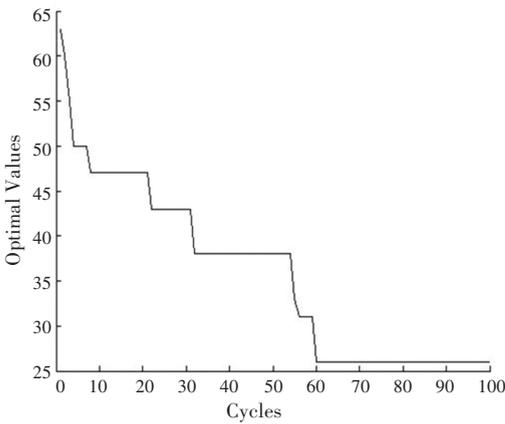


图2 实验二中IABC算法迭代收敛过程

Fig. 2 IABC algorithm iterative convergence process in the second test

表2 实验二中3种算法所得结果数据对比

Tab. 2 Comparison of the results obtained by the three algorithms in the second test

算法	最小值	最大值	用时/s
改进的匈牙利算法	27	30	0.168 9
蚁群算法	26	27	4.283 2
IABC	26	26	0.266 8

5 结束语

匈牙利算法是任务指派问题的标准算法,但该