

多编组协同任务分配模型及DLS-QGA算法求解

万路军^{1a,1b}, 姚佩阳^{1a}, 周翔翔², 税冬东^{1a}

(1. 空军工程大学 a. 信息与导航学院, b. 空管领航学院, 西安 710077; 2. 95616 部队, 成都 611531)

摘要: 为解决多智能体编组协同任务分配问题, 定义任务、智能体编组和相关的分配过程变量, 建立以最高任务执行效率为目标的数学模型. 在问题模型中设计考虑资源损耗的编组资源能力更新机制, 提出用于求解该模型的动态列表规划和量子遗传算法的混合任务分配算法, 使用动态列表规划选择处理的任務, 利用量子遗传算法为选定任务分配最合适编组. 最后通过算例表明, 所提出的方法在解决时序逻辑任务分配时能够得到更优更稳定的方案.

关键词: 智能体编组; 任务分配; 动态列表规划; 量子遗传算法

中图分类号: TP391; TP237.5

文献标志码: A

Cooperative task allocation methods in multiple groups using DLS-QGA

WAN Lu-jun^{1a,1b}, YAO Pei-yang^{1a}, ZHOU Xiang-xiang², SHUI Dong-dong^{1a}

(1a. College of Information and Navigation, 1b. College of Air Control and Navigation, Air Force Engineering University, Xi'an 710077, China; 2. Unit 95616 of PLA, Chengdu 611531, China. Correspondent: WAN Lu-jun, E-mail: pandawlj@126.com)

Abstract: To solve the cooperative task allocation of multiple agent groups, the task, agents group and process variables are defined. The mathematical model in pursuit of maximal task execute efficiency is established. The wastage of group resource capability is introduced in the process of problem modeling. A hybrid approach to the problem based on dynamic list scheduling(DLS) and the quantum genetic algorithm(QGA) is proposed. In the DLS-QGA algorithm, DLS is used to select the task needed to dispose, then QGA is used to select the best group for the selected task. Finally, the superiority and applicability of this approach are illuminated by the simulation of campaign assumption, and the scheme can be better and more steadily in task allocation with time-logic.

Key words: agent groups; task allocation; dynamic list scheduling; quantum genetic algorithm

0 引言

现代战场上具有信息系统的空中作战智能体(ACA)应用得越来越广泛, 但单个ACA所能发挥的作战功能有限, 需要多个ACA形成混合集群执行任务, 以发挥更佳的综合作战效能^[1-3]. 混合集群执行作战任务时, 通常根据任务属性特征形成多个编组, 将编组视为一个整体, 各编组分别执行属性不相一致的一项或多项任务. 所要执行的复杂序列任务间具有一定的时序关系, 多编组协同任务分配便是将编组合理地配置到存在时序逻辑关系的任务中, 这属于时间扩展型的任务分配^[4-5], 是NP完全问题. 对于该类型问题, 若直接使用智能优化算法求解则无法保证满足任务时序关系约束; 精确方法面向较大规模的任务集和编组集时, 无法在有效时间内实现优化求解; 启

发式方法可以避免上述不足, 是解决该类问题的有效途径, 动态列表规划(DLS)^[6]作为一种最常用的启发式策略, 具有求解效率高、可以获得最优解或满意解的优点. 在基于DLS的扩展启发式算法中, 多维动态列表规划(MDLS)算法^[7]和多优先级列表动态规划(MPLDS)算法^[8]最为常用. 前者存在优先权函数设定不够合理的问题, 两种算法的局部搜索策略都是基于贪心策略的, 无法保证以稳定的高概率寻到最优任务分配方案. 针对MDLS算法和MPLDS算法存在的不足, 文献[9]设计了DLS和遗传算法(GA)的分配策略, 使用全局搜索GA为选定的任务分配最佳的平台(编组), 提高了任务分配搜索结果趋近于最优的概率, 但传统GA寻优时收敛速度慢, 使得该方法每次局部任务编组匹配耗时较长. 量子遗传算法(QGA)^[10]

收稿日期: 2013-07-01; 修回日期: 2013-09-17.

基金项目: 国家自然科学基金项目(61273048).

作者简介: 万路军(1986—), 男, 博士, 讲师, 从事指挥控制理论与方法的研究; 姚佩阳(1960—), 男, 教授, 博士生导师, 从事指挥控制理论与技术等研究.

是在遗传算法中引入量子计算的概念,当应用于0/1背包问题时,QGA相比于GA在收敛速度和全局搜索能力上具有明显的优势^[11-12].因此,围绕多编组任务分配问题,需要寻找一种新的满足任务时序关系的全局优化机制,既能寻到最优分配结果,又可以相对减少任务与编组匹配时耗费的时间.

基于以上分析,本文首先建立空中多作战智能体编组协同任务分配数学模型,然后针对所建模型具有时间扩展的特点,设计以DLS算法为主的基本求解框架,嵌套QGA为选定任务选择最优编组的多编组协同任务分配策略.

1 多编组协同任务分配的问题描述

1.1 问题背景和基本概念

由多个属性基本一致的ACA形成编组,考虑编组差异即为不同ACA的位置、距离、移动速度、功能和执行能力水平等差异,多编组协同任务分配(MGCTA)便是对多个序列任务进行快速任务ACA编组匹配.首先对任务和智能体编组进行描述,给出任务和ACA编组有效匹配为目的的属性.

任务(T):由使命任务分解得到所需要执行的任务序列.记包含 N 个任务的集合为 $T = \{T_i\} (i = 1, 2, \dots, N)$,每个任务 T_i 有如下属性:任务处理时间 $dt_i = ft_i - st_i$,其中 ft_i 为任务 T_i 的结束时间, st_i 为任务 T_i 的开始时间;任务 T_i 的地理坐标位置 $TP_i = (x_i, y_i)$;任务的资源需求向量 $R_i = \{R_{il}\} (l = 1, 2, \dots, L)$, R_{il} 为成功处理任务 T_i 所需的第 l 种类型的资源能力值, L 为资源能力类型的数量.

ACA编组(AG):有人和无人作战智能体组成的编组,是资源能力的载体.记包含 M 个编组的ACA编组集为 $AG = \{AG_m\} (m = 1, 2, \dots, M)$,每个编组 AG_m 具有如下属性:编组 AG_m 的平均移动速率为 v_m ; AG_m 具备的初始资源能力矢量 $GR_m = \{GR_{ml}\}$, GR_{ml} 为 AG_m 所具备的第 l 种类型的资源能力值;ACA编组地理位置变量 $(x_{AG}^{(m)}, y_{AG}^{(m)})$ 为在战场各作战任务区域进行迁移并执行任务的编组,在整个任务过程中不断变化的地理位置.

1.2 任务分配过程的变量定义

任务分配过程的变量定义如下:

1) 编组-任务的分配变量 φ_{im} .若 AG_m 分配给 T_i ,则 $\varphi_{im} = 1$,否则 $\varphi_{im} = 0$.

2) 编组在任务间的迁移变量 x_{ijm} .若 AG_m 在执行 T_i 后分配给 T_j ,则 $x_{ijm} = 1$,否则 $x_{ijm} = 0$.编组在同一任务间不存在迁移关系,当 $i = j$ 时, $x_{ijm} = 0$.

3) 任务执行序列变量 e_{ij} .若 T_i 必须在 T_j 开始前完成,则 $e_{ij} = 1$,否则 $e_{ij} = 0$.

4) 任务集 T 的持续时间是从小于第1个执行任务的起始时间到最后1个任务的结束时间.若 T 中第1个任务的起始时间为 $\min_{T_i \in T} st_i$,最后1个任务的结束时间为 $\max_{T_i \in T} ft_i$,则所有任务处理结束时间 $Tft(T, G_T)$ 为

$$Tft(T, G_T) = \max_{T_i \in T} ft_i - \min_{T_i \in T} st_i.$$

实际计算时,通常假设 $\min_{T_i \in T} st_i$ 从时间轴0时刻开始,这样 $Tft(T, G_T) = \max_{T_i \in T} ft_i$.

5) 任务 T_i 对 l 种类型的资源需求满足度 $Ta(i, l)$ 可以利用其获得的资源能力 \tilde{R}_{il} 与资源需求 R_{il} 的比值表示,即

$$Ta(i, l) = \begin{cases} 1, & \tilde{R}_{il} \geq R_{il}; \\ \tilde{R}_{il}/R_{il}, & \tilde{R}_{il} < R_{il}; \end{cases}$$

$$\tilde{R}_{il} = \sum_{m=1}^M \varphi_{im} \cdot R_{ml} \quad (1)$$

当 T_i 获取的资源能力不小于其需求时,资源需求满足度为1;当 T_i 获取的资源能力小于其需求时,资源需求满足度线性减小.记 $A(i) = \{k | R_{ik} > 0\}$ 为 T_i 所需资源类型的集合, $|A(i)|$ 为 T_i 所需资源类型的能力值,则任务 T_i 的任务资源满足度 Ta_i 为 T_i 对于所有类型资源需求满足度的均值,即

$$Ta_i = \sum_{k \in A(i)} Ta(i, k) / |A(i)| \quad (2)$$

6) 任务执行质量 Tq_i 与 Ta_i 有直接函数关系, Tq_i 是对任务 T_i 完成概率的度量. $Tq_i \in (0, 1)$, Tq_i 越大表明任务 T_i 的完成概率越大, Tq_i 大小由 Ta_i 决定,变化趋势由中间阈值 ϑ 控制,并呈现如下函数变化特点:

① 当 Ta_i 较小时, $Ta_i < \vartheta$, Tq_i 随着 Ta_i 的提升而提高的速度较慢,当 $Ta_i \rightarrow 0$ 时, $Tq_i \rightarrow 0$;

② 当 Ta_i 较大时, $Ta_i > \vartheta$, Tq_i 随着 Ta_i 的提升而缓慢提高,当 $Ta_i \rightarrow 1$ 时, $Tq_i \rightarrow 1$;

③ 当 Ta_i 处于中间阈值 ϑ 附近时, Tq_i 随着 Ta_i 的提升而快速提高.

综上 Tq_i 随 Ta_i 的变化特点,设定 $\vartheta = 0.5$,采用下式近似表示 Tq_i 与 Ta_i 之间的函数关系^[13]:

$$Tq_i = [1 - \exp(8 \cdot (1 - 2 \cdot Ta_i))]^{-1} \quad (3)$$

1.3 编组资源能力的更新模型

记 AG_m 执行任务前具有资源能力矢量 $P_m = \{R_{ml}\}$,在实际的空中任务执行中,随着任务的依次进行,编组的资源能力随执行任务数量和强度的增加逐渐递减.因此,在编组执行完某一任务后,应对其资源能力进行更新^[9]如下:

$$GR_{ml}^{\text{renewed}} = GR_{ml} \left(1 - \omega_l \frac{GR_{ml}^{\text{real}}}{GR_{ml}}\right), l \in [1, L] \quad (4)$$

其中: ω_l 为第 l 种资源能力的损耗系数,设定 ω_l 为第

l 种资源类型的一种自身属性, 与编组本身和执行的任务无关; GR_{ml} 为 AG_m 在执行某一任务前所具备的第 l 种类型的资源能力值; GR_{ml}^{renewed} 为其执行该任务后第 l 种类型的资源能力值; GR_{ml}^{real} 为 AG_m 在执行该任务过程中第 l 种类型的资源能力的实际消耗值. 任务完成后对编组的资源能力进行更新, 更新后的编组 AG_m 资源能力值为

$$GR_m^{\text{renewed}} = \{GR_{m1}^{\text{renewed}}, GR_{m2}^{\text{renewed}}, \dots, GR_{mL}^{\text{renewed}}\}.$$

任务执行过程中, AG_m 变化的资源能力值为

$$GR'_m = \{GR'_{m1}, GR'_{m2}, \dots, GR'_{mL}\}.$$

1.4 数学模型的分析 and 建立

1.4.1 协同任务分配的约束分析

1) 编组任务的分配约束.

若编组 AG_m 与任务 T_i 存在分配关系, 则 AG_m 在执行完某一任务 $T_{i'}$ 后被分配执行 T_i , 同时 AG_m 在执行完 T_i 后将会被分配执行下一任务 T_j , 满足

$$\sum_{i'=0}^N x_{i'im} - \varphi_{im} = 0, \\ i', i = 1, 2, \dots, N, m = 1, 2, \dots, M; \quad (5)$$

$$\sum_{j=0}^N x_{ijm} - \varphi_{im} = 0, \\ i, j = 1, 2, \dots, N, m = 1, 2, \dots, M. \quad (6)$$

2) 任务的资源能力需求约束.

如果执行任务 T_i 的资源能力过低, 则任务执行效果会远低于期望效果, 因此要保证每个任务的资源满足度 Ta_i 高于某一下限阈值 ϕ , 即

$$Ta_i \geq \phi, i = 1, 2, \dots, N. \quad (7)$$

其中 ϕ 由领域专家结合任务特点给出, 作为模型的输入参数.

3) 任务间执行顺序关系约束.

以 x_{ijm} 的值判断编组 AG_m 在任务 T_i 与 T_j 之间存在的关系, 即每个任务的处理必须在所有使能(前导)任务都处理完成之后. 若 $x_{ijm} = 1$, 则任务 T_j 的执行开始必须在任务 T_i 完成之后, 满足

$$st_j \geq st_i + dt_i, i, j = 1, 2, \dots, N. \quad (8)$$

设定任务的执行需要协作的所有编组都到达任务发生区域之后方能开始, 那么先到达编组需要等待, 因此, 任务的执行开始时间不小于分配给其的所有编组的到达时间, 即

$$st_j \geq \max_{i \in \{k | \varphi_{km}=1, k \in [1, N]\}} \left(st_i + dt_i + x_{ijm} \frac{d_{ij}}{v_m} \right), \quad (9)$$

其中 d_{ij} 为 T_i 与 T_j 之间的空间距离, 满足

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}.$$

4) 编组单任务处理约束.

编组 AG_m 在任意时刻只处理一个任务, 满足

$$\sum_{j=1}^N x_{ijm} \leq 1, i, j = 1, 2, \dots, N. \quad (10)$$

1.4.2 目标函数

多编组协作执行序列任务要综合考虑时效性和有效性两个方面的需求, 多编组协同任务分配一是需要考虑整个任务集的任务执行质量, 二是考虑整体任务的完成时间. 以整体任务执行质量和任务完成时间的比值为目标函数, 记为任务执行效率 $Te(T, G_T)$, 则

$$Te(T, G_T) = \prod_{T_i \in T} Tq_i [Tft(T, G_T)]^{-1}. \quad (11)$$

1.4.3 数学模型的建立

MGCTA 问题的数学模型描述如下:

$$\max Te(T, G_T) = \prod_{T_i \in T} Tq_i [Tft(T, G_T)]^{-1}.$$

$$\text{s.t. } \sum_{i'=0}^N x_{i'im} - \varphi_{im} = 0,$$

$$i', i = 1, 2, \dots, N, m = 1, 2, \dots, M;$$

$$\sum_{j=0}^N x_{ijm} - \varphi_{im} = 0,$$

$$i, j = 1, 2, \dots, N, m = 1, 2, \dots, M;$$

$$Ta_i \geq \phi, i = 1, 2, \dots, N;$$

$$st_j \geq \max_{i \in \{k | \varphi_{km}=1\}} \left(st_i + dt_i + x_{ijm} \frac{d_{ij}}{v_m} \right),$$

$$k = 1, 2, \dots, N;$$

$$\sum_{j=1}^N x_{ijm} \leq 1, i, j = 1, 2, \dots, N,$$

$$x_{i'im}, x_{ijm}, \varphi_{im} \in \{0, 1\};$$

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}.$$

2 多编组协同任务分配问题描述

2.1 算法思想及流程

MGCTA 问题的数学描述是一个混二元线性规划问题, 模型中包含编组-任务的分配变量 φ_{im} 和编组在任务间的转移变量 x_{ijm} 两个变量, 且两个变量之间具有复杂的关联关系.

采用列表规划方法 (DLS) 求解该问题的步骤如下.

Step 1: 计算所有任务的优先权系数;

Step 2: 在前导任务已完成任务中, 选择优先权系数最大的一个任务;

Step 3: 为该任务分配一个或多个编组, 以满足任务的功能能力需求;

Step4: 判断是否所有任务已完成分配, 如果是则终止算法, 否则转入 Step 1.

可见, DLS 通过多步局部匹配合并构造完整的分配方案. 本文基于 DLS 的思想, 提出了基于 DLS-QGA 的混合编组任务分配算法, 算法流程如图 1 所示. 该算法主要包含两个关键环节: 一是使用 DLS 在待选任务集中选择一个当前需要处理的任務; 二是利用 QGA 为选定任务分配最佳编组.

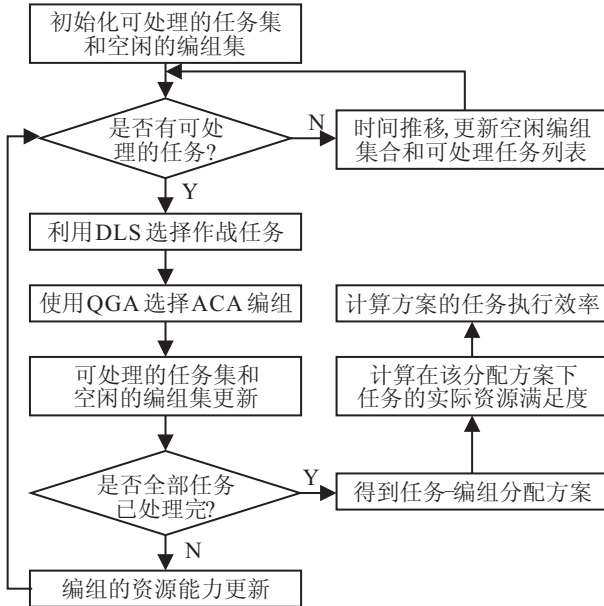


图 1 基于 DLS-QGA 的任务分配算法流程

2.2 作战任务选择

当一个任务 T_i 的所有直接前导任务都已处理完成时, 该任务 T_i 便进入可处理的任務集 T_{ready} 中, 在 T_{ready} 中根据任务的优先级系数 p_T 选择任务进行编组分配. 计算任务优先级系数 p_T 的依据是任务本身的预计执行时间与任务之间的序列关系^[8], 任务的优先级系数越大, 表示该任务的优先级越高, 该任务越需要优先被选择分配编组. 任务优先级的最直接影响因素是任务的预计处理时间, 同时也要考虑任务的后续任务影响. 任务优先级系数定义如下:

$$p_T(i) = dt_i + \frac{\sum_{j \in OUT(i)} p_T(j)}{\max_{j \in OUT(i)} p_T(j)} \quad (12)$$

其中: $OUT(i)$ 为 T_i 的直接后续任务集, dt_i 为任务 T_i 的预计处理时间.

只要任务的预计处理时间与任务之间的序列关系确定, 即可确定出每个任务的任务优先级系数 p_T , 且 p_T 在整个任务分配过程中一直保持不变.

2.3 基于 QGA 的编组资源选择

从当前处于空闲状态的编组集 AG_{free} 中, 为选定的任务 T_C ($C \in \{1, 2, \dots, N\}$) 选择最佳编组, 其

实质是多维 0-1 背包问题. 由于 QGA 具有种群规模小、收敛速度快和全局搜索能力强等特点, 且在 0-1 背包问题解决上具有显著优势, 采用 QGA 为选定任务分配编组.

2.3.1 染色体编码方式

对于某个选定的任务 T_C , 若 N_{free} 个编组处于空闲状态, 则一个染色体包含 N_{free} 位置量子位, 编码形式为

$$s = \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_{N_{free}} \\ \beta_1 & \beta_2 & \dots & \beta_{N_{free}} \end{bmatrix} \quad (13)$$

其中: α_k 和 β_k 分别为第 k 位置量子位状态 0) 和 1) 的概率幅, $|\alpha_k|^2$ 为对该量子位测量时得到 0 的概率, $|\beta_k|^2$ 为得到 1 的概率, α_k 和 β_k 满足归一化条件 $|\alpha_k|^2 + |\beta_k|^2 = 1, k = 1, 2, \dots; N_{free}$ 为默认的编组标号. 采用量子位编码后, QGA 中的染色体种群可以表示为

$$S(g) = \{s_1^{(g)}, s_2^{(g)}, \dots, s_{N_{size}}^{(g)}\}, \quad (14)$$

$$s_p^{(g)} = \begin{bmatrix} \alpha_{p1}^{(g)} & \alpha_{p2}^{(g)} & \dots & \alpha_{pN_{free}}^{(g)} \\ \beta_{p1}^{(g)} & \beta_{p2}^{(g)} & \dots & \beta_{pN_{free}}^{(g)} \end{bmatrix} \quad (15)$$

其中: g 为进化代数, N_{size} 为种群规模, $s_p^{(g)}$ ($p = 1, 2, \dots, N_{size}$) 为第 g 代种群的第 p 个染色体.

2.3.2 种群测量及旋转角机制

对种群 $S(g)$ 中的每个个体实施一次测量, 得到一组状态 $P(g)$, 为二进制解, 其中一个二进制解 $p_p^{(g)}$ ($p = 1, 2, \dots, N_{size}$) 由长度为 N_{free} 的二进制串组成. $p_p^{(g)}$ 中第 k 位的取值通过第 k 位置量子位的概率 $|\alpha_{pk}^{(g)}|^2$ ($k = 1, 2, \dots, N_{free}$) 确定, 有

$$p_{pk}^{(g)} = \begin{cases} 0, & |\alpha_{pk}^{(g)}|^2 < \text{rand}(0, 1); \\ 1, & |\alpha_{pk}^{(g)}|^2 \geq \text{rand}(0, 1). \end{cases} \quad (16)$$

其中: $p_{pk}^{(g)}$ 为第 g 代种群第 p 个染色体的第 k 位置量子位真实状态, 1 表示为选定任务分配标号为 k 的编组, 0 表示没有为选定任务分配编组; $\text{rand}(0, 1)$ 为随机产生的 $[0, 1]$ 之间的数.

QGA 采用量子旋转门更新策略完成种群的更新操作, 而旋转角度定义是基于查表机制的, 因此表参数选取是算法的关键. 本文采用文献[14]的查询表数值, 包括旋转角的方向和步长等参数.

2.3.3 适应度函数

由测量得到的 $P(g)$ 中的一个二进制解 $p_p^{(g)}$ 代表 AG_{free} 中的一种编组组合方式. 记编组组合为 $X^{(AG)}$, 群体规模为 N_{size} , $X_n^{(AG)}$ ($n = 1, 2, \dots, N_{size}$) 的相应编组数为 $N_{X^{(AG)}}$. 适应度函数的构造需要与编组分配的整体目标一致, 即任务执行效率最高. 因此, 考虑任务选择编组的时间优先权和编组资源的能力优先权构造适应度函数.

1) 时间优先权系数 CT 计算.

编组组合 $X_n^{(AG)}$ 开始执行任务的时间是 $X_n^{(AG)}$ 中最迟到达任务处理地点的编组到达时间, 即

$$st(X_n^{(AG)}) = \max_{m \in X_n^{(AG)}} \left(st_{x(m)} + dt_{x(m)} + \frac{d_{x(m),C}}{v_m} \right), \quad (17)$$

其中 $x(m)$ 为 AG_m 最后处理的任务. 种群中最优个体对应的任务开始时间最小, 为

$$st_{\min} = \min_{n=1,2,\dots,N_{\text{size}}} (st(X_n^{(AG)})). \quad (18)$$

对该种群中所有 $X_n^{(AG)}$ 的 $st(X_n^{(AG)})$ 进行归一化处理, 并将结果作为时间优先权系数 CT, 有

$$CT(X_n^{(AG)}) = \frac{st(X_n^{(AG)}) - st_{\min}}{\sum_{n=1}^{N_{\text{size}}} (st(X_n^{(AG)}) - st_{\min})}. \quad (19)$$

2) 编组资源能力优先权系数 AGR 计算.

编组资源能力优先权系数是考察所选择的编组应具有较小的功能能力冗余, 同时对其他候选列表中任务的资源满足程度较低. 设 $X_n^{(AG)}$ 分配给任务 T_C 的资源能力冗余程度为 $R_D(X_n^{(AG)})$, 定义为

$$R_D(X_n^{(AG)}) = \sum_{l=1}^L \sum_{m \in X_n^{(AG)}} (GR'_{ml} - R_{Cl}). \quad (20)$$

$X_n^{(AG)}$ 中的各个编组相对于 T_{ready} 中其他任务的平均资源满足度定义为

$$R_Q(X_n^{(AG)}) = \frac{1}{N_{X^{(AG)}}} \sum_{m \in X_n^{(AG)}} \sum_{j \in T_{\text{ready}} \setminus T_C} \sum_{l=1}^L \min(GR'_{ml}, R_{jl}). \quad (21)$$

种群中最优的个体所对应的 $R_D(X_n^{(AG)})$ 值和 $R_Q(X_n^{(AG)})$ 值最小, 记为

$$RS_{\min} = \min_{n \in [1, N_{\text{size}}]} (R_D(X_n^{(AG)}) + R_Q(X_n^{(AG)})). \quad (22)$$

对种群中所有 $X_n^{(AG)}$ 的 $R_D(X_n^{(AG)}) + R_Q(X_n^{(AG)})$ 进行归一化处理, 并将其作为编组资源能力优先权系数 AGR, 有

$$AGR(X_n^{(AG)}) = \frac{R_D(X_n^{(AG)}) + R_Q(X_n^{(AG)}) - RS_{\min}}{\sum_{n=1}^{N_{\text{size}}} (R_D(X_n^{(AG)}) + R_Q(X_n^{(AG)}) - RS_{\min})}. \quad (23)$$

综合式 (19) 和 (23), 构造适应度函数为

$$f(X_n^{(AG)}) = (\omega_1 \cdot CT(X_n^{(AG)}) + \omega_2 \cdot AGR(X_n^{(AG)}))^{-1}, \quad (24)$$

其中 $\omega_1 + \omega_2 = 1$, ω_1 和 ω_2 表示规划者在两个因素上的决策偏好. 适应度函数值越小, 为任务 T_C 选择的编组组合越适合于任务 T_C .

2.3.4 QGA 算法实现步骤

Step 1: 根据 N_{free} 确定量子位染色体位数, 设置合适的种群规模 N_{size} , 包括 N_{size} 个个体的种群 S 和量子位编码个体 s_p .

Step 2: 根据 S 中各个体的概率幅, 对种群 S 各个体实施一次测量构造出量子叠加态的测量态 P .

Step 3: 对测量态进行适应度评估, 保留精英个体, 并判断是否满足终止条件.

Step 4: 利用动态调整量子旋转角机制对种群中所有个体进行更新.

Step 5: 进化代数增加一次, 算法转至 Step 2 继续执行, 直至达到某种停止准则. 本文的停止准则为最大迭代次数.

3 算例分析

以空中联合突击离岛上多个目标的想定为算例, 作战任务序列如图 2 所示. 所需要执行的作战任务数量 $N = 20$, 能够使用的编组数量为 $M = 24$. 任务的属性和编组资源的属性如表 1 和表 2 所示.

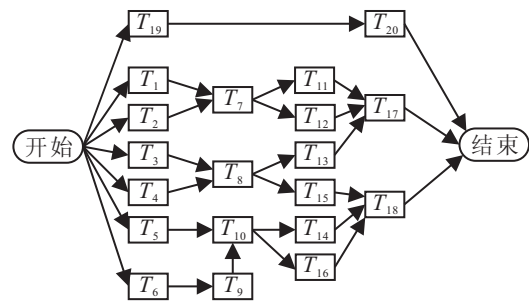


图 2 任务之间的序列关系

表 1 作战任务的属性

任务	资源需求向量						处理时间	位置	
	R_1	R_2	R_3	R_4	R_5	R_6		x	y
T_1	5	0	0	4	4	10	15	60	60
T_2	2	0	0	4	4	6	10	40	70
T_3	2	0	0	4	4	6	10	18	25
T_4	2	0	0	4	4	6	10	60	38
T_5	2	0	0	4	4	6	10	50	90
T_6	2	0	0	4	4	6	10	5	82
T_7	4	2	0	0	6	0	10	38	58
T_8	12	6	0	0	6	4	25	28	83
T_9	5	0	0	6	4	0	15	18	78
T_{10}	5	0	0	6	4	0	15	6	94
T_{11}	0	6	0	0	2	0	5	65	26
T_{12}	0	8	0	0	2	0	15	50	25
T_{13}	8	0	0	0	4	0	20	25	45
T_{14}	8	0	0	0	4	0	20	37	42
T_{15}	2	0	0	0	4	0	5	25	40
T_{16}	2	0	0	0	4	0	5	30	92
T_{17}	0	8	0	0	2	0	15	5	39
T_{18}	4	4	0	0	2	0	5	25	28
T_{19}	0	0	16	2	6	10	20	65	10
T_{20}	10	0	8	0	4	0	20	10	15

表2 编组资源的属性

编组	编组拥有的资源能力向量						速度
	GR ₁	GR ₂	GR ₃	GR ₄	GR ₅	GR ₆	
AG ₁	0	0	0	0	10	6	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
AG ₆	8	0	0	2	0	0	4
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
AG ₁₀	2	6	0	4	0	2	4.5
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
AG ₁₄	6	2	0	0	0	0	2.5
AG ₁₅	6	2	0	0	0	0	2.5
AG ₁₆	0	10	0	4	0	0	3.5
AG ₁₇	0	10	0	4	0	0	3.5
AG ₁₈	0	0	0	0	8	0	2.75
AG ₁₉	0	0	0	0	8	0	2.75
AG ₂₀	0	0	0	0	0	8	2.5
AG ₂₁	0	0	0	0	0	8	2.5
AG ₂₂	0	4	6	0	0	2	3
AG ₂₃	0	5	10	0	0	0	3.25
AG ₂₄	0	5	10	0	0	0	3.25

仿真结果中,基于MPLDS得到的最优协同任务分配方案,整个任务集的完成时间为107.0491 min,耗时10.1780 s;基于DLS-GA得到的最优协同任务分配方案,整个任务集的完成时间为93.3340 min,耗时43.8641 s.

DLS-QGA算法得到最优多作战智能体编组的协同任务分配方案,整个任务集的完成时间为91.0501 min,算法耗时20.4519 s. DLS-QGA和DLS-GA得到的执行方案更为紧凑,有效节省了全部任务集的处理时间.相比于DLS-GA, DLS-QGA的时间开销要小得多.图3给出了DLS-QGA与DLS-GA在解决第1局部任务到编组分配时的最佳适应度变化曲线, DLS-QGA在种群规模较小的情形下,具有更快的收敛速度.从全局看,在一定的任务执行质量下, DLS-QGA的全局寻优能力更好.对算法运行100次得到方案的平均效果如图4所示.仿真结果表明,基于DLS-QGA的协同任务分配方法在统计意义上能够得到全局更优的任务分配方案,从而提高整个任务集的执行质量,改善编组资源的利用率.

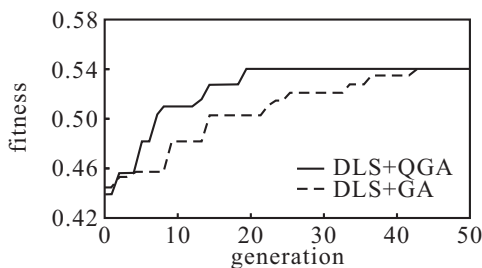


图3 DLS-QGA和DLS-GA在任务分配的最佳适应度

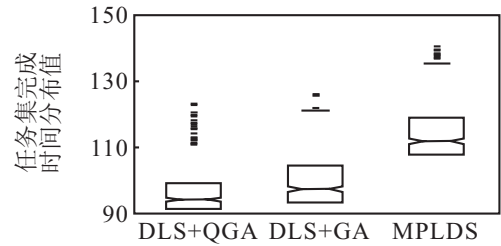


图4 3种方法得到的任务完成时间的盒须图分布

表3 案例的任务分配方案结果对比

编组	不同算法下的任务方案分配结果		
	MPLDS	DLS-GA	DLS-QGA
AG ₁	(6, 0)(8, 23.80) (14, 82.05)	(1, 0)(17, 78.33)	(6, 0)(9, 26.57) (16, 68.31)
AG ₂	(1, 0)(12, 38.33)	(19, 24.44)	(1, 0)
AG ₃	(1, 0)(12, 38.33) (17, 78.33)	(6, 0)	(5, 0)(8, 17.70) (14, 62.29)
AG ₄	(5, 0)(12, 38.33)	(2, 0)(5, 22.02) (16, 71.01)	(1, 0)
AG ₅	(2, 0)(7, 20.52) (15, 61.56)	(8, 23.80)	(2, 0)
AG ₆	(4, 0)(7, 20.52) (10, 51.91)(16, 75.67)	(6, 0)(9, 29.27) (20, 65.44)	(6, 0)(8, 17.70) (14, 62.29)
AG ₇	(4, 0)(8, 23.80) (13, 58.33)(18, 102.05)	(4, 0)(8, 23.80) (13, 58.33)(18, 84.99)	(6, 0)(8, 17.70) (20, 64.34)
AG ₈	(6, 0)(8, 23.80) (13, 58.33)	(4, 0)(9, 29.27) (10, 49.99)(16, 71.01)	(1, 0)(7, 17.70) (13, 42.79)(18, 82.29)
AG ₉	(1, 0)(7, 20.52) (10, 51.91)(14, 82.05)	(4, 0)(8, 23.80) (13, 58.33)	(4, 0)(9, 26.57) (10, 47.29)(16, 68.31)
AG ₁₀	(5, 0)(9, 31.20) (11, 61.78)(18, 102.05)	(1, 0)(12, 35.43) (15, 56.91)(14, 64.98)	(19, 41.24)(17, 76.05)
AG ₁₁	(1, 0)(7, 20.52) (12, 38.33)(20, 81.51)	(1, 0)(5, 22.03) (11, 46.63)(14, 64.98)	(1, 0)(12, 35.43) (17, 76.05)
AG ₁₂	(2, 0)(12, 38.33) (20, 81.51)	(2, 0)(19, 24.44) (15, 56.91)	(5, 0)(11, 35.43) (15, 52.27)(20, 64.34)
AG ₁₃	(19, 41.24)(20, 81.51)	(1, 0)(5, 22.02) (10, 49.99)(18, 84.99)	(2, 0)(8, 17.70) (15, 52.27)(20, 64.34)
AG ₁₄	(3, 0)(9, 31.20) (15, 64.56)(14, 82.05)	(3, 0)(7, 25.43) (20, 65.44)	(3, 0)(7, 25.43) (13, 42.79)
AG ₁₅	(3, 0)(9, 31.20) (20, 81.51)	(3, 0)(12, 35.43) (14, 64.98)	(3, 0)(12, 35.43) (14, 62.29)
AG ₁₆	(6, 0)(8, 23.20) (17, 78.33)	(3, 0)(9, 29.27) (10, 49.99)	(3, 0)(9, 26.57) (10, 47.29)
AG ₁₇	(3, 0)(9, 31.20) (10, 51.91)	(6, 0)(8, 23.80) (17, 78.33)	(4, 0)(9, 26.57) (10, 47.29)
AG ₁₈	(4, 0)(10, 51.91) (16, 75.67)	(4, 0)(7, 25.43) (20, 65.44)	(3, 0)(10, 47.29)
AG ₁₉	(3, 0)(13, 58.33)	(3, 0)(9, 29.27) (13, 58.33)	(4, 0)(7, 25.43) (13, 42.79)(18, 82.29)
AG ₂₀	(3, 0)(19, 41.24)	(3, 0)	(3, 0)(19, 31.24)
AG ₂₁	(4, 0)(19, 41.24)	(4, 0)(19, 24.44)	(4, 0)(19, 31.24)
AG ₂₂	(6, 0)(19, 41.24)	(6, 0)(8, 23.80)	(6, 0)(19, 31.24)
AG ₂₃	(8, 23.20)(20, 81.51)	(19, 24.44)	(8, 17.70)(20, 64.34)
AG ₂₄	(19, 41.24)	(19, 24.44)(20, 65.44)	(19, 31.24)(18, 82.29)

4 结 论

本文在空中突击作战的背景任务下,将多 ACA 编组视为具有资源承载能力的整体,针对多编组任务分配问题,建立了时间扩展型的协同任务分配模型.结合 DLS 和 QGA 提出了一种多编组协同任务分配算法,使用 DLS 选择当前需要处理的作战任务,QGA 为选定任务分配最佳的编组.案例的仿真计算表明,所建模型和所提出算法可以有效应用于战前空中作战智能体任务联盟中编组层面的任务分配.仿真结果中,DLS-QGA 可以计算得到统计意义上更优更稳定的分配方案.本文所建模型和所设计算法能够为任务联盟新作战样式的战前规划提供新的研究思路,方法具有较好的可靠性和时效性.在该方案的基础上,结合时域滚动策略和快速模拟退火算法,研究作战过程中应对不确定突发事件的任务分配方案动态调整策略是下一步的研究方案.

参考文献(References)

- [1] Deng Y Y. Task allocation and path planning for acoustic networks of AUVs[D]. Boca Raton: School of Graduate, Florida Atlantic University, 2010.
- [2] 杜继永,张凤鸣,杨骥,等.多UCAV协同任务分配模型及粒子群算法求解[J].控制与决策,2012,27(11):1751-1755.
(Du J Y, Zhang F M, Yang J, et al. Cooperative task assignment for multiple UCAV using particle swarm optimization[J]. Control and Decision, 2012, 27(11): 1751-1755.)
- [3] 孙黎阳,林剑柠,毛少杰,等.基于改进粒子群优化算法的网络化仿真任务共同体服务选择[J].兵工学报,2012,33(11):1393-1403.
(Sun L Y, Lin J N, Mao S J, et al. Service selection of network simulation task community based on improved particle swarm optimization algorithm[J]. Acta Armamentarii, 2012, 33(11): 1393-1403.)
- [4] Bui H, Han X, Mandal S, et al. Optimization-based decision support algorithms for a team-in-the-loop planning experiment[C]. Proc of the 2009 IEEE Int Conf on Systems, Man and Cybernetics. San Antonio, 2009: 4684-4689.
- [5] Mandal S, Han X, Pattipati K R, et al. Agent-based distributed framework for collaborative planning[C]. Proc of the 2010 IEEE Int Conf on Aerospace. Big Sky, 2010: 1-11.
- [6] Levchuk G M, Levchuk Y N, Luo J, et al. A library of optimization algorithms for organizational design[C]. Proc of the 2000 Int Command and Control Research and Technology Symposium. Monterey: NPS, 2000: 1-40.
- [7] Levchuk G M, Levchuk Y N, Luo J, et al. Normative design of organization: Mission planning[J]. IEEE Trans on Systems, Man and Cybernetics, Part A: Systems and Humans, 2002, 32(3): 346-359.
- [8] 阳东升,张维明,刘忠,等.战役任务计划的数学描述与求解算法研究[J].系统工程理论与实践,2006,26(1):26-34.
(Yang D S, Zhang W M, Liu Z, et al. Research on mathematical description and solving algorithms of tasks scheduling for campaign[J]. Systems Engineering-Theory & Practice, 2006, 26(1): 26-34.)
- [9] 张杰勇,姚佩阳,周翔翔,等.基于DLS和GA的作战任务-平台资源匹配方法[J].系统工程与电子技术,2012,34(5):947-954.
(Zhang J Y, Yao P Y, Zhou X X, et al. Approach to operation task and platform resource matching based on DLS and GA[J]. Systems Engineering and Electronics, 2012, 34(5): 947-954.)
- [10] Narayanan A, Moore M. Quantum inspired genetic algorithm[C]. Proc of the IEEE Int Conf on Evolutionary Computation. Nagoya, 1996: 61-66.
- [11] 李欣,程春田,曾筠.基于改进量子遗传算法的过程神经元网络训练[J].控制与决策,2009,24(3):347-351.
(Li X, Cheng C T, Zeng Y. Training of process neural networks based on improved quantum genetic algorithm[J]. Control and Decision, 2009, 24(3): 347-351.)
- [12] 刘文杰,马廷淮,闫莽莽,等.NIQGA改进及基于可变角距离的新型量子进化算法[J].东南大学学报:自然科学版,2011,41(5):487-491.
(Liu W J, Ma T H, Yan Q Q, et al. Improvement on NIQGA and novel quantum-inspired evolutionary algorithm based on variable angle-distance[J]. J of Southeast University: Natural Science Edition, 2011, 41(5): 487-491.)
- [13] 牟亮.不确定使命环境下C2组织结构动态适应性优化方法研究[D].长沙:国防科技大学研究生院,2011.
(Mu L. Dynamic adaptive optimization methodology of C2 organization structure under uncertainty mission environment[J]. Changsha: School of Graduate, National University of Defense Technology, 2011.)
- [14] 邢焕来,潘炜,邹喜华.一种解决组合优化问题的改进型量子遗传算法[J].电子学报,2007,35(10):1999-2002.
(Xing H L, Pan W, Zou X H. A novel improved quantum genetic algorithm for combinatorial optimization problems[J]. Acta Electronica Sinica, 2007, 35(10): 1999-2002.)

(责任编辑:郑晓蕾)