

基于鲁棒多目标优化的智能体群组协同任务规划

孙昱¹, 姚佩阳¹, 孙鹏¹, 任高明²

(1. 空军工程大学信息与导航学院, 西安 710077; 2. 95019部队, 湖北 襄阳 441802)

摘要: 为了解决不确定性条件下的智能体群组协同任务规划问题, 从提高任务分配方案鲁棒性的角度出发, 建立以最小化任务完成时间和最大化任务完成质量为目标的时间规划模型, 提出可直接求解模型的区间型非支配排序算法. 算法定义区间目标函数间的占优支配关系, 在编码空间通过组合使用随机遗传算子和启发式算子引导种群进化, 在解码空间采用循环拥挤距离排序淘汰染色体保持种群规模. 实验结果表明, 所提出的方法可行有效, 在不确定性条件下能得到鲁棒优质的任务分配方案.

关键词: 智能体群组; 任务分配; 鲁棒性; 区间规划; 多目标

中图分类号: TP18; TP301.6

文献标志码: A

Cooperative task scheduling method for agent group using robust multi-objective optimization approach

SUN Yu¹, YAO Pei-yang¹, SUN Peng¹, REN Gao-ming²

(1. Information and Navigation College, Air Force Engineering University, Xi'an 710077, China; 2. Unit 95019 of PLA, Xiangyang 441802, China. Correspondent: SUN Yu, E-mail: suny.z@qq.com)

Abstract: To solve the cooperative task scheduling problem under uncertain conditions for the agent group, an interval programming mathematical model is established from the aspect of improving robustness of task allocation scheme. A non-dominated sorting algorithm in the interval pattern is proposed for solving this model. Since the model aims to minimize task completion time while maximize task completion quality, the dominance relationship between different individuals is defined. Then the algorithm guides the evolution of population by using the random genetic operator and heuristics operator in the coding space, and removes chromosomes to keep the size of population by adopting the circular crowded sorting strategy in the decoding space. Experiment results show that the proposed method is feasible and effective, and can obtain robust and superior task allocation schemes under uncertain conditions.

Keywords: agent group; task scheduling; robustness; interval programming; multi-objective

0 引言

现代社会中智能体如机器人、无人机得到了广泛的应用, 但是面对复杂多样的任务, 单个智能体的功能和能力往往有限, 需要多个智能体构成混合群组协同执行任务, 以发挥一加一大于二的整体优势^[1-2]. 智能体群组执行的任务之间通常具有一定的时序约束关系, 且不同任务的性质互异, 如何合理地将任务分配给群组中的各个智能体, 使整个群组又好又快地完成所有任务是智能体群组协同任务规划需要考虑的问题^[3].

现有的智能体群组协同任务规划方法大多假定

任务执行环境是静态的和确定的^[4-7], 对任务执行环境的不确定性考虑得较少, 而在实际的任务执行过程中, 参数扰动、任务执行偏差等不确定性因素很可能导致预先制定的任务分配方案失效, 进而导致智能体群组行动的失败. 因此, 在对智能体群组进行协同任务规划时, 需要事先考虑这些不确定性因素以提高任务分配方案的鲁棒性. 文献[8]综合考虑了任务成功不确定性、任务执行时间不确定性等因素, 建立了以最大化使命成功概率为目标的机会约束规划模型; 文献[9]采用冗余策略, 通过在任务分配方案中增加时间缓冲区和资源缓冲区减少不确定性因素的干扰; 文

收稿日期: 2015-12-21; 修回日期: 2016-03-04.

基金项目: 国家自然科学基金项目(61573017).

作者简介: 孙昱(1989—), 男, 博士生, 从事指挥控制系统建模与仿真的研究; 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.

献[10]在遂行协同搜索任务时先将不确定性环境参量转化为探测概率,再以此为依据合理地智能体规划任务;文献[11]提出了复杂动态环境中的任务分配框架,并结合环境信息和智能体能力等要素计算了智能体的任务成功概率;文献[12]进一步研究了信息不完全条件下的任务规划方法.这些研究工作虽然能在一定程度上提高任务分配方案的鲁棒性,但通常需要预先估计一些不确定性变量的分布律,这在复杂多变的任务执行环境中比较困难,因此其实用性仍然有限.

本文将区间规划^[13-14]思想引入智能体群组的协同任务规划中,以最小化所有任务的完成时间和最大化所有任务的完成质量为目标建立相应的不确定性多目标优化模型,并提出一种可以直接求解该模型的含启发式算子的区间型非支配排序算法.本文方法无需提前预计不确定性变量的分布律,只需简单估计不确定性变量的上下界即可,通过定义区间目标函数之间的占优支配关系保证最终解既优质又鲁棒.本文首先描述和分析智能体群组协同任务规划问题,然后建立相应的数学模型,设计求解算法,最后通过实验验证所提方法的有效性和优越性.

1 问题描述

1.1 基本概念

任务(Task, T): 任务是智能体群组为达成目的而必须采取的行动.智能体群组的任务集记为 $T = \{T_1, T_2, \dots, T_n\}$, n 为集合中任务的数量.对于 $\forall T_i \in T$, 其属性包括: 任务坐标位置 (X_{T_i}, Y_{T_i}) ; 任务持续时间 TT_i ; 任务的能力需求向量 $(DT_{i1}, DT_{i2}, \dots, DT_{il})$, 其中 l 为不同能力类型的数量, DT_{ik} ($1 \leq k \leq l$) 为处理任务 T_i 时需要的第 k 项能力的大小,若 $DT_{ik} = 0$, 则处理任务 T_i 并不需要第 k 项能力.集合 T 中各任务之间通常具有时序约束关系,如图 1 所示.

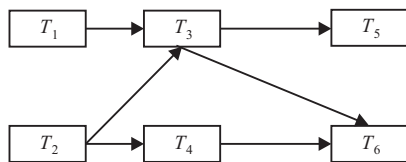


图 1 任务间的时序约束关系

任务集 T 中任务间的时序约束关系可由矩阵 $M^{T-T} = (m_{ij}^{T-T})_{n \times n}$ 表示,其中若任务 T_i 是任务 T_j 的前导任务,则 $m_{ij}^{T-T} = 1$, 否则 $m_{ij}^{T-T} = 0$. 一个任务必须同时满足以下 3 个条件才能开始被处理: 1) 该任务的所有前导任务都已经完成; 2) 分配处理该任务的所有智能体都到达了该任务所在的位置; 3) 处理该任务需要的各项能力大小得到一定满足.

智能体(Agent, A): 智能体是资源的载体,是任务的直接处理者.智能体群组中所有智能体构成的集合

记为 $A = \{A_1, A_2, \dots, A_v\}$, v 为智能体的数量.对于 $\forall A_j \in A$, 其属性包括: 智能体的初始坐标位置 (X_{A_j}, Y_{A_j}) ; 智能体的移动速度 VA_j ; 智能体的能力向量 $(OA_{j1}, OA_{j2}, \dots, OA_{jl})$, 其中 l 为不同能力类型的数量, OA_{jk} ($1 \leq k \leq l$) 为智能体 A_j 第 k 项能力的大小,若 $OA_{jk} = 0$, 则智能体 A_j 不具备第 k 项能力.

智能体群组的任务分配方案可由矩阵 $M^{T-A} = (m_{ij}^{T-A})_{n \times v}$ 表示,其中若任务 T_i 分配给智能体 A_j 处理,则 $m_{ij}^{T-A} = 1$, 否则 $m_{ij}^{T-A} = 0$. 当一个智能体需处理多个任务时,该智能体将按照任务的优先级大小依次处理各个任务,智能体群组任务集 T 中任务的优先级可采用加权长度法^[15]确定. 当一个任务由于不满足条件而无法开始被处理时,已经到达该任务位置的智能体将在此等待直到该任务被完成.

1.2 不确定性分析

智能体群组在执行任务时存在的不确定性因素主要有以下 3 类:

- 1) 任务不确定性. 主要是任务新增、取消和变更的不确定性.
- 2) 智能体不确定性. 主要是智能体由于内外部原因损毁、故障等不确定性.
- 3) 参数不确定性. 主要是任务和智能体的参数受环境干扰、执行偏差等因素影响而引起的不确定性.

以上不确定性因素的存在导致智能体群组在完成其任务时无法严格按照预先的任务分配方案进行,甚至可能引起整个行动的失败.因此,在进行智能体群组协同任务规划时,必须事先将这些不确定性因素纳入考虑.

本文主要考虑智能体群组协同任务规划中的参数不确定性问题,包括任务持续时间 TT_i ($1 \leq i \leq n$) 不确定、任务能力需求向量中各分量 DT_{ik} ($1 \leq i \leq n, 1 \leq k \leq l$) 不确定、智能体移动速度 VA_j ($1 \leq j \leq v$) 不确定和智能体能力向量中各分量 OA_{jk} ($1 \leq j \leq v, 1 \leq k \leq l$) 不确定.在复杂多变的任务执行环境中,这些参数的分布律难以准确统计,但是估计它们的范围却相对简单,因此本文引入区间数表征这些参数的不确定性.

1.3 协同任务规划模型

对于 $\forall T_i \in T$, 其开始时间记为 BT_i , 结束时间记为 $ET_i = BT_i + TT_i$, 其前导任务构成的集合记为 $S_{T_i}^T = \{T_k | m_{ki}^{T-T} = 1, 1 \leq k \leq n\}$, 处理任务 T_i 时需要的能力类型构成的集合记为 $R_{T_i} = \{k | DT_{ik} > 0, 1 \leq k \leq l\}$, 处理 T_i 的智能体构成的集合记为 $S_{T_i}^A = \{A_k | m_{ik}^{T-A} = 1, 1 \leq k \leq v\}$.

当智能体群组按任务分配方案执行各项任务时, 其完成所有任务的时间为

$$TM = \max_{T_i \in T} (ET_i). \quad (1)$$

其中: TM 越小, 智能体群组完成所有任务耗时越短.

对于 $\forall T_i \in T$, 智能体群组完成 T_i 的质量定义为

$$QT_i = \left(\prod_{k \in R_{T_i}} Z_{ik} \right)^{1/\|R_{T_i}\|}. \quad (2)$$

其中: $Z_{ik} = \min \left(\sum_{A_j \in S_{T_i}^A} OA_{jk} / DT_{ik}, 1 \right)$ 为 T_i 的第 k 项能力需求被满足的程度, $\|R_{T_i}\|$ 为 R_{T_i} 中元素的数量. 智能体群组完成所有任务的平均质量定义为

$$QM = \left(\prod_{i=1}^n QT_i \right)^{1/n}. \quad (3)$$

显然 $0 \leq QM \leq 1$, QM 越大, 智能体群组完成所有任务的质量越好.

智能体群组协同任务规划的目标是使得整个群组又好又快地完成其任务集 T 中的所有任务, 故协同任务规划模型的目标函数为

$$\min TM; \max QM. \quad (4)$$

考虑到 $0 \leq QM \leq 1$, 因此式 (4) 可改写为

$$\min f = (TM, 1 - QM). \quad (5)$$

下面考虑协同任务规划模型的约束条件. 对于 $\forall A_k \in S_{T_i}^A$, 若其在处理 T_i 之前没有处理其他任务, 则其到达 T_i 所在位置的时间为 $TIME_{ki} = ((XA_k - XT_i)^2 + (YA_k - YT_i)^2)^{1/2} / VA_k$; 若其在处理 T_i 之前处理的最后一个任务为 T_j , 则其到达 T_i 所在位置的时间为 $TIME_{ki} = ((XT_j - XT_i)^2 + (YT_j - YT_i)^2)^{1/2} / VA_k + ET_j$. 当任务 T_i 开始被处理时, 必须满足 1.1 节中描述的 3 个条件, 因此有

$$ST_i = \max \left(\max_{T_k \in S_{T_i}^T} (ET_k), \max_{A_k \in S_{T_i}^A} (TIME_{ki}) \right), \quad (6)$$

$$QT_i > 0. \quad (7)$$

综上所述, 智能体群组协同任务规划模型为

$$\min f = (TM, 1 - QM).$$

$$\text{s.t.} \begin{cases} m_{ij}^{T-A} \in \{0, 1\}, 1 \leq i \leq n, 1 \leq j \leq v; \\ ST_i = \max \left(\max_{T_k \in S_{T_i}^T} (ET_k), \max_{A_k \in S_{T_i}^A} (TIME_{ki}) \right), \\ 1 \leq i \leq n; \\ QT_i > 0, 1 \leq i \leq n. \end{cases} \quad (8)$$

其中 m_{ij}^{T-A} 为该模型的决策变量. 由 1.2 节的分析可知, 该模型的许多参数均为表征不确定性的区间数, 因此式 (8) 是一个体现了参数不确定性的多目标优化模型.

2 问题求解

NSGA-II^[16] 因能较好地解决确定性多目标优化问题而得到广泛使用, 其基本框架如图 2 所示.

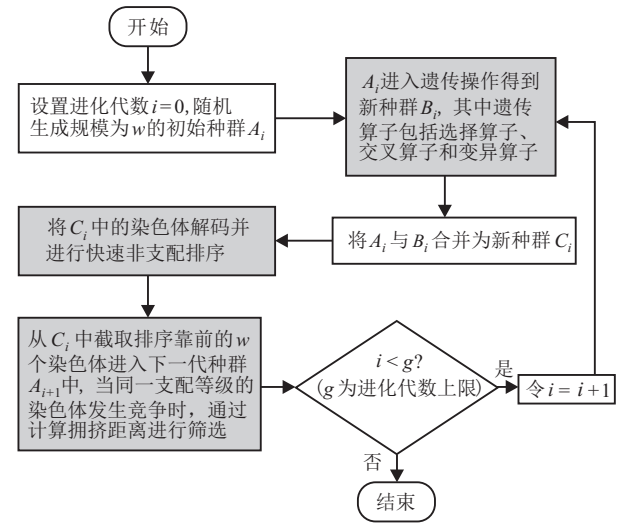


图 2 NSGA-II 的基本框架

本文在此基础上提出一种区间型非支配排序算法求解式 (8). 与 NSGA-II 相比, 本文对图 2 中的阴影部分进行改进, 包括以下 4 点:

- 1) 提出满足模型约束的染色体解码方式, 保证算法的可达空间与模型的可行域相对应;
- 2) 定义基于区间可能度的区间数占优支配关系, 用于模型目标的比较排序;
- 3) 采用区间距离公式计算解的拥挤距离, 并使用循环拥挤排序淘汰多余的解;
- 4) 设计启发式交叉算子引导种群进化, 提高算法的收敛速度.

下面从以上 4 个方面介绍本文求解算法.

2.1 染色体编、解码

染色体 X 采用二进制编码方式, 由 n 行 v 列的 0-1 矩阵 $CM = (cm_{ij})_{n \times v}$ 表示, 矩阵 CM 的每一行为染色体的一个基因. 由于模型存在约束, 任务分配方案 M^{T-A} 直接设置为 CM 不一定是可行解, 需先对染色体进行解码, 具体步骤如下.

Step 1: 初始化 $i = 0$.

Step 2: 令 $i = i + 1$, 若 $i < n + 1$, 则设置 $k = 0$ 并执行 Step 3, 否则解码过程结束.

Step 3: 令 $k = k + 1$, 若 $k < l + 1$, 则执行 Step 4, 否则设置 $j = 0$ 并执行 Step 5.

Step 4: 若 $DT_{ik} > 0$ 且 $\sum_{\substack{cm_{ip}=1 \\ 1 \leq p \leq v}} OA_{pk} = 0$, 则说明

任务分配方案 $M^{T-A} = CM$ 时, 约束 $QT_i > 0$ 将不被满足, 此时在智能体集合 A 中找出具有第 k 项能力且序号值最小的智能体 A_q , 并令 $cm_{iq} = 1$, 然后返回

Step 3, 否则直接返回 Step 3.

Step 5: 令 $j = j + 1$, 若 $j < v + 1$, 则执行 Step 6, 否则返回 Step 2.

Step 6: 若 $cm_{ij} = 1$, 则执行 Step 7, 否则返回 Step 5.

Step 7: 令 $M^{T-A} = CM$, 并根据 M^{T-A} 计算 $c_1 = QT_i$, 然后令 $m_{ij}^{T-A} = 0$, 再次根据 M^{T-A} 计算 $c_2 = QT_i$. 若 $c_1 = c_2$, 则说明任务 T_i 分配给智能体 A_j 处理并不能提高 T_i 的完成质量, 相反还可能会延后 T_i 的开始时间, 因此令 $cm_{ij} = 0$ 并返回 Step 5, 否则直接返回 Step 5.

染色体 $X = CM$ 经过解码后是模型的可行解, 此时任务分配方案 M^{T-A} 可以设置为 CM .

2.2 基于区间可能度的占优支配关系

若区间数 $A = [a^L, a^R]$, $B = [b^L, b^R]$, 则基于区间可能度的概率 $P(A \leq B)$ 的计算公式^[17]为

$$P(A \leq B) = \begin{cases} 0, b^R \leq a^L; \\ 0.5 \times \frac{b^R - a^L}{a^R - a^L} \frac{b^R - a^L}{b^R - b^L}, b^L \leq a^L < b^R \leq a^R; \\ \frac{0.5b^L - a^L + 0.5b^R}{a^R - a^L}, a^L < b^L < b^R \leq a^R; \\ \frac{b^L - a^L}{a^R - a^L} + \frac{a^R - b^L}{a^R - a^L} \times \frac{b^R - 0.5a^R - 0.5b^L}{b^R - b^L}, \\ \quad a^L < b^L \leq a^R < b^R; \\ \frac{b^R - 0.5a^R - 0.5a^L}{b^R - b^L}, b^L \leq a^L < a^R < b^R; \\ 1, a^R < b^L. \end{cases} \quad (9)$$

考虑到由式 (9) 计算得到的概率 $P(A \leq B)$ 具有以下性质: 1) $0 \leq P(A \leq B) \leq 1$; 2) $P(A \leq B) + P(B \leq A) = 1$; 3) $P(A \leq B) = P(B \leq A) = 0.5 \Leftrightarrow a^L = b^L, a^R = b^R$. 本文在此基础上定义基于区间可能度的区间数占优支配关系.

定义 1 若区间数 $A = [a^L, a^R]$, $B = [b^L, b^R]$, 则当 $P(A \leq B) \geq 0.5$ 时, 称区间数 A 在区间意义下不大于区间数 B ; 当 $P(A \leq B) > 0.5$ 时, 称区间数 A 在区间意义下小于区间数 B ; 当 $P(A \leq B) \leq 0.5$ 时, 称区间数 A 在区间意义下不小于区间数 B ; 当 $P(A \leq B) < 0.5$ 时, 称区间数 A 在区间意义下大于区间数 B .

定义 2 若染色体 X_1 解码后计算得到的目标函数值为 $TM_1, 1 - QM_1$, 染色体 X_2 解码后计算得到的目标函数值为 $TM_2, 1 - QM_2$, 则有:

1) 当 $TM_1, 1 - QM_1$ 在区间意义下不大于 $TM_2, 1 - QM_2$, 且满足 TM_1 在区间意义下小于 TM_2 或者 $1 - QM_1$ 在区间意义下小于 $1 - QM_2$ 时, 称 X_1 优于 X_2 , 记作 $X_1 \succ X_2$;

2) 当 $TM_1, 1 - QM_1$ 在区间意义下不小于 $TM_2, 1 - QM_2$, 且满足 TM_1 在区间意义下大于 TM_2 或者 $1 - QM_1$ 在区间意义下大于 $1 - QM_2$ 时, 称 X_1 劣于 X_2 , 记作 $X_1 \prec X_2$;

3) 当 X_1 既不优于 X_2 , 也不劣于 X_2 时, 称 X_1 无差别于 X_2 , 记作 $X_1 \sim X_2$.

定义 3 若种群 P 是由一定数量的染色体构成的集合, $X_i \in P$, 且当 $j \neq i$ 时, 有

$$\nexists X_j \in P: X_j \succ X_i \quad (10)$$

成立, 则称 X_i 为种群 P 中的 Pareto 解.

2.3 循环拥挤距离排序

由于种群规模有限, 当处于某一支配等级的染色体无法全部进入到下一代种群时, 需要计算这些染色体的拥挤距离并淘汰一些密集的解. 考虑到目标函数 TM 和 $1 - QM$ 都是区间数, 在计算染色体的拥挤距离时需用到区间距离公式, 本文采用的区间距离公式^[18]为

$$d(A, B) = \frac{((a^L + a^R - b^L - b^R)^2/4 + ((a^R - a^L)^2 + (b^R - b^L)^2)/12 - F^2/6)^{1/2}}{F} \quad (11)$$

其中: $A = [a^L, a^R]$, $B = [b^L, b^R]$; F 为区间 $A \cap B$ 的长度, 当 $A \cap B = \phi$ 时, $F = 0$. 需要指出的是, 目标函数 TM 与 $1 - QM$ 的量纲不同, 故在计算拥挤距离前需将目标函数进行归一化处理.

与 NSGA-II 算法一次性计算染色体的拥挤距离不同, 本文通过循环拥挤距离排序选择进入下一代种群的染色体, 即首先计算各备选染色体的拥挤距离, 然后将最稀疏(拥挤距离最大)的染色体选入下一代种群中并将其从备选染色体中删除, 重复进行上述步骤直至下一代种群的规模达到上限. 采用循环拥挤距离排序虽然在一定程度上提高了算法的时间复杂度, 但能更好地保持种群中染色体的多样性, 避免算法过早收敛陷入局部最优当中.

2.4 遗传算子

种群进化时采用的遗传算子主要包括选择算子、变异算子和交叉算子.

选择算子用于从上一代种群中挑选染色体执行交叉和变异操作, 本文选择算子采用二进制锦标赛选择策略.

变异算子通过基因突变产生新染色体, 本文变异算子采用均匀变异策略.

交叉算子通过重组一对染色体的基因产生新染色体, 本文交叉算子分为随机交叉算子和启发式交叉

算子两种, 其中随机交叉算子采用均匀交叉策略, 启发式交叉算子构造新染色体的方式如下。

Step 1: 选定参与交叉的染色体 $X_1 = CM_1, X_2 = CM_2$, 初始化新染色体 $X_3 = CM_3$, 其中矩阵 CM_3 的所有元素均为 1, 设置 $i = 1$ 。

Step 2: 若 $i < n + 1$, 则执行 **Step 3**, 否则结束并将 X_3 作为构造出的新染色体输出。

Step 3: 将 X_3 的第 i 个基因分别用 $X_1、X_2$ 的第 i 个基因替代, 形成的新染色体分别记为 $X_{31}、X_{32}$ 。若 $X_{31} \succ X_{32}$, 则令 $X_3 = X_{31}$; 若 $X_{31} \prec X_{32}$, 则令 $X_3 = X_{32}$; 若 $X_{31} \sim X_{32}$, 则从 $X_{31}、X_{32}$ 中随机选择一个更新 X_3 。令 $i = i + 1$, 返回 **Step 2**。

随机交叉算子本质上是在解空间中进行随机搜索, 容易导致算法的收敛速度过慢, 而引入启发式交叉算子可以不断得到一些较好的解, 从而引导搜索在一些更为有利的方向上进行, 进而提高算法的收敛速度。组合使用这两种交叉算子的方式如下: 设置启发式算子触发率 $\varepsilon (0 \leq \varepsilon \leq 1)$, 当需要使用交叉算子时, 产生区间 $[0, 1]$ 内的随机数 δ , 若 $\delta \leq \varepsilon$, 则采用启发式交叉算子, 否则采用随机交叉算子。

2.5 算法步骤

本文提出的区间型非支配排序算法的具体步骤如下。

Step 1: 设置进化代数上限 g , 种群规模 w , 交叉概率 p_c , 变异概率 p_m , 启发式算子触发率 ε 。设置 $i = 0$, 初始化初代种群 A_i 。

Step 2: 种群 A_i 采用遗传算子进化, 得到的新种群记为 B_i 。

Step 3: 将 A_i 与 B_i 合并为 C_i , 采用非支配排序和循环拥挤距离排序从 C_i 中选出染色体进入下一代种群 A_{i+1} 中。

Step 4: 令 $i = i + 1$ 。

Step 5: 若 $i < g$, 则执行 **Step 2**, 否则执行 **Step 6**。

Step 6: 将种群 A_g 中的所有 Pareto 解输出, 算法结束。

由于算法通过非支配排序将每一代种群中的“精英”保留到下一代种群, 且模型可行域中的任意两个解经过交叉和变异都是可达的, 算法具有全局收敛性。

3 实验结果及分析

以智能体群组协同任务规划为例在 Pentium(R) Dual-Core CPU 3.0 GHz 计算机上进行仿真实验, 群组中智能体数量 $v = 20$, 需要完成的任务数量 $n = 18$, 任务间的时序约束关系如图 3 所示, 任务和智能体属性分别如表 1 和表 2 所示。

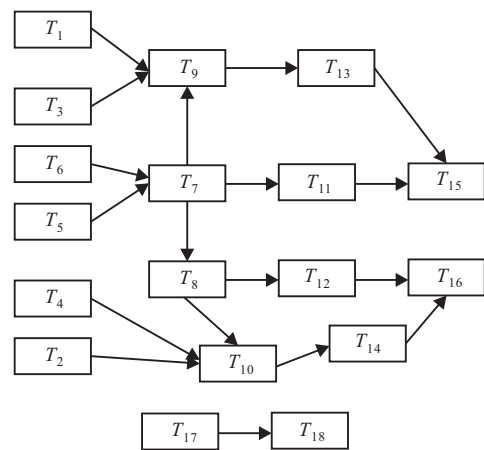


图 3 任务时序约束

表 1 任务属性表

任务 编号	能力需求向量								持续时间 TT_i	位置	
	DT_{1i}	DT_{2i}	DT_{3i}	DT_{4i}	DT_{5i}	DT_{6i}	DT_{7i}	DT_{8i}		XT_i	YT_i
T_1	[5,6]	[3,3.6]	[10,12]	0	0	[8,9.6]	0	[6,7.2]	[30,36]	70	15
T_2	[5,6]	[3,3.6]	[10,12]	0	0	[8,9.6]	0	[6,7.2]	[30,36]	64	75
T_3	0	[3,3.6]	0	0	0	0	0	0	[10,12]	15	40
T_4	0	[3,3.6]	0	0	0	0	0	0	[10,12]	30	95
T_5	0	[3,3.6]	0	0	0	0	[10,12]	0	[10,12]	28	73
T_6	0	0	0	[10,12]	[14,16.8]	[12,14.4]	0	0	[10,12]	24	60
T_7	0	0	0	[10,12]	[14,16.8]	[12,14.4]	0	0	[10,12]	28	73
T_8	0	0	0	[10,12]	[14,16.8]	[12,14.4]	0	0	[10,12]	28	83
T_9	[5,6]	0	0	0	0	[5,6]	0	0	[10,12]	28	73
T_{10}	[5,6]	0	0	0	0	[5,6]	0	0	[10,12]	28	83
T_{11}	0	0	0	0	0	[10,12]	[5,6]	0	[10,12]	25	45
T_{12}	0	0	0	0	0	[10,12]	[5,6]	0	[10,12]	5	95
T_{13}	0	0	0	0	0	[8,9.6]	0	[6,7.2]	[20,24]	25	45
T_{14}	0	0	0	0	0	[8,9.6]	0	[6,7.2]	[20,24]	5	95
T_{15}	0	0	0	[20,24]	[10,12]	[4,4.8]	0	0	[15,18]	25	45
T_{16}	0	0	0	[20,24]	[10,12]	[4,4.8]	0	0	[15,18]	5	95
T_{17}	0	0	0	0	0	[8,9.6]	0	[4,4.8]	[10,12]	5	60
T_{18}	0	0	0	[8,9.6]	[6,7.2]	0	[4,4.8]	[10,12]	[20,24]	5	60

表 2 智能体属性表

智能体 编号	能力向量								速度 VA_j	初始位置	
	OA_{1j}	OA_{2j}	OA_{3j}	OA_{4j}	OA_{5j}	OA_{6j}	OA_{7j}	OA_{8j}		XA_j	YA_j
A_1	[8,10]	[8,10]	[0.8,1]	0	[7.2,9]	[4,5]	0	0	[1.6,2]	85	40
A_2	[0.8,1]	[3.2,4]	[8,10]	0	[3.2,4]	[2.4,3]	0	0	[1.6,2]	85	40
A_3	[8,10]	[8,10]	[0.8,1]	0	[7.2,9]	[1.6,2]	0	0	[1.6,2]	85	40
A_4	0	0	0	[1.6,2]	0	0	[4,5]	0	[3.2,4]	85	40
A_5	[0.8,1]	0	0	[8,10]	[1.6,2]	[1.6,2]	[0.8,1]	0	[1.08,1.35]	85	40
A_6	[4,5]	0	0	0	0	0	0	0	[3.2,4]	85	40
A_7	[2.4,3]	[3.2,4]	0	0	[4.8,6]	[8,10]	[0.8,1]	0	[3.2,4]	85	40
A_8	[0.8,1]	[2.4,3]	0	0	[8,10]	[6.4,8]	[0.8,1]	0	[3.2,4]	85	40
A_9	[0.8,1]	[2.4,3]	0	0	[8,10]	[6.4,8]	[0.8,1]	0	[3.2,4]	85	40
A_{10}	[0.8,1]	[2.4,3]	0	0	[8,10]	[6.4,8]	[0.8,1]	0	[3.2,4]	85	40
A_{11}	[4.8,6]	[0.8,1]	0	0	[0.8,1]	[0.8,1]	0	0	[3.6,4.5]	85	40
A_{12}	[4.8,6]	[0.8,1]	0	0	[0.8,1]	[0.8,1]	0	0	[3.6,4.5]	85	40
A_{13}	[4.8,6]	[0.8,1]	0	0	[0.8,1]	[0.8,1]	0	0	[3.6,4.5]	85	40
A_{14}	0	0	0	0	0	0	[8,10]	0	[1.6,2]	85	40
A_{15}	0	0	0	0	0	0	0	[4.8,6]	[4,5]	85	40
A_{16}	0	0	0	0	0	0	0	[4.8,6]	[5.6,7]	85	40
A_{17}	0	0	0	[4.8,6]	[4.8,6]	0	[0.8,1]	[8,10]	[2.2,5]	85	40
A_{18}	[0.8,1]	0	0	[8,10]	[1.6,2]	[1.6,2]	[0.8,1]	0	[1.08,1.35]	85	40
A_{19}	[0.8,1]	0	0	[8,10]	[1.6,2]	[1.6,2]	[0.8,1]	0	[1.08,1.35]	85	40
A_{20}	[0.8,1]	0	0	[8,10]	[1.6,2]	[1.6,2]	[0.8,1]	0	[1.08,1.35]	85	40

实验 1 为了验证算法的可行性, 利用其求解算例, 其中进化代数上限 $g = 200$, 种群规模 $w = 100$, 交叉概率 $p_c = 0.5$, 变异概率 $p_m = 0.05$, 启发式算子触发率 $\epsilon = 0.05$, 实验结果如图 4 所示, 其中每个矩形代表一个 Pareto 最优解。

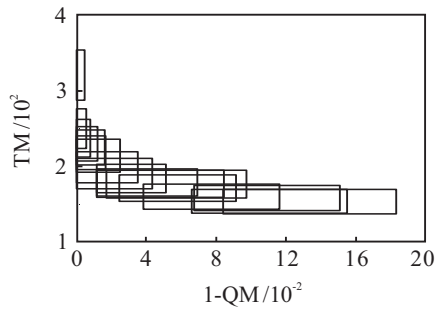


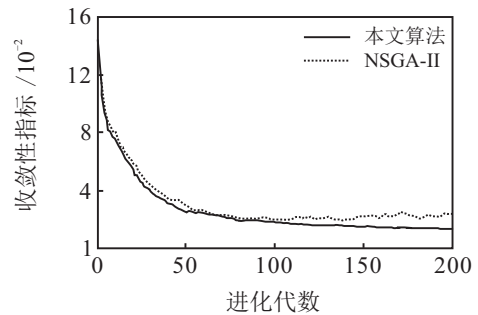
图 4 Pareto 最优解

由图 4 可以看出: 算法得到了一组分布相对均匀的 Pareto 解, 即算法是可行的, 可求解这类含区间参数的不确定性多目标优化模型. 图 4 中典型解对应的任务分配方案如表 3 所示。

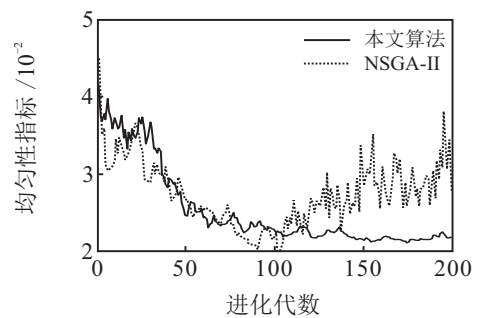
表 3 典型的任务分配方案

任务	分配给智能体	任务	分配给智能体
T_1	A_2, A_7, A_{15}	T_{10}	A_3, A_9
T_2	A_2, A_7, A_{11}, A_{16}	T_{11}	A_8, A_9, A_{14}
T_3	A_3	T_{12}	A_8, A_9, A_{14}
T_4	A_1	T_{13}	A_8, A_{13}, A_{15}
T_5	A_3, A_4, A_{14}	T_{14}	A_1, A_{17}
T_6	A_9, A_{10}, A_{17}	T_{15}	$A_5, A_{10}, A_{18}, A_{19}$
T_7	A_5, A_9, A_{17}, A_{18}	T_{16}	$A_5, A_{10}, A_{18}, A_{19}$
T_8	$A_8, A_{10}, A_{19}, A_{20}$	T_{17}	A_2, A_7, A_{16}
T_9	A_8, A_{12}, A_{13}	T_{18}	$A_4, A_{13}, A_{15}, A_{17}, A_{20}$

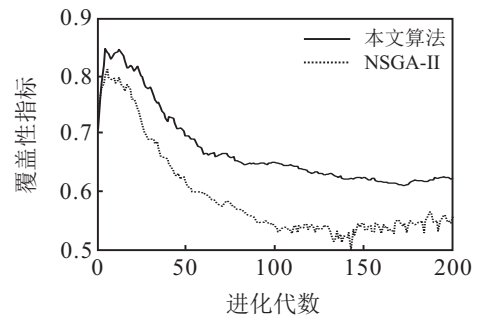
TM=[153.11,188.39] QM=[0.91,0.98]



(a) 收敛性指标比较



(b) 均匀性指标比较



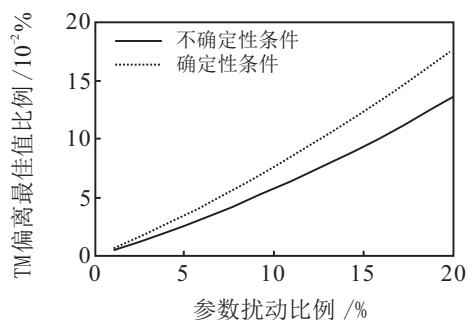
(c) 覆盖性指标比较

图 5 算法比较

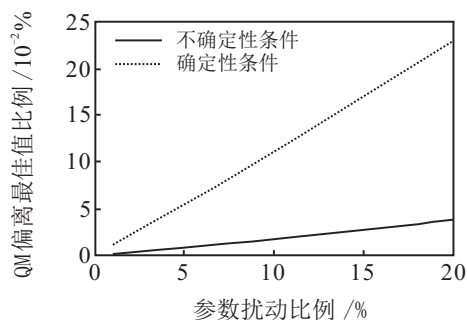
实验 2 为了验证算法的优越性, 将传统 NSGA-II 算法扩展为区间型, 并与本文算法进行比较. 衡量多目标进化算法性能的指标主要包括收敛性指标、均匀性指标^[19]和覆盖性指标^[14]. 其中: 收敛性指标、均匀性指标越小越好, 覆盖性指标越大越好. 为了消除不同优化目标量纲的影响, 将目标函数进行归一化处理后计算算法性能评价指标. 分别运行 30 次本文算法和 NSGA-II 算法, 取平均值作为最终结果. 为了保证比较的公平性, NSGA-II 算法的参数设置与本文算法保持一致, 实验结果如图 5 所示.

比较图 5 中本文算法与 NSGA-II 算法可知: 前者得到的 Pareto 解集在收敛性、均匀性和覆盖性方面均优于后者. 当进化结束时, 本文算法的收敛性指标相对于 NSGA-II 优 41.05%, 均匀性指标优 21.36%, 覆盖性指标优 11.72%. 在图 5(a) 中, NSGA-II 算法进化到 100 代之后出现“早熟”, 而本文算法仍然能继续朝全局最优的方向进化; 在图 5(b) 和图 5(c) 中, 本文算法对应的曲线不仅更优, 相对 NSGA-II 算法也更平滑, 说明本文算法在稳定性上也强于 NSGA-II.

实验 3 为了观察不确定性条件下获得的解的鲁棒性能, 对智能体和任务的参数进行扰动, 观察目标函数偏离其最佳值的大小, 并与确定性条件下获得的解进行对比. 因为算法获得的是包含多个解的 Pareto 解集, 故在比较时取解集中所有解的均值, 实验结果如图 6 所示.



(a) 任务完成时间方面的比较



(b) 任务平均完成质量方面的比较

图 6 不同条件下解的鲁棒性能比较

参数的扰动变化将导致解的质量下降, 从图 6 中可以看到: 确定性条件下获得的解对参数的扰动比不

确定性条件下获得的解更为敏感, 特别是在任务的平均完成质量方面. 由此可知, 本文获得的任务分配方案更具鲁棒性, 可以更好地应对不确定性的任务执行环境.

4 结 论

智能体群组协作执行任务过程中可能出现的不确定性因素将直接影响行动的成败. 为了提高任务分配方案的鲁棒性, 在进行任务规划时必须将不确定性因素考虑在内. 本文采用区间规划思想建立了不确定性条件下的智能体群组协同任务规划模型, 设计了一种区间型非支配排序求解算法. 实验结果表明, 所提出算法不仅可行、优越, 而且能有效提高解的鲁棒性, 使其对任务执行环境中的不确定性因素具有更强的抗干扰能力. 本文存在的不足是仅考虑了参数的不确定性, 为处理任务临时新增、智能体故障等不确定性因素, 在此基础上研究相应的动态规划方法将是下一步的工作重点.

参考文献(References)

- [1] 姚敏, 王绪芝, 赵敏. 无人机群协同作战任务分配方法研究[J]. 电子科技大学学报, 2013, 42(5): 723-727.
(Yao M, Wang X Z, Zhao M. Cooperative combat task assignment optimization design for unmanned aerial vehicles cluster[J]. J of University of Electronic Science and Technology of China, 2013, 42(5): 723-727.)
- [2] 颜骥, 李相民, 刘波. 应用离散粒子群-郭涛算法分配多无人机协同任务[J]. 国防科技大学学报, 2015, 37(4): 165-171.
(Yan J, Li X M, Liu B. Cooperative task allocation of multi-UAVs with mixed DPSO-GT algorithm[J]. J of National University of Defense Technology, 2015, 37(4): 165-171.)
- [3] Yu F, Tu F, Pattipati K R. Integration of a holonic organizational control architecture and multi-objective evolutionary algorithm for flexible distributed scheduling[J]. IEEE Trans on System, Man, and Cybernetics—Part A: System and Humans, 2008, 38(5): 1001-1016.
- [4] 张杰勇, 姚佩阳, 李凡. 完成时间限制下的任务-平台关系设计模型及算法[J]. 系统工程与电子技术, 2012, 34(8): 1621-1629.
(Zhang J Y, Yao P Y, Li F. Task-platform relation design model and its algorithm under completion time constraint[J]. Systems Engineering and Electronics, 2012, 34(8): 1621-1629.)
- [5] Han X, Bui H, Mandal S. Optimization-based decision support software for a team-in-the-loop experiment: Asset

- package selection and planning[J]. IEEE Trans on Systems, Man, and Cybernetics: System, 2013, 43(2): 237-251.
- [6] Han X, Mandal S, Pattipati K R, et al. An optimization-based distributed planning algorithm: A blackboard-based collaborative framework[J]. IEEE Trans on Systems, Man, and Cybernetics: System, 2014, 44(6): 673-686.
- [7] 万路军, 姚佩阳, 周翔翔, 等. 多编组协同任务分配模型及 DLS-QGA 算法求解[J]. 控制与决策, 2014, 29(9): 1562-1568.
(Wan L J, Yao P Y, Zhou X X, et al. Cooperative task allocation methods in multiple groups using DLS-QGA[J]. Control and Decision, 2014, 29(9): 1562-1568.)
- [8] 牟亮, 张维明, 陈涛, 等. 不确定性下 C2 组织结构的“任务-平台”关系设计模型及算法[J]. 系统工程与电子技术, 2010, 32(12): 2576-2583.
(Mu L, Zhang W M, Chen T, et al. Task-platform relation design model and algorithm for C2 organization structure under uncertainty[J]. Systems Engineering and Electronics, 2010, 32(12): 2576-2583.)
- [9] 张迎新, 陈超, 刘忠, 等. 资源不确定军事任务计划预测调度模型与算法[J]. 国防科技大学学报, 2013, 35(3): 30-35.
(Zhang Y X, Chen C, Liu Z, et al. Method for modeling and solving military mission planning with uncertain resource availability[J]. J of National University of Defense Technology, 2013, 35(3): 30-35.)
- [10] Mishra M, An W, Han X, et al. Decision support software for anti-submarine warfare mission planning within a dynamic environment context[C]. Proc of IEEE Int Conf on Systems, Man, Cybernetics. New York: IEEE Press, 2014: 3390-3393.
- [11] Ayala D F M, Sidoti D, Mishra M, et al. Context-based models to overcome operational challenges in maritime security[C]. Proc of IEEE Int Symposium on Technologies for Homeland Security. New York: IEEE Press, 2015: 1-6.
- [12] Avvari G V, Sidoti D, Mishra M, et al. Dynamic asset allocation for counter-smuggling operations under disconnected, intermittent and low-bandwidth environment[C]. Proc of IEEE Symposium on Computational Intelligence for Security and Defense Applications. New York: IEEE Press, 2015: 1-6.
- [13] Bhurjee A K, Panda G. Efficient solution of interval optimization problem[J]. Mathematical Methods of Operations Research, 2012, 76(3): 273-288.
- [14] 陈志旺, 陈林, 白铎, 等. 求解约束多目标区间优化的交互多属性决策 NSGA-II 算法[J]. 控制与决策, 2015, 30(5): 865-870.
(Chen Z W, Chen L, Bai X, et al. Interactive multi-attribute decision-making NSGA-II for constrained multi-objective optimization with interval numbers[J]. Control and Decision, 2015, 30(5): 865-870.)
- [15] Levchuk G M, Levchuk Y N, Luo J, et al. Normative design of organizations — Part I: Mission planning[J]. IEEE Trans on System, Man, and Cybernetics — Part A: System and Humans, 2002, 32(3): 346-357.
- [16] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multi-objective genetic algorithm: NSGA-II[J]. IEEE Trans on Evolutionary Computation, 2002, 6(2): 182-197.
- [17] Jiang C, Han X, Liu G P, et al. A nonlinear interval number programming method for uncertain optimization problems[J]. European J of Operational Research, 2008, 188 (1): 1-13.
- [18] 李霞, 张绍林, 张淼, 等. 基于新距离测度的区间数排序[J]. 西华大学学报: 自然科学版, 2008, 27(1): 87-90.
(Li X, Zhang S L, Zhang M, et al. Rank of interval numbers based on a new distance measure[J]. J of Xihua University: Natural Science Edition, 2008, 27(1): 87-90.)
- [19] Van V D A, Lamont G B. On measuring multi-objective evolutionary algorithm performance[C]. Proc of Congress on Evolutionary Computation. New York: IEEE Press, 2000: 204-211.

(责任编辑: 闫 妍)