一种基于可变粒度搜索的面向高维数据分类的多目标特征选择算法

摘要

进化算法（EAs）可以在解决特征选择问题中有较为良好的表现。现如今，绝大多数使用进化算法的特征选择方法使用的是一位代表一个特征。当特征数量不断增加时，如果使用这一方法将可能会导致搜索空间呈指数型增长。所以，这样可能并不适用与高维数据分类。为克服这一难题，该篇文章提出了一种基于可变粒度搜索的多目标进化算法，名为VGS-MOEA，用于高维特征选择。在该算法中，个体表示中的每一位代表一组特征，这大大减小了搜索空间。具体而言，一开始，VGS-MOEA 的搜索粒度较粗，即一位代表大量的特征，这有助于所提出的算法快速检测出潜在的优质特征子集。随着进化的持续进行，搜索粒度逐渐细化，一位所代表的特征数量逐渐减少，直至其只代表一个特征。得益于这种粒度分解，可以进行更精细的搜索，从而使 VGS-MOEA 能够获得质量更高的特征子集。

关键词：特征选择；高维数据分类；进化算法；多目标优化

1 引言

特征选择任务是机器学习以及深度学习的重要环节之一。现如今，在高维数据中计算量通常非常庞大，特征选择可以有效减少特征的数量，从而显著降低计算开销，提升训练和推理的效率。进化算法在高维特征选择任务中表现良好。本次复现论文选择的是“A Variable Granularity Search Based Multi-Objective Feature Selection Algorithm for High-Dimensional Data Classification” [2]，该篇文章描述了一个多目标特征选择算法，可以有效的提升特征选择任务在高维数据的情况下的性能。

2 相关工作

在这个部分，将首先介绍了解到的相关基于进化计算的特征选择算法与多目标优化任务，并且介绍最近几年的基于进化计算的高维特征选择算法。

2.1 进化计算的特征选择算法

在机器学习中，特征选择是从原始数据集中选择一个较小的、重要的特征子集的过程。目的是去除冗余和不相关的特征，以提高模型的性能、减少计算成本、减少过拟合，并提升模型的解释性。特征选择是一个 NP 难题，特别是当特征空间非常庞大时，传统方法可能不适用。

基于进化计算的特征选择方法通过模拟自然进化过程，利用进化计算方法在特征选择问题上进行全局优化。它能够处理高维和复杂的数据集，具有较强的全局搜索能力和适应性，尤其在特征选择的精确性和自动化方面表现突出。然而，计算开销大和收敛速度慢等挑战也需要在实际应用中进行权衡和优化。近期研究表明，由于进化计算具有强大的搜索能力，其在特征选择方面极具应用前景 [18]。

在进化计算应用于特征选择任务中，最为经典的应用之一是 Tan 等人 [13] 提出了一种基于遗传算法的特征选择方法，在该方法中，遗传算法与多种特征选择标准相结合，以获得规模更小且分类准确率更高的特征子集。

2.2 多目标优化问题

在实际工程应用中，人们面临的很多优化问题都是多目标优化问题。如何获得这些问题的最优解决方案一直是学术界和工程界关注的焦点。通过模仿某种自然现象或过程建立的智能优化算法是一类优化方法 [7]。与数学规划方法相比，智能优化算法更适合求解多目标优化问题 [8] 。

与单目标优化问题不同，多目标问题之间追求的目标往往会相互影响：一个目标的评估指数提高往往伴随着另一个目标评估指数的降低。所以，多目标问题并不能达到所有目标全部最优的状态。我们所能做的就是对各个目标进行调节，使所有目标函数尽可能达到最佳状态，并求得问题的最优解。有多类传统的多目标优化研究方法，比如分量加权和法、理想点法、主要目标与评价函数法等等。这些方法都是基于单目标来寻找解决方案的。帕累托解集是近来在多目标优化问题领域中广泛应用的一种目标优化问题解决方案。其特点是能够利用帕累托优胜关系来比较优劣。在选择复现的这篇文章中，帕累托解集、帕累托前沿是重要的选择解的方法。

2.3 面向高维数据分类的基于进化计算的特征选择算法

基于进化计算（EC）的算法在解决特征选择问题方面已展现出其竞争力。然而，在一些实际的特征选择（FS）应用中，数据中的特征数量非常庞大，这给现有的基于进化计算的特征选择算法带来了巨大挑战。为应对这一挑战，近期人们已经付出诸多努力来设计用于高维特征选择的基于进化计算的算法，其中一系列具有代表性的成果是基于粒子群优化（PSO）框架的算法 [2]。

粒子群优化算法（PSO）并不适用于多目标选择任务。在单目标优化的情境下，粒子运动轨迹存在明确导向性，其个体历史进程中所抵达的最优位置，会径直成为后续行动的个体最优参照点；与此同时，群体在当下时刻汇聚而成的全局最优位置，也将精准锚定粒子下一步的全局最优走向。然而一旦切换至多目标优化场景，情况便截然不同。此时，由于问题的复杂性与目标的多元性交织，最优解不再单一呈现，而是呈现出多样化的态势，散布于一个复杂的解空间之中。这就使得在单目标优化里能够直接获取的个体与全局最优位置这两个关键量，在多目标优化的迷宫中失去了直接判定的明晰路径，变得难以简单、直接地确定。

于是有一种专门的基于 PSO 的多目标算法：在 Coello, Carlos A. Coello 提出的方法中 [3] ，多目标粒子群优化算法采用了多目标进化算法中常用的归档机制存在一个外部归档库，它始终保存着整个迭代过程中的非劣解。群体中每个个体的全局最优位置是由这个集合随机生成的。这里的外部文件就是所有非劣解的集合，也被称作非支配解。如果一个解不受解集中

任何其他解的支配，那么就称该解相对于这个解集是非劣的或非受支配的。多目标粒子群优化算法的研究过程就是为每一代种群寻找其当前最优解，也就是帕累托解，然后通过计算适应度来更新粒子的速度、位置以及非劣解集，使粒子的位置逐渐趋近于帕累托最优前沿，最终实现优化。

Huang Fang 等人提出了一种基于岛屿种群模型的并行粒子群优化算法 [6] ，该种算法与选择复现的这篇文章具有一定的相似性。将单个群体划分为多个子群体，采用分而治之的方法；二是对子群体之间的信息交换进行控制和管理，不同的划分方法会产生不同的算法结构。结构上的差异导致了诸如主从模型、岛屿模型等不同的并行群体模型的出现。这样一种并行粒子群优化算法，在设计之初就锚定了一个目标，那就是要以更短的耗时、更少的计算资源投入，挖掘出数量更多且质量更优的可行解，进而在多目标问题的求解成效上超越传统的多目标优化算法。其主要呈现出两大显著特性 [6] ：一方面，是把单一的大群体拆解为若干个子群体，运用化整为零、分而治之的策略，将复杂的优化任务分摊开来；另一方面，则聚焦于对子群体间信息交互流程的把控与统筹，不一样的子群体划分模式，催生出各异的算法内在结构。而正是这些结构层面的差别，衍生出了像主从模型、岛屿模型这类风格迥异的并行群体模型。

大多数现有的用于高维特征选择的进化算法（EAs）是通过开发各种问题导向的算子或策略来减小巨大的搜索空间，以此解决“维数灾难”问题，然而，它们的搜索粒度是固定的，设定为一（即一位代表一个特征）。与上述提到的并行粒子群优化算法的思想相似：将单个群体划分为多个子群体，采用分而治之的方法；二是对子群体之间的信息交换进行控制和管理，不同的划分方法会产生不同的算法结构 [6]，选择复现的这篇文章是一种基于可变粒度搜索的多目标进化算法（VGS-MOEA），在该算法的进化过程中采用不同大小的搜索粒度。这确保了 VGS-MOEA 能在更短的运行时间内，用更小的特征子集实现更高的分类准确率。下面，我们将详细探讨所提出的 VGS-MOEA [2]。

3 本文方法

3.1 本文方法概述

此部分对本文将要复现的工作进行概述，图的插入如图 3.4所示：

本文提出了一种基于可变粒度搜索的多目标进化算法（VGS-MOEA）用于高维数据分类中的特征选择，总体框架如下：

以准确率和所选特征数量作为优化目标，通过初始化阶段（包括粒度初始化、种群初始化和存档初始化）、主循环阶段（包括遗传操作、种群更新和存档更新）和输出阶段来实现算法流程。在主循环中，结合存档与当前种群形成交配池，根据个体粒度选择遗传算子产生后代，每　代进行粒度细化，直到满足停止准则，最后输出存档中的个体。

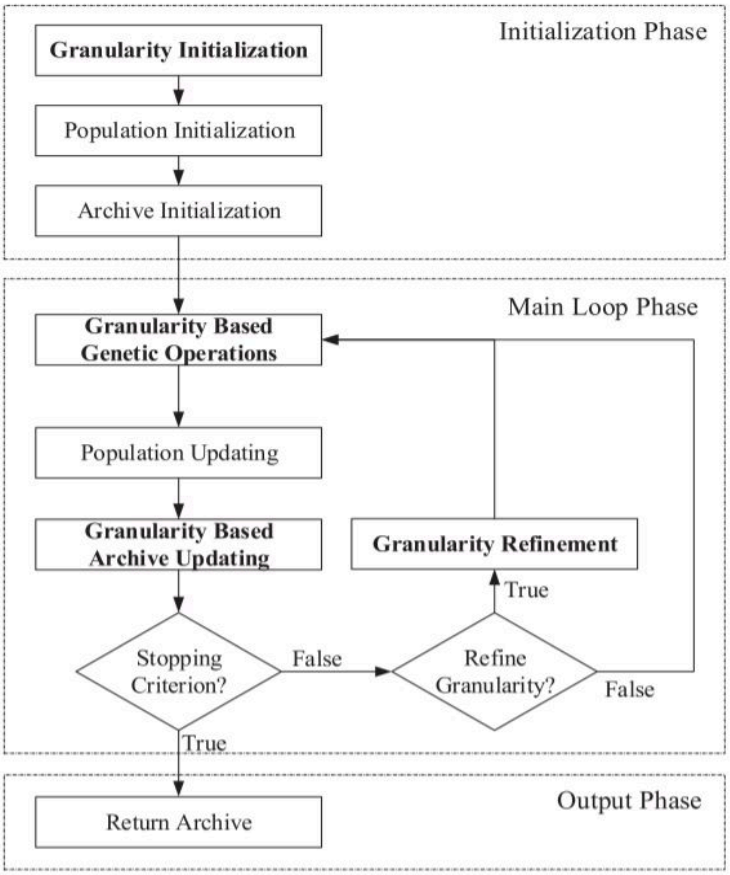

图 1. 所提出的 VGS-MOEA 的总体流程图

3.2 可变粒度搜索方案

大多数现有的基于进化算法（EA）的特征选择算法，如前所述，都是用一位来表示一个特征。这就导致在总特征数量庞大时，个体的搜索空间会变得极为巨大，给现有算法带来严峻挑战。针对此问题，提出了 VGS 可变粒度搜索方案。其基本思路为，鉴于高维特征选择问题中存在大量不相关或冗余特征，在搜索初始阶段，将搜索粒度设置得较为粗放，即一位代表一组特征，如此一来，搜索空间会大幅缩小，有助于快速定位潜在的优质特征子集。随着进化进程推进，搜索粒度逐步细化，直至最终每位仅代表一个特征，确保每个特征都能在进化过程中得到考量。通过这种从大特征组逐步细化到单个特征的方式，VGS 既保证了搜索效率，又确保了所得特征子集的质量。

具体而言，所提出的 VGS 方案采用二进制编码作为个体表示。与现有编码方案不同的是，在 VGS 编码方案中，一位 i 代表一组特征。如果 $i = 1$，则表示这组特征全部被选中；如果 $i = 0$，则表示该组所有特征均未被选中。所谓的可变粒度意味着在进化过程中一位所代表的特征数量从大到小变化，通过这种方式搜索操作从粗粒度逐渐过渡到细粒度。图 1 给出了一个示例来说明具有八个特征的 VGS 编码方案。假设存在一个个体 p^t，在开始时 p^t 的粒度较粗，p^t 中的每一位代表四个特征。因此，对于总共八个特征，个体 p^t 由两位表示。其中，第一位代表f_1, f_2, f_3, f_4 ，，第二位代表f_5, f_6, f_7, f_8 。第一位的值为 1，这意味着在个体 p^t 中f_1, f_2, f_6, f_7；而第二位的值为 0, 这代表f_3, f_4, f_5, f_8 没有被选中。然后，随着进化的继续（例如经过 α 代），搜索粒度变小，当前个体 $p^{t+\alpha}$ 中的每一位代表两个特征，并且搜索操作比

p^t 时更加精细。再经过若干次迭代后，搜索粒度进一步细化，此时个体 $p^{t+2\alpha}$ 中的一位代表一个特征，这确保了原始数据中的每个特征都被考虑到，从而保证最终获得的特征子集具有高质量。

从图 2 的示例中，我们可以发现，通过使用所提出的 VGS 方案，搜索空间大大减少，搜索效率显著提高。此外，随着进化的进行，搜索粒度逐渐变小，直到一位仅代表一个特征。这些细粒度搜索确保数据中的每个特征都被考虑到，最终实现良好的解决方案。然而，这个示例也表明，要实施所提出的 VGS 方案，有两个问题需要解决。

- 在进化开始时（粗粒度搜索），哪些特征可以组合在一起并用一位来表示?
- 随着进化的继续，如何分解搜索粒度，并逐步进行细粒度搜索?

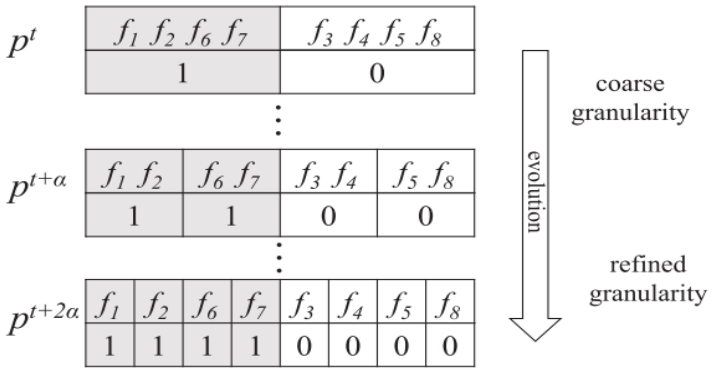

图 2: 一个具有八个特征的 VGS 编码方案的示例，其中 p^t 是第 t 代的个体。α 是两次转换之间的代数，它将个体从粗粒度变为细粒度。将文中的英文缩写 EA 展开可变粒度搜索方案的优势有哪些？用可变粒度搜索方案解决具体问题的案例

1) 粗粒度初始化: 为了解决第一个问题，设计了一种特征聚类方法来初始化粗粒度，属于同一聚类的特征将由一位表示。具体来说，在这项工作中，一个特征 f 由向量 $< SU_f, PCC_f >$ 来表征。SU_f 表示对称不确定性，用于衡量特征 f 与标签之间的非线性关系 [15]。PCC_f 是皮尔逊相关系数，用于捕捉特征 f 与标签之间的线性关系 [10]。

基于向量表示 $< SU_f, PCC_f >$，我们利用一种聚类算法（例如 K - 均值算法 [43]）将所有特征聚类成 k 个组，然后，属于同一组的特征将由一位表示。粗粒度初始化的具体过程在算法 1 中描述。图 4 则是一个简单的描述。

Algorithm 1 GranularityInitialization($data$, $label$, k)

Input: $data$: the data set; $label$: the labels; k: the group number;
Output: G: the initialized coarse granularity;
1: $SU_f \leftarrow$ Get the normalized symmetric uncertainty for all features in $data$;
2: $PCC_f \leftarrow$ Get the normalized pearson correlation coefficient for all features in $data$;
3: $g_1, g_2, ..., g_k \leftarrow$ KMeans($\langle SU_f, PCC_f \rangle, k$);
4: $G = \{g_1, g_2, ..., g_k\}$;

图 3: 算法 1

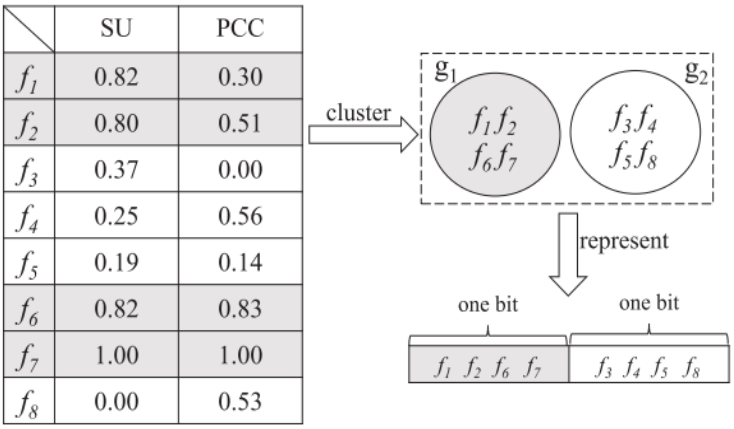

图 4: 粗粒度初始化的示例

2）粒度细化: 为了解决在进化过程中如何进行细粒度搜索的难题，提出了一种简单而有效的粒度细化方法，并应用于种群中的所有个体。该方法将一个粗粒度分解为两个相同大小的细粒度，直到最后一位仅代表一个特征。

3.3 基于粒度的遗传算子

通过所提出的可变粒度搜索（VGS）方案，我们可以有效地解决高维特征选择中的“维数灾难”问题。然而，必须承认 VGS 也带来了一些新的挑战。例如，传统的遗传算子（如交叉、变异）并不适用，因为交配池中的个体可能具有不同的粒度大小。此外，在高维特征选择问题中，最终选择的特征数量与原始数量相比非常少。因此，如何在可变粒度下生成稀疏后代是另一个需要解决的挑战。为了应对上述挑战，开发了两种基于粒度的遗传算子（即基于粒度的交叉和基于粒度的变异），其中前者操作用于探索，后者用于开发。

对于具有不同粒度大小的两个个体，首先将粗粒度个体转换为与细粒度个体具有相同粒度的个体。具体而言，假设个体是一个粗粒度个体，而是一个细粒度个体。在转换过程中，则采用了在之前提出的粒度细化策略。

基于粒度的变异算子的方法概述如下：在交叉操作之后，执行变异操作。该文章所提出的变异操作采用了与 NSGA - II [4] 中类似的位变异程序。然而，在 NSGA - II 中，每位的变异概率是相同的。这对于具有不同粒度的个体来说并不适用，因为粗粒度个体中的一位所代表的特征比细粒度个体中的一位代表的特征更多。所以还设计了一种基于粒度的变异算子。所设计算子的基本思路源于以下：从粗粒度父代中被选中（等于 1），但在细粒度父代中未被选中（等于 0）的位，意味着它可能包含冗余特征。因此，来自粗粒度父代的该位从“1”变为“0”的变异概率可能更高。

3.4 基于粒度的归档策略

通过使用所提出的可变粒度搜索方案以及基于粒度的遗传算子，我们能够高效地解决高维特征选择问题。此外，还开发了一种基于粒度的存档策略，利用存档 A 来进一步提升所提出的 VGS - MOEA（可变粒度搜索多目标进化算法）的性能。在所提出的 VGS - MOEA 中，存档 A 有两个用途：(1) 在第 t 次进化期间，存档 A_t 与当前种群 P_t 相结合，以构建交配池，

交配池进而被用于生成子代种群 O_t。(2) 在 VGS - MOEA 算法结束时，当前存档中的个体将作为最终解决方案输出。

具体而言，有两类个体被考虑存储到存档中。一类是种群中的非支配个体。另一类是具有最高准确率的个体，因为对于特征选择（FS）问题来说，准确率比所选特征的数量更受重视 [11]。需要注意的是，可能存在不止一个具有相同最高准确率的个体，在提出的存档策略中，所有这些个体都将被添加到存档里。在第 t 代期间，存档 A_t 中的个体与当前种群 P_t 中的个体相结合，以此构建交配池，然后在交配池上运用所提出的基于粒度的交叉和变异算子，来生成用于下一代 P_{t+1} 的子代 O_t。之后，通过将 O_t 中上述两类个体合并来更新存档，从而构建出下一代存档 A_{t+1}。

在更新存档时，存档中相同的个体仅保留一个，其余的则被删除，这是存档更新过程中一种常见的操作（去除冗余个体）。然而，在所提出的 VGS - MOEA（可变粒度搜索多目标进化算法）中，一种新型的“基于粒度的冗余个体”也会从存档中被移除，这能避免存档规模变得过大。所谓的“基于粒度的冗余个体”是由于存档中存储了许多具有不同粒度的个体，在这些不同粒度的个体所代表的特征子集中可能存在包含关系。因此，在存档中只保留准确率最高的那个个体。图 5 给出了一个示例用以说明“基于粒度的冗余个体”的情况。

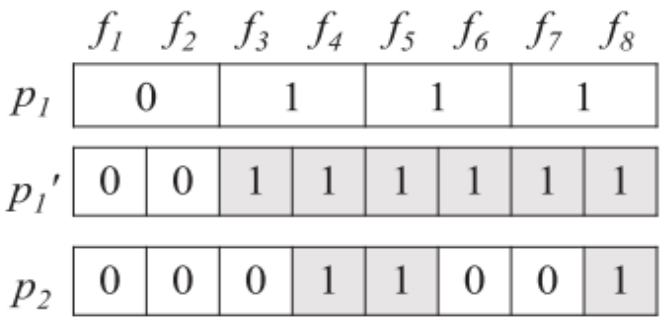

图 5: 一个用于说明基于粒度的冗余个体的实例

4 复现细节

4.1 与已有开源代码对比

在本次复现任务中，参考的源码时该篇论文的源码。完成了复现任务，在复现完成后，考虑了相应的改进措施与创新，但在实现的过程中并没有取得更好的效果，故最终并未实现有额外效果的改进或者创新。在接下来的部分，本次复现任务选择了原文中提到的对比方法、选择了该篇文章选择对比的相关算法。

4.2 实验环境搭建

采用和原文相同的实验环境，并且与原文对比的算法进行对比。

1）对比算法：该篇文章将其所提出的 VGS-MOEA 与七种当时（2023）最先进的基于进化计算（EC）的特征选择算法进行比较，即 VLPSO [15]、VS - CCPSO [12]、MOFS - BDE [17]、MOEA/D - DYN [11]、MOABC [5]、DAEA [16] 和 MOEA/PSL [14]，其中前两种算法

是单目标进化算法，后五种是多目标进化算法。在本次实验中，只实现了部分的对比，即与高维特征选择算法进行了对比，但这些高维特征选择算法只是单目标算法。

原文之所以挑选 VLPSO 和 VS - CCPSO 这两种单目标算法作为比较对象，是鉴于它们是近期为应对高维特征选择问题而新提出的算法，且二者均以粒子群优化（PSO）作为其优化架构。VLPSO 作为 PSO 的改进版本，创新地设计了一种可变长度表示形式，此形式能赋予粒子各异且更短的长度，大幅削减了搜索空间，进而契合高维特征选择的需求。而 VS - CCPSO 作为一种基于合作协同进化的特征选择算法，运用“分而治之”的策略来攻克特征选择中的“维数诅咒”难题。为了落实这一策略，它构建了空间划分方案、子群规模自适应调节机制，以及粒子删除与生成策略。

鉴于所提出的 VGS - MOEA 属于多目标进化算法范畴，文章选取的是五种当时（2023）最新的多目标进化算法当作基线算法。这些算法均具备处理高维特征选择难题的实力。具体而言，MOFS - BDE 乃是一种借助二进制差分进化的多目标特征选择算法。在 MOFS - BDE 里，精心设计了三个全新算子，即二进制变异算子、一位净化搜索算子以及高效非支配排序算子，旨在获取优质的特征子集。第二种基于 MOEA 的对比算法运用了基于多参考点分解的特征选择策略，其中构建了静态机制（MOEA/D - STAT）与动态机制（MOEA/D - DYN），目的在于攻克特征选择过程中帕累托前沿高度不连续以及偏好不平衡的难题。已有实证研究显示，MOEA/D - DYN 所获取的特征子集质量优于 MOEA/D - STAT，因此，在实现复现的该篇文章的实验中，将 MOEA/D - DYN 设定为基线算法。第三种基于 MOEA 的对比算法为 MOABC，它采用人工蜂群算法作为进化计算优化框架，以此解决特征选择任务。在 MOABC 中，开发出了二进制版本和连续版本这两种实现方式。在实现复现的该篇文章的实验中，之所以选用二进制版本，是因为其能够得出比连续版本更优的解。第四种算法是 DAEA，这是近期提出的一种双目标特征选择算法，该算法设计了三项重要改进措施，以便获取高质量的特征子集。此外，由于高维情况下的多目标特征选择属于稀疏大规模多目标问题（Sparse LSMOPs），于是增添了一种近期提出的名为 MOEA/PSL 的算法作为另一对比算法，此算法在解决稀疏大规模多目标问题方面已彰显出竞争力。

4.3 数据集与参数设置

数据集设置：为了检验不同算法的性能，在 12 个高维数据集上开展了一系列实验，这些数据集的特征数量在 1,024 到 22,283 之间。这些数据集在之前的一些关于高维特征选择的研究中已被采用 [15] [1] [9]。表 I 列出了 12 个实验数据集的详细特征，其中 # 特征、# 实例和 # 类别分别表示特征的数量、实例的数量和类别的数量。

5 实验结果分析

从图 6 实验结果及与其他算法对比能够看出，依据测试准确率来评判，所提出的 VGS - MOEA 在 11/12 个数据集上达成了最高的准确率。威尔科克森检验结果有力地证实了 VGS - MOEA 的卓越之处，它始终显著优于或者能够与另外两种高维特征选择算法相媲美，并且一次也没有表现得更差。从特征子集规模的角度考量，我们的方法所选取的特征数量相较于 VS - CCPSO 要少很多，尤其是在那些维度极高的数据集上体现得更为明显。例如，针对 GLI 85 数据集（包含 22,283 个特征），VS - CCPSO 挑选了大约五千个特征，然而 VGS - MOEA 仅

仅选取约 40 个特征就能获取更高的准确率。此外，威尔科克森统计结果还表明，VLPSO 所得到的特征数量与 VGS - MOEA 相近，这是因为它们在 12 个数据集上的“正 / 负 / 约等”分布情况为 4/4/4。VLPSO 表现良好的主要原因在于其可变长度（VL）表示形式，这种表示形式能够缩小搜索范围，从而得到规模极小的特征子集。不过，我们也留意到这种 VL 表示法存在局限性。在 VLPSO 中，一旦表示长度被缩短，就再也无法恢复。这也就意味着如果某些特征被去除，它们在最终的解决方案中就再也不会被纳入考虑范围。所以，在全部 12 个数据集上，VLPSO 的分类准确率都远远不及 VGS - MOEA。最后，第五列的统计数据显示，VGS - MOEA 的运行时间比 VLPSO 和 VS - CCPSO 都要短，这充分体现了所提出方法在计算效率方面的优势。综上所述，所有对比结果都确凿地表明，与其他高维特征选择算法相比，VGS - MOEA 具备更为出色的性能表现。

Data Set	Algorithm	Accuracy	Size	Time
COIL20	VLPSO	0.9951 ± 0.0053(+)	199.37 ± 78.88(+)	3296.03 ± 594.45(+)
	VS-CCPSO	0.9933 ± 0.0067(+)	**57.23** ± 13.25(≈)	653.14 ± 52.25(+)
	VGS-MOEA	**0.9988** ± 0.0022	59.03 ± 25.91	**305.79** ± 32.04
SRBCT	VLPSO	0.9173 ± 0.0492(+)	**5.73** ± 1.41(-)	**8.65** ± 1.65(-)
	VS-CCPSO	0.8667 ± 0.0767(+)	167.00 ± 21.52(+)	60.57 ± 2.24(+)
	VGS-MOEA	**0.9920** ± 0.0194	29.33 ± 21.57	10.78 ± 1.23
PCMAC	VLPSO	0.8394 ± 0.0283(+)	140.23 ± 84.12(+)	8386.67 ± 2774.83(+)
	VS-CCPSO	0.7672 ± 0.0206(+)	578.30 ± 50.00(+)	5046.88 ± 414.30(+)
	VGS-MOEA	**0.8626** ± 0.0149	**50.33** ± 10.67	**1159.96** ± 110.24
lymphoma	VLPSO	0.8632 ± 0.0704(+)	96.47 ± 88.54(≈)	34.78 ± 6.10(+)
	VS-CCPSO	0.9115 ± 0.0402(+)	593.27 ± 87.68(+)	110.87 ± 3.07(+)
	VGS-MOEA	**0.9529** ± 0.0429	**70.70** ± 47.53	**23.31** ± 3.68
GLIOMA	VLPSO	0.6956 ± 0.1397(+)	106.33 ± 249.41(≈)	29.66 ± 7.63(+)
	VS-CCPSO	0.7467 ± 0.0900(≈)	544.20 ± 130.27(+)	111.36 ± 2.17(+)
	VGS-MOEA	**0.7622** ± 0.1259	**23.97** ± 26.06	**12.59** ± 1.10
BASEHOCK	VLPSO	0.9225 ± 0.0287(≈)	110.20 ± 36.33(+)	10920.02 ± 2476.77(+)
	VS-CCPSO	0.8671 ± 0.0157(+)	871.63 ± 62.58(+)	7660.20 ± 533.72(+)
	VGS-MOEA	**0.9231** ± 0.0145	**65.57** ± 19.56	**1760.67** ± 236.49
TOX_171	VLPSO	0.8647 ± 0.0508(+)	297.97 ± 128.10(+)	116.23 ± 20.47(+)
	VS-CCPSO	0.8500 ± 0.0638(≈)	583.60 ± 100.54(+)	214.60 ± 5.37(+)
	VGS-MOEA	**0.8814** ± 0.0498	**102.07** ± 72.61	**33.83** ± 2.44
Brain1	VLPSO	0.8160 ± 0.0910(≈)	**35.70** ± 33.94(≈)	32.76 ± 9.96(+)
	VS-CCPSO	**0.8607** ± 0.0517(≈)	1027.00 ± 200.30(+)	169.14 ± 5.12(+)
	VGS-MOEA	0.8452 ± 0.0665	36.47 ± 31.56	**22.79** ± 3.99
leukemia	VLPSO	0.9288 ± 0.0593(+)	**7.47** ± 4.69(-)	**24.89** ± 5.10(-)
	VS-CCPSO	0.9061 ± 0.0642(+)	1090.70 ± 170.46(+)	202.88 ± 4.09(+)
	VGS-MOEA	**0.9970** ± 0.0115	24.17 ± 18.69	35.24 ± 1.72
ALLAML	VLPSO	0.9182 ± 0.0613(+)	**8.70** ± 9.99(-)	**21.91** ± 8.67(-)
	VS-CCPSO	0.8848 ± 0.0582(+)	1080.93 ± 257.83(+)	205.61 ± 6.42(+)
	VGS-MOEA	**0.9924** ± 0.0172	20.77 ± 14.99	35.20 ± 2.84
Brain2	VLPSO	0.6267 ± 0.1169(+)	43.60 ± 72.28(≈)	69.27 ± 14.96(+)
	VS-CCPSO	0.6467 ± 0.1402(+)	1586.00 ± 362.27(+)	353.52 ± 9.31(+)
	VGS-MOEA	**0.7356** ± 0.1013	**42.60** ± 35.10	**31.31** ± 2.67
GLI_85	VLPSO	0.8282 ± 0.0698(+)	**12.20** ± 13.76(-)	126.48 ± 54.64(+)
	VS-CCPSO	0.8615 ± 0.0594(+)	5153.53 ± 1022.90(+)	1115.53 ± 29.35(+)
	VGS-MOEA	**0.9013** ± 0.0531	39.63 ± 42.76	**96.81** ± 22.25
+/≈/-	VLPSO	10/2/0	4/4/4	9/0/3
	VS-CCPSO	9/3/0	11/1/0	12/0/0
	VGS-MOEA	-	-	-

图 6. 实验结果及与其他算法对比

6 总结与展望

在本次复现任务中，仅复现了代码与原文所述的相关算法的对比。曾考虑过将该算法应用于多任务特征选择，但实验效果并不好，甚至比较低效。其主要原因时没有很好的策略将可变粒度的种群初始化应用于多任务，因为原文章是使用一种特征聚类方法来初始化粗粒度，属于同一聚类的特征将由一位表示，在多任务特征选择算法中，将哪些特征聚类是一个较为困难的事情。在未来的研究中，将这一方法应用于多任务特征选择中仍然是一个可以考虑的方向，我将继续在这个方向进行研究与思考。

参考文献

[1] K. Chen, B. Xue, M. Zhang, and F. Zhou. An evolutionary multitasking-based feature selection method for high-dimensional classification. *IEEE Transactions on Cybernetics*, PP(1):1–15, 2020.

[2] Fan Cheng, Junjie Cui, Qijun Wang, and Lei Zhang. A variable granularity search-based multiobjective feature selection algorithm for high-dimensional data classification. *IEEE Transactions on Evolutionary Computation*, 27(2):266–280, 2023.

[3] Carlos A. Coello Coello, David A. Van Veldhuizen, and Gary B. Lamont. *Evolutionary algorithms for solving multi-objective problems*. Kluwer Academic Publishers, New York, 2002.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[5] E. Hancer, B. Xue, M. Zhang, D. Karaboga, and B. Akay. Pareto front feature selection based on artificial bee colony optimization. *Information Sciences*, 422:462–479, 2018.

[6] Fang Huang and Xiao-ping Fan. Parallel particle swarm optimization algorithm with island population model. *Control and Decision*, 21(2):175–179, 2006.

[7] Wu Jian, XinHua Tang, and Cao Yong. The research of parallel multi-objective particle swarm optimization algorithm. In *2014 IEEE 5th International Conference on Software Engineering and Service Science*, pages 300–304, 2014.

[8] Deming Lei and Ping Yan. *Algorithm and its application to intelligent multi-objective optimization*. Science Press, Beijing, 2009.

[9] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, 50(6):94, 2018.

[10] Z. Li and A. G. Bors. Selection of robust and relevant features for 3-d steganalysis. *IEEE Transactions on Cybernetics*, 50(5):1989–2001, 2020.

[11] B. H. Nguyen, B. Xue, P. Andreae, H. Ishibuchi, and M. Zhang. Multiple reference points-based decomposition for multiobjective feature selection in classification: Static and dynamic mechanisms. *IEEE Transactions on Evolutionary Computation*, 24(1):170–184, 2020.

[12] X. F. Song, Y. Zhang, Y. N. Guo, X. Y. Sun, and Y. L. Wang. Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data. *IEEE Transactions on Evolutionary Computation*, 24(5):882–895, 2020.

[13] F. Tan, X. Fu, Y. Zhang, and A. Bourgeois. A genetic algorithm-based method for feature subset selection. *Soft Computing*, 12:111–120, 2008.

[14] Y. Tian, C. Lu, X. Zhang, K. C. Tan, and Y. Jin. Solving large-scale multiobjective optimization problems with sparse optimal solutions via unsupervised neural networks. *IEEE Transactions on Cybernetics*, 51(6):3115–3128, 2021.

[15] B. Tran, B. Xue, and M. Zhang. Variable-length particle swarm optimization for feature selection on high-dimensional classification. *IEEE Transactions on Evolutionary Computation*, 23(3):473–487, 2019.

[16] H. Xu, B. Xue, and M. Zhang. A duplication analysis-based evolutionary algorithm for biobjective feature selection. *IEEE Transactions on Evolutionary Computation*, 25(2):205–218, 2021.

[17] Y. Zhang, D. W. Gong, X. Z. Gao, T. Tian, and X.-Y. Sun. Binary differential evolution with self-learning for multi-objective feature selection. *Information Sciences*, 507:67–85, 2020.

[18] Junhai Zhou, Quanwang Wu, Mengchu Zhou, Junhao Wen, Yusuf Al-Turki, and Abdullah Abusorrah. Lagam: A length-adaptive genetic algorithm with markov blanket for high-dimensional feature selection in classification. *IEEE Transactions on Cybernetics*, 53(11):6858–6869, 2023.