

基于协方差矩阵调整的多目标多任务优化算法

邱鸿辉, 刘海林, 陈磊

(广东工业大学 应用数学学院, 广州 510520)

摘要: 多任务进化(EMT)是进化计算领域的一个新兴研究方向,区别于传统的单任务搜索算法,EMT通过在任务间传递有用知识,对多个任务同时实施进化搜索,以提升多个任务的收敛性能。目前,大多数进化算法只考虑了知识迁移而忽略了任务间的联系。提出一种多目标多任务优化算法,结合迁移学习的思想,采用任务间种群的协方差矩阵差异表示任务间种群分布特征差异,使用任务间种群均值的距离表示任务间种群的分布距离,并通过任务间种群的分布特征差异和分布距离表示任务间的相似度。对于某个目标任务,将其最相似任务中的解集实施K最近邻分类,以筛选出对目标任务有价值的解,并使其迁移到目标任务中。实验结果表明,与EMTSD、MaTEA、MO-MFEA-II等多目标多任务优化算法相比,所提算法具有较佳的收敛性能,平均运行效率约提高了66.62%。

关键词: 多目标多任务优化;进化算法;多任务进化;迁移学习;协方差矩阵

开放科学(资源服务)标志码(OSID):



中文引用格式:邱鸿辉,刘海林,陈磊.基于协方差矩阵调整的多目标多任务优化算法[J].计算机工程,2022,48(8):306-312.

英文引用格式:QIU H H, LIU H L, CHEN L. Multi-objective multi-tasking optimization algorithm based on adjustment of covariance matrix[J]. Computer Engineering, 2022, 48(8): 306-312.

Multi-Objective Multi-Tasking Optimization Algorithm Based on Adjustment of Covariance Matrix

QIU Honghui, LIU Hailin, CHEN Lei

(School of Applied Mathematics, Guangdong University of Technology, Guangzhou 510520, China)

[Abstract] Evolutionary Multi-Tasking (EMT) is an emerging research direction in the field of evolutionary computations. Differing from a traditional single-task search algorithm, EMT implements an evolutionary search for multiple tasks simultaneously by transferring useful knowledge between tasks, thereby improving the convergence of such tasks. At present, most evolutionary algorithms focus on knowledge transferring and ignore the connections between tasks. Therefore, by combining the ideas of transfer learning, a Multi-Objective Multi-Tasking Optimization (MTO) algorithm is proposed. First, the difference in the covariance matrix between tasks represents the difference in population distribution characteristics between tasks, and the distance between the means of different tasks is used to represent the distribution distance of the population between tasks. Thus, the difference in population distribution characteristics and the population distribution distance between tasks are used to express the similarity between tasks. For the target task, K-Nearest Neighbor (KNN) classification of the solutions to the most similar task is conducted, so as to screen out valuable solutions to target task and then transfer them to the target task. The experimental results show that, compared with EMTSD, MaTEA, MO-MFEA-II, and other MTO algorithms, the proposed algorithm achieves a better convergence, and the average running efficiency is increased by approximately 66.62%.

[Key words] Multi-Objective Multi-Tasking Optimization (MTO); evolutionary algorithm; Evolutionary Multi-Tasking (EMT); transfer learning; covariance matrix

DOI: 10.19678/j.issn.1000-3428.0062365

0 概述

进化算法是一类基于种群个体随机搜索的算法^[1-2],由于其强大的搜索能力及简单易用的特性,在路径规

划^[3]、软件工程^[4]等现实问题中被广泛的应用,并取得了良好的效果。

近年来,研究人员受到人类大脑能同时处理多个任务的启发,提出多任务进化(Evolutionary Multi-

基金项目:国家自然科学基金(62006044);广东省科技计划项目(2021A0505110004);广东省自然科学基金(2022A1515010130)。

作者简介:邱鸿辉(1996—),男,硕士研究生,主研方向为智能计算;刘海林,教授、博士生导师;陈磊,讲师、博士。

收稿日期:2021-08-16 修回日期:2021-09-22 E-mail: hlliu@gdut.edu.cn

Tasking, EMT)^[5-6]的研究方向, 传统的进化搜索方法一次只能解决一个任务, 然而EMT方法一次运行能够同时解决多个任务。研究表明, 与传统的单任务进化算法相比, 由于EMT方法利用了任务间潜在的协同作用, 使其在解决优化问题时具有更好的性能^[7]。EMT包括单目标多任务优化(Single-Objective Multi-Tasking Optimization, STO)^[8]和多目标多任务优化(Multi-Objective Multi-Tasking Optimization, MTO)。本文研究的是求解MTO问题^[9]的算法性能。

当前, 求解MTO问题的算法大致可以分为基于单个种群进化的算法和基于多个种群进化的算法。前者的典型代表是文献[9]提出的解决MTO问题的MO-MFEA算法, 该算法通过采用单个种群来同时优化多个相关的任务, 并以预先设定的概率决定是否要实施任务间的知识迁移。文献[10]在MO-MFEA算法的基础上引入粒子群优化^[11-12]和差分进化策略^[13-14], 提出解决MTO问题的新算法。实验结果表明, 上述多任务优化算法比其对应的单任务优化算法的性能更好。后者的典型代表是文献[15]提出的全新多种群多任务框架, 使用多个子种群同时解决多个任务, 每个子种群对应于解决单个任务。文献[16]在多个种群的条件下, 采用一种高效的知識迁移策略来促进多个任务的同时优化。大量实验表明, 多种群MTO框架由于能在多个种群之间互相独立, 可以灵活解决多个任务的优化问题, 因此比单种群MTO框架更具有优势。尽管当前解决MTO问题的算法较多, 但这些算法涉及到衡量任务间相似度的研究很少, 有些算法虽然包含衡量任务间相似度的策略, 但步骤较繁琐, 计算量较大, 难以推广应用。

本文提出一种多目标多任务优化算法, 基于自适应协方差矩阵的进化策略, 从某个正态分布中采样产生种群个体, 使种群分布特征具有规律性, 并采用任务间种群的分布特征差异和分布距离衡量任务间的相似程度。设计一种从相似任务中寻找有价值的解, 进而实施知识迁移的方法, 并基于迁移学习的思想, 对相似任务中的解实施K最近邻(K-Nearest Neighbor, KNN)分类, 以筛选出对目标任务更有价值的解。

1 相关工作

1.1 多目标多任务优化算法

多目标多任务优化是进化计算领域的一个重要研究方向。但目前大部分的多目标多任务优化算法主要解决2个待优化任务的问题, 较少涉及到研究超多任务优化问题。文献[17]阐明了使用进化算法解决现实生活中超多任务优化问题的可行性, 当待优化的任务数量超过2个时, 为避免过高的时间开销, 对于某个给定的目标任务, 从其余任务中选择一个与目标任务最相似的辅助任务, 进而实施任务间的知识迁移, 因此衡量任务间的相似性是解决超多任务优化问题的核心。文献[18]从最优解之间的距

离、适应度等级的相关性和适应值函数范围分析3个方面衡量任务间的相似性。文献[19]提出一个协同性度量指标量化不同任务间的联系, 这一度量指标可以从侧面证明任务间的知识迁移能够同时促进多个待优化任务的收敛性能。

基于任务间的相似性, 为求解超多任务优化问题, 文献[20]采用轮盘赌的方法为目标任务选择一个合适的辅助任务, 某个与目标任务最相似的任务有较大的概率被选为辅助任务, 从而通过迁移有用的知识来减少负迁移的产生。文献[21]通过同时考虑任务间的相似性和进化过程中知识迁移的累积奖励值, 提出一种自适应辅助任务选择机制, 采用KL散度衡量任务间的相似性, 并基于强化学习的思想, 构建关于种群进化过程中任务间知识迁移的累积奖励系统, 选择出合适的辅助任务, 并通过交叉的方式实现知识迁移。文献[22]基于任务间种群概率分布的差异自适应地学习任务间的联系, 并根据进化过程中求得的不同任务种群间的联系, 自适应调整知识迁移的强度。文献[21]和文献[22]中的算法均采用高斯分布来近似模拟种群的真实概率分布, 但实际上, 种群的真实分布通常是随机且不可预测的, 且算法花费大量的计算资源在衡量任务间的相似度上, 导致最终运行的效率较低。

1.2 自适应协方差矩阵进化算法CMA-ES

CMA-ES算法^[23-24]是目前性能较好、应用较多的进化策略之一, 在中等规模的复杂优化问题处理上具有良好的效果。CMA-ES算法的核心思想是: 每一代种群的所有个体均从某个正态分布群体 $N(m, \sigma^2 C)$ 中采样产生, 并通过调整协方差矩阵 C 和步长参数 σ , 使产生的种群个体尽可能往产生好解的搜索方向搜索, 从而增强算法的收敛速率。CMA-ES算法的具体步骤如下:

1) 采样产生新的种群个体。在CMA-ES算法中, 进化过程中的每一代种群均从 $N(m, \sigma^2 C)$ 产生进化的种群个体, 个体 x_i 更新的表达式如式(1)所示:

$$x_i = m + \sigma_i y_i, y_i \sim N(0, C) \quad (1)$$

其中: y_i 通过协方差矩阵 C 的特征分解 $C = BD^2B^T$ 得到, 即:

$$y_i = BDz_i, z_i \sim N(0, I) \quad (2)$$

其中: B 为正交矩阵; D 为对角矩阵且其对角元素值为矩阵 C 特征值的平方根; I 为单位矩阵。

2) 进化路径的更新。在CMA-ES算法中, 每一代种群的进化路径均通过如下的方式构造^[24]:

$$p_{t+1} = (1-c)p_t + \sqrt{c(2-c)} \sqrt{\mu_w} \frac{m_{t+1} - m_t}{\sigma_t} \quad (3)$$

其中: p_t 表示在第 t 代的进化路径; c 表示进化路径更新的累积学习率; μ_w 表示均值的有效选择质量。

式(3)描述了种群分布均值的移动方向, 将每一代种群进化过程中的移动方向做加权平均, 使得种群中每个个体相反的进化方向分量相互抵消, 相同的进化方向分量相互叠加, 从而保证种群沿着最优的方向搜索。

3) 协方差矩阵的更新。通过增大沿历史成功搜索最优方向的方差,即增大沿这些最优搜索方向的采样概率,避免算法把搜索资源浪费在无效的区域上。协方差矩阵的更新原理如式(4)所示^[24]:

$$\mathbf{C}_{t+1} = (1 - c_1 - c_\mu) \mathbf{C}_t + c_1 \mathbf{p}_{t+1} \mathbf{p}_{t+1}^T + c_\mu \sum_{i=1}^{\mu} \mathbf{y}_i \mathbf{y}_i^T \quad (4)$$

其中: \mathbf{y}_i 表示当前种群中较优个体的搜索方向; c_1 和 c_μ 表示协方差矩阵更新的学习率。

本文采用下述方法衡量任务间的相似度: 基于CMA-ES算法的思想, 每一代种群在进化过程中, 均从正态分布群体采样产生种群个体, 即种群的均值和协方差矩阵可以反映当前种群的特征, 因此通过种群的均值和协方差矩阵来表示任务之间的联系, 从而为目标任务选择出一个最合适的辅助任务来实施知识迁移, 加速目标任务的收敛进程。

2 本文算法

2.1 种群编码和解码

为便于在任务间高效地实施知识迁移, 目前许多任务优化算法采用统一空间表示法对不同任务的种群实施编码^[9]。具体为, 首先把所有任务的解集均编码到一个统一的搜索空间中。例如, 给定 K 个待优化的任务 $\{T_1, T_2, \dots, T_K\}$, 将第 i 个任务决策空间的维度记为 D_i ($i=1, 2, \dots, K$), 定义所有任务统一的搜索空间维度为 D_{\max} , 且 $D_{\max} = \max_i \{D_i\}$, 在种群进化的初始阶段, 给所有任务种群中的每个个体都分配一个 D_{\max} 维的随机向量, 该向量每个维度的变量范围为 $[0, 1]$, 通常把这个向量称为包含该个体所有遗传物质的染色体, 该编码策略把所有任务中的种群都统一到共同的搜索空间中, 以便于在不同的任务之间实施知识迁移。本文提出的衡量任务间相似度策略以及任务间知识迁移的方式均基于该编码策略, 且均在统一的搜索空间中进行。

当要对某个特定的任务 T_i 中的个体实施评估时, 必须把该个体的染色体解码为任务 T_i 对应问题的解。同时, 对于任务 T_i 中的个体, 只使用其对应染色体的前 D_i 维变量。本文仅考虑连续优化的情形, 例如对于任务 T_i 中某个解的第 i 维变量 y_i , 其取值范围是 $[L_i, U_i]$, 若其染色体前 D_i 维中对应的变量值为 x_i , 则二者通过如式(5)所示的方式实现转换:

$$y_i = L_i + (U_i - L_i) \times x_i \quad (5)$$

该解码策略把实际问题中的解与统一搜索空间中的染色体一一对应, 通过在统一搜索空间中进行染色体间的杂交变异, 从而在实际问题中增加每个任务最优解的多样性。

2.2 任务间相似度衡量

若有 K 个待优化的任务 $\{T_1, T_2, \dots, T_K\}$, 对于某个目标任务 T_i , 使用一个任务间的自适应相似度衡量策略来决定是否需要为目标任务 T_i 分配一个辅助任务, 以实施知识迁移, 促进目标任务的优化进程。该策略包含 2 个部分: 任务间种群分布特征差异的衡量及任务间种群分布距离的衡量。根据CMA-ES算

法的原理, 每个任务的种群个体经过正态分布采样后, 用种群的协方差矩阵表示种群的分布特征, 用任务间种群均值之间的距离表示任务间的分布距离, 从而衡量出任务间的相似性。

2.2.1 任务间种群分布特征差异

给定任务 T_1 的种群 $\mathbf{P} = \{\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_n\}$ 以及任务 T_2 的种群 $\mathbf{Q} = \{\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_n\}$, 他们分别从正态分布 $N_1(m_1, \sigma_1^2 \mathbf{C}_1)$ 和 $N_2(m_2, \sigma_2^2 \mathbf{C}_2)$ 中采样而来。每个任务中种群的协方差矩阵可以近似看作种群沿着最优解搜索方向的集合, 因此衡量任务间种群的协方差矩阵之间的差异可以度量任务间种群分布特征差异(种群中的所有个体均沿着最优解方向搜索)。基于主成分分析(Principal Component Analysis, PCA)的思想衡量任务 T_1 和任务 T_2 中种群协方差矩阵之间的差异。

首先, 对任务 T_1 和任务 T_2 中种群的协方差矩阵分别实施特征分解:

$$\mathbf{E}_1^T \mathbf{C}_1 \mathbf{E}_1 = \mathbf{\Delta}_1 = \text{diag}(\lambda_1^p, \lambda_2^p, \dots, \lambda_d^p) \quad (6)$$

$$\mathbf{E}_2^T \mathbf{C}_2 \mathbf{E}_2 = \mathbf{\Delta}_2 = \text{diag}(\lambda_1^q, \lambda_2^q, \dots, \lambda_d^q) \quad (7)$$

其中: \mathbf{E}_1 和 \mathbf{E}_2 分别由 \mathbf{C}_1 和 \mathbf{C}_2 中所有的特征向量构成; $\mathbf{\Delta}_1$ 和 $\mathbf{\Delta}_2$ 分别由 \mathbf{C}_1 和 \mathbf{C}_2 中所有的特征值 $\{\lambda_1^p, \lambda_2^p, \dots, \lambda_d^p\}$ 和 $\{\lambda_1^q, \lambda_2^q, \dots, \lambda_d^q\}$ 构成对角矩阵。

接着, 分别从 \mathbf{E}_1 和 \mathbf{E}_2 中选出 k 个特征向量, 这些特征向量对应 \mathbf{C}_1 和 \mathbf{C}_2 中最大的 k 个特征值, 根据PCA中的可解释方差比例指标, 若保留的这 k 个主成分的累积方差贡献率大于等于 95%, 即选出的主成分至少能分别保留 \mathbf{C}_1 和 \mathbf{C}_2 中 95% 的信息, 使用从 \mathbf{E}_1 和 \mathbf{E}_2 中选出的 k 个特征向量来构建种群 \mathbf{P} 和 \mathbf{Q} 的子空间 \mathbf{P}_s 和 \mathbf{P}_t , 即:

$$\mathbf{P}_s = \mathbf{E}_1^{\text{sort}}(1:k) \quad (8)$$

$$\mathbf{P}_t = \mathbf{E}_2^{\text{sort}}(1:k) \quad (9)$$

其中: $\mathbf{E}_{i(2)}^{\text{sort}}(1:k)$ 把 $\mathbf{E}_{i(2)}$ 中的特征向量按照其对应的特征值降序排列, 并保留最前的 k 个特征向量。

最后, 由于 \mathbf{P}_s 和 \mathbf{P}_t 中的列向量均为单位向量, 故对 \mathbf{P}_s 和 \mathbf{P}_t 中的列向量之间做内积, 表示 \mathbf{P}_s 和 \mathbf{P}_t 中的列向量之间角度的余弦值。若 \mathbf{P}_s 中的一个列向量为 \mathbf{e}_i , 从 \mathbf{P}_t 中找到一个列向量 \mathbf{f}_i , 使 \mathbf{e}_i 和 \mathbf{f}_i 之间的余弦值最大(夹角最小), 并记 \mathbf{e}_i 和 \mathbf{f}_i 之间的夹角为 θ_i 。照此步骤执行 k 次, 对应的夹角分别记为 $\theta_1, \theta_2, \dots, \theta_k$, 采用这些夹角正弦值的平均值来表示任务间种群协方差矩阵之间的差异, 即:

$$\text{Covd}(T_1, T_2) = \frac{\sin \theta_1 + \sin \theta_2 + \dots + \sin \theta_k}{k} \quad (10)$$

其中: $\text{Covd}(T_1, T_2)$ 的值越小, 表明任务 T_1 和任务 T_2 中种群的分布特征越相似, 即任务 T_1 和任务 T_2 中种群搜索最优解的方向越相似。

2.2.2 任务间种群分布距离

若任务 T_1 的种群 \mathbf{P} 和任务 T_2 的种群 \mathbf{Q} 分别从正态分布 $N_1(m_1, \sigma_1^2 \mathbf{C}_1)$ 和 $N_2(m_2, \sigma_2^2 \mathbf{C}_2)$ 中采样产生, 则任务 T_1 和任务 T_2 种群的均值可以分别看作任务 T_1 和任务 T_2 中种群的中心, 因此任务间种群均值间的距离可以近似看作任务间种群的分布距离。

任务 T_1 和任务 T_2 之间的种群分布距离的表达式如式(11)所示:

$$\text{Dis}(T_1, T_2) = \|m_1 - m_2\|_2^2 \quad (11)$$

其中: 任务 T_1 和任务 T_2 的种群均值间的距离(种群中心之间的距离)表示任务 T_1 和任务 T_2 的种群分布间的交叉程度。若任务 T_1 和任务 T_2 之间的种群分布距离较近, 则在任务 T_1 和任务 T_2 之间实施知识迁移能够促进每个任务的优化进程。

2.2.3 任务间种群相似度衡量

基于CMA-ES算法的思想, 每个任务中的种群均通过正态分布采样产生, 在种群进化的前期阶段, 每个任务中的种群均朝着各自的最优解方向搜索, 种群的协方差矩阵可以反映种群沿着最优解方向搜索时的分布特征, 因此在种群进化的前期, 任务间种群的相似度主要由任务间种群分布特征的差异决定。在种群进化的后期阶段, 大部分任务中的种群已经收敛到了最优解, 此时任务间种群的相似度主要由任务间种群分布的距离决定。采用式(12)来表达种群进化全过程中, 任务 T_1 和任务 T_2 之间种群的相似度:

$$\text{Sim}(T_1, T_2) = (1 - \lambda) \text{Covd}(T_1, T_2) + \lambda \text{Dis}(T_1, T_2) \quad (12)$$

其中: λ 表示任务间种群的分布特征差异和分布距离之间的权重值, 令 $\lambda = \frac{g}{m}$, g 表示当前的进化代数, m 表示进化的最大代数。式(12)表示在种群进化的前期阶段, 任务间种群的相似度主要由任务间种群分布特征的差异决定, 在种群进化的后期阶段, 任务间种群的相似度主要由任务间种群分布的距离决定。若 $\text{Sim}(T_1, T_2)$ 的值较小, 则表明任务 T_1 和任务 T_2 的 Pareto 最优解的方向和位置较接近, 在任务 T_1 和任务 T_2 之间实施知识迁移能够有效促进每个待优化任务的收敛进程。

2.3 任务间知识迁移策略

图1所示为本文算法任务间知识迁移策略的具体流程图。可以看到, 在种群进化的第 g 代, 对于某个待优化的目标任务 T_i , 根据本文策略可以找到与目标任务 T_i 最相似的任务是 T_a 。依据迁移学习的思想, 把目标任务 T_i 种群中的解分成两类, 一类是非支配解, 标记为“1”, 另一类是支配解, 标记为“0”, 把这些解作为给任务 T_a 中解分类的源标签, 本文采用K最近邻(KNN)分类, 具体的分类方式为, 对于任务 T_a 中的某个解 P , 找出目标任务 T_i 中距离解 P 最近的2个解 Q_1 和 Q_2 , 并记录他们的标签为 L_a 、 L_b , 若 $L_a + L_b = 2$, 则把解 P 标记为“1”, 否则记为“0”。接着, 对于任务 T_a 中标记为“1”的解(近似目标任务中的非支配解), 将他们迁移到目标任务 T_i 的种群中, 从而促进目标任务的优化进程。若根据式(12)计算得到目标任务 T_i 与任务 T_a 的相似度在逐代降低, 表明此时所

有任务均与目标任务 T_i 不相似, 为避免无效的迁移, 应停止知识迁移, 让目标任务 T_i 中的种群个体之间独立进化产生后代。

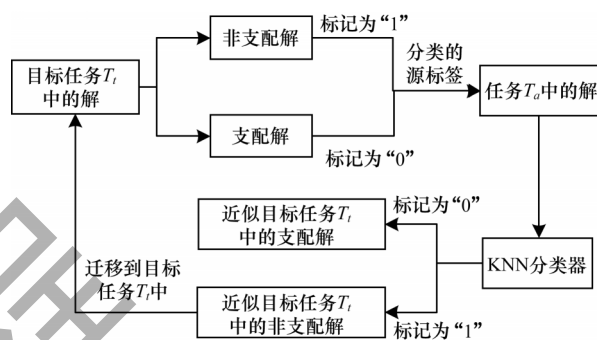


图1 任务间知识迁移策略流程

Fig.1 Procedure of knowledge transfer strategy between tasks

2.4 算法整体框架

综合以上过程, 本节给出解决多目标多任务优化问题的 EMTSD 算法。若给定一个有 K 个任务的多目标多任务优化问题, EMTSD 算法通过进化 K 个独立的种群来同时优化这些任务。首先, EMTSD 算法初始化 K 个种群 P_1, P_2, \dots, P_K , 并按照 2.1 节中的策略将这 K 个种群中的所有个体均编码到一个统一的空间中, 便于各个任务之间的知识实施互相迁移。然后, 对于某个特定的目标任务 T_i , 按照 2.2 节的过程, 根据式(12)从其余 $K-1$ 个任务中找出当前与目标任务 T_i 最相似的任务 T_a , 并按照 2.3 节的策略在目标任务 T_i 与任务 T_a 之间实施知识迁移, 从而促进目标任务 T_i 的收敛进程。若在进化过程中发现目标任务 T_i 与其余任务的相似度降低, 则停止知识迁移, 让目标任务 T_i 种群内部中的个体独立产生后代。EMTSD 算法的具体框架如图 2 所示。

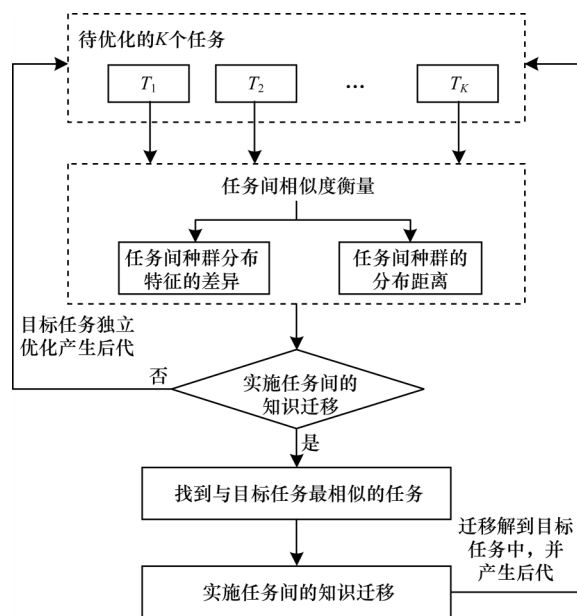


图2 EMTSD算法的整体框架

Fig.2 Overall framework of the EMTSD algorithm

3 实验结果与分析

为验证本文算法EMTSD的性能,本文使用文献[25]提出的多目标多任务测试集,包括CIHS、CIMS、PIHS、PIMS等,每个测试集由2个多目标优化问题(2个任务)构成。通过对这些测试集的Pareto Sets进行旋转和平移,得到4组新的多目标多任务测试集,分别记为MATP1~MATP4,这4组测试集是基于任务间联系的先验知识创建的,任务间的联系通过多目标优化问题的Pareto Sets重合程度来衡量。每个测试集包含5个任务,这些测试问题的详细信息如表1所示。

表1 多目标多任务测试问题信息
Table 1 Information about multi-objective multi-tasking test problems

测试集	函数形式	函数特征
MATP1	CIHS-T1	单峰,可分离
	CIHS-T2	单峰,可分离
	CIHS-T1+Shifted+Rotated	单峰,可分离
	CIHS-T2+Shifted+Rotated	单峰,可分离
	CIHS-T1+Shifted+Rotated	单峰,可分离
MATP2	CIMS-T1	多峰,不可分离
	CIMS-T2	单峰,不可分离
	CIMS-T1+Shifted+Rotated	多峰,不可分离
	CIMS-T2+Shifted+Rotated	单峰,不可分离
	CIMS-T1+Shifted+Rotated	多峰,不可分离
MATP3	PIHS-T1	单峰,可分离
	PIHS-T2	多峰,可分离
	PIHS-T1+Shifted+Rotated	单峰,可分离
	PIHS-T2+Shifted+Rotated	多峰,可分离
	PIHS-T1+Shifted+Rotated	单峰,可分离
MATP4	PIMS-T1	单峰,不可分离
	PIMS-T2	多峰,不可分离
	PIMS-T1+Shifted+Rotated	单峰,不可分离
	PIMS-T2+Shifted+Rotated	多峰,不可分离
	PIMS-T1+Shifted+Rotated	单峰,不可分离

在上述4组测试函数集的基础上,将本文算法EMTSD与MO-MFEA-II^[22]、MaTEA^[21]、MO-MFEA^[9]等多目标多任务算法进行比较。其中:MO-MFEA-II算法采用最小化任务间KL散度的方式找出任务间的联系;MaTEA算法采用KL散度和自适应奖励策略的方式为目标任务选择一个最合适的辅助任务;MO-MFEA算法采用随机迁移的方式实施任务间的知识迁移。

3.1 实验参数设置

将实验中每个任务的种群规模设置为100,算法的最大进化代数为500代,算法的独立运行次数为20次。对于多目标多任务优化算法,多目标优化算法是NSGA-II。在DE算法中,令 $F=0.5$ 、 $C_{CR}=0.4$ 。根据文献[24],CMA-ES算法中的步长参数 $\sigma_r=0.3$,步长更新的阻尼参数 $d_r=1$ 。根据文献[9],将MO-MFEA算法中的随机交配概率值

设置为0.3。根据文献[21],将MaTEA算法中的知识迁移率设置为 $\alpha=0.1$,令精英集更新率 $U_{UR}=0.2$,精英集最大规模 $A_{AcS}=300$ 。MO-MFEA-II算法中使用的概率模型为多元正态分布。

3.2 算法性能表征

本文采用反转世代距离(Inverted Generational Distance,IGD)来表征算法的性能。IGD的原理是计算Pareto最优解集中的点到算法最终得到的解之间的平均距离。IGD的值越小,表明算法最终得到的解接近问题的Pareto前沿界面且具有良好的分布性。IGD值通过式(13)计算:

$$I_{IGD} = \frac{\sum_{x \in P^*} \text{dist}(x, A)}{|P^*|}$$

(13)

其中: $\text{dist}(x, A)$ 表示Pareto前沿界面上的参考点 x 到算法最终求解得到的解之间的最小距离; P^* 表示Pareto前沿界面上预设的参考点集合。

3.3 结果分析

表2所示为EMTSD算法、MaTEA算法、MO-MFEA-II算法与MO-MFEA算法在4组多任务测试集下运行20次得到的平均IGD值,表中加粗数字表示该组数据最大值。

表2 不同算法在测试集下运行的平均IGD值对比
Table 2 Comparison of average IGD value of different algorithms running on the test suites

测试集	任务序号	EMTSD 算法	MaTEA 算法	MO-MFEA-II 算法	MO-MFEA 算法
MATP1	T_1	1.82E-04	4.88E+01	1.95E-04	2.53E-03
	T_2	1.82E-04	1.44E+00	1.19E-03	8.62E-03
	T_3	1.82E-04	6.52E+01	2.39E-04	4.92E-03
	T_4	2.77E-04	1.56E+00	1.73E-03	1.36E-02
	T_5	1.05E-03	1.61E+00	3.13E-03	7.62E-02
MATP2	T_1	1.75E-04	1.67E-04	1.68E-04	1.65E-04
	T_2	1.63E-04	1.73E-04	1.87E-04	1.77E-04
	T_3	1.75E-04	1.65E-04	1.68E-04	1.63E-04
	T_4	1.72E-04	1.62E-04	1.78E-04	1.65E-04
	T_5	1.69E-04	1.61E-04	1.62E-04	1.60E-04
MATP3	T_1	1.83E-04	1.93E+00	2.72E-02	2.16E-02
	T_2	1.85E-04	3.50E-04	3.05E-04	8.99E-04
	T_3	8.86E+00	1.85E+00	1.36E-03	2.09E-02
	T_4	6.28E-03	3.48E-04	2.65E-04	8.15E-04
	T_5	2.46E-04	3.58E-04	3.62E-04	1.18E-03
MATP4	T_1	1.69E-04	2.67E-04	3.00E-04	6.17E-03
	T_2	2.50E+00	3.35E+00	7.61E-01	2.74E+00
	T_3	1.80E-04	2.65E-04	2.94E-04	5.43E-03
	T_4	2.06E-01	2.94E+00	1.08E+00	3.52E+00
	T_5	2.60E+00	3.27E+00	3.91E+00	8.78E+00

从表2中可以看到,与其他算法相比,EMTSD算法在大多数测试问题中均显示出卓越的性能,这是因

为在每一代种群进化过程中, EMTSD 算法均从正态分布群体中采样产生新的种群个体, 通过任务间种群的分布特征差异和分布距离找到任务间相似度, 进而实施任务间的知识迁移, 促进种群中解的多样性并加速目标任务的收敛。MaTEA 算法和 MO-MFEA-II 算法均采用多元高斯分布来近似模拟种群的真实分布, 然而这些算法中的种群个体通过杂交变异产生, 并不像 EMTSD 算法通过在多元高斯分布中采样产生种群个体, 这样会导致真实的种群分布是随机且不可预测的, 因此多元高斯分布通常很难准确近似为真实的种群分布。对于 MO-MFEA 算法, 由于缺乏衡量任务间相似度的步骤, 因此在任务间随机进行知识迁移时, 难以保证知识迁移会对目标任务起作用。然而, EMTSD 算法在 MATP2- T_1 、MATP2- T_3 、MATP2- T_4 、MATP2- T_5 、MATP3- T_3 、MATP3- T_4 、MATP4- T_2 这 7 个任务中的性能比其他算法差。在实验中发现, 测试问题集 MATP2 和 MATP3 中各个测试问题的 Pareto Sets 之间较相似, 因此在其他算法中, 针对某个目标任务, 从最相似的任务中随机选择解来实施知识迁移的策略, 能够帮助目标任务跳出局部最优, 找到更多分布性良好的最优解。

表 3 所示为 EMTSD 算法、MaTEA 算法、MO-MFEA-II 算法与 MO-MFEA 算法在 4 组多任务测试集下运行 20 次的平均运行时间对比。由于 EMTSD 算法、MaTEA 算法和 MO-MFEA-II 算法衡量任务间相似度的方式不同, 因此每个算法在衡量任务间相似度方面花费的计算资差异直接影响了各个算法最终运行时间的差异。

表 3 不同算法在测试集运行时的平均运行时间对比

Table 3 Comparson of average running time of different algorithms

测试函集	EMTSD 算法	MaTEA 算法	MO-MFEA-II 算法	MO-MFEA 算法
MATP1	330.57	739.91	1 492.57	34.19
MATP2	119.56	588.76	1 471.38	30.31
MATP3	331.22	957.78	1 426.90	33.85
MATP4	327.53	1 035.58	1 442.67	36.44

由表 3 可知, EMTSD 算法的运行时间比 MaTaE 算法和 MO-MFEA-II 算法都要短, MO-MFEA-II 算法运行的时间最长。这主要是因为 MO-MFEA-II 算法需要构建概率模型来近似种群的真实分布, 而求解该概率模型需要花费大量的计算资源。此外, EMTSD 算法在 4 组多任务测试集下的平均运行时间比 MaTEA 算法缩短约 66.62%。尽管 MO-MFEA 算法只需要很短的运行时间, 但它缺乏衡量任务间相似度这一步骤, 随机实施知识迁移的做法会导致算法的性能较差。

为验证 EMTSD 算法在衡量任务间相似度方面的有效性, 表 4 和表 5 分别列出了 EMTSD 算法在 MATP1 和 MATP2 测试集中计算得到的任务间相似度, 其中“—”表示忽略该栏的值。由表 4 和表 5 可知, 在 MATP1 测试集中, 任务 T_1 与任务 T_2 较相似, 任务 T_3 与任务 T_4 较相似, 任务 T_5 与其他任务之间的相似度较低。在 MATP2 测试集中, 任务 T_1 与任务 T_2 之间、任务 T_3 与任务 T_4 之间的相似程度均较高, 任务 T_5 与其他任务之间的相似度较低。实验得出的结果与事先已知的任务间相似度高度吻合, 因此可以说明 EMTSD 算法衡量任务间相似度的方法是有效的。

表 4 EMTSD 算法在 MATP1 测试集中得到的任务间相似度

Table 4 Similarity between tasks in the MATP1 test set obtained by EMTSD algorithm

任务序号	T_1	T_2	T_3	T_4	T_5
T_1	—	0.003	1.000	1.000	0.959
T_2	0.003	—	1.000	1.000	0.959
T_3	0.707	0.707	—	0.001	1.000
T_4	0.707	0.707	0.001	—	1.000
T_5	0.678	0.678	1.000	1.000	—

表 5 EMTSD 算法在 MATP2 测试集中得到的任务间相似度

Table 5 Similarity between tasks in the MATP2 test set obtained by EMTSD algorithm

任务序号	T_1	T_2	T_3	T_4	T_5
T_1	—	0.023	0.999	1.000	1.000
T_2	0.023	—	1.000	1.000	1.000
T_3	0.500	0.500	—	0.016	1.000
T_4	0.500	0.500	0.016	—	1.000
T_5	0.500	0.500	1.000	1.000	—

4 结束语

本文提出一种基于自适应协方差矩阵调整的多目标多任务优化算法, 令所有任务中的种群个体均通过正态分布采样产生, 并采用不同任务种群间的分布特征差异和分布距离得到任务间的相似度, 从而找到与目标任务最相似的任务并实施知识迁移。采用种群的协方差矩阵反映种群的分布特征, 使用不同种群均值之间的距离反映任务间种群的分布距离。实验结果表明, 与 EMTSD、MaTEA、MO-MFEA-II 等多目标多任务优化算法相比, 本文算法具有较好的收敛性能, 平均运行效率提高了约 66.62%。下一步将结合迁移学习中特征迁移的相关理论, 使本文算法能应用于复杂的多目标多任务优化问题。

参考文献

- [1] PISARSKI S, RUGALA A, BYRSKI A, et al. Evolutionary multi-agent system in hard benchmark continuous optimisation[M]. Berlin, Heidelberg: Springer, 2013: 132-141.
- [2] 李智翔, 李赞, 褚衍杰. 基于改进平衡策略的多目标分解优化算法[J]. 计算机工程, 2019, 45(3): 155-161.
- LI Z X, LI Y, CHU Y J. Decomposition multi-objective optimization algorithm based on improved balancing strategy[J]. Computer Engineering, 2019, 45(3): 155-161. (in Chinese)
- [3] ZHU Z X, WANG F X, HE S, et al. Global path planning of mobile robots using a memetic algorithm[J]. International Journal of Systems Science, 2015, 46(11): 1982-1993.
- [4] HIERONS R M, LI M Q, LIU X H, et al. SIP: optimal product selection from feature models using many-objective evolutionary optimization[J]. ACM Transactions on Software Engineering and Methodology, 2016, 25(2): 17-29.
- [5] ONG Y S, GUPTA A. Evolutionary multitasking: a computer science view of cognitive multi-tasking[J]. Cognitive Computation, 2016, 8(2): 125-142.
- [6] XU Q Z, WANG N, WANG L, et al. Multi-task optimization and multi-task evolutionary computation in the past five years: a brief review[J]. Mathematics, 2021, 9(8): 864-873.
- [7] YANG C E, DING J L, JIN Y C, et al. Multitasking multi-objective evolutionary operational indices optimization of beneficiation processes[J]. IEEE Transactions on Automation Science and Engineering, 2019, 16(3): 1046-1057.
- [8] BALI K K, GUPTA A, FENG L, et al. Linearized domain adaptation in evolutionary multitasking[C]//Proceedings of 2017 IEEE Congress on Evolutionary Computation. Washington D. C., USA: IEEE Press, 2017: 1295-1302.
- [9] GUPTA A, ONG Y S, FENG L, et al. Multi objective multifactorial optimization in evolutionary multitasking[J]. IEEE Transactions on Cybernetics, 2017, 47(7): 1652-1665.
- [10] FENG L, ZHOU L, ZHONG J H, et al. Evolutionary multitasking via explicit autoencoding[J]. IEEE Transactions on Cybernetics, 2019, 49(9): 3457-3470.
- [11] SHEN Y, LI Y, KANG H W, et al. Research on swarm size of multi-swarm particle swarm optimization algorithm[C]//Proceedings of the 4th International Conference on Computer and Communications. Washington D. C., USA: IEEE Press, 2018: 37-43.
- [12] 邱宁佳, 李娜, 胡小娟, 等. 基于粒子群优化的朴素贝叶斯改进算法[J]. 计算机工程, 2018, 44(11): 27-32, 39.
- QIU N J, LI N, HU X J, et al. Improved native Bayes algorithm based on particle swarm optimization[J]. Computer Engineering, 2018, 44(11): 27-32, 39. (in Chinese)
- [13] 刘树强, 秦进. 一种求解动态优化问题的改进自适应差分进化算法[J]. 计算机工程, 2021, 47(4): 84-91, 99.
- LIU S Q, QIN J. An improved self-adaptive differential evolution algorithm for solving dynamic optimization problem[J]. Computer Engineering, 2021, 47(4): 84-91, 99. (in Chinese)
- [14] 吕铭晟, 沈洪远, 李志高, 等. 多变异策略差分进化算法的研究与应用[J]. 计算机工程, 2014, 40(12): 146-150.
- LÜ M S, SHEN H Y, LI Z G, et al. Research and application of differential evolution algorithm under multiple mutation strategy[J]. Computer Engineering, 2014, 40(12): 146-150. (in Chinese)
- [15] CHEN Y L, ZHONG J H, TAN M K. A fast memetic multi-objective differential evolution for multi-tasking optimization[C]//Proceedings of 2018 IEEE Congress on Evolutionary Computation. Washington D. C., USA: IEEE Press, 2018: 1-8.
- [16] LIN J B, LIU H L, TAN K C, et al. An effective knowledge transfer approach for multi objective multitasking optimization[J]. IEEE Transactions on Cybernetics, 2021, 51(6): 3238-3248.
- [17] SAGARNA R, ONG Y S. Concurrently searching branches in software tests generation through multitask evolution[C]//Proceedings of 2016 IEEE Symposium Series on Computational Intelligence. Washington D. C., USA: IEEE Press, 2016: 46-56.
- [18] ZHOU L, FENG L, ZHONG J H, et al. A study of similarity measure between tasks for multifactorial evolutionary algorithm[C]//Proceedings of the Genetic and Evolutionary Computation Conference Companion. New York, USA: ACM Press, 2018: 39-46.
- [19] GUPTA A, ONG Y S, DA B, et al. Landscape synergy in evolutionary multitasking[C]//Proceedings of 2016 IEEE Congress on Evolutionary Computation. Washington D. C., USA: IEEE Press, 2016: 112-129.
- [20] LIANG J, QIAO K J, YUAN M H, et al. Evolutionary multi-task optimization for parameters extraction of photovoltaic models[J]. Energy Conversion and Management, 2020, 207: 112509-112524.
- [21] CHEN Y L, ZHONG J H, FENG L, et al. An adaptive archive-based evolutionary framework for many-task optimization[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2020, 4(3): 369-384.
- [22] BALI K K, GUPTA A, ONG Y S, et al. Cognizant multitasking in multiobjective multifactorial evolution: MO-MFEA-II[J]. IEEE Transactions on Cybernetics, 2021, 51(4): 1784-1796.
- [23] JIA Y H, ZHOU Y R, LIN Y, et al. A distributed cooperative co-evolutionary CMA evolution strategy for global optimization of large-scale overlapping problems[J]. IEEE Access, 2019, 7: 19821-19834.
- [24] HANSEN N. The CMA evolution strategy: a comparing review [EB/OL]. [2021-07-09]. https://link.springer.com/chapter/10.1007/3-540-32494-1_4.
- [25] YUAN Y, ONG Y S, FENG L, et al. Evolutionary multitasking for multiobjective continuous optimization: benchmark problems, performance metrics and baseline results [EB/OL]. [2021-07-09]. <https://arxiv.org/abs/1706.02766>.