

第3章 遗传算法的数学基础

遗传算法在机理方面具有搜索过程和优化机制等属性,数学方面的性质可通过模式定理和构造块假设等分析加以讨论,Markov 链也是分析遗传算法的一个有效工具。就遗传算法的计算机理而言,在人工智能中光束搜索作为最佳优先搜索法,对搜索空间中计算量压缩有一定的效果,采用 open 表的数据结构,该表长度反映了光束的幅度,在搜索过程中不断地以新的更好的结点来调整该表。遗传算法的群体规模与光束搜索中的光束幅度类似。优化理论中的单纯形法采用称为反射的操作迭代进行的直接搜索方法。遗传算法的交叉与单纯形法的反射操作类似,群体的规模与形成单纯形的端点数类似。模拟退火法具有跳出局部解的能力,它比爬山法有所改进,通过温度降低工序的仿真,动态地对选择概率实施控制。遗传算法的选择操作是在个体适应度基础上以概率方式进行的,在概率选择方式上与模拟退火法有些相似。

自治分布式系统(autonomous distributed system)包括个体的自律性、个体(部分)之间相互作用的非决定性、有序的形态及对环境变化的适应性等。类似的概念还有全息系统(holosystem)、多 agent 系统(multiagent system)等。在遗传算法中,各个个体由其基因型结构决定其表现型性状,主要的操作属于非确定性过程,其中交叉为个体间的相互作用,根据环境中的适应度取值的不同,劣质的个体被淘汰,优秀的个体得以保留,在该选择作用下自适应性也得到体现,同时在种群中也形成一定程度的有序状态。随机搜索特性使得种群保持了一定的分散性,进化选择机制又通过环境下适应度评价的方法,完成种群的自适应优化过程。所以遗传算法也可以从自律分布系统的角度来进行分析和研究。遗传算法具有丰富的动态特性,从数学机理上加以探讨,有助于遗传算法的理论研究和应用。

本章将较全面地介绍遗传算法的基本数学理论和分析工具,包括验证基本遗传算法(SGA)的有效性的模式定理、分析遗传算法过程的 Walsh 模式变换方法、遗传算法的欺骗问题以及遗传算法的动态分析工具——Markov 链分析。

3.1 模式定理

3.1.1 模式

遗传算法的执行过程中包含了大量的随机性操作,因此有必要对其数学机理进行分析,为此首先引入“模式”(schema)这一概念。

我们将种群中的个体即基因串中的相似样板称为“模式”,模式表示基因串中某些特征位相同的结构,因此模式也可解释为相同的构形。它描述的是一个串的子集,在二进制编码的串中,模式是基于三个字符集(0, 1, *)的字符串,符号*代表任意字符,即 0 或 1。例如模式*1*描述了一个四个元的子集{010, 011, 110, 111}。

对于二进制编码串,当串长为 l 时,共有 3^l 个不同的模式,遗传算法中串的运算实际上是

模式的运算。如果各个串的每一位按等概率生成 0 或 1, 则规模为 n 的种群模式总数的期望值为

$$\sum_{i=0}^l C_l^i 2^i (1 - (1 - (1/2)^i)^n) \quad (3.1)$$

种群最多可以同时处理 $n \cdot 2^l$ 个模式。从图 3.1 可以看出其内含并行性(implicit parallelism)。如果独立地考虑种群中的各个串, 则仅能得到 n 条信息。然而, 当把适应值与各个串结合起来考虑, 发掘串群体的相似点, 我们就得到大量的信息来帮助指导搜索, 相似点的大量信息包含在规模不大的种群中。

遗传算法是如何利用这些信息的呢? 必须考察选择、交叉和变异对模式的作用效果。模式定理给出了这一问题的答案。

3.1.2 模式阶和定义距

所有的模式并不是以同等机会产生的。有些模式比起其他模式更确定, 如与模式 $0 * * * *$ 相比, 模式 $0 1 1 * *$ 在相似性方面有更明确的表示。有些模式的跨度要比其他的长, 如与模式 $1 * 1 * *$ 相比, $1 * * * *$ 要跨越整个串长更大的部分。为了定量地描述模式, 我们介绍模式中包含的两个重要参数: 模式阶(schema order)和定义距(defining length)。

定义 3.1 模式 H 中确定位置的个数成为模式 H 的模式阶, 记作 $O(H)$ 。例如, $O(0 1 1 * 1 *) = 4$ 。

模式阶用来反映不同模式间确定性的差异, 模式阶数越高, 模式的确定性就越高, 所匹配的样本个数就越少。

定义 3.2 模式 H 中第一个确定位置和最后一个确定位置之间的距离称为模式的定义距, 记作 $\delta(H)$ 。例如, $\delta(0 1 1 * 1 * *) = 4$ 。

在遗传操作中, 即使阶数相同的模式, 也会有不同的性质, 而模式的定义距就反映了这种性质的差异。

3.1.3 模式定理

假定在给定时间步 t , 一个特定的模式 H 有 m 个代表串包含在种群 $A(t)$ 中, 记为 $m = m(H, t)$, 在再生阶段, 每个串根据它的适应值进行复制, 一个串 A_i 的再生概率为

$$p_i = \frac{f_i}{\sum_{j=1}^n f_j} \quad (3.2)$$

当采用非重叠的 n 个串的种群代替种群 $A(t)$, 可以得到下式:

$$m(H, t+1) = m(H, t) \cdot n \cdot \frac{f(H)}{\sum_{j=1}^n f_j} \quad (3.3)$$

其中 $f(H)$ 是在时间步 t 表示模式 H 的串的平均适应度, 整个种群的平均适应度可记成:

$$\bar{f} = \frac{\sum_{i=1}^n f_i}{n} \quad (3.4)$$

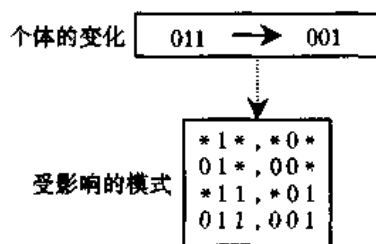


图 3.1 内含并行性

故在基本遗传算法的结构条件下,若遗传操作只选择转移到下一世代的话,则下式成立:

$$m(H, t+1) = m(H, t) \frac{f(H)}{\bar{f}} \quad (3.5)$$

这表明,一个特定的模式按照其平均适应度值与种群的平均适应度值之间的比率生长。换言之,那些适应度值高于种群平均适应值的模式在下一代中将会有更多的代表串,而对于适应度值低于种群平均适应值的模式,它们在下一代中的代表串将会减少。假设从 $t=0$ 开始,某一特定模式适应度值保持在种群平均适应度值以上一个 $c\bar{f}$, c 为一常数,则模式的选择生长方程变为

$$m(H, t+1) = m(H, t) \frac{(\bar{f} + c\bar{f})}{\bar{f}} = (1+c) \cdot m(H, t) = (1+c)^t \cdot m(H, 0) \quad (3.6)$$

(3.6)式表明,在种群平均值以上(以下)的模式将按指数增长(衰减)的方式被复制。在一定程度上这种复制算子在种群中并行地采样,许多不同的模式按照相同的规则增长或衰减。仅仅靠复制过程无助于检测搜索空间中新的区域,因为复制并没有搜索新的相似点。因而需要采取交叉操作,交叉是两个串之间随机地进行信息交换。这里仅考虑单点交叉的场合。

为了搞清楚模式受交叉影响的方式和程度,我们以一个串长为7的特定串和包含在其中的两个具有代表性的模式为例:

$$\begin{aligned} A &= 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \\ H_1 &= * \ 1 \ * \ * \ * \ * \ 0 \\ H_2 &= * \ * \ * \ 1 \ 0 \ * \ * \end{aligned}$$

设串 A 被选择来进行交叉,假设随机地选择一个交叉位置(在串长为7时仅有6个可选位置),如果选定位置在3和4之间,考察一下交叉对模式 H_1 和 H_2 的作用。其中交叉位置用“|”标记。

$$\begin{aligned} A &= 0 \ 1 \ 1 \ | \ 1 \ 0 \ 0 \ 0 \\ H_1 &= * \ 1 \ * \ | \ * \ * \ * \ 0 \\ H_2 &= * \ * \ * \ | \ 1 \ 0 \ * \ * \end{aligned}$$

显然,如果 A 的交叉对象在模式 H_1 的确定位置上与 A 不同,模式 H_1 将被破坏。而对于相同的交叉位置,模式 H_2 将保留到一个子代串中。虽然这里取的交叉位置比较特别,但是很明显,在交叉过程中模式 H_1 比模式 H_2 更不易生存,这是因为交叉点一般更易落在距离最远的确定位置之间。一般地,模式 H 被破坏的概率为 $\delta(H)/(l-1)$, 模式 H 生存概率为 $1 - \delta(H)/(l-1)$ 。模式 H_1 被破坏的概率为 $5/6$ (生存概率为 $1/6$);模式 H_2 被破坏的概率为 $1/6$ (生存概率为 $5/6$)。考虑到交叉本身是以随机方式进行的,即以概率 p_c 进行的交叉,因此对于模式 H 的生存概率可以计算如下:

$$p_s \geq 1 - p_c \cdot \frac{\delta(H)}{l-1} \quad (3.7)$$

同时考虑选择、交叉操作对模式的影响,由于选择与交叉操作是不相关的,可以得到子代模式的估计:

$$m(H, t+1) \geq m(H, t) \cdot \frac{f(H)}{\bar{f}} \left[1 - p_c \frac{\delta(H)}{l-1} \right] \quad (3.8)$$

(3.8)式表明,模式增长或衰减依赖于两个因素,一个因素是模式的适应值是在平均适应值之上还是在平均适应值之下,另一个因素是模式具有相对长还是相对短的定义距。那些既在种群平均适应度值之上同时又具有短的定义距的模式将按指数增长率被采样。

下面再考察变异操作对模式的影响。变异操作是以概率 p_m 随机地改变一个位上的值,为了使得模式 H 可以生存下来,所有特定的位必须存活。因为单个等位基因存活的概率为 $(1 - p_m)$,并且由于每次变异都是统计独立的,因此,当模式 H 中 $O(H)$ 个确定位都存活时,这时模式 H 才被保留下来,存活概率为

$$(1 - p_m)^{O(H)} \approx 1 - O(H) \cdot p_m \quad (p_m \ll 1) \quad (3.9)$$

因此,在考虑选择、交叉和变异操作的作用下,一个特定模式在下一代中期望出现的数目可以近似地表示为

$$m(H, t+1) \geq m(H, t) \cdot \frac{f(H)}{\bar{f}} \left[1 - p_c \frac{\delta(H)}{l-1} - O(H)p_m \right] \quad (3.10)$$

式中, $m(H, t+1)$ —— 表示在 $t+1$ 代种群中存在模式 H 的个体数目;

$m(H, t)$ —— 表示在 t 代种群中存在模式 H 的个体数目;

$f(H)$ —— 表示在 t 代种群中包含模式 H 的个体平均适应度;

\bar{f} —— 表示在 t 代种群中所有个体的平均适应度;

l —— 表示个体的长度;

p_c —— 表示交叉概率;

p_m —— 表示变异概率。

对于 k 点交叉的场合,上式可以作如下改变(证明从略):

$$m(H, t+1) \geq m(H, t) \cdot \frac{f(H)}{\bar{f}} \left[1 - p_c \frac{C_l^* - C_{l-1-\delta(H)}^*}{C_{l-1}^*} - O(H)p_m \right] \quad (3.11)$$

综上所述,我们可以得到遗传算法的一个非常重要的结论——模式定理(schema theorem)。

定理 3.1 在遗传算子选择、交叉、变异的作用下,具有低阶、短定义距以及平均适应度高于种群平均适应度的模式在子代中呈指数增长。

模式定理是遗传算法的基本理论,保证了较优的模式(遗传算法的较优解)的数目呈指数增长,为解释遗传算法机理提供了一种数学工具。

3.1.4 积木块假设

在模式定理中所指的具有低阶、短定义距以及平均适应度高于种群平均适应度的模式被定义为积木块(building block)。它们在遗传算法中很重要,在子代中呈指数增长,在遗传操作下相互结合,产生适应度更高的个体,从而找到更优的可行解。

积木块假设(building block hypothesis) 遗传算法通过短定义距、低阶以及高平均适应度的模式(积木块),在遗传操作作用下相互结合,最终接近全局最优解。

满足这个假设的条件比较简单,包括两方面:

- ① 表现型相近的个体,其基因型类似;
- ② 遗传因子间相关性低。

下面考虑一个整数解的最优化问题,采用 4 位二进制基因码。表现型 7 对应基因型

0111;8 对应 1000。假定 7 是问题的最优解,种群中有 8 对应的个体 1000,用一般的遗传算子不易生成最优解 7 对应的个体 0111,但采用逆位(inversion)操作,则 1000 很容易变为 0111。图 3.2 表示了种群中的积木块。

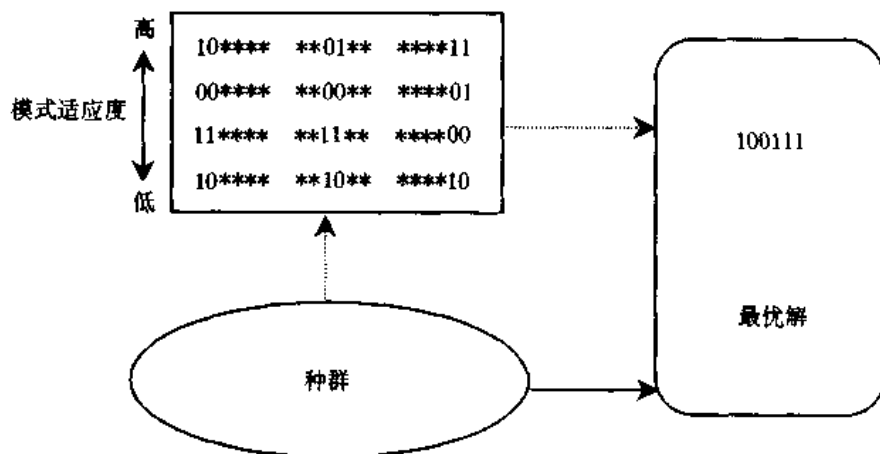


图 3.2 适于逆位操作的积木块

目前大量的实践支持积木块假设,它在许多领域内都取得成功,例如平滑多峰问题、带干扰多峰问题以及组合优化问题等。模式定理保证了较优模式(遗传算法的较优解)的样本数呈指数增长,从而满足了求最优解的必要条件,即遗传算法存在找到全局最优解的可能性;而积木块假设指出,遗传算法具备寻找全局最优解的能力,即积木块在遗传算子的作用下,能生成低阶、短距、高平均适应度的模式,最终生成全局最优解。

虽然模式定理在一定意义上解决了基本遗传算法(SGA)的有效性,但它仍有以下的缺点:

- ① 模式定理只对二进制编码适用,对其他编码方案尚没有相应的结论成立。
- ② 模式定理只给出了在下世代包含模式 H 的个体数的下限。我们无法据此推断算法的收敛性。
- ③ 模式定理没有解决算法设计中控制参数选取等问题。

3.2 Walsh 模式变换

在数字信号处理中, Walsh 函数(Walsh, 1923 年)是基函数完备的正交集,可提供类似傅里叶变换的表示方式。A. D. Bethke 的博士论文“作为函数优化器的遗传算法”(密歇根大学, 1980 年),首次提出了应用 Walsh 函数进行遗传算法的模式处理,并引入 Walsh 模式变换的概念,采用 Walsh 函数的离散形式有效地计算模式的平均适应度,从而对遗传算法的优化过程的特征进行分析。在 Bethke 的工作的基础上,Goldberg(1989 年)进一步发展了这一分析方法。

3.2.1 Walsh 函数

Walsh 函数的基函数,只有两个值 +1 和 -1。Bethke 试图用这些基函数来构造遗传算法困难程度的函数,为此将 Walsh 函数离散化,构成值域 $\{0, 1\}^l$ 的正交基。对 $j = \{0, 1, \dots, 2^l - 1\}$ 的 l 位二进制编码统一定义为 $j = \langle j_{l-1}j_{l-2} \dots j_0 \rangle$ 。

定义 3.3 (Walsh 函数) Walsh 函数 $\Psi_j(X)$ ($X = (x_{l-1}x_{l-2}\cdots x_0)$, $x_i = \{0, 1\}$, $i = 0, 1, \dots, l-1$) 定义为

$$\Psi_j(X) = \prod_{i=0}^{l-1} (1 - 2x_i)^{j_i} \quad (3.11)$$

即 $\Psi_j(X) = (-1)^\eta$, $\eta = \sum_{i=0}^{l-1} j_i x_i$ 。例如, x 为 3 位时 Walsh 函数 $\Psi_0(X), \Psi_1(X), \dots, \Psi_7(X)$ 的计算式列于表 3.1。

表 3.1 Walsh 函数

$\Psi_0(X)$	$\Psi_1(X)$	$\Psi_2(X)$	$\Psi_3(X)$
1	$1 - 2x_1$	$1 - 2x_2$	$(1 - 2x_1)(1 - 2x_2)$
$\Psi_4(X)$	$\Psi_5(X)$	$\Psi_6(X)$	$\Psi_7(X)$
$1 - 2x_3$	$(1 - 2x_1)(1 - 2x_3)$	$(1 - 2x_2)(1 - 2x_3)$	$(1 - 2x_1)(1 - 2x_2)(1 - 2x_3)$

Walsh 函数的离散值计算, 如图 3.3 所示, 图中阴影部分位置的函数值为 1, 空白部分为 0。

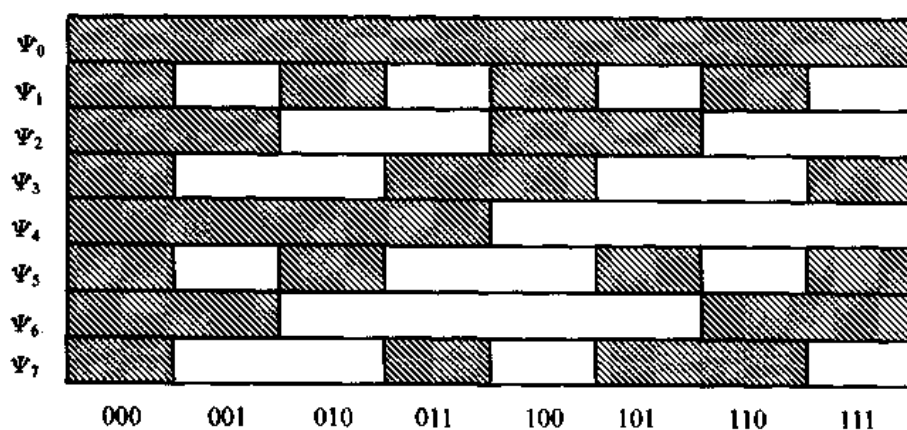


图 3.3 Walsh 函数的图示

定理 3.2 (Walsh 函数的正交性) Walsh 函数的基函数为正交函数, 满足

$$\sum_{X=0}^{2^l-1} \Psi_i(X) \Psi_j(X) = \begin{cases} 2^l, & i = j \\ 0, & i \neq j \end{cases} \quad (3.12)$$

定理 3.3 (任意函数的 Walsh 函数展开式) 设 $X = (x_{l-1}x_{l-2}\cdots x_0)$, $x_i = \{0, 1\}$, $i = 0, 1, \dots, l-1$, 则 X 定义域为 $\{0, 1, \dots, 2^l - 1\}$ 的任意函数 $f(X)$ 可以展开表示成 Walsh 多项式:

$$f(X) = \sum_{X=0}^{2^l-1} \omega_j \Psi_j(X) \quad (3.13)$$

式中 X 为一长度为 l 的二进制位串, ω_j 为 Walsh 系数

$$\omega_j = \frac{1}{2^l} \sum_{X=0}^{2^l-1} f(X) \Psi_j(X) \quad (3.14)$$

3.2.2 用 Walsh 函数表示模式平均适应度

考虑模式 H 的平均适应度

$$\begin{aligned} f(H) &= \frac{1}{|H|} \sum_{X \in H} f(X) = \frac{1}{|H|} \sum_{X \in H} \sum_{j=0}^{2^l-1} \omega_j \Psi_j(X) \\ &= \sum_{j=0}^{2^l-1} \omega_j \frac{1}{|H|} \sum_{X \in H} \Psi_j(X) \end{aligned} \quad (3.15)$$

设 $S(H, j) = \frac{1}{|H|} \sum_{X \in H} \Psi_j(X)$, 其值域为 $\{-1, 0, 1\}$ 。例如, 当 $l=3$ 时计算如表 3.2 所示, 表中 $y_i = 1 - 2x_i$ 。

表 3.2 $S(H, j)$ 的值

J	0	1	2	3	4	5	6	7
H	1	y_0	y_1	$y_0 y_1$	y_2	$y_0 y_2$	$y_1 y_2$	$y_0 y_1 y_2$
***	1	0	0	0	0	0	0	0
**0	1	1	0	0	0	0	0	0
* * 1	1	-1	0	0	0	0	0	0
* 0 *	1	0	1	0	0	0	0	0
* 0 0	1	1	1	0	0	0	0	0
* 0 1	1	-1	1	-1	0	0	0	0
* 1 *	1	0	-1	0	0	0	0	0
* 1 0	1	1	-1	-1	0	0	0	0
* 1 1	1	-1	-1	1	0	0	0	0
0 **	1	0	0	0	1	0	0	0
0 * 0	1	1	0	0	1	1	0	0
0 * 1	1	-1	0	0	1	-1	0	0
0 0 *	1	0	1	0	1	0	1	0
0 0 0	1	1	1	1	1	1	1	1
0 0 1	1	-1	1	-1	1	-1	1	-1
0 1 *	1	0	-1	0	1	0	-1	0
0 1 0	1	1	-1	-1	1	1	-1	-1
0 1 1	1	-1	-1	1	1	-1	-1	1
1 **	1	0	0	0	-1	0	0	0
1 * 0	1	1	0	0	-1	-1	0	0
1 * 1	1	-1	0	0	-1	1	0	0
1 0 *	1	0	1	0	-1	0	-1	0
1 0 0	1	1	1	1	-1	-1	-1	-1
1 0 1	1	-1	1	-1	-1	1	-1	1
1 1 *	1	0	-1	0	-1	0	1	0
1 1 0	1	1	-1	-1	-1	-1	1	1
1 1 1	1	-1	-1	1	-1	1	1	-1

由于 $f(H) = \sum_{j \in J(H)} \omega_j S(H, j)$, $J(H)$ 为 H 包含的模式序号, 例如, $H = 01*$ 所包含的模式有 $h_0 = ***$, $h_2 = *f*$, $h_4 = f**$, $h_6 = ff*$, 则 $J(H) = \{0, 2, 4, 6\}$ 。模式 H 的平均适应

度 $f(H)$ 是 Walsh $|\omega_i|$ 与 $|-1, 0, 1|$ 求积的和。

综上所述, 模式平均适应度的值计算分为三个步骤:

- ① 计算 $f(X)$ 的值;
- ② 由 Walsh 变换(3.15)式求 Walsh 系数;
- ③ 根据 Walsh 系数与模式变换表, 计算模式平均适应度的值。

Goldberg 对第③步作了改进, 采用类似快速傅里叶变换 FFT 的方法, 由函数值求取 Walsh 系数, 如图 3.4 所示。

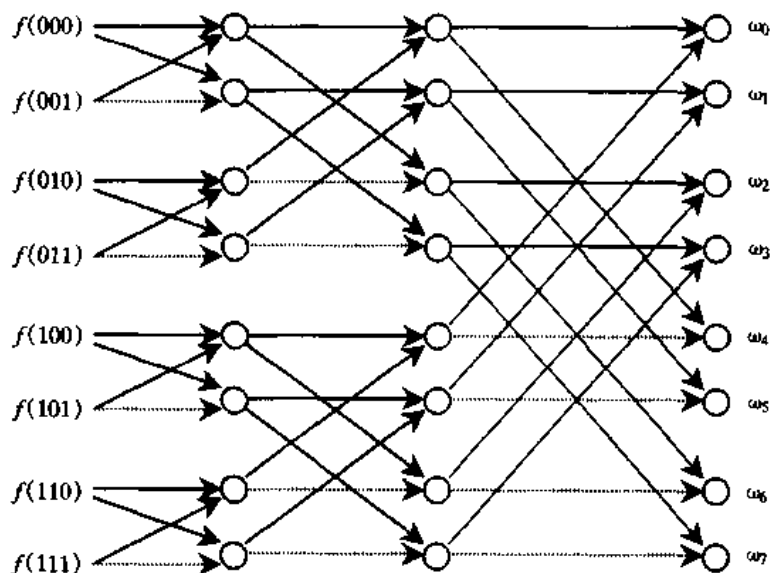


图 3.4 函数值与 Walsh 系数的关系

图 3.5 表示了模式 $1**$, $**0$, $1*0$ 对应的位串的分布。

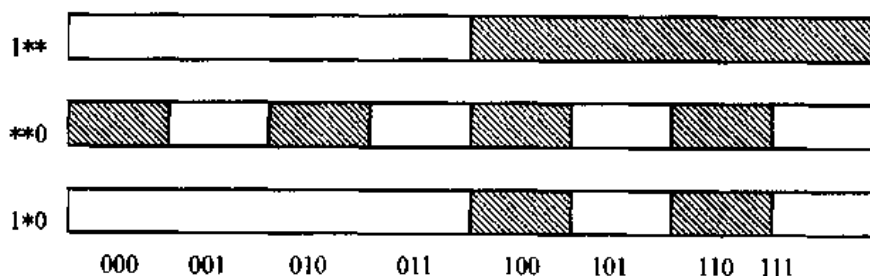


图 3.5 模式对应的位串

3.2.3 Walsh 系数与异位显性(epistasis)

模式定理认为, 具有低阶、短定义距以及平均适应度高于种群平均适应度的模式, 即所谓的积木块, 对遗传算法非常重要, 如果很容易地获得, 说明遗传算法的效果好。

我们知道, 0 阶模式代表了种群中所有的个体, 模式 $**1$, $*0*$ 等表示 1 阶模式的积木块。根据 Walsh 模式变换可以由低阶的模式适应度估计高阶模式的适应度值。

上述的例子中, 0 阶估计 $f(***)$ 是由式 (3.13) 和 (3.14) 很容易求得的。这里 $f(***)=\omega_0$, 由 0 阶估计推算 1 阶估计的方法是: 如果确定位在上位, 取适应度的偏差值为 $\pm\omega_4$; 如果确定位在中位, 取适应度的偏差值为 $\pm\omega_2$; 如果确定位在下位, 取适应度的偏差值为 $\pm\omega_1$ 。

$$f'(1**)=\omega_0-\omega_4$$

$$f'(0**)=\omega_0+\omega_4$$

2 阶和 3 阶的适应度估计值也可以类似地求得, 例如:

$$f'(11*)=\omega_0-\omega_2-\omega_4$$

$$f'(01*)=\omega_0-\omega_2+\omega_4$$

$$f'(1*1)=\omega_0-\omega_1-\omega_4$$

$$f'(*11)=\omega_0-\omega_1-\omega_2$$

$$f'(011)=\omega_0-\omega_1-\omega_2+\omega_4$$

上述估计值与准确值存在一些差别, 如根据表 3.2 得到:

$$f(11*)=\omega_0-\omega_2-\omega_4+\omega_6$$

$$f(01*)=\omega_0-\omega_2+\omega_4-\omega_6$$

$$f(1*1)=\omega_0-\omega_1-\omega_4+\omega_5$$

偏差值的 walsh 系数项的序号是: 将模式中确定位均置为 1, 不确定位 * 均置 0, 得到一个二进制数的数值; 偏差值的符号根据模式中确定位 1 的总个数的奇偶性确定。

我们将模式适应度估计值与准确值的差异称为异位显性(epistasis), 在生物学中异位显性是指一个基因对另一个非等位基因表现的显性现象。如果异位显性很大, 便不能解释仅借助低次模式的积木块生成最优解的机制。

假设 $f(x_3x_2x_1)=x_3-2x_2-x_1$, $x_i \in \{0, 1\}$, 模式的 walsh 系数计算如表 3.3 所示。

如果异位显性为正值, 表示由低阶积木块生成较高适应度的高阶模式是困难的; 相反, 如果异位显性为负值, 表示由低阶积木块生成较高适应度的高阶模式是容易的。也就是说, 对于最优解对应的可能模式而言, 低阶估计值比其实际计算值越高时说明其获得最优解可能性越大, 反之获得最优解的可能性越小。这与模式定理的结论是一致的。

为了说明 Walsh 模式变换的应用, 作为示例, 我们来分析另一种与普通二进制编码不同的 Gray 编码方式的适用性问题。

Gray 编码($g_{l-1}g_{l-2}\cdots g_0$)与普通二进制编码($b_{l-1}b_{l-2}\cdots b_0$)的关系, 可以用下式表示:

$$g_k = \begin{cases} b_{l-1}, & k = l-1 \\ b_{k+1} \oplus b_k, & k \leq l-2 \end{cases} \quad (3.16)$$

上式中 \oplus 为排斥律逻辑谓词。Gray 编码与普通二进制编码的逆关系可以表示为:

表 3.3 Walsh 系数

j	$x_3x_2x_1$	$f(x_3x_2x_1)$	ω_j
0	000	0	-0.5
1	001	-1	0.5
2	010	-2	1
3	011	-3	0
4	100	1	-0.5
5	101	0	0
6	110	-1	0
7	111	-2	0

$$b_k = \sum_{i=k}^{l-1} g_i \pmod{2} \quad k = 0, 1, \dots, l-1 \quad (3.17)$$

表 3.4 为一个三位二进制编码与其对应的 Grey 编码的示例。

表 3.4 二进制编码与其对应的 Grey 编码

十进制数	二进制编码	Grey 编码
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

对于某些优化问题,采用 Grey 编码比二进制编码优越。下面从模式分析的角度作一剖析。例如,对于最小化问题 $f(x) = (x-4)^2$,分别采用二进制编码和 Grey 编码,分别计算适应度值和 Walsh 系数,如表 3.5 所示。表 3.6 列出了模式的评价值。

表 3.5 Walsh 系数

Walsh 函数		二进制编码			Grey 编码		
编码	编号	x	$f(x)$	ω	x	$f(x)$	ω
000	0	0	16	44	0	16	44
001	1	1	9	4	1	9	8
010	2	2	4	8	3	1	32
011	3	3	1	8	2	4	16
100	4	4	0	16	7	9	16
101	5	5	1	16	6	4	0
110	6	6	4	32	4	0	8
111	7	7	9	0	5	1	4

表 3.6 模式评价值

二进制编码		Grey 编码	
H	$f(H)$	H	$f(H)$
* * 0	48	* * 0	48
* 0 *	52	* 1 *	12
* 0 0	64	* 1 0	4
1 * *	28	1 * *	28
1 * 0	16	1 * 0	36
1 0 *	4	1 1 *	4
1 0 0	0	1 1 0	0

从模式评价可知, Grey 编码的中间位突出了最优模式。比较定义距为 1 的模式 ff^* 和 *ff 的评价值, Grey 编码较普通二进制编码优越; 比较 2 阶以上的 $\omega_j (j \neq 1, 2, 4)$ 的值, 一般 Grey 编码比较小。这些说明了 Grey 编码是支持积木块假设的。

3.3 非均匀 Walsh 模式变换

上一节中, 我们介绍的 Walsh 模式中未考虑个体数的分布问题, 模式的平均评价值计算时, 假定个体服从均匀分布, 这不适用于遗传算法初期个体随机产生的情况。因此有必要考虑模式个体数的非均匀 Walsh 模式变换。

设个体的二进制编码 $X = (x_{l-1}x_{l-2}\cdots x_0)$, $x_i \in \{0, 1\}$ 。模式 $H = (h_{l-1}h_{l-2}, \cdots, h_0)$, $h_i \in \{0, 1, *\}$ 。 $f(X)$ 为 X 的适应度函数, $P(X)$ 为 X 在种群 Ω 中所占的比例。

$$\sum_{X \in \Omega} P(X) = 1 \quad (3.18)$$

属于模式 H 的个体数比例为 $P(H)$

$$P(H) = \sum_{X \in H} P(X) \quad (3.19)$$

模式 H 的适应度平均值为

$$f(H) = \frac{\sum_{X \in H} f(X)P(X)}{P(H)} \quad (3.20)$$

由于属于模式 H 的个体数目为 $2^{O(H)}$, 假设 $P(H) = 2^{-O(H)}$, 这时 $f(H)$ 的评价值为

$$\bar{f}(H) = f(H) \frac{P(H)}{2^{-O(H)}} \quad (3.21)$$

将式(3.20)代入(3.21), 可得

$$\bar{f}(H) = \frac{1}{2^{l-O(H)}} \sum_{X \in H} f(X)P(X)2^l \quad (3.22)$$

因此, 将下式

$$\bar{f}(X) = f(X)P(X)2^l \quad (3.23)$$

代替 Walsh 模式变换公式中的 $f(X)$, 即可得到非均匀 Walsh 变换。因而, 考虑个体数分布, 进行模式分析也是可行的。

3.4 欺骗问题

在遗传算法中, 将所有妨碍评价高的个体生成从而影响遗传算法正常工作的问题统称为欺骗问题(deceptive problem)。遗传算法运行过程具有将高于平均适应度、低阶和短定义距的模式重组成高阶模式的趋势。如果在低阶模式中包含了最优解的话, 则遗传算法就可能找出它来。但是低阶、高适应度的模式可能没有包含最优串的具体取值, 于是遗传算法就会收敛到一个次优的结果。下面给出有关欺骗性的概念。

定义 3.4(竞争模式) 若模式 H 与 H' 中, $*$ 的位置完全一致, 但任一确定位的编码均不

同,则称 H 与 H' 互为竞争模式。

定义 3.5(欺骗性) 假设 $f(X)$ 的最大值对应的 X 集合为 X^* , H 为一包含 X^* 的 m 阶模式。 H 的竞争模式为 H' , 而且 $f(H) > f(H')$, 则 f 为 m 阶欺骗。

在上述定义中, 当 $m = 1$ 时, 由于模式中不包含 $*$, 自然不存在欺骗性问题。

例如, 对于一个 3 位二进制编码的模式, 如果 $f(111)$ 为最大值时, 下列 12 个不等式中任意一个不等式成立, 则存在欺骗性问题。

模式阶数为 1 时 $f(* * 1) < f(* * 0)$, $f(* 1 *) < f(* 0 *)$, $f(1 * *) < f(0 * *)$

模式阶数为 2 时 $f(* 11) < f(* 00)$, $f(1 * 1) < f(0 * 0)$, $f(11 *) < f(00 *)$

$f(* 11) < f(* 01)$, $f(1 * 1) < f(0 * 1)$, $f(11 *) < f(01 *)$

$f(* 11) < f(* 10)$, $f(1 * 1) < f(1 * 0)$, $f(11 *) < f(10 *)$

阶数 1 的条件分别与 $\omega_1 > 0$, $\omega_2 > 0$, $\omega_3 > 0$, $\omega_4 > 0$ 等价。

对于阶数 2 的条件, 例如 $f(1 * 1) < f(0 * 1)$ 可化为

$$\omega_0 - \omega_1 - \omega_4 + \omega_5 < \omega_0 - \omega_1 + \omega_4 - \omega_5$$

即 $\omega_4 < \omega_5$ 。其他条件也可作类似的推导和化简。

3.4.1 欺骗函数的类型

Goldberg 曾研究用适应度函数的非单调性来研究欺骗性问题。考虑一个 2 位二进制最大化问题, 假定“11”对应最优解, 若 $H(0 *) > H(1 *)$, 则欺骗性存在。该条件可化为

$$\frac{f(00) + f(01)}{2} > \frac{f(10) + f(11)}{2} \quad (3.24)$$

设 $r = f(11)/f(00)$, $c = f(01)/f(00)$, $c' = f(10)/f(00)$, 则(3.24)式约简为

$$r < 1 + c - c' \quad (3.25)$$

Goldberg 将 $c > 1$ 的情况称为 I 类欺骗问题, $c \leq 1$ 的情况称为 II 类欺骗问题, 如图 3.6 所示。对遗传算法而言, II 类欺骗问题的求解比 I 类欺骗问题更困难。在图 3.7 中, 若以 (f_{00}, f_{00}) 为原点时, 在第一象限, 因为适应度随每个遗传因子单调增加, 叫单调问题。在二、四象限成为 I 类欺骗问题的存在领域, 在第三象限成为 II 类欺骗问题的存在领域。I 类和 II 类虚拟

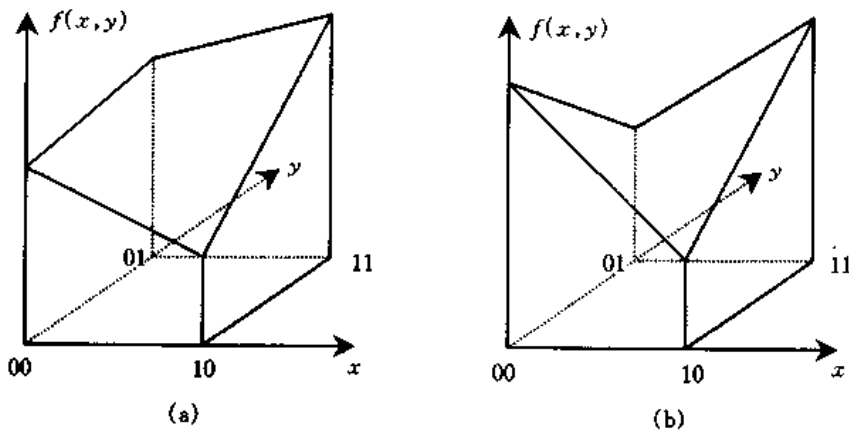


图 3.6 欺骗问题的两种类型

(a) I 类欺骗问题; (b) II 类欺骗问题

问题应该称为非单调问题,在非单调问题中同时存在欺骗和非欺骗问题。

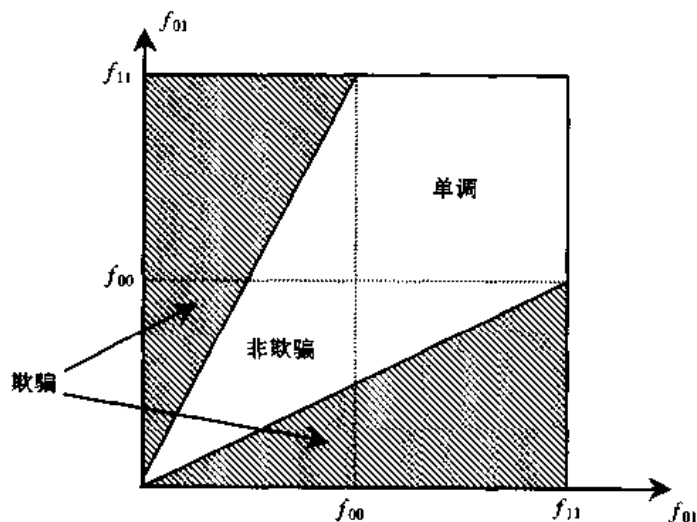


图 3.7 2 位两个个体的欺骗示域图

过去,将适应度函数的非单调问题与欺骗问题同等看待,认为遗传算法只有在单调问题里有效。但是,如果单调问题不使用遗传算法或者不使用概率搜索,一般的搜索法可能是适用的,没有遗传算法存在的必要。即使是非单调,只有存在需要高机能交叉操作(非单调且非欺骗问题),才能使遗传算法存在有意义,这不外乎使交叉操作成为遗传算法本质作用的一个证明。

3.4.2 欺骗性的化解

遗传算法中欺骗性的产生往往与适应度函数确定和调整、基因编码方式相关。下面以合适的编码方式为例,说明如何化解和避免欺骗问题。

一个 2 位编码的适应度函数

$$f(x) = 4 + \frac{11}{6}x - 4x^2 + \frac{7}{6}x^3 \quad (3.26)$$

采用二进制编码,计算个体的函数值如表 3.7 所示。这时,存在所谓的 II 类欺骗问题。

采用 Grey 编码,计算个体的函数值如表 3.8 所示。这时,II 类欺骗问题化解为 I 类欺骗问题。最优解编码为 10。

表 3.7 二进制编码函数值

编码	对应整数解	函数值
00	0	4
01	1	3
10	2	1
11	3	5

表 3.8 Grey 编码函数值

编码	对应整数解	函数值
00	0	4
01	1	3
11	2	1
10	3	5

下面的例子说明了采用适当的适应度函数调整方法可以避免欺骗问题。

设目标函数 $g(00) = 128, g(01) = 1, g(10) = g(11) = 32$, 如果适应度函数 $f(x) = g(x)$,

则 $f(0*) = 64.5$, $f(1*) = 32$ 。这时, 存在欺骗问题。

如果用适应度函数的调整方法, $f(x) = \log_2 g(x)$, 则

$$f(00) = 7, f(01) = 0, f(11) = f(10) = 5$$

得到 $f(0*) = 3.5$, $f(1*) = 5$, 从而不会产生欺骗问题。

3.4.3 遗传算法的困难问题

我们将采用基本的遗传操作包括选择和再生、交叉与变异, 以及标准的操作参数, 进行模拟进化的过程求解最优解很容易的场合, 称为遗传算法的容易问题; 反之, 称为遗传算法的困难问题。遗传算法的欺骗性并不一定是导致遗传算法的困难问题, 同样的非欺骗性并不能说明会产生遗传算法的容易问题。遗传算法的欺骗性与遗传算法的困难性不存在等价的关系, 这是由于遗传算法的欺骗性是从静态的超平面分析中给出的, 并且假定个体数无偏差, 而遗传算法的困难性来源于不适当的问题表示、交叉和变异的扰动作用、有限的种群大小、复杂的多模型状态图等。

例如, 下面的欺骗问题, 却是遗传算法的容易问题, 如图 3.8 所示。

$$\max_{0 \leq x_1, x_2 \leq 1} f(x_1, x_2) = \begin{cases} x_1^2 + 10x_2^2, & x_2 < 0.995 \\ 2(1 - x_1)^2 + 10x_2^2, & \text{其他} \end{cases} \quad (3.27)$$

这里 x_1, x_2 分别用 10 位二进制编码表示。很明显, 最优解为 (0, 1)。

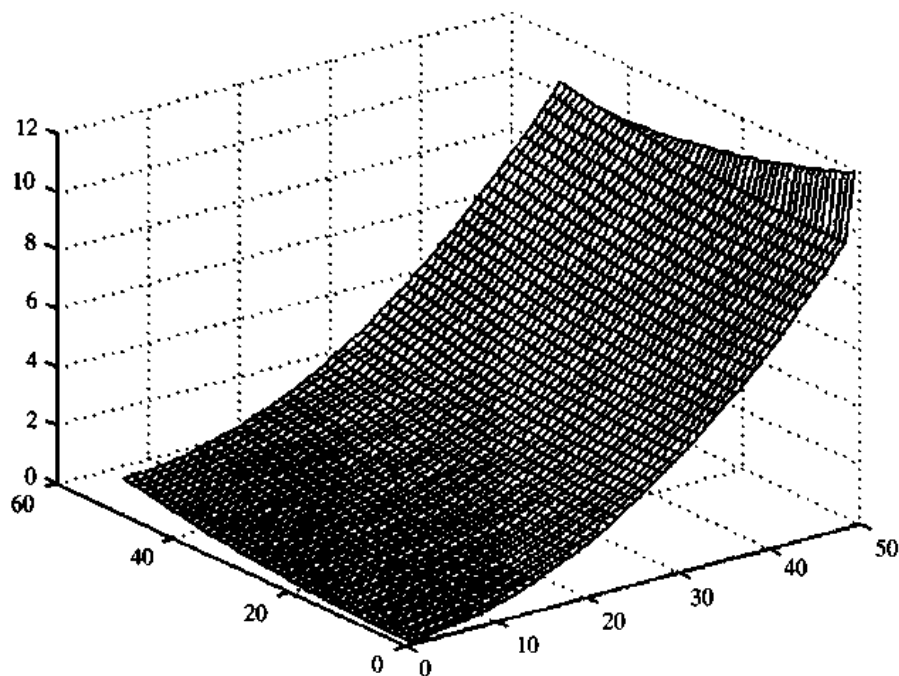


图 3.8 一个遗传算法容易产生欺骗问题的函数图形

模式(1111111111#####)包含最优解, 它比问题解(1, #)的适应度高, 因而存在欺骗问题。但对遗传算法而言却是容易问题。

下面的问题属于非欺骗问题, 但却是遗传算法的困难问题。

$$\max_{0 \leq x \leq 1} f(x) = \begin{cases} 2^{L+1}, & x = 0 \\ x^2, & \text{其他} \end{cases} \quad (3.28)$$

上式中 x 的二进制码长为 L 。据 T. Kuo(1993 年)对这类遗传算法困难问题的研究, 可以采用破坏性选择方法(disruptive selection)的遗传算法来解决。

3.5 遗传算法动态分析

从随机过程和数理统计角度探讨遗传算法较为一般的规律, 有助于较好地把握遗传算法的特性, 以提高求解效率和改善求解效果。铃木(1995 年)从 Markov 链的角度分析了基本遗传算法(SGA)的统计规律, 并得出了一些有意义的结论。

SGA 的当前种群只与前一代种群有关, 因此 SGA 可以用一个 Markov 链来描述。

定义 3.6 (有限 Markov 链) 设 $\{x_t, t \geq 0\}$ 是一列取值为有限状态空间 $S = \{s_1, s_2, \dots, s_n\}$ 上的随机变量, 若 x_{k+1} 所在的状态只与 x_k 有关, 而与 x_0, x_1, \dots, x_{k-1} 无关, 即对于任意 $k \geq 0$ 的正整数 $i_0, i_1, \dots, i_k, i_{k+1}$, 有

$$P(x_{k+1} = s_{i_{k+1}} | x_0 = s_{i_0}, \dots, x_k = s_{i_k}) = P(x_{k+1} = s_{i_{k+1}} | x_k = s_{i_k}) \quad (3.29)$$

成立, 则称 $\{x_t, t \geq 0\}$ 为 Markov 链, $P(x_{t+1} = s_j | x_t = s_i)$ 称为在时刻 t 由状态 s_i 转移到状态 s_j 的转移概率, 记为 $p_{ij}(t)$ 。若转移概率与时间无关, 则称 Markov 链为齐次的。

SGA 的当前种群只与前一代种群有关, 因此 SGA 可以用一个 Markov 链来描述。

令个体 i 的适应度值为 $f(i)$, 种群规模记为 M , 个体串长为 L , 并将个体表示为如下形式:

$$a_{i1}a_{i2}\cdots a_{iL} \in A^L$$

这里, $A = \{0, 1, \dots, \alpha\} (\alpha \geq 2)$, a_{ij} 取 c 进制值 $j = 1, 2, \dots, L$ 。每个个体 i 也可记为无符号整数形式 $\sum_{j=1}^L a_{ij}\alpha^{L-j}$ 。每个种群 k 记为向量形式:

$$[Z(0, k), Z(1, k), \dots, Z(\alpha^L - 1, k)]$$

这里, $Z(i, k)$ 表示标记为 $i = 0, 1, \dots, \alpha^L - 1$ 的个体出现次数。

为简化讨论起见, 令 $\alpha = 2$, 选择按下列概率 $p(i, k)$ 进行:

$$p(i, k) = \frac{f(i)Z(i, k)}{\sum_{h=0}^{2^L-1} f(h)Z(h, k)} \quad (3.30)$$

交叉概率记为 x , 形式为单点交叉, 即从 $(1, L-1)$ 中以均匀概率选择交叉点。突变概率为 μ , 对个体的位值进行翻转操作。每次保留 1 个具有最高适应度的个体作为良种, 记为 $i^*(k)$ 。

种群的初始分布可记为

$$q^{(0)} = (q_1^{(0)} q_2^{(0)} \cdots q_N^{(0)})$$

令 Q 表示转移矩阵, 则 n 代时的种群为

$$q^{(n)} = (q_1^{(n)} q_2^{(n)} \cdots q_N^{(n)}) = q^{(0)} Q^n \quad (3.31)$$

引理 3.1 存在满足下式的系数 $\rho_{k,v}^{(t)} (t = 1, 2, \dots, N)$

$$q_{k,v}^{(n)} = \sum_{i=1}^N \rho_{k,v}^{(i)} \lambda_i^n \quad (3.32)$$

当且仅当转移矩阵 $Q = (q_{k,v})$ 是本原的(不可约和非周期的)或者不可分解的(可约为只有一个非周期递归类的情况), 其中

$$1 = |\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_N|$$

而且 $\lambda_i (i=1, 2, \dots, N)$ 是转移矩阵的特征值, $\rho_{k,v}^{(i)}$ 是种群 k 的不可约值, 并与种群 $v=1, 2, \dots, N$ 的平稳概率 $q_v^{(\infty)}$ 相一致。

引理 3.2 在修改的良种策略中, 从种群 k 到 v 的转移矩阵 $Q = (q_{k,v})$ 具有 2^L 个子矩阵 $Q(i)$ 。 $Q(i)$ 的大小为 $N(i) \times N(i)$, $i=0, 1, \dots, 2^L-1$, 它的对角元素、对角线右上角的元素均为零, 其中, $N(i)$ 是种群 k 的数量, 这里 $i = i^*(k)$ 。

定理 3.4

$$N(i) = C_{M-1+2^L-i}^{M-1} \quad (3.33)$$

引理 3.3 每一子矩阵 $Q(i)$ ($i=0, 1, \dots, 2^L-1$) 的 $N(i)$ 个特征值与矩阵 Q 的 $N = \sum_{i=0}^{2^L-1} N(i)$ 个特征值是同一的。

定理 3.5 存在满足下式的常数 C ,

$$\sum_{k \in K} a_k^{(n)} \geq 1 - C |\lambda_*|^n \quad (3.34)$$

其中 $|\lambda_*| = \max_{1 \leq i \leq 2^L} \max_{1 \leq j \leq N(i)} |\lambda_{i,j}| < 1$, 且 $\lambda_{i,j} (j=1, 2, \dots, N(i))$ 表示子矩阵 $Q(i)$ 的 $N(i)$ 个特征值 ($i=0, 1, \dots, 2^L-1$)。

定理 3.6 对于 λ_* , 存在不依赖于变异概率 μ 且满足下式的常数 A ,

$$|\lambda_*| \leq 1 - A\mu^\delta (1-\mu)^{L-\delta} \quad (3.35)$$

其中突变阶 δ 是个体 i 和 $j (< i)$ 之间的海明(Hamming)距离 $d(i, j)$ 的极值。海明距离是指两个个体 L 位中位值不一致的个数。突变阶 δ 可被认为是陷入局部搜索的程度。

推论 3.1 如果突变概率 μ 满足 $\mu = \delta/L$ 时, $1 - A\mu^\delta (1-\mu)^{L-\delta}$ 最小。

从此推论中可以获知, 如果变异概率与突变阶之间存在正比关系, $|\lambda_*|$ 获得最小。

以上基于有限种群模型的 3 个定理, 有助于对遗传算法内在机理的探讨, 所得的结论适用于良种选择策略(精英主义策略)引导下的遗传算法。在选择操作中保留当前最好解的基本遗传算法(SGA)能以概率收敛到最优解。

参考文献

- [1] Forrest S, Mitchell M. What Makes a Problem Hard for a Genetic Algorithm? Some Anomalous Results and Their Explanation. In: Proceedings of the Fourth International Conference on Genetic Algorithms, CA, Morgan Kaufman, 1991, 120~131
- [2] Goldberg D E. Genetic Algorithms and Walsh Functions: Part I, A Gentle Introduction, Complex Systems 1989(3): 129~152

- [3] Goldberg D E. Genetic Algorithms and Walsh Functions: Part II, Deceptive and its Analysis, Complex Systems, . 1989(3): 153~171
- [4] De Jong, Spears W M. D. F. Gordon: Using Markov Chains to Analysis GAFOs. In: Foundations of Genetic Algorithms, Morgan Kaufmann Publishers, 1994, 115~137
- [5] Goldberg D E. Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, MA, Addison - Wisely, 1989
- [6] Kuo T, Hwang S Y. A Genetic Algorithm with Deceptive Selection in Genetic Algorithms. In: Proceedings of the Fifth International Conference on Genetic Algorithms, Morgan Kaufmann, 1993, 65~69
- [7] Deb K, Goldberg D E. Sufficient Conditions for Deceptive and Easy Binary Functions. Annals of Mathematics and Artificial Intelligence, 1994, 10(4): 385~408
- [8] Suzuki J. A Markov Chain Analysis on Simple Genetic Algorithm. IEEE Trans. SMC, 1995, 25(4): 655~659
- [9] Matsui K, Kosugo Y. An Analysis on Genetic Algorithms Using Markov Processes with Rewards. In: Proceedings of the 1996 IEEE Signal Processing Society Workshop, Sixth in a Series of Workshops Organized by the IEEE Signal Processing Society Neural Networks Technical Committee, 1996, 130~139
- [10] Mahfoud S W. Population Size and Genetic Drift in Fitness Sharing, In: Foundations of Genetic Algorithms, Morgan Kaufmann Publishers, 1994, 185~223
- [11] Blichle T, Thiele L A. Mathematical Analysis of Tournament Selection. In: Proceedings of the 6th International Conference(ICGA95), Eshelman L(ed). Morgan Kaufmann Publishers, 1995
- [12] 潘正军, 康立山 等. 演化计算. 清华大学出版社, 广西科学技术出版社, 1998
- [13] 史忠植. 高级人工智能. 北京: 科学出版社, 1998
- [14] 张文修, 梁怡. 遗传算法的数学基础. 西安: 西安交通大学出版社, 2000
- [15] 刘健勤. 人工生命理论及其应用. 北京: 冶金工业出版社, 1997
- [16] 刘勇, 康立山 等. 非数值并行计算(第2册)——遗传算法. 北京: 科学出版社, 1995