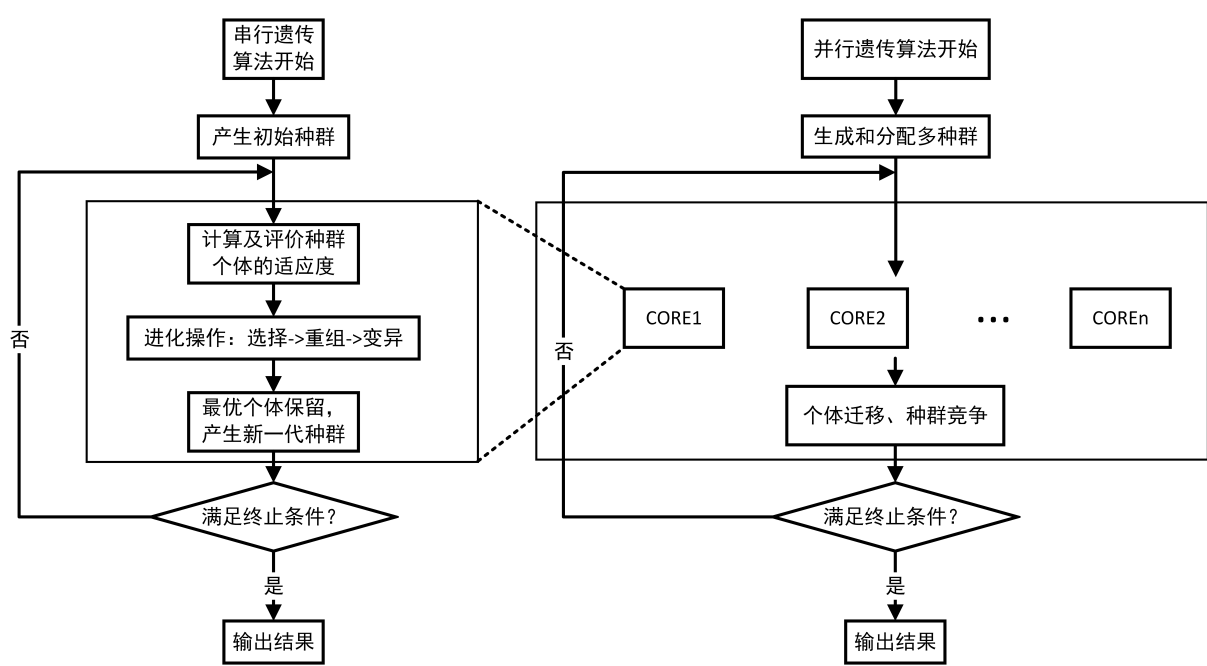


1 遗传算法概述

自然界生物在周而复始的繁衍中，基因的重组、变异等，使其不断具有新的性状，以适应复杂多变的环境，从而实现进化。遗传算法精简了这种复杂的遗传过程而抽象出一套数学模型，用较为简单的编码方式来表现复杂的现象，并通过简化的遗传过程来实现对复杂搜索空间的启发式搜索，最终能够在较大的概率下找到全局最优解，同时与生俱来地支持并行计算。

下图展示了常规遗传算法 (左侧) 和在并行计算下的遗传算法 (右侧) 的流程。



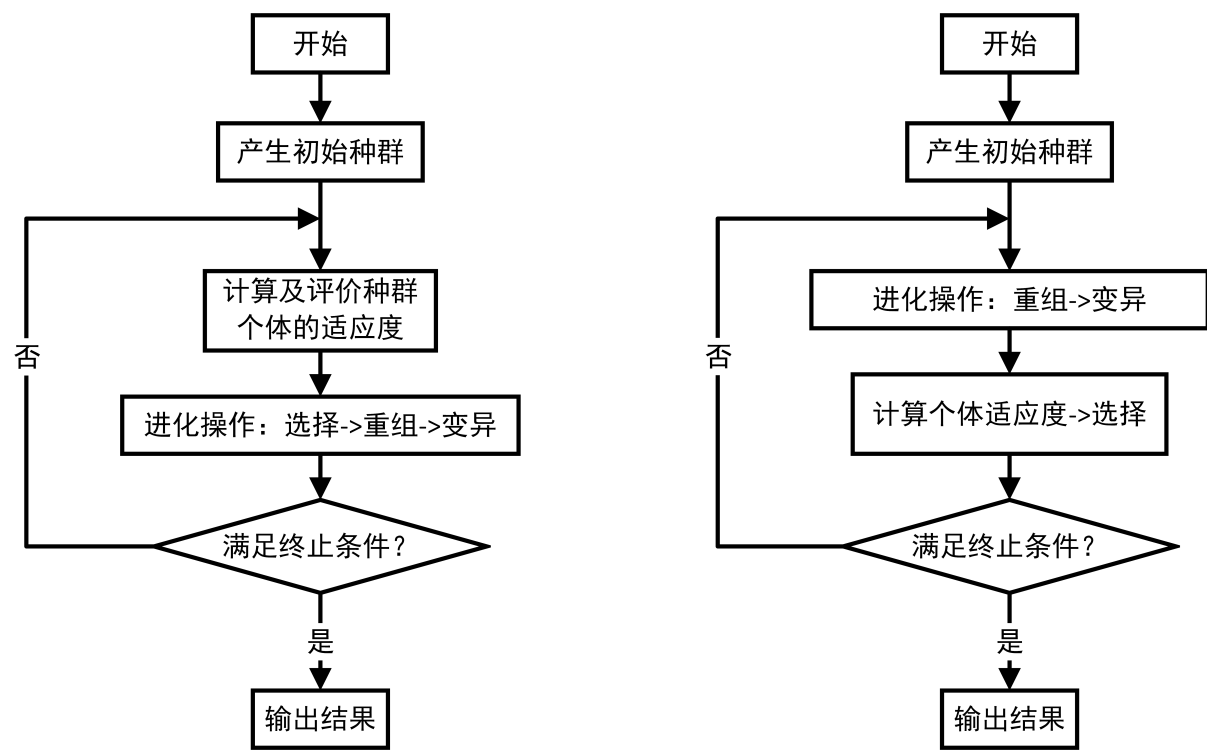
图中 $CORE_n$ 表示计算核心。不同的计算核心处理不同的单个子种群 (当然也可以处理多个子种群)，种群间互相独立进行进化 (区域模型)，种群间进行个体迁移和种群竞争。这只是其中一种并行计算遗传算法，另外还有全局模型和本地模型。

除了种群间可以并行计算外，中群内的若干个个体也可以利用矩阵化来实现并行计算。因此遗传算法具有很强的并行性。

值得注意的是：遗传算法中，重组和交叉并不是同一个概念，交叉是重组的一种。

对于常规遗传算法，在计算开始时，根据设计的编码规则随机初始化许多个体 (形成一个或多个种群)，然后评估种群中个体的适应度，并根据适应度来选择一些个体到交配池，然后对交配池中的个体进行一定概率的重组和变异产生育种后代，最后把育种后代插入到父代种群，淘汰父代中的个体，最终得到新一代种群。

当然，你也可以调整一下上面的顺序，比如重组和变异操作后不进行重插入而直接得到新一代种群；或者是不经过选择直接对父代种群的所有个体进行重组和变异操作，然后才是进行选择操作，最后把选择操作的结果作为新一代种群。如下图所示：



实际上，重插入操作可以保持种群的规模，有利于算法的稳定运行。若把选择操作放在重组和变异之后，就难以控制重组和变异的规模。但在差分进化算法里正是把选择操作放在重组和变异之后，这并不表明它因此是不好的，而是因为它在具体算法上的差异，这里就不对差分进化算法进行展开描述了。

由此我们可以看出，以遗传算法为例，进化算法和传统的搜索和优化算法有着显著不同，最明显的差异是：

- 进化算法具有与生俱来的并行性，它可以并行地搜索一组点，而不是一个点。
- 进化算法使用的是概率转换规则，并非确定性转换规则。
- 进化算法不需要额外的信息，只有目标函数和相应的适应度影响搜索方向。
- 进化算法鲁棒性强，可以与各种算法轻松地结合在一起。
- 进化算法可以整合其他优化算法的优点，比如利用其他优化算法的优化结果来生成初始种群，这种二次搜索方式在很多场合下可以大幅度提高搜索效率。
- 进化算法可以给特定的问题提供多样化的搜索结果，让用户自己选择。比如在多目标优化的进化算法里，算法给出的是一组帕累托最优解。这些最优解可以作为多组备选方案。

选择、重组和变异是遗传算法提供的经典操作算子。很多改进的遗传算法都是围绕他们展开的。