A fast and elitist multiobjective genetic algorithm- NSGA-II

作者	
发表时间	
来源	
评级	★ ★ ★ ★ ★

第 2 页今天 11:17

In a naive approach,in order to identify solutions of the first nondominated front in a population of size ,each solution can be compared with every other solution in the population to find if it is dominated.

幻 今天 11:17

这句话的意思是，在一个拥有一定数量个体的群体中，为了找到第一个非支配前沿的解决方案，可以采用一种天真的方法，即将每个解决方案与群体中的每个其他解决方案进行比较，以确定它是否被其他解决方案支配。

在这个上下文中，"front" 可以翻译为 "前沿" 或 "前线"，通常用于描述多目标优化问题中的 Pareto 前沿。Pareto 前沿是一组解决方案，其中没有一个解决方案在所有目标上都被其他解决方案支配。所以，"第一个非支配前沿" 意味着找到第一个在所有目标上都不被其他解决方案支配的一组解决方案。

第 3 页今天 11:24

Thus,the worst case is when there are fronts and there exists only one solution in each front.This requires an overall computations.

第 3 页今天 11:30

we describe a fast nondominated sorting approach which will require computations

幻 今天 11:30

传统的非支配排序算法时间复杂度：O(MN^2)
文中提出的快速非支配排序算法时间复杂度：O(MN^2)

第 3 页今天 11:52

fast-non-dominated-sort(P)	
for each $p \in P$	
$S_p = \emptyset$	
$n_p = 0$	
for each $q \in P$	
if $(p \prec q)$ then	If p dominates q
$S_p = S_p \cup \{q\}$	Add q to the set of solutions dominated by p
else if $(q \prec p)$ then	
$n_p = n_p + 1$	Increment the domination counter of p
if $n_p = 0$ then	p belongs to the first front
$p_{\text{rank}} = 1$	
$\mathcal{F}_1 = \mathcal{F}_1 \cup \{p\}$	
$i = 1$	Initialize the front counter
while $\mathcal{F}_i \neq \emptyset$	
$Q = \emptyset$	Used to store the members of the next front
for each $p \in \mathcal{F}_i$	
for each $q \in S_p$	
$n_q = n_q - 1$	
if $n_q = 0$ then	q belongs to the next front
$q_{\text{rank}} = i + 1$	
$Q = Q \cup \{q\}$	
$i = i + 1$	
$\mathcal{F}_i = Q$	

幻　今天 11:58

首先，对于每个解（通常用 "p" 表示），我们计算两个指标：1) 支配计数 np，即支配解 p 的解的数量，和 2) Sp，即由解 p 支配的一组解。这需要执行 O(MN^2) 次比较操作。

第一个非支配前沿中的所有解将具有支配计数为零。现在，对于每个支配计数 np = 0 的解 p，我们访问其集合 Sp 中的每个成员（q），并将其支配计数减一。在这个过程中，如果对于任何成员 q，支配计数变为零，我们将其放入一个单独的列表 Q 中。这些成员属于第二个非支配前沿。现在，上述过程继续对 Q 中的每个成员进行，以识别第三个前沿。这个过程一直持续下去，直到识别出所有前沿。

对于第二个或更高级别的非支配中的每个解 p，支配计数 np 最多为 N -1。因此，每个解 p 在其支配计数变为零之前最多会被访问 N-1 次。在这一点上，该解将被分配一个非支配级别，并且将不再被访问。

由于最多有 N - 1 个这样的解，因此总的复杂度为 O(N^2)。因此，该过程的总体复杂度是 O(MN^2)。另一种计算这个复杂度的方法是意识到第一个内循环的主体（对于每个 p 属于 Fi）将被执行恰好 N 次，因为每个解最多可以是一个前沿的成员，第二个内循环（对于每个 q 属于 Sp）可以对每个解最多执行 (N-1) 次[每个解最多支配 (N-1) 个解，每个支配检查最多需要 M 次比较]，从而导致总体上是 O(MN^2) 次计算。值得注意的是，尽管时间复杂度降低到 O(MN^2)，但存储需求增加到 O(N^2)。

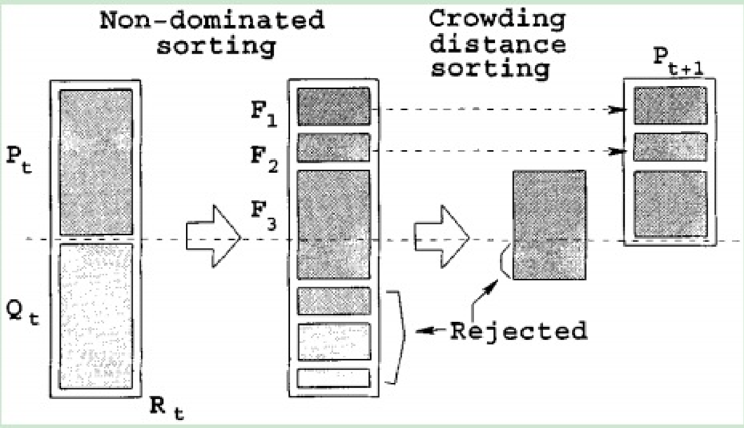
第 3 页今天 12:01

The original NSGA used the well-known sharing function approach,which has been found to maintain sustainable diversity in a popula tion with appropriate setting of its associated parameters.

第 4 页今天 14:17

crowding-distance-assignment$(\mathcal{I})$	
$l = \lvert\mathcal{I}\rvert$	number of solutions in $\mathcal{I}$
for each i, set $\mathcal{I}[i]_{\text{distance}} = 0$	initialize distance
for each objective m	
$\mathcal{I} = \text{sort}(\mathcal{I}, m)$	sort using each objective value
$\mathcal{I}[1]_{\text{distance}} = \mathcal{I}[l]_{\text{distance}} = \infty$	so that boundary points are always selected
for $i = 2$ to $(l - 1)$	for all other points
$\mathcal{I}[i]_{\text{distance}} = \mathcal{I}[i]_{\text{distance}} + (\mathcal{I}[i+1].m - \mathcal{I}[i-1].m)/(f_m^{\max} - f_m^{\min})$	

第 5 页今天 16:04



第 5 页今天 15:05

Since all previous and current population members are included in ,elitism is ensured.Now,solutions belonging to the best non dominated set are of best solutions in the combined popu lation and must be emphasized more than any other solution in the combined population.

幻 今天 15:05

将父代和子代连接成一个2N的种群是一种实现精英主义的策略，因为这允许最优秀的解决方案在下一代中得到保留。这个策略有以下几个原因：

1. **保留精英解**：在下一代中，最佳的解决方案通常是当前种群中的一部分。通过将父代和子代合并成一个更大的种群，最优秀的解决方案不会在选择操作中被遗漏，因为它们仍然存在于种群中。这确保了精英解的保留。

2. **更多的选择机会**：在选择操作中，更大的种群意味着每个解决方案有更多的机会被选择为父代，因为选择通常是基于解决方案的适应度值。最佳解决方案在适应度值上通常具有竞争优势，因此它们更有可能成为父代。

3. **多样性维护**：合并父代和子代有助于保持种群的多样性。子代通常包含一些新的解决方案，而父代包含了遗传信息。通过将它们合并在一起，可以确保下一代的多样性，因为新解决方案和经过选择的解决方案都有机会进入下一代。

4. **简化算法设计**：将父代和子代合并成一个种群简化了算法的设计和实现。它避免了必须管理两个不同的种群，从而减少了复杂性。

总之，将父代和子代连接成2N种群是一种有效的策略，旨在确保最佳解决方案的保留并促进种群的多样性，同时简化了算法的实现。这有助于提高算法的性能，特别是在多目标优化问题中，其中保持 Pareto 前沿的多样性和高质量解很重要。

幻 今天 16:05

实际上，我们还可以连接多代（3N）或者更多。
连接多代（如父代、子代和孙代）形成一个更大的种群（3N或更多）确实是一种可能的策略，被称为多代连接。这种策略有其优点和缺点，取决于具体的问题和算法设计。

优点：

1. **更大的多样性**：多代连接可以引入更多的多样性，因为它包括了多代的遗传信息。这有助于更好地探索解空间，特别是在具有复杂和多模态解的问题中。

2. **更多的机会**：更大的种群允许更多的解决方案参与竞争和选择过程，从而增加了每个解决方案被选择为父代的机会。

3. **全局搜索**：多代连接可以提供更广泛的全局搜索，因为它可以跨越多个代际传播有用的遗传信息。

缺点：

1. **复杂性增加**：管理更大的种群会增加算法的复杂性，包括选择、交叉、变异等操作。此外，需要更多的计算资源。

2. **过多的多样性**：在某些情况下，引入过多的多样性可能导致解决方案过于分散，而不容易收敛到 Pareto 前沿。

3. **计算成本**：连接多代的计算成本更高，因为需要维护更多的解决方案和评估它们的适应度。

选择是否使用多代连接策略取决于具体问题和算法的需求。对于某些复杂的多模态问题，多代连接可能是一个有用的策略，可以提高全局搜索的性能。但需要权衡复杂性和计算成本，以确保算法的有效性。多代连接通常在高度复杂的多目标优化问题或特定领域的应用中使用，而在其他问题中可能不太常见。

The overall complexity of the algorithm is ,which is governed by the nondominated sorting part of the algorithm.

幻　今天 15:30

该算法的总体复杂度为O(MN^2)，主要由算法中非支配排序的部分决定。如果谨慎执行，不需要根据非支配性对大小为2N的完整种群进行排序。一旦排序过程已找到足够数量的前沿以获得Pt+1中的N个成员，就没有继续排序的必要。

非支配解之间的多样性是通过使用拥挤比较过程引入的，该过程在锦标赛选择和种群缩减阶段中使用。由于解决方案根据其拥挤距离（解决方案在邻域中的密度度量）进行竞争，因此不需要额外的拥挤参数（例如在NSGA中需要的Oshare参数）。尽管拥挤距离是在目标函数空间中计算的，但如果需要，它也可以在参数空间中实现。然而，在本研究中执行的所有模拟中，我们使用了目标函数空间的多样性引入方法。