

函件

将多目标优化问题分解为多个简单的多目标子问题

刘海林、顾方庆、
张庆福, IEEE 高级会员

摘要 – 这封信提出了一种将多目标优化问题 (MOP) 分解为一组简单的多目标优化子问题的方法。利用该方法, 提出了基于多目标优化进化算法的新版本 MOEA/D-M2M。该算法以协同方式解决了这些子问题。每个子问题都有自己的群体, 并在每一代都接受计算工作。这样, 就可以保持种群多样性, 这对于解决一些 MOP 至关重要。已经进行了实验研究, 将 MOEA/D-M2M 与经典的 MOEA/D 和 NSGA-II 进行了比较。这封信认为, 在处理某些 MOP 的多目标进化算法中, 种群多样性比收敛更重要。这也解释了为什么 MOEA/D-M2M 表现更好。

索引项 – 分解、混合算法、多目标优化。

一、引言

这封信考虑了以下连续多目标优化问题 (MOP) :

最小化 $F(x) = (f_1(x), \dots, f_m(x))$
受制于 $x \in \bigcap_{i=1}^n [a_i, b_i]$

哪里 $\bigcap_{i=1}^n [a_i, b_i]$ 是决策空间。 $f_i : \bigcap_{i=1}^n [a_i, b_i] \rightarrow \mathbb{R}$ 由 m 实值连续目标函数 f_1, \dots, f_m 。 R 是目标空间组成。由于这些目标相互冲突, 因此没有单一的解决方案可以同时优化它们。设 $u = (u_1, \dots, u_m)$ 和 $v = (v_1, \dots, v_m)$, u 主导 v 当且仅当 $u_i \leq v_i$ 对于每个 i 并且存在一个索引 j , 使得 $u_j < v_j$ 。如果没有 $x \in [a, b]$, 使得 $F(x)$ 主导 $F(x)$, 则 x 称为 (全局) 帕累托最优点, $F(x)$ 是帕累托最优目标向量。换句话说, 在帕累托最优点上对一个目标的任何改进都必须导致至少一个其他目标的恶化。帕累托最优解是最优解

所有目标之间的权衡候选者。所有帕累托最优点的集合称为帕累托集 (PS), 所有帕累托最优目标向量的集合称为帕累托前沿 (PF) [3]。在温和的条件下, PS 和 PF 都是 $(m - 1) - D$ 分段连续流形[1]。很多时候, 决策者需要与 PF 进行近似值, 以便对问题有很好的洞察力并做出最终选择。

已经开发了许多多目标优化进化算法 (MOEA), 用于在单次运行中找到一组近似 PF 的解[2] - [6]。大多数 MOEA, 如 NSGA-II [4], 主要依靠帕累托优势来指导他们的搜索, 特别是他们的选择运算符。相比之下, 基于分解的多目标进化算法

(MOEA/D) [6]利用传统的聚合方法将近似 PF 的任务转化为多个单一目标优化子问题。然后, 采用基于群体的算法以协作方式解决这些子问题。已经提出了一些 MOEA/D 变体来处理各种 MOPs [7]、[8]。MOEA/D 也被用作一些混合算法的基本元素 [9] - [11]。

通常, 有效的 MOEA 需要良好的种群多样性, 因为它们的目标是近似一个集合而不是单个点。当前的 MOEA/D 算法通过其子问题的多样性来实现其种群多样性。这些算法在选择时采用精英策略。新解是否取代旧解完全由其聚合函数值决定。在某些情况下, 这种替代会导致种群多样性的严重丧失。因此, 他们可能会错过某些搜索区域并减慢搜索速度。另一个缺点是, MOEA/D 在特定问题上的良好性能通常需要用户选择合适的聚合方法和合适的权重向量设置, 这可能并不总是一件容易的事。为了减少这些缺点, 这封信对原始的 MOEA/D 进行了推广, 并提出了一种将 (1) 分解为许多简单的多目标优化子问题的方法。基于这种分解, 设计了一种新的 MOEA/D 算法 MOEA/D-M2M。我们表明, 在我们实验研究中使用的一组测试实例中, MOEA/D-M2M 的性能可以显著优于 MOEA/D 和 NSGA-II。以一个实例为例, 我们解释了为什么 MOEA/D-M2M 比其他两种算法能够保持更好的种群多样性。

稿件于 2012 年 7 月 18 日收到;2013 年 3 月 13 日修订;2013 年 4 月 23 日接受。出版日期: 2013 年 9 月 11 日;当前版本的日期为 2014 年 5 月 27 日。这项工作得到了中国自然科学基金 60974077 资助、广东省自然科学基金资助 S2012010008813 资助、广东省科学技术计划资助 2012B091100033 资助、广东省教育厅科技计划资助 2012KJX0042 资助。
H. L. Liu 和 F. Gu 在广东工业大学工作, 广东 510520 (电子邮件: hlliu@gdut.edu.cn; gufangqing84@163.com)。
Q. 张先生现任教于香港特别行政区九龙市立大学计算机科学系 (电子邮件: qingfu.zhang@cityu.edu.hk)。他正在英国科尔切斯特 C04 3SQ 埃塞克斯大学计算机科学与电子工程学院休假 (电子邮件: qzhang@essex.ac.uk)。
本文中个或多个图形的彩色版本可在 <http://ieeexplore.ieee.org> 在线获取。

二、主要思想和算法

A. 分解

为简单起见, 这封信假设所有单独的目标函数 f_1, \dots, f_m 都是非负的。因此, 所有目标向量以及 (1) 的 PF 都在 R 中。

否则, 我们可以将 f 替换为 $f+M$, 其中 M 是一个足够大的正数, 因此所有 i 的 $f_i + M > 0$ 。

数字对象标识符 10.1109/TEVC.2013.2281533

1089-778X © 2013 IEEE. 允许个人使用, 但重新发布/重新分发需要 IEEE 许可。有关详细信息, 请参阅 http://www.ieee.org/publications_standards/publications/rights/index.html。

MOEA/D-M2M 背后的主要思想是将 (1) 分解为一组简单的多目标优化子问题，然后在一次运行中解决它们。为此，我们首先选择 K 单位向量 $v_1, \dots, v_K \in R$ 。然后，将 R 分为 K 个子区域 R_1, \dots, R_K ，其中 $R_k = \{u \in R \mid \langle u, v_j \rangle \leq \langle u, v_k \rangle \text{ 对于任何 } j = 1, \dots, K, j \neq k\}$ 为

其中 $\langle u, v \rangle$ 是 u 和 v 之间的锐角。换句话说，当且仅当 v 在所有 K 方向向量中与 u 的角度最小时， u 在 R_k 。基于此划分，(1) 可以转化为 K 约束多目标优化子问题。子问题 k 是

最小化 $F(x) = (f_1(x), \dots, f_K(x))$
受制于 $x \in \bigcap_{i=1}^K [a_i, b_i]$
 $F(x) \in R_k$

我们想对上述转变发表以下评论。

- 1) 从某种意义上说，这种转换是等价的，因为所有这些子问题的 PF 构成了 (1) 的 PF。相比之下，原始 MOEA/D 中使用的变换是近似的，因为所有单一目标优化子问题的最优解只是 (1) 的 PF 的近似值。
- 2) 即使 (1) 的 PS 具有非线性几何形状，由于约束 $F(x) \in R_k$ ，(3) 的 PS 也可能接近线性，因为它只是 (1) PS 的一小部分。因此，(3) 可能比 (1) 更简单，至少在 PS 形状方面是这样。请注意，基于帕累托的 MOEA，如 NSGA-II，适用于处理简单的 PS [1]，因此，这些算法中的元素可用于解决这些子问题。
- 3) 此转换不需要任何聚合方法。用户需要做的是选择一组方向向量。在某种程度上，它比原始 MOEA/D 框架中使用的转换需要更少的人力。

B. 算法

MOEA/D-M2M 采用上述分解策略，将 (1) 分解为 K 个多目标优化子问题，并协同求解。在每一代中，MOEA/D-M2M 保持 K 个亚群： P_1, \dots, P_K ，其中 P_k 表示子问题 k 。每个子群体都包含 S 个单独的解。还记录了每个解的 F 值（即目标函数向量）。MOEA/D-M2M 在算法 1 中进行了描述。算法 1 中的第 1 行和第 13 行都使用许多单独的解决方案来设置 P_1, \dots, P_K 。我们使用算法 2 中描述的简单方法来做到这一点。算法 2 确保每个 P_k 在每一代都有 S 解，从而在搜索过程中促进种群多样性。当某些多目标子问题比其他子问题困难得多时，这一点非常重要。[1] 中的 MOEA/D 变体和大多数帕累托优势算法可能无法有效地求解此类 MOP。经济部/D

算法 1: MOEA/D-M2M

输入:
1) 澳门币 (1); 2) 停止标准;
3) K : 子问题的数量;
4) K 单位方向向量: v_1, \dots, v_K ;
5) 亚种群的大小; 6) 遗传操作员及其相关参数。输出: 一组非支配解初始化: 均匀随机选择 $K \times S$ 点

1 从 $[a, b]$ 计算它们的 F 值，然后使用它们来设置 P_1, \dots, P_K 。
2 当不满足停止标准时，do
3 新解决方案的生成:
4 设置 $R = \emptyset$;
5 对于 $k \leftarrow 1$ 到 K do
6 foreach $x \in P_k$ do
7 从 P_k 中随机选择 y ;
8 将 x 和 y 上的遗传算子应用于
9 生成新的解决方案 z ; 计算 $F(z)$;
10 $R := R \cup \{z\}$;
11 结束
12 $Q := R \cup (\cup P_k)$;
13 使用 Q 设置 P_1, \dots, P_K 。
14 结束
15 在 $\cup P_k$ 和
16 输出它们。

算法 2: 将个体分配给亚种群输入: $Q: [a, b]$ 中的一组个体解及其 F 值。

输出: P_1, \dots, P_K 。

1 对于 $k \leftarrow 1$ 到 K do
2 初始化 Q 中的 F 值为
3 在;
4 如果 $|P_k| < S$ 然后
5 随机选择 $S - |P_k|$ 和
6 将它们添加到 P_k 。
7 结束
8 如果 $|P_k| > S$ 然后
9 使用非支配的
10 排序方法 [4] 并从 P_k 中删除 $S - |P_k|$ 排名最低
11 的解决方案。
12 结束
13 结束

在 [1] 中使用聚合函数进行选择，这些帕累托优势算法在其选择中进行全局非优势排名。因此，一些硬的多目标子问题可能很少甚至没有为下一代提供解决方案，如果很可能发生，其可行区域中的当前解决方案完全由以下因素主导

其他解决方案。结果, 人口多样性将大大丧失。

算法 2 的第 7 行使用 NSGA-II 中的非支配排序方法。实际上, 任何其他选择策略都可以用于此目的。

一些并行的 MOEA 和多目标粒子群优化[12] - [15]以及基于多种群的 MOEA[16] - [19]也利用多个种群来显式或隐式地靶向 PF 的不同部分。在这些现有算法中, 锥分离 NSGA-II (CS-NSGA-II) [19]的工作方式与我们提出的 MOEA/D-M2M 类似。CS-NSGA-II 和 MOEA/D-M2M 之间的主要区别包括以下几点。

- 1) MOEA/D-M2M 中的分解方法比 CS-NSGA-II 中的分解方法要简单得多。CS-NSGA-II 对目标函数进行归一化, 使所有当前解都在一个单位 m -D 立方体中, 然后将该立方体划分为几个锥体。Branke 等[19]仅解释了在两个和三个目标的情况下如何做到这一点。在超过三个目标的情况下, 使用他们的方法进行空间划分并非易事。而 MOEA/D-M2M 中的分解非常简单, 只需要几个单位方向向量。
- 2) CS-NSGA-II 中的空间划分适用于具有多个处理器的并行计算环境。在某些世代中, 处理器与其他一些处理器共享其不可行的解决方案。MOEA/D-M2M 中的分解主要是为了平衡多样性和收敛性, 而在每一代中, 它都会合并所有子种群, 然后将它们重新分配给不同的子问题。

三、实验研究

我们在实验研究中将 MOEA/D-M2M 与 MOEA/D-DE 和 NSGA-II 进行了比较。MOEA/D-DE 是[1]中提出的 MOEA/D 对连续 MOP 的高效和有效的实施。

A. 实验环境

- 1) [20]中具有相同控制参数的交叉和突变算子用于生成新解的三种算法。
- 2) 对于双目标实例, NSGA-II 和 MOEA/D-DE 的总体规模均为 100。就这三个目标而言, NSGA-II 和 MOEA/D-DE 的目标为 300。
- 3) 对于双目标实例, MOEA/D-M2M 中的 $K = S = 10$ 。对于三目标实例, MOEA/D-M2M 中的 $K = S = 17$ 。
- 4) 停止标准: 所有三种算法在 3000 代后停止。
- 5) MOEA/D-DE 中的其他控制参数: $t = 20$, $\delta = 0.9$ 和 $n = 2$, 与[1]相同。
- 6) K 方向矢量从 MOEA/D-M2M 中第一个八分位的单位球体中均匀选择。

显然, 在上述设置下, 在三个目标的情况下, MOEA/D-M2M 在每一代评估的解决方案比 MOEA/D 和 NSGA-II 略少。

B. 绩效指标

IGD 度量: IGD 度量用于测量我们实验中溶液集 P 的质量。假设 P 是一组点, 在物镜空间中沿 PF 均匀分布, P 是 PF 的近似值。 P 和 P 之间的距离可以定义为:

$$IGD(P, P) = \frac{\sum_{v \in P} d(v, P)}{|P|}$$

其中 $d(v, P)$ 是从点 v 到 P 的最小欧几里得距离。显然, IGD 的值越小, 算法的性能就越好。双目标测试实例的 500 点和三目标测试实例的 1000 点在 PF 上均匀采样以形成 P 。

H 度量: 设 $y = (y_1, \dots, y_n)$ 是目标空间中的一个点, 该点由任何帕累托最优目标向量主导。设 S 是目标空间中 PF 的近似值。那么, S 的 H 值 (相对于 y) 是区域的体积, 该区域由 S 主导并主导 y 。在我们的实验中, $y = (1, \dots, 1)$ 。 H 度量越大, 近似值越好。

C. 测试实例

使用以下修改后的 ZDT 和 DTLZ 实例[21]。我们修改后的实例中使用的 $g(x)$ 函数与其原始版本中的函数不同。它们的搜索空间是 $[0, 1]$ 。除非另有说明, 否则 $n = 10$ 。所有这些实例都是为了最小化。

澳门币 1

$$\begin{cases} f_1(x) = (1 + g(x))x \\ f_2(x) = (1 + g(x))x \end{cases}$$

哪里

$$g(x) = 2 \sin\left(\sum_{j=2}^n \pi x_j\right) - 0.9t + |t|$$

$t = x - \sin(0.5\pi x)$; 其 PF 为 $f=1$; 它的 PS 是 $\{(x, \dots, x) \mid 0 < x < 1, x = \sin(0.5\pi x); j = 2, \dots, n; \text{或 } x = 0, 1\}$ 。

澳门币 2:

$$\begin{cases} f_1(x) = (1 + g(x))x \\ f_2(x) = (1 + g(x))x \end{cases}$$

哪里

$$g(x) = 10 \sin\left(\sum_{j=2}^n \pi x_j\right) - t$$

$t = x - \sin(0.5\pi x)$; 其 PF 为 $f=1-f, 0 \leq f \leq 1$; 它的 PS 是 $\{(x, \dots, x) \mid 0 < x < 1, x = \sin(0.5\pi x); j = 2, \dots, n; \text{或 } x = 0, 1\}$ 。

澳门币 3:

$$\begin{cases} f_1(x) = (1 + g(x)) \\ \cos(\cdot) f_2(x) = (1 + g(x)) \end{cases}$$

哪里

$$g(x) = 10 \sin\left(\sum_{j=2}^n \pi x_j\right) - t$$

$t = x - \sin(0.5\pi x)$; 其 PF 为 $0 \leq f \leq 1$; 它的 PS 是 $\{(x, \dots, x) \mid 0 < x \leq 1, x = \sin(0.5\pi x); j = 2, \dots, n; \text{或 } x = 0, 1\}$ 。

表一

20 次独立运行中 MOEA/D-M2M、MOEA/D-DE 和 NSGA-II 的平均值和最佳 IGD 度量值

对于每个测试实例

IGD-metric	MOEA/D-M2M		MOEA/D-DE		NSGA-II	
Instance	best	mean	best	mean	best	mean
MOP1	0.0151	0.0179	0.2897	0.3239	0.2129	0.2206
MOP2	0.0103	0.0118	0.2167	0.2342	0.2103	0.2121
MOP3	0.0116	0.0123	0.4437	0.4798	0.2611	0.2660
MOP4	0.0091	0.0102	0.2662	0.2738	0.2745	0.2826
MOP5	0.0153	0.0209	0.2657	0.2925	0.2419	0.2442
MOP6	0.0513	0.0526	0.3039	0.3040	0.3040	0.3044
MOP7	0.0623	0.0780	0.3507	0.3507	0.3505	0.3505

表二

20 次独立运行中 MOEA/D-M2M、MOEA/D-DE 和 NSGA-II 的平均值和最佳 H 度量值

对于每个测试实例

H-metric	MOEA/D-M2M		MOEA/D-DE		NSGA-II	
Instance	best	mean	best	mean	best	mean
MOP1	0.6500	0.6499	0.1326	0.1208	0.1862	0.1503
MOP2	0.3057	0.2947	0.0812	0.0678	0.0570	0.0375
MOP3	0.1871	0.1837	0.0012	0.0004	0.0545	0.0286
MOP4	0.5079	0.5077	0.1346	0.1026	0.1641	0.1326
MOP5	0.6361	0.6340	0.2143	0.1801	0.2156	0.1841
MOP6	0.7532	0.7257	0.5433	0.5431	0.5432	0.5404
MOP7	0.3773	0.3762	0.2014	0.2014	0.2014	0.2014

澳门币 4 元:
$$\begin{cases} f(x) = (1 + g(x)) \times f(x) = \\ (1 + g(x)) (1 - x \cos(2\pi x)) \end{cases}$$
 哪里
$$g(x) = 10 \sin\left(\sum_{i=2}^{n+1} x_i\right) -$$

$$t = x - \sin(0.5\pi x)$$
; 它的 PF 是不连续的; 它的 PS 是 $\{(x, \dots, x) \mid 0 < x < 1, x = \sin(0.5\pi x); j = 2, \dots, n; \text{或 } x = 0, 1\}$ 。

澳门币 5 元:
$$\begin{cases} f(x) = (1 + g(x)) \times x \\ f(x) = (1 + g(x)) \times x \end{cases}$$
 哪里
$$g(x) = 2 \cos\left(\sum_{i=2}^{n+1} x_i\right) - 0.9t + |t|$$
;
$$t = x - \sin(0.5\pi x)$$
; 其 PF 为 $f \neq 1$; 它的 PS 是 $\{(x, \dots, x) \mid 0 \leq x \leq 1, x = \sin(0.5\pi x); j = 2, \dots, n; \text{或 } x = 0.5\}$ 。

澳门币 6 元:
$$\begin{cases} f(x) = (1 + g(x)) \times x \\ f(x) = (1 + g(x)) \times x \\ (1 - x) f(x) = (1 + \end{cases}$$
 哪里
$$g(x) = 2 \sin\left(\sum_{i=3}^{n+1} x_i\right) - 0.9t + |t|$$
;
$$t = x - xx$$
; 其 PF 为 $f + f + f = 1, 0 \leq f, f, f \leq 1$; 它的 PS 是 $\{(x, \dots, x) \mid 0 < x < 1, 0 \leq x \leq 1; x = xx, j = 3, \dots, n; \text{或 } x = 0, 1\}$ 。

澳门币 7 元:
$$\begin{cases} f(x) = (1 + g(x)) \cos(\cdot) \\ \cos(\cdot) f(x) = (1 + g(x)) \\ \cos(\cdot) \sin(\cdot) f(x) = (1 + \end{cases}$$
 哪里
$$g(x) = 2 \sin\left(\sum_{i=3}^{n+1} x_i\right) - 0.9t + |t|$$
;
$$t = x - xx$$
; 其 PF 为 $f + f + f = 1, 0 \leq f, f, f \leq 1$; 它的 PS 是 $\{(x, \dots, x) \mid 0 < x < 1, 0 \leq x \leq 1; x = xx, j = 3, \dots, n; \text{或 } x = 0, 1\}$ 。

D. 结果

表 I 显示了 20 次独立运行中每个测试实例的算法的最小（即最佳）和平均值。表 II 显示了最大（即最佳）

以及 20 次运行中 H 度量值的平均值。从这两张表中可以清楚地看出，MOEA/D-M2M 在所有测试实例上的这两个指标上都比 MOEA/D-DE 和 NSGA-II 好得多。

图 1 在目标空间中绘制了运行中获得的最终解决方案的分布，以及每个测试实例中每种算法的 IGD 度量值。显然，MOEA/D-DE 和 NSGA-II 都无法在任何实例上定位全局 PF。相比之下，MOEA/D-M2M 可以很好地近似这些实例的 PF。

为了理解为什么 MOEA/D-M2M 的性能优于其他两种算法，我们以 $n = 2$ 的 MOP1 为例，分析了搜索空间中所有点之间的帕累托优势关系[0, 1]。在图 2 中，搜索空间中的区域 A 对应于目标空间中包括 (1, 0) 在内的非常小的区域，A 中的每个点都支配着 B 区域中的任何点。由于 MOEA/D 和 NSGA-II 都采用收敛优先、多样性第二选择策略，因此很容易被困在 A 区。相比之下，MOEA/DM2M 总是为每个子问题分配相同数量的解，因此，它不会在 A 区花费最多的精力，从而能够找到全局 PF。

$N = K \times S$ 可视为 MOEA/D-M2M 中的整个人口规模。如果我们固定 N ，则与 NSGA-II 相比，MOEA/D-M2M 多了一个控制参数 K 。为了研究该参数对算法性能的影响，我们在 MOP1 上进行了不同 K 值的实验。在我们的实验中， S 设置为 100 K 保持 $N = 100$ ，所有其他设置与第 III-A 节相同。 S 的最小值为 2，因为交叉算子在每个子群体中至少需要两个不同的解。因此， K 的最大值为 50。20 次运行的实验结果总结如图 3 和图 4 所示。

从这两个数字可以明显看出，当 $K \geq 10$ 时，算法的性能不会发生巨大变化。这些结果表明，MOEA/D-M2M 对该参数的敏感性不是很强。我们应该指出的是， K 的大值需要大量的方向向量，这会增加为子问题分配单个解的计算开销。

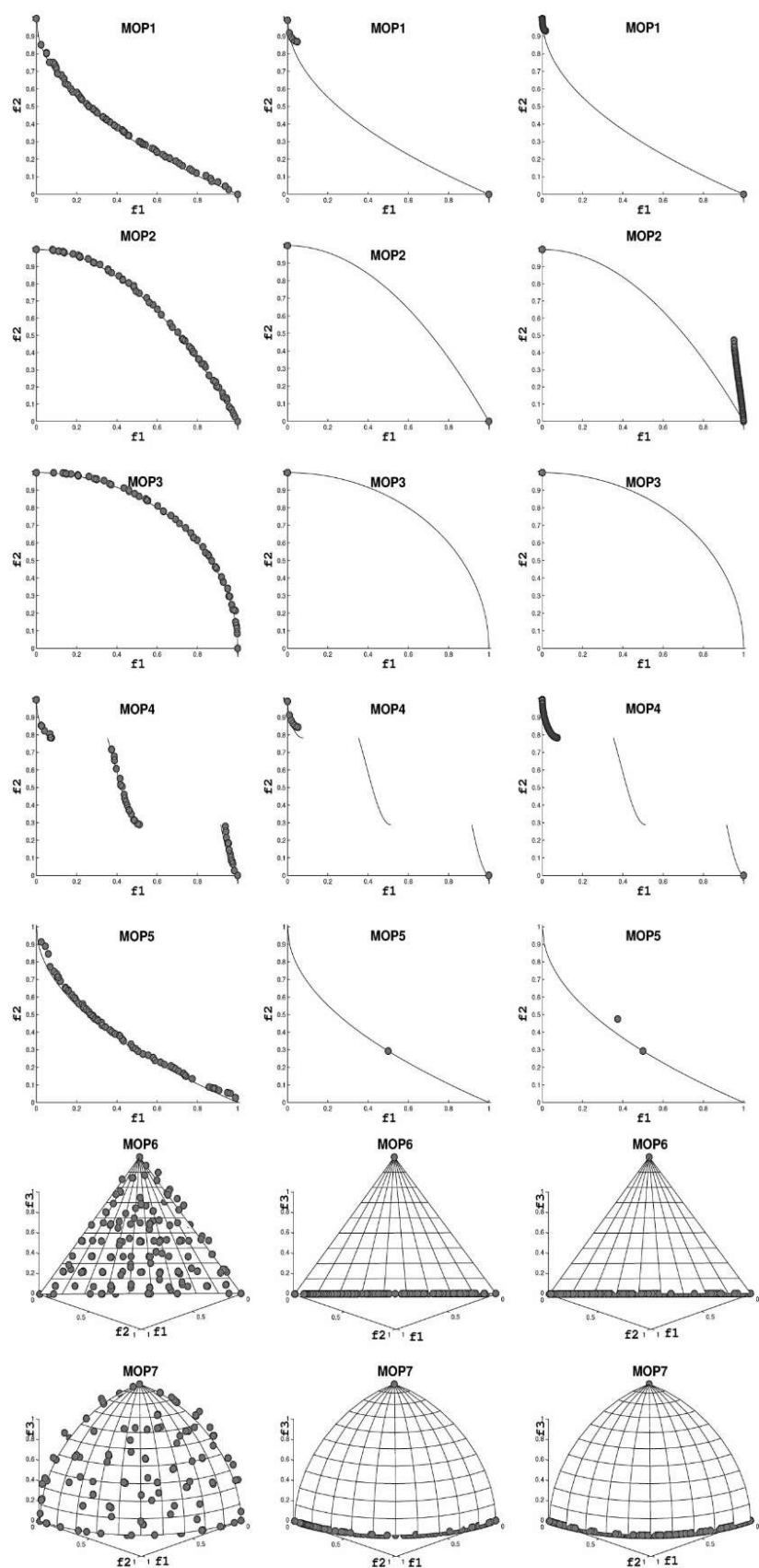


图 1. 非支配前沿的图, 以及 MOEA/D-M2M (左图)、MOEA/D-DE (中图) 和 NSGA-II (右图) 发现的 IGD 度量值中值。

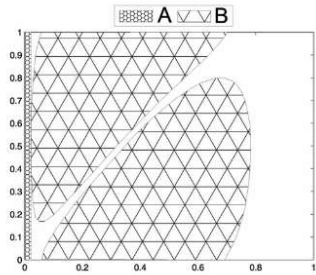


图 2. $n = 2$ 的 MOP1 搜索空间图。区域 A 中的一个点主导区域 B 中的任何点。

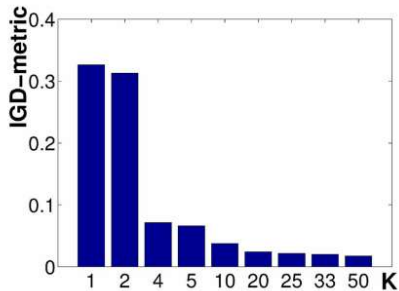


图 3. MOP1 不同 K 值的 IGD 度量条形图。

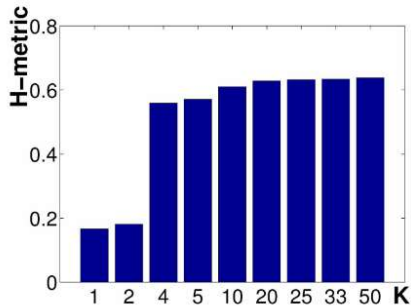


图 4. MOP1 的不同 K 值的 H 度量条形图。

四、结语

这封信提出了一种简单的方法，将 MOP 分解为许多简单的多目标子问题。它还设计了一种名为 MOEA/D-M2M 的算法，该算法对 MOEA/D 框架进行了推广。MOEA/D-M2M 旨在以协作方式解决这些多目标子问题。每个子问题都有一个子群体，并且在搜索过程中始终需要进行计算工作。这样，就可以实现良好的种群多样性，这对于解决一些 MOP 至关重要。对一组测试实例的实验研究表明，MOEA/D-M2M 在一组测试实例上可以显著优于 MOEA/D-DE 和 NSGA-II。未来的工作将包括将 MOEA/D-M2M 与其他进化搜索技术和调查相结合

其在其他硬多目标优化问题上的性能。

引用

[1] H. Li and Q. Zhang, “复杂帕累托集的多目标优化问题, MOEA/D 和 NSGA-II”, IEEE Trans. Evol.Comput., 第 13 卷, 第 2 期, 第 284-302 页, 2009 年 4 月。
[2] K. Deb, 使用进化算法进行多目标优化。美国纽约州纽约: Wiley, 2001 年。
[3] K. Miettinen, 非线性多目标优化。美国马萨诸塞州诺威尔: Kluwer, 1999 年。
[4] K. Deb, A. Pratap, S. Agarwal 和 T. Meyarivan, “一种快速而精英主义的多目标遗传算法: NSGA-II”, IEEE Trans.Comput., 第 6 卷, 第 2 期, 第 182-197 页, 2002 年 4 月。
[5] E. Zitzler, M. Laumanns 和 L. Thiele, “SPEA2: 提高多目标优化的强度帕累托进化算法”, 论文集。
进化。甲基苯丙胺。设计优化。控制应用。Ind. Problems, 2001 年, 第 95-100 页。
[6] Q. Zhang 和 H. Li, “MOEA/D: 基于分解的多目标进化算法”, IEEE Trans. Evol.Comput., 第 11 卷, 第 6 期, 第 712-731 页, 2007 年 12 月。
[7] Y. Mei, K. Tang 和 X. Yao, “用于多目标电容电路路由问题的基于分解的模因算法”, IEEE Trans. Evol.Comput., 第 15 卷, 第 2 期, 第 151-165 页, 2011 年 4 月。
[8] V. A. Shim, K. C. Tan 和 C. Y. Cheong, “用于解决多目标多旅行推销员问题的分布算法与分解的混合估计”, IEEE Trans. Syst., Man, Cybern. C, 第 42 卷, 第 5 期, 第 682-691 页, 2012 年 9 月。
[9] S. Z. Martinez 和 Carlos. A. C. Coello, “基于分解的多目标进化算法的直接局部搜索机制”, 在 IEEE 世界大会论文集。计算。Intell., 2012 年 6 月, 第 3431-3438 页。
[10] K. Sindhya, K. Miettinen 和 K. Deb, “混合框架进化多目标优化”, IEEE Trans. Evol.Comput., 第 17 卷, 第 4 期, 第 495-511 页, 2012 年 8 月。
[11] H. Ishibuchi, Y. Sakane, N. Tsukamoto 和 Y. Nojima, “使用两种邻域结构对多目标优化的细胞进化算法性能的影响”, 在 IEEE Congr 论文集。进化。Comput., 2009 年 5 月, 第 2508-2515 页。
[12] T. Hiroyasu, M. Miki 和 S. Watanabe, “并行的新模型多目标优化问题中的遗传算法, 划分范围多目标遗传算法”, 载于 Proc. Congr. 进化。Comput., 第 1 卷。2000 年 7 月, 第 333-340 页。
[13] A. L. Jaimes 和 C. A. C. Coello, “MRMOGA: 使用多种分辨率的并行进化多目标优化”, 载于 Proc. Congr. 进化。Comput., 第 3 卷。2005 年, 第 2294-2301 页。
[14] D. VanVeldhuizen, J. Zydallis 和 G. Lamont, “工程并行多目标进化算法的考虑”, IEEE Trans. 进化。Comput., 第 7 卷, 第 2 期, 第 144-173 页, 2003 年 4 月。
[15] S. Mostaghim, J. Branke 和 H. Schneck, “计算机网格上的多目标粒子群优化”, 载于 Proc. Genetic Evol. 计算。会议, 2007 年, 第 869-875 页。
[16] T. Hiroyasu, M. Miki 和 S. Watanabe, “在多目标优化问题中采用新共享方法的分布式遗传算法”, 在会议程序中。进化。Comput., 第 1 卷。1999 年, 第 69-76 页。
[17] K. Deb, P. Zope 和 A. Jain, “帕累托最优的分布式计算具有进化算法的解决方案”, 载于 Proc. 2th Conf. Evol.MultiCriterion Optimiz., 2003 年, 第 534-549 页。
[18] H. Liu 和 F. Gu, “基于次区域搜索的改进 NSGA-II 算法”, Proc. Congr. 进化。Comput., 2011 年, 第 1906-1911 页。
[19] J. Branke, H. Schneck, K. Deb 和 M. Reddy, “并行化多目标进化算法: 锥体分离”, 载于 Proc. Congr. 进化。Comput., 2004 年, 第 1952-1957 页。
[20] H. Liu 和 X. Li, “基于确定权重和次区域搜索的多目标进化算法”, 载于 Proc. Congr. 进化。Comput., 第 15 卷。2009 年, 第 361-384 页。
[21] S. Huband, P. Hingston, L. Barone 和 L. 而 “回顾多目标测试问题和可扩展的测试问题工具包”, IEEE Trans. Evol.Comput., 第 10 卷, 第 5 期, 第 477-506 页, 2006 年 10 月。