

模拟二进制交叉

连续搜索空间

卡利扬莫伊 • 德布和拉姆 • 布桑 • 阿格拉瓦尔

IITK/ME/SMD-94027

1994 年 11 月

召集人、技术报告

机械工程系

印度理工学院

坎普尔, UP208 016

印度

连续搜索空间的模拟二进制交叉

卡利扬莫伊·德布和拉姆·布山·阿格拉瓦尔

机械工程系

印度理工学院

印度坎普尔, 上角 208 016

电子邮件: deb@iitk.ernet.in

摘要

二进制编码遗传算法 (GAs) 在具有离散搜索空间的问题上的成功, 在很大程度上取决于用于表示问题变量的编码, 以及将构建块从父字符串传播到子字符串的交叉操作符。在求解具有连续搜索空间的优化问题时, 二进制编码的 GAs 通过使用二进制字符串中的问题变量的编码来离散搜索空间。然而, 在有限长度字符串中对实值变量的编码使得许多 difficulties — inability 在得到的解中达到任意精度, 问题变量的固定映射, 与二进制编码相关的固有汉明悬崖问题, 以及在连续搜索空间中对荷兰图式的处理。虽然开发了一些真实编码的 GAs 来解决具有连续搜索空间的优化问题, 但这些交叉算子的搜索能力是不够的。本文根据从给定的一对父解中生成任意子解的概率, 定义了交叉算子的搜索能力。由于二进制编码 GAs 在离散搜索空间问题中的成功, 我们开发了一个实编码交叉 (我们称为模拟二叉交叉或 SBX) 算子, 其搜索能力与二进制编码 GAs 中使用的单点交叉相似。对一些不同难度和维数的实值测试问题的仿真结果表明, 具有 SBX 算子的实编码 GAs 能够与具有单点交叉的二进制编码 GAs 一样好或更好。SBX 被发现在具有一个狭窄的全局盆地的多个最优解的问题和在全局最优的下界和上界事先未知的问题中特别有用。此外, 对双变量阻塞函数的模拟表明, SBX 的实编码算法是戈德堡所建议的, 在大多数情况下, SBX 的实编码算法的性能与单点交叉的二进制 GAs 相似。基于这些令人鼓舞的结果, 本文提出了对本研究的一些扩展。

1 介绍

交叉算子被认为是遗传算法 (GA) 作为优化工具的主要搜索算子 (Goldberg, 1989)。交叉运算符的目的是双重。为了创建好的字符串, 必须彻底搜索表示问题变量的初始随机字符串。此后, 这些字符串的好部分必须组合在一起, 以形成更好的字符串。遗传算法文献中存在许多交叉算子; 然而, 实现上述两个方面的搜索能力因不同交叉而异。GAs 的成功操作还取决于用于表示问题变量的编码机制 (Kargupta, Deb, 和戈德堡, 1992 年; 拉德克里夫, 1993 年)。除非好的构建块被严格编码为 \wedge , 否则交叉操作符不能将构建块组合在一起 (戈德堡、Korb 和 Deb, 1989)。这种编码-交叉交互作用对气体的成功工作非常重要。二进制编码 GAs 在离散搜索空间问题上的成功很大程度上是由于对编码的仔细选择和交叉操作符的仔细选择, 在连续几代中传播有用的构建块时补充彼此的交互。问题变量的紧密或松散编码问题在很大程度上被称为连锁问题, 最近通过使用混乱的二进制编码的遗传算法来解决 (戈德堡, Korb, Deb, 1989; 戈德堡, 等。然而, 在真实编码的气体中, 这些变量被直接使用。因此, 构成一个变量的位的紧密或松散编码是不会出现的, 但不同变量的紧密或松散编码仍然存在。是否真的存在一个真实编码的遗传算法来解决这个连锁问题是一个有趣的进一步研究的问题。然而, 在本文中, 我们认识到这方

面的编码对气体成功工作中多变量固有问题的重要性，但我们没有解决这个问题。相反，我们设计了一个真实编码的交叉算符，它寻址了二进制编码气体在处理连续搜索空间问题时固有的其他困难。

在解决使用二进制编码的气体的具有连续搜索空间的问题时，通常选择二进制字符串来编码问题变量。当二进制（离散）编码用于连续搜索空间时，会出现许多困难。一个困难是与某些字符串相关联的汉明悬崖，从它过渡到一个邻近的解决方案（在真实空间中）需要改变许多位。二进制编码中的汉明悬崖人为地阻碍了连续搜索空间中的逐渐搜索。另一个困难是无法在最优解中实现任何任意的精度。在二进制编码的 GAs 中，必须预先选择字符串长度，以使 GAs 在解决方案中达到一定的精度。所需的精度越高，字符串的长度就越大。对于大的字符串，总体规模的要求也很大(戈德堡, Deb, 和 Clark, 1992)，从而增加了算法的计算复杂度。由于固定的，映射编码用于编码变量，变量边界必须使它们括号最优变量值。由于在许多问题中，这些信息通常不是事先知道的，这可能会导致在这些问题中使用二进制编码的气体有一些困难。此外，仔细考虑二进制字符串中的模式处理就会发现，并非所有荷兰的模式在具有连续搜索空间的大多数问题中都同样重要。对于一个连续的搜索空间，有意义的模式是那些表示搜索空间的连续区域的模式。模式（1）？例如，表示离散化搜索空间中的所有其他点。虽然这个模式在某些周期或振荡函数中可能有用，但在大多数问题中，模式（1****）表示一个更有意义的模式，表示搜索空间的右半部分。因此，需要重新设计在二进制编码中使用的交叉操作符，以增加与连续搜索空间有关的更有意义的模式的传播。

然而，出于二进制气体在离散搜索空间问题和实现需要一个有效的真实编码遗传算法来消除上述困难的气体连续搜索空间问题，我们设计了一个真正的交叉操作符只是模拟二进制交叉操作符但不使用变量的编码。为了实现这一点，我们根据由任意两个给定的父字符串创建的任意子字符串的概率分布来定义交叉算子的搜索能力¹首先计算了单点交叉算子的搜索能力。随后，模拟的二叉交叉(SBX)被开发为具有类似于单点交叉的搜索能力。的区别实现的真实编码气体 SBX 和二进制编码气体单点交叉是在前一种方法消除编码变量和子字符串从概率分布取决于父字符串的位置。

虽然在文献中存在许多其他真实编码的气体(埃舍尔曼和谢弗, 1993; Wright, 1991)，在论文中定义，这些交叉运算符的搜索能力是不够的。其中，埃舍尔曼和谢弗的研究特别重要，因为他们引入了与连续搜索空间相关的图式（他们称之为区间图式）的概念。这里可以提到的是，间隔图式在概念上类似于戈德堡（1991）引入的虚拟字母和拉德克里夫（1991）定义的局部格式。虚拟字母、局部格式或区间图式表示搜索空间中的连续区域。虽然埃舍尔曼和谢弗设计了一个混合交叉(BLX)算子，但他们并没有从交互模式处理的角度来构建交叉算子。在后面一节中，我们观察到他们最好发现的交叉 BLX-0.5 并不能很令人满意地将间隔图式从父点传播到子点。有趣的是，我们发现一种特殊情况下的 SBX 操作符处理区间模式比他们的最佳报告的 BLX-0.5 操作符更好。

在关于进化策略的文献中，直接使用实值参数，但在这些研究中使用突变作为主要的搜索算符(Rechenberg, 1973)。直到最近，交叉操作符才被用于进化策略研究中(Back, 霍夫迈斯特和施韦费尔, 1991 年)。对于每个变量，离散交叉选择其中一个父值作为子变量，中间交叉

使用父值的平均值作为子变量。虽然引入了其他类型的交叉算符，但这些算符是确定性的，只能创建一个搜索点。因此，这些操作员没有足够的搜索能力。然而，如果考虑到交叉和后续的突变算子的影响，它们的组合搜索能力将变得显著。由于本文主要讨论和比较交叉算子，因此不考虑进化策略方法中的交叉和突变算子的联合作用。

■交叉操作符的设计依赖于许多其他因素，如拉德克里夫（1991）提出的尊重，分类和格式（相似模板）的传递性。这里设计的交叉操作符满足雷德克里夫⁶的设计标准，但它不是一个 R^3 （随机、尊重、重组）操作员。相反，偏离儿童的随机分布和访问搜索点的能力（拉德克里夫的遍历性准则）被用来定义交叉算子的搜索能力。

采用具有单点交叉的二进制编码气体和采用 BLX-0.5 运算符的真实编码气体。测试函数还包括一个由戈德堡（1991）引入的双变量、双侧主动阻塞函数。戈德堡认为，重新编码的 GAs 的工作是这样的，在这些函数中，总体首先收敛到虚拟字母（高于平均点），然后收敛到全局最优被两个局部最优阻塞。我们使用 SBX 操作符的模拟支持虚拟字母的戈德堡⁷理论，在这个简单的阻塞函数上的模拟结果表明，使用 SBX 的真实编码 GAs 与使用单点交叉的二进制编码 GAs 相似。仿真结果还表明，使用 SBX 算子的实编码 GAs 可以克服汉明悬崖问题、精度问题和固定映射问题，在困难问题上执行类似于单点交叉的二进制编码 GAs（但优于 BLX0.5 的真实编码 GAs）。这些结果是令人鼓舞的，并建议使用真实编码的 GAs 与 SBX 算子对其他具有连续变量的优化问题。基于本文的这些结果，本文概述了对这项工作的一些扩展。

2 二进制交叉运算符

在二进制编码的 GAs 的许多应用中，使用了单点交叉。在单点交叉中，沿着字符串的长度选择一个随机的交叉点，并且交叉点一侧的位在两个父字符串之间交换。在单变量优化问题中，交叉的操作是从两个父字符串中创建两个新的子字符串。在一个多变量优化问题中，每个变量通常被编码为一定数量的位（统称为子字符串），这些子字符串被组合形成一个更大的字符串。由于在单点交叉算符中，只从完整的字符串中选择一个交叉点，因此单点交叉的作用可能会同时破坏一个变量或连续变量的组合。在下面，我们计算在单点交叉下从两个选择的点创建一个子点的概率。

2.1 搜索能力

交叉算子的搜索能力在这里被定义为对算子在搜索空间中创建任意点的灵活性的度量。拉德克里夫（1991）提出了成功设计交叉操作员的一些标准。该研究表明了现有的不同交叉操作符如何满足这些标准，但该研究没有建议，当他们满足所有要求的标准时，采取任何比较操作符的措施。这里值得一提的是，拉德克里夫还定义了一个遍历性标准，即通过来自任何种群的遗传操作符访问任何搜索点的能力。摘要利用拉德克里夫遍历性判据，将交叉算子的搜索能力定义为从两个给定的父点在搜索空间中创建任意点的概率。为了使分析更简单，我们假设该函数是一个单变量函数。稍后，我们将讨论如何扩展对多变量函数的分析。

布隆			&	设计	布隆			A_0	设
1 0 1 0		1	0 1	85	1 0 1 0		0	1 1	8

$$\begin{array}{ccc|ccc} 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ B_1 & & & A_2 & & & \end{array} \quad \text{平} \quad \frac{51}{68} = \begin{array}{ccc|ccc} 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ B_1 & & & & & & \end{array} \quad \text{平} \quad \frac{5}{6}$$

图 1: 显示了在两个随机字符串上的单点交叉的作用。DV 代表已解码后的参数值。注意, 交叉操作前后解码参数值的平均值是相同的。

当两个长度为 ℓ 的父二进制字符串在任意站点上交叉时, 交叉站点两侧的内容将在父站点之间交换。让我们想象交叉位是字符串右边的 k 位。我们还假设最左边的位是底层二进制编码中最高位。如果第一父位中最右边的 k 位的解码值为 A_1 , 其余 $(\ell - k)$ 位的解码值为 B_1 , 则该字符串的解码值如下:

$$x_1 = B_1 + A_1. \quad (1)$$

第二个父字符串的解码值也可以分别通过知道和作为该字符串的右 k 位和其余 $(\ell - k)$ 位的解码值来计算: $x_2 = B_2 + A_2$ 。图 1 显示了对任意字符串的单点交叉操作以及在特定站点交叉获得的子字符串。单点交叉运算符的效果是交换值 A_1 和两个父点之间。因此, 子字符串的解码值如下:

$$x_1' = B_1 + A_2, \quad (2)$$

$$x_2' = B_2 + A_1, \quad (3)$$

上述方程提出了单点交叉的一个重要性质。注意, 父字符串 $(x_1 + x_2)/2$ 的解码值的平均值与子字符串 $(x_1' + x_2')$ 的平均值相同。这也可以在图 1 中通过计算父字符串和子字符串的解码参数值来实现。值得注意的是, 解码参数值的这种特性对于线性映射规则的父点和子点的真实参数值也很有效。假设有一个线性映射规则, 可以得到以下父点:

$$P_i = m x_i + t \quad i = 1, 2, \quad (4)$$

其中, m 和 t 是两个常数。还可以使用以下相同的映射规则计算子点:

$$C_i = m x_i' + t \quad i = 1, 2. \quad (5)$$

使用上面的两个方程和之前对情况的观察, 很难证明父点的平均值与子点的平均值是相同的。换句话说, 子点与父点的中点是等距的。我们将利用这个特性来开发实编码交叉运算符。此外, 我们观察到子点可能位于父字符串边界的区域内部或外部, 这取决于字符串和交叉站点的位置。为了定义子点相对于父字符串的扩展, 我们定义了一个扩展因子 α , 作为子点与父点的扩展的比率:

$$\alpha = (C_i - P_2) / (P_1 - P_2)$$

根据上述对扩散因子的定义, 我们将交叉分为三个不同的类别: 定义 1 所有具有扩散因子 $\alpha < 1$ 的交叉操作都是收缩交叉。当子点被父点包围时, 子点的分布 (C_i 和 C_2 值的绝对差) 小于父点的分布。因此, 扩散因子 α 小于 1。由于在非重叠遗传算法中, 子点通常取代父点, 这种交叉似

乎对父点有收缩效应。

定义 2 所有具有扩展因子 >1 的交叉操作都在扩展交叉。

当子点包围父点时，子点的绝对差异大于父点的绝对差异。这种交叉可以扩展父点以形成子点。

定义 3 所有具有扩散因子的交叉操作 $\beta=1$ 是平稳交叉。

在平稳交叉点中，子点与父点相同。传播因子 0 的一个有趣特征是，它隐藏了子点的真实传播的影响。传播因子 β 表示子点相对于父点的传播。对于两个相近的子点，如果父点彼此靠近，则可以得到两个固定值的子点，或者如果父点彼此远离，则可以得到两个遥远的子点。在单点交叉中，收缩、扩展或平稳交叉的发生取决于父串和交叉点的位置。

根据上述不同类型交叉的交叉的定义，现在让我们计算任意两个随机父字符串中三种不同类型的交叉出现的概率。我们首先考虑两个父字符串在从最右边的位开始的一个位点 $k \in (0, L-1)$ 上发生交叉。同样，我们假设最左边的位是最重要的位。将两个父字符串表示为 q 和 r ，我们将解码后的值写入如下：

$$\hat{q}_1 = \sum_{i=0}^{L-1} Z_i q_i \quad (7)$$

$$\hat{r}_2 = \sum_{i=0}^{L-1} Z_i r_i \quad (8)$$

$$y_2 = \sum_{i=0}^{k-1} \epsilon_i \quad (9)$$

$$y_2 = \sum_{z=0}^{k-1} \epsilon_z \quad (10)$$

利用方程 7 和公式 8 中给出的单点交叉算子的性质，我们得到了子串的解码值如下：

对于线性映射规则（方程 9 和 10），式 6 中定义的扩展因子可以用父串和子串的解码值来表示。

因此，方程 7 到 10 可以用来写出扩展因子 0。但是我们通过假设所有的 $z=0, 1, \dots, k-1$ ，都有一个参数 U_i ，(四-饥)来简化扩散因子的表达式 $U_i = 1$ 。请注意，每个 U_i 可以取一个值 -1、0 和 1。

$$X_1 = X_2$$

$$E_{ti} = E_{-o} \text{ 妃}$$

$$\frac{i - g}{i + g}$$

其中 $\langle \& \rangle = \epsilon$ 冒如終， $\%2$ 气上述术语对 $\&$ 对任何两个父字符串 (或换句话说，对于任何 U_i 值) 和任何交叉站点 (或任何 k) 有效。但是，让我们首先考虑收缩交叉点 (其中为 $0 < \dots < 1$)。为了简化我们的分析

此外，我们假设父字符串是两个极端字符串，具有所有的 $(111\dots 1)$ 和所有的 $(000\dots 0)$ 。在这种情况下，每个位的等位基因值的差异是 1 (或 $U_i=1$ 对于 $z=0, 1, \dots, k-1$)。将上述表达式中的 $U_i=1$ 替换为 $=1$ ，我们得到以下结果

表达式

$$= 1 - 2(2^k - 1)/(2^n - 1) \text{ 幻 } 1 \text{ } 2^{-(n+1)} \quad (11)$$

图 2 绘制了两个长度为 20 的极端串的上述扩散因子。从图中可以清楚地看出，对于大多数交叉站点 fe ，传播因子 (3^1) 。

通过将扩展因子的分布作为交叉点的函数，我们现在可以计算从两个给定的父字符串中创建任意字符串 (代表 0) 的概率。让我们再次考虑一下图 2。对于一个特定的值，我们考虑一个小范围的 dp ，并找到有多少个交叉操作 (dfe) 将在该范围内产生子点。这可以通过计算上述表达式在相应处的斜率来实现

$$dk = \frac{1}{WT^{d(3-1)}} \quad (12)$$

用 $C(1/3)$ 表示 0 出现的概率，并且知道 0 在收缩交叉的 0 和 1 之间，我们得到了 dpt_0 的分布：

$$= \left(\frac{1}{L} \right)^{1/L} \cdot \left(\frac{1}{I} \right)^{1/I} \text{ 制 } j \text{ 衣} \quad (13)$$

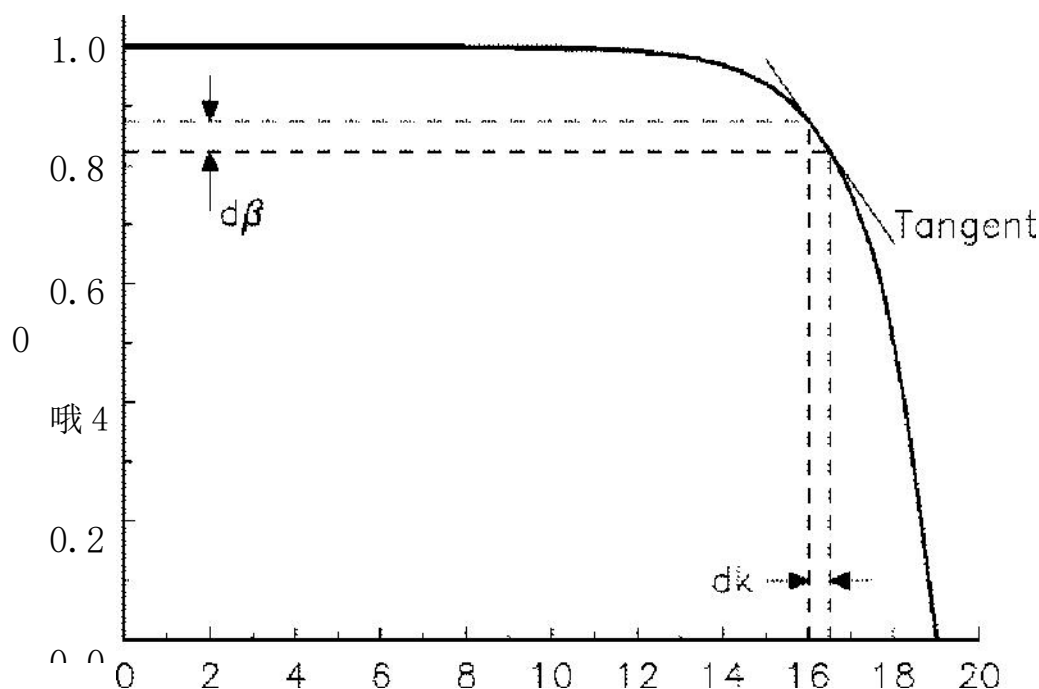


图 2: 显示了 $f=20$ 的 $f(3)$ 与 k 的分布。

右侧表达式的分母是一个常数量，它使分布 $C(3)$ 成为一个概率分布。在这里，我们对精确地计算上述概率分布不感兴趣，而是感兴趣的是寻找概率分布 $C(3)$ 与 0 的函数关系。因此，在这里只计算上述表达式的分子就足够了。通过将方程 11 对 k 进行微分，并将 k 的表达式代入，得到方程 13 中右侧表达式的分子如下：

$$\frac{dC(3)}{dk} = f \quad (14)$$

其中， k 是一个常数项。上述分布的函数形式如图 3 所示（对于 $0 < 3 < 1$ ）。图中显示，当 3 向 1 增加时，概率增加。因此，创建更接近父点的子点的概率要大于远离父点的子点的概率。

²扩展交叉点的概率分布 ($3 > 1$) 可以通过考虑其他两个父字符串和使用上述过程来得到。然而，通过实现一个简单的概率事件，也可以从收缩交叉点中得到概率分布。如果在一个收缩交叉中发现的两个子字符串在同一站点上进行交叉，则得到相同的父字符串。因此，对于每一个具有扩散因子的收缩交叉，都存在一个扩散因子为 $1/3$ 的扩展交叉。换句话说，我们可以认为发生收缩交叉的概率具有一个扩散的 fac —

在 0 和 $(3+d[3])$ 之间的 tor 与在 $1/(3, +d[3])$ 和 $1/3$ 之间的概率相同。在 $0 < 3 < 1$ 范围内，将所有收缩交叉的概率相加

我们得出结论，总体概率

³收缩交叉与扩展交叉是相同的。等于展开交叉的概率 ($f(3)$) 扩散系数在 $(3$ 和 $(3+3)$ 和收缩交叉 ($C(3)$) 之间，扩散系数在 $1/(3+3)$ 和 $1/3$ 之间，我们得到以下结果：

²注意，在收缩交叉中考虑了两个极端串，扩展交叉是不可能的。

³当然，静止的交叉点可以被看作是收缩的或扩展的。为了便于论证，我们可以假设一半的平稳交叉是收缩的，其余一半的平稳交叉是膨胀的。

$$p_{3+} = \frac{1}{2} \quad (15)$$

重新排列项和在极限 $p \rightarrow 0$ 中，我们得到

$$(16)$$

有了收缩交叉和扩张交叉的发生概率之间的这种关系，我们只需要知道一种交叉的概率分布。由于它们的和是所有交叉的总体概率，所以收缩或扩展交叉的总体概率必须等于 0.5。对于方程 14 中给出的概率分布，我们利用上述方程得到了一个展开交叉的概率分布：

• (17)

这个分布绘制在图 3 中的 $p > 1$ 区域。该图显示，创建具有大 p 的子字符串的概率很小。

回想一下，方程 14 和 17 只是为两个极端的二进制串导出的。为了研究上述其他父对的分布和概率分布，我们创建了所有特定长度的二进制字符串对，并在所有可能的交叉位置进行交叉它们。对于每个交叉，计算相应的扩散因子 p ，并按升序排列。实验继续进行，测试琴弦的长度分别为 10、15 和 30。图 4 显示了 $L=15$ 的分布图。图中显示，出现 1 的概率比其他任何一个 p 值。如果父母的距离更近，这两个可能的孩子的传播范围也会更小。另一方面，如果父母的传播更多，可能的孩子的传播也很大。交叉算子的这一方面将其与突变算子区分开来。在突变操作符中，靠近父字符串的突变字符串的概率更高，但这个概率通常是常数的，并且只依赖于一个字符串。

3 个模拟的二进制交叉 (SBX)

为了直接模拟在实变量上的单点二元交叉的操作，我们设计了一个使用相似的概率分布的交叉算子

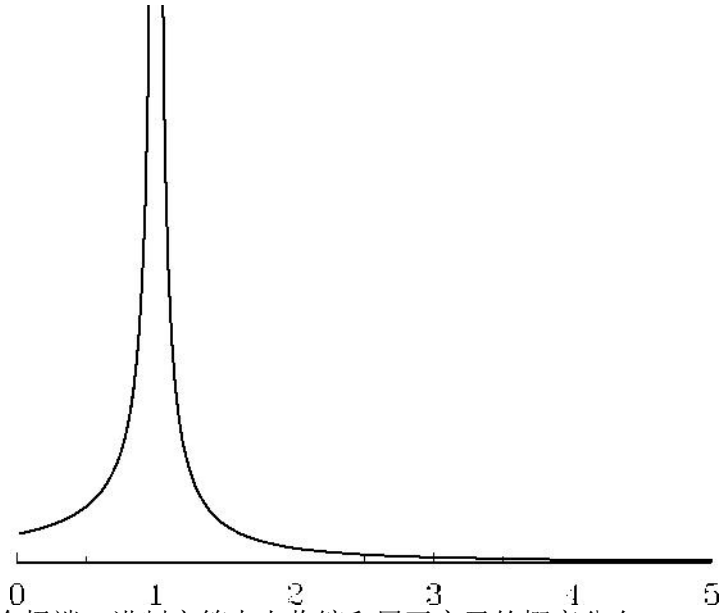


图 3: 显示了在两个极端二进制字符串上收缩和展开交叉的概率分布。

上一节。概率分布的一个性质是，所有可能状态的累积概率等于 1。但是方程 14 中的概率分布对 0 的所有值的积分并不是一个有限的数，这是因为在 $x=1$ 时，该概率不存在。此外，请记住，在方程 14 中得到的分布仅为两个极端二进制串作为父项。对于收缩交叉，我们假设有一个简单的多项式概率分布，它类似于方程 13 中的分布和图 4 中所示的分布：

$$C(0) = 0.5(n + 1)/3^n, \quad (18)$$

其中 n 是任意非负实数。因子 0.5 使在 $x=1$ 时的累积概率等于 0.5。当 n 值大时， n 值可以使方程 14 更好地近似上述分布。 n 值越小，在 $0 < x < 1$ 范围内分布更加均匀。 $n=0$ 的值在该范围内呈均匀分布（图 5）。 $n(2^5)$ 的中等值与二进制编码的 GAs 中单点交叉的模拟结果非常吻合。

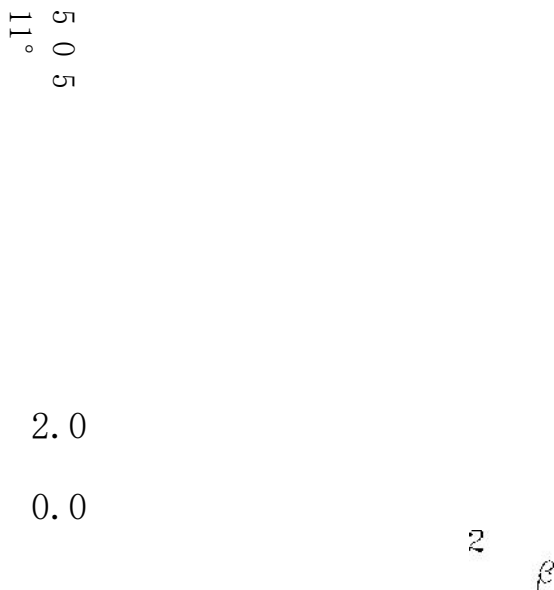


图 4: 显示了在所有长度为 15 的随机二进制字符串对上的收缩和展开交叉的概率分布。

对于展开交叉，概率分布可以通过公式 16 和 18 推导出：

$$= 0.5(n+1)^{-\frac{1}{n+2}} \quad (19)$$

值得注意的是，所有扩展交叉的概率（可以通过计算 $\mathcal{E}(0)$ 找到）的概率为 0.5，等于收缩交叉的概率。从两个方程中得到的平稳交叉的概率也是相同的。

不同值为 n 时的收缩和扩张交叉的概率分布如图 5 所示。很明显，这个分布在很大程度上取决于指数 n 。对于 n 的小值，可能选择远离亲本的点，而对于 n 的大值，可能只选择靠近亲本的点。在某种意义上，指数 n 类似于模拟退火算法中使用的温度参数 (T) 的倒数 (Aarts 和 Korst, 1989)。理想情况下，一个好的搜索算法必须在早期有一个广泛的搜索 (大 T 或小 n)，随着世代的进行，搜索必须集中在一个狭窄的区域 (有小 T 或大 n)，以获得更好的精度。

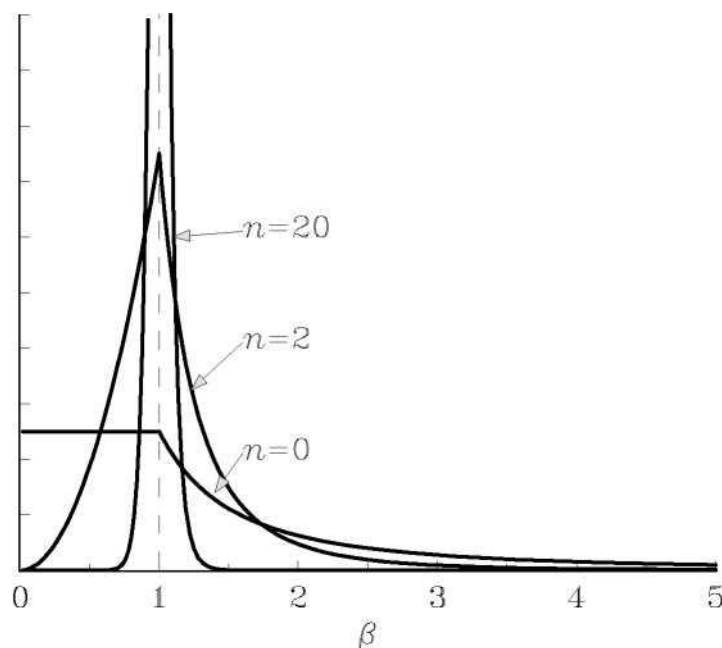


图 5: 所提出的多项式模型的收缩和展开交叉的概率分布显示。

解决办法然而，为了简洁起见，我们在整个模拟中使用了一个固定的妥协 n 。

图 5 显示了 SBX 交叉的搜索实际上可以扩展到真实空间中的任何一个点。如果该变量的下界和上界已知，则上述分布可以被限制在该区域内，我们将在第 6 节中讨论。

值得注意的是，方程 18 中给出的多项式概率分布是收缩交叉的许多可能的概率分布之一。对于不同的分布，可以用公式 16 计算出展开交叉的相应概率分布。值得一提的是，也可以首先假设一个扩展交叉的分布，然后用 C (公式 16 中的 (3)) 替换 $E(J3)$ 来计算相应的收缩交叉的概率分布。在所有这些概率分布中，在父点附近创建子点的概率必须大于创建远离父点的子点的概率（这模拟了如图 4 所示的单点交叉的效果）。然而，在本文的其余部分中，我们使用方程 18 和 19 中给出的多项式概率分布。

下面，我们将讨论如何将上述分析扩展到多变量优化问题中。

3.1 对多变量问题的扩展

使用二进制编码对多个变量进行编码的最常见的方法是使用映射后的连接字符串 (Goldberg, 1989)。在单点交叉中，沿着整个字符串选择一个交叉站点，并且交叉点右侧的内容在父站点之间进行交换。当使用二进制编码的 GAs 来解决 j_v 变量函数时，当交叉位置落在代表从左边开始的第 k 个变量的位上时，单点交叉创建第 k 个变量的新值，并在父变量之间将 $\{k+1\}$ -st 交换到第 n 个变量。同样的操作可以通过使用 SBX 算子对第 k 个变量 (k 在 1 到 J_v 之间随机选择)，并在父变量之间交换 $\{f+\}$ -st 到第 n 个变量来模拟。其他地方已经表明 (Booker, 1993)，单点交叉的这种特征导致某些变量组合从父母传播到孩子的很大的位置偏差。在本研究中，我们选择了一种类似于多变量均匀交叉的机制 (Syswerda, 1989) 来选择要交叉的变量。然而，我们在这里认识到，这一假设与多变量编码的重要连锁问题有关，而我们的机制并不能缓解连锁问题。

为了简化 SBX 算子的实现，我们选择对每个变量进行概率为 0.5 的 SBX 交叉。但在单变量函数的情况下，由于只有一个变量，SBX 执行的概率为 1.0。

交叉概率的概念 (p_c) 仍然适用于这个重新编码的遗传算法。这个概率意味着在一个平均的 p 上，交配池中的 N 个点参与了交叉。

4 个现有的实编码交叉操作符

对真实编码的气体的研究还没有得到遗传算法界的广泛关注。然而，埃谢尔曼和谢弗（1993）对真实编码 GAs 的不同方法进行了很好的描述。Wright（1991）的早期工作建议使用一个线性交叉算子，它只比较三个点 $(p_i + \text{rand}) \cdot (1.5p_i - 0.5p_2)$ 和 $(-0.5p_i + 1.5p_2)$ ，并选择最好的两点。这项工作还提出了一个交叉算子，通过在其范围内随机扰动单个变量来创建一个点。戈德堡在真实编码的 GAs 环境中引入了虚拟字母的概念（戈德堡，1991）。虽然他没有提出任何特定的交叉操作符，但他建议阻塞功能，这可能导致真实编码的气体困难。埃谢尔曼和谢弗（1993）引入了真实编码遗传算法的区间图式的概念，并提出了一个混合交叉 (BLX-a) 算子。对于两个父点， p_i 和 p_2 （假设是 $\langle \text{rand} \rangle$ ）BLX-a 在该范围内随机选取一个点 $(P_i - a(P_2 - P_i), P_2 + (P_2 - P_i))$ 。如果 a 为零，这个交叉会在范围内创建一个随机点 (p_i, p_2) ——在一些测试问题中，他们报告说 BLX-0.5 ($=0.5$) 比使用任何其他值的 BLX 操作符表现更好。这里值得一提的是，区间分割图在概念上与表示搜索空间中连续区域的戈德堡虚拟 alphabets 一 both 相似。

4.1 搜索能力

对于上述真实编码的交叉，我们研究前一节定义的搜索能力。赖特的搜索能力⁵ 线性交叉显然是不够的（太确定性），因为只有三个子点可以从两个父串获得。莱特均匀交叉或埃谢尔曼和谢弗的 BLX-0.0 操作符具有创建由父点包围的任何点的灵活性。正如埃舍尔曼和谢弗（1993）所讨论的，这种交叉保留了在搜索空间中表示连续区域的区间模式。下面，我们将进一步研究 BLX 操作符下的间隔模式处理。

考虑一个范围为 $[0, 8]$ 的变量。根据埃谢尔曼和谢弗的公式，上述变量范围有 45 个区间模式（假设区间模式所表示的点仅以整数为界）。让我们还分别考虑两个父点 $p_i=2$ 和 $p_2=5$ 。第一个点代表总共 $(2+1) \cdot (9-2)$ 或 21 个间隔模式，第二个点代表 $(5+1) \cdot (9-5)$ 或 24 个间隔模式。这些模式如下表所示：

Number of interval schemata					
Parent point at 2			Parent point at 5		
(2,2)			(5,5)		
(1,2)	(2,3)		(4,5)	(5,6)	
(0,2)	(1,3)	(2,4)	(3,5)	(4,6)	(5,7)
(0,3)	(1,4)	(2,5)	(2,5)	(3,6)	(4,7)
(0,4)	(1,5)	(2,6)	(1,5)	(2,6)	(3,7)
(0,5)	(1,6)	(2,7)	(0,5)	(1,6)	(2,7)
(0,6)	(1,7)	(2,8)	(0,6)	(1,7)	(2,8)
(0,7)	(1,8)		(0,7)	(1,8)	
(0,8)			(0,8)		

两个父点之间的公共模式显示在方框中，有 12 个这样的模式。成功的编码-交叉交互的一个潜在特征是，交叉必须将两个参与的父母之间的共同图式（适合于编码）传播给他们的孩子。二进制编码的 GAs 中的单点交叉具有传播荷兰图式的特性。在真实编码的 GAs 中，如果我们假设间隔模式是被处理的，那么就可以假设创建一个子点的概率会与子点和父点之间的共同模式的数量成比例地变化。在某种意义上，这个参数表示更靠近父点的点（在区间模式的上下文中）更有可能被选为子点。有了这个论证，我们现在可以找到两个被选择的父本 $p_1=2$ 和 $p_2=5$ 的一个子点的概率分布。

范围 (2, 5) 中的任何点都包含两个父元素的所有 12 个通用模式。因此，在 (2, 5) 范围内的任何子点都是等可能的。因此，我们可以认为，BLX-0.0 或 Wright 均匀交叉（收缩交叉）将所有共同的区间图式从父母传播给孩子。但是，当使用 BLX-a (a 为正) 时，两个父点中的所有公共间隔模式都不会传播到由父点所包含的范围之外的子点。对于这些交叉（扩展交叉），可以将概率分配到与子点和父点之间的公共间隔图式的数量线性依赖。在上面的例子中，点 $c_i=1$ 将只代表上述公共模式中的 8 个，点 $c_i=0$ 将代表上述公共模式中的 4 个，以此类推。一般来说，任意三点之间的公共间隔模式数是 $\{x_i+1\}$ (JV-册)？其中 x_i 和 x_h 是三个点的最低值，最高值，N 是搜索空间的整个范围。在上述问题中，由于考虑了范围 $[0, 8]$ ， $N=9$ 。对于子点 $x < 2$ ，公共图式的数量为 $(x+1)(9-5)$ 或 $4(x+1)$ 。对于子点 $2 < x < 5$ ，公共模式的数量为 $(2+1)(9-5)$ 或 12（常数），对于子点 $x > 5$ ，公共模式的数量为 $(2+1)(9-x)$ 或 $3(9-x)$ 。这些数字被绘制在图 6 中。如果根据这些数字对创建一个子点的概率进行分配，那么对于由父点包围的点的分布是均匀的，而对于外部点的分布是线性变化的。在扩展交叉的情况下，埃舍尔曼和谢弗（1993）提出的 BLX-0.5 交叉偏离了这一观察结果 (BLX-0.5 算子具有均匀分布，如图 7 所示)。

值得注意的是，当这些概率分布被绘制为扩散因子 0 的函数时（图 7），也出现了类似的概率分布。收缩交叉对于远离父点的父点包围的区域内的任何子点具有均匀的概率分布，对于子点的概率减小。同样有趣的是，上述概率分布是上一节中开发的 SBX 算子的一个特例。对于 $n=0$ ，所有收缩交叉都具有均匀的概率，而扩展交叉的概率随着扩散因子的增加而减小。这两种分布之间唯一的差异是，在 $n=0$ 的 SBX 算子中，概率以非线性方式降低，而在理想的区间模式处理

中，概率线性降低。然而，我们可以认为，具有 $n=0$ 的 SBX 算子比 BLX-0.5 算子更好地尊重区间模式处理。然而，其他具有非零 n 的 SBX 算子不尊重区间模式。以上分析表明，其他 SBX 操作符可能尊重其他一些代表连续点的模式，但具有不同的重要性。由于这种图式将是间隔图式或局部性形式的变体，它们也将满足拉德克里夫的设计标准（拉德克里夫，1991 年）。然而，这些图式的形式化只是一个数学练习，这对目前的研究并不重要。因此，我们避免了这个练习，并给出了 SBX 算子对一些测试问题的仿真结果。

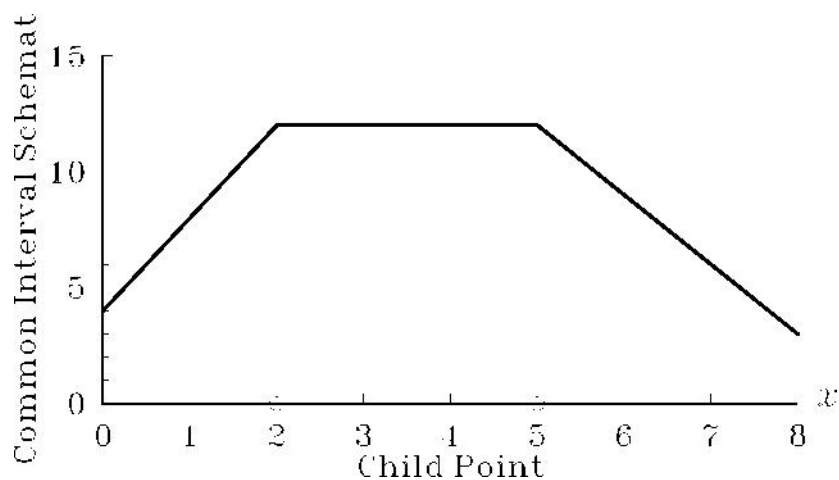


图 6: 显示了子点和父点之间的公共间隔模式的数量。父点分别在第 2 点和第 5 点。

5 模拟结果

为了研究 SBX 算子的有效性，我们选择了一个包含 12 个函数的测试台，其中 9 个取自现有文献。测试函数的范围从单变量函数到多变量函数，从单模态函数到多模态函数，以及从确定性函数到随机函数。最后，加入一个双变量阻塞函数来比较真实编码的气体与 SBX 和二进制编码的气体在一个困难函数上的性能。

利用 C 语言实现了一种实编码的遗传算法。变量向量的初始填充是在用户提供的初始变量范围内创建的。然而，在后代中，并不能保证种群被限制在给定的范围内，因为 SBX 算子创建远离父点的概率是非零（尽管很小）的子点。这在事先不知道真正最优值的位置的问题上是有利的。我们演示了这种灵活性

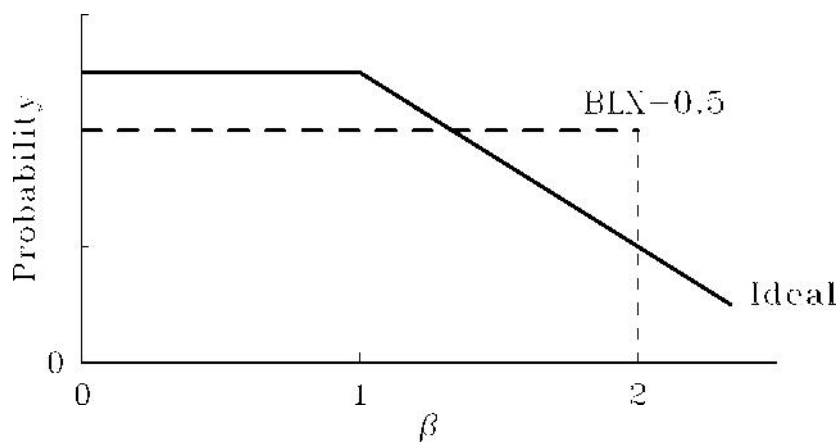


图 7: BLX-0.5 的概率分布和理想分布以扩散因子(3)。

在测试问题中，用 SBX 算子初始化远离真实最优的总体。然而，对于变量的下限和上限是绝对已知的问题，SBX 算子的这个特性是冗余的。我们在第 6 节中讨论了对这个 SBX 算子的一个简单的修改，以将搜索限制在给定的范围内。在我们所有的模拟中，我们允许 SBX 在真实空间中创建任何点，并使用方程 18 和 19 中给出的概率分布。为了研究在 SBX 算子中收缩和扩展交叉的概率分布的影响，我们提出了一些不同的指数值(n)的模拟运行。这里可以记住，对于指数 n 的小值，可以得到远离父点的子点，但对于大 n ，很可能只得到更接近父点的点。突变并不是主要用来单独研究 SBX 操作符的搜索能力的。使用了四个不同的标准来终止搜索过程。如果满足其中任何一个（或多个），则终止模拟运行。相关标准如下：

1. 种群中最好的个体最接近于真正的最优值

所有问题变量中的终止边距为 6。在这种情况下，我们假设遗传算法已经找到了真正的最优优点。

2. 总体中最佳个体的函数值小于或等于一个目标函数值/ y 。在这种情况下，我们假设遗传算法已经收敛到真最优解附近。
3. 种群中的所有个体都收敛到一个小的邻域（在所有问题变量的小边界 6 内）到全局最优优点以外的一个点。在这种情况下，会发生一个过早的收敛。
4. 生成数已超过指定的大数 7^{\wedge} 。在这种情况下，遗传算法根本没有收敛到任何解。

除非明确说明，否则所有模拟使用的目标函数值/ y 等于全局最优优点处的函数值，另一个值等于 200。将使用 SBX 操作符的真实编码 GAs 的性能与 BLX-0.5 和单点交叉进行了比较。在所有的模拟中，使用了没有替换的二进制比赛选择和交叉概率为 1.0。成功解决一个问题（当上述标准函数 1 或 2 得到满足时）所需的平均函数评估数为每个问题制成表格。仿真结果中提到了其他遗传算法参数。

5. 1V 功能

该函数是一个定义在 $0 < x < 1$ 范围内的单变量函数（埃谢尔曼和谢弗，1993）。该函数在 $x=0.5$ 处只有一个最小点：

$$f_1(x) = -0.5|x|. \quad (20)$$

首先，使用二进制编码的气体。字符串长度为 30，总体大小为 50，使用单点交叉操作符的二进制编码气体找到一个小数后 6 位的点 ($c=10^{-6}$)，100 次中只有 23 次。在其余的运行中，GAs 过早地收敛到接近真正最优的点，但不接近最优的 6 以内。在所有 100 次运行中，最佳个体的平均函数值为 0.0015，这与真实最优值（片=0.0）非常接近。值得注意的是，上述=函数的最小点对应于所使用的二进制编码中的汉明悬崖，但使用 $e=10^{-6}$ 大约有 2147 个字符串代表该区域（0.5-6, 0.5+6）。然而，正如其他研究人员所遇到的那样，汉明悬崖的存在是二进制编码的问题之一 (Davis, 1991)。

接下来将应用带有 SBX 操作符的真实编码的气体。模拟结果的汇总表见表 1。终止范围为 $6=10^{-6}$ 已使用。在这个问题中，实际编码的气体比二进制编码的气体的优势很大程度上是由于在最优优点上的汉明悬崖问题。结果还表明，具有 SBX 算子的实编码气体在所有的仿真中都找到了最优优点。当使用更小的终止裕量（达到机器精度）时，也可以观察到 SBX 的类似性能。这证明了真实编码的气体能够克服二进制编码的气体的精度问题。请注意，对于指数 n 的不同值，SBX 操作符的性能保持相同。但是对于一个较大的 n ，需要更多的函数计算来找到最优优点。

BLX-0.5 算子在这个问题上的成功率与 SBX 同样很好。由于 SBX 操作符执行扩展交叉的时间约为一半的时间，并且搜索了比 BLX-0.5 更多的区域，因此在 BLX 操作符的情况下，解决上述问题所需的函数评估较少。但是，当在范围（0.9, 1.0）内初始化一个总体来解决相同的问题时，SBX 的性能远远优于 BLX-0.5 算子。SBX 的成功率与之前相同，但 BLX-0.5 的成功率降低

到 25%左右。回想一下，最小点并不在创建初始总体的范围内。为了找到最优点，搜索操作符必须在此范围之外进行搜索。显然，在这种情况下，二进制编码的最优气体将无法找到该变量的固定映射编码。SBX 的搜索能力通过从该地区被限制的种群开始 (0.9999, 1.0000) 进行了进一步的检验。在本例中使用的终止裕量也是 10^{-6} 。有了 SBX，扩展的交叉在初始范围之外找到更好的点，整个种群向真正的最优迁移。在这种情况下，SBX 的成功率再次为 100%，而 BLX-0.5 的表现较差。

表 1：使用不同交叉算子的 GAs 在 v 函数上的模拟结果被制成表格。第六列中括号中的数字表示 GAs 根本没有收敛到任何一个点的次数。

经营者	范围	Pop 大小	n 或 ϵ	100 次运行的性能		平均。试验
				成功	Prem. 组。	
1-pt	(0, 1)	50	30	23	77	834.8
波段 应答机	(0, 1)	50	0	100	0	929.5
			2	100	0	748.5
			5	100	0	818.5
		100	0	100	0	1739.0
			2	100	0	1396.0
			5	100	0	1321.0
BLX-0.5	(0, 1)	50		100	0	746.0
		100		100	0	1368.0
波段 应答机	(0.9, 1)	50	0	100	0	1279.5
			2	100	0	2018.5
BLX-0.5	(0.9, 1)	50		24	76	2604.2
波段 应答机	(0.9999, 1)	50	0	100	0	1790.0
			2	100	0	4318.0
BLX-0.5	(0.9999, 1)	50		0	61 (39)	

5.2 V-cliff 功能

这个函数与 V 函数相似，除了这个函数在最小点 ($x=0.5$) 处有一个不连续点 (埃谢尔曼和谢弗，

$$\begin{aligned} & 0.6 - \text{如果} \\ & x - x < 0.5; \text{否} \end{aligned} \quad (21)$$

1991) :

公差参数为 $6=10^{-6}$ 再次使用。表 2 显示了具有单点交叉的双链抑制 GAs 和具有 SBX 和 BLX-0.5 的真实编码 GAs 在该功能上的性能。具有单点交叉的二进制编码气体无法找到解决方案

表 2: 使用不同交叉算子的 GAs 在 V-Cliff 函数上的模拟结果被制成表格。第六列中括号中的数字表示 GAs 根本没有收敛到任何一个点的次数。

经营者	范围	Pop 大小	n 或 ϵ	100 次运行的性能		平均。试验
				成功	Prem. 组。	
1-pt	(0, 1)	50	30	16	84	840.6
波段	(0, 1)	50	0	88	0 (12)	2,015.9
			2	100	0	1,382.0
			5	58	42	1,191.4
BLX-0.5	(0, 1)	50		100	0	1,726.0
波段	(0.9999,	50	0	88	0 (12)	2,795.5
			1	100	0	3,253.0
			2	74	2 (24)	7,147.3
BLX-0.5	(0.9999,	50		0	62 (38)	

接近真正的最优在 100% 的模拟。二进制编码的 GAs 的性能比简单的 v 函数差，因为在最优点的汉明悬崖和选择的终止准则 (实际上只能从最优点的一侧发生收敛)。另一方面，带有 SBX 算符的实编码 GAs 在指数 n 的小值时表现良好。n=0 和大 n 的 SBX 性能恶化是由于最优点的不连续。然而，BLX-0.5 的性能为 100%，其功能评价略高于具有 n 个=2 的 SBX。表清楚地表明，当初始化远离最小点的总体时，SBX 优于 BLX-0.5。

5.3 双峰等分布函数

这里使用的双峰函数有两个最小 points — one 比另一个好。每个最小点的吸引盆地与另一个吸引点相同：

$$\begin{aligned} & (-\exp(-(x - 0.25)^2), \text{如果 } x < 0.5; \\ & [-0.5\exp(-(x - 0.75)^2/0.01), \text{否则。} \end{aligned} \quad ()$$

这个函数在 $x=0.75$ 处有一个局部最小值，在 $x=0.25$ 处有一个全局最小值。在局部和全局最小值处的函数值分别为 -0.5 和 -1.0。选择这个函数来测试使用 SBX 的真实编码的 GAs 是否可以找到全局最小点。表 3 比较了两种 GAs 在终止边际为 $6=10$ 时的性能⁴。由于二进制编码的 GAs 的性能问题，低于 SBX 或 BLX-0.5 的真实编码的 GAs。在二进制编码的 GAs 中，需要更大的总体

来实现用更大字符串编码的问题的期望解决精度 (Goldberg, Deb, 和 Clark, 1992)。同样, 同时使用 SBX 和 BLX-0.5 的真实编码 GAs 100% 运行。尽管存在局部最优, 真实编码的 GAs 以期望的精度找到真正的最优点。

表 3: 使用不同交叉算子的 GAs 在函数上的仿真结果被制成表格。

经营者	范围	Pop 大小	n 或 ϵ	50 次运行的性能		平均。试验
				成功	Prem. 组。	
1-pt	(0.0, 1.0)	10	2	4	1	76
		0	0	9	2	5.3
波段 应答机	(0.0, 1.0)	10	0	5	0	87
			2	0	0	0.0
			5	5	0	66
BLX-0.5	(0.0, 1.0)	10		5	0	67
波段 应答机	(0.5, 1.0)	10	0	4	4	450
			2	9	6	0.0
			5	1	4	140
BLX-0.5	(0.5, 1.0)	10		0	5	

这两种真实编码的 GAs 的性能都不适合于不利的初始种群 (种群在当地盆地初始化) 的运行。由于总体已在范围 (0.5, 1.0) 中初始化, 因此全局最小点 ($x^*=0.25$) 不包括在该范围中。因此, 二进制编码的 GAs 将无法找到具有上述初始种群的这个最小点 (固定的、映射编码的问题)。使用 SBX 的真实编码 GAs 的性能略优于使用 BLX-0.5。由于大约 50% 的点是在父系所包围的范围之外创建的, 因此全球盆地中的几个点预计将由 SBX 创建。由于二进制比赛的选择方案和一个 p_c 在上述模拟中使用了 $=1$, 选择压力不足以维持全球盆地中的这些点。当 SBX 的交叉概率降低为 $p_c=0.5$, 好点的中断减少, 在 50 次运行中, SBX 的成功率 (有 2006 个功能评估) 增加到 34 次)。

5.4 双峰、不等分布函数

此函数是对前面的函数的修改。这里，全局最小值的吸引力范围小于局部最小值的盆地：

$$f(x, z) = \begin{cases} \exp(-(x - 0.2)/0.04)^2), & \text{如果 } x < 0.3; \\ -0.5 \exp(-(z - 0.6)/0.4)^2), & \text{否则} \end{cases} \quad (23)$$

表 4 比较了单点交叉、SBX 和 BLX-0.5 在该函数上的性能。与之前的函数相比，该函数中具有单点交叉的二进制编码气体的性能略差，因为全局盆地相对较小。使用 SBX 算子的真实编码 GAs 的性能保持不变，除了具有小指数 n 的 SBX 不能很好地工作。其失败的原因与之前的初始不利功能相同。虽然在全球盆地中产生了一些点，但选择压力并不足以在后代中保持它们。

表 4：使用不同交叉算子的 GAs 在函数上的仿真结果被制成表格。第六列中括号内的数字表示

经营者	范围	Pop 大小	n 或 ℓ	执行成功	50 次的好成绩	平均。试验
1-pt	(0, 1)	10	2	4	5	817.8
波段	(0, 1)	10	0	8	0(42)	450.0
			2	4	6(4)	1117.5
			5	5	0	730.0
			1	5	0	688.0
BLX-0.5	(0, 1)	10		4	45(1)	1570

遗传算法根本不收敛到任何点的次数。

在这个功能中，SBX 和 BLX-0.5 的性能差异很明显。由于局部盆地占据了大部分的搜索空间，所以大部分的初始随机种群是在局部盆地中创建的。在使用 BLX-0.5 进行的 50 个遗传算法模拟中，约有 45 个局限于局部盆地，最终收敛到局部最优解。尽管有些点是在全球盆地中创建的，但由于与 BLX-0.5 操作符相关的大中断率，它们很难在人口中维持下来。另一方面，由于 SBX 有一个中等的 n ，使用 SBX 创建的子点的传播并不大。种群逐渐向全球盆地转移，最终收敛到全球最优点。

5.5 极问题

极点问题是一个有四个最小点的两个变量函数，其中一个是全球最小值。该函数假设在每个极点上都有一个光极

具有一定照明能力的最小点。照明在极处最大，在远离极处减少。当搜索空间中存在多个极点时，混合照明的结果。对于搜索空间中给定的照明水平分布，问题的目标是找到照明最大的点。

$$\text{最大化 } f(x) = g \text{ 时 } \hat{x} \text{ 岩 } \text{ } \quad (24)$$

这个问题可以用数学方法写成如下：

式中表示两极的坐标， C_i 表示第 i 极的光强度， h_i 表示第 i 极的高度。在优化的说法中，参数， (x, g) 表示第 z 个最优点， q_i 表示第 i 个峰处的函数值， h_i 表示第 i 个峰的吸引盆地。在上述问题中，有四个峰值，其参数值如下：

	i			
	1	2	3	4
x	0.	0.	0.	0
y	0.	0.	0.	0
z	1.	1.	1.	1.12
h	0.	0.	0.	0.07

该优化问题在 $(0.8, 0.8)$ 处有一个全局最大点。由于这是一个双变量优化问题，所以在这里使用了第 3.2 节中描述的技术。对每个变量执行 SBX 算子，其概率为 0.5。这导致大约一半的变量（平均）进行交叉操作。我们对所有的多变量函数和 BLX-0.5 操作符都遵循这个策略。表 5 显示了单点、SBX 和 BLX 在终止边缘问题上的性能³。

表 5：使用不同交叉算子的 GAs 在函数上的仿真结果被制成表格。第五列中括号内的数字表示遗传算法根本不收敛到任何点的次数。

经营者	Pop 大小	n 或 ℓ	10 次运行的性能		
			成功	Prem. 组。	平均。试验
1-pt	200	2	6	4	366
		3	9	1	397
		4	8	2	297
波段	200	0	0	9(1)	
		2	8	2	3, 37
		5	9	1	3, 20
BLX-0.5	200		3	7	2, 93

我们观察到，使用 SBX 算符的真实编码 GAs (n 在 2 到 5 之间) 与使用单点交叉的二进制编码 GAs 一样好。然而，使用 BLX-0.5 的真实编码 GAs 的性能不如其他两个 GAs。

表 6：不同交叉算子的 DeJong 五个函数的模拟结果。第 7 列中括号内的数字表示遗传算法根本没有收敛到任何一个点的次数。

功能	公差	经营者	Pop 大小	n 或 ϵ	执行, 50 次运行		平均。试验
					成功	过早组。	
函数 I	10 ⁻³	1-pt	100	3	2	2	2532
		波段	100	0	5	0	2556
				4	5	0	2190
				1	2	2	2420
功能 II	10 ⁻³	1-pt	100	24-40	0	5	
		波段		0-20	0	5	
功能故障	1.0	1-pt	100	2	5	0	560
				5	5	0	720
		波段		0	5	0	1256
				5	5	0	722
功能 IV	0.16	1-pt	100	240	4	42 (15200
		波段		0	0	0 (5	
				2	5	0	17032
				5	5	0	9372
V 函数	0.1	1-pt	100	2	4	2	756
				4	4	1	836
		波段		0	7	2 (4	5014
				5	3	1	1097
				1	3	1	891

从上述模拟中，我们观察到 BLX-0.5 算子只在具有连续搜索空间的表现良好的单峰函数上表现良好，但在具有多个最优点的问题上表现不佳。此外，本文的重点是比较真实编码气体中使用的 SBX 算子和二进制编码气体中使用的单点交叉算子。因此，在随后的模拟中，我们只比较单点和 SBX 算子。

5.6 德容函数

接下来将尝试所有五个德容的测试函数 (DeJong, 1975)。单点算法和 SBX 算子算法的仿真结果如表 6 所示。

函数 I 是一个三变量单峰函数，在 (0, 0, 0) 处有一个最小点^T。在 DeJong 的研究中，该函数被定义为 $i=1, 2$ 和 3 的范围为 $-5.12 < x_i < 5.12$ 。在具有 SBX 操作符的实际编码 GAs 中，也可以创建此范围之外的点。因此，我们假设该函数也被定义在上述范围之外，但初始总体是在上述范围内创建的。仿真结果（见表 6）表明，在 n 值较小的情况下，采用 SBX 的 GAs 都能找到最小点后两位的最小点。具有单点交叉的二进制编码 GAs 仅在 50% 的模拟中找到最小点。SBX 的结果是预期的，因为该函数在真实搜索空间中是单峰的和连续的。然而，在二进制编码的 GAs 中，最小点位于汉明悬崖中，这导致在 100% 的模拟中难以解决这个问题。

函数 II 是一个具有全局最小值的二维最小化问题^T。该函数最初定义在范围

$-2.048 < X^2 < 2.048$ 为二进制编码的 GAs (DeJong, 1975)。对于真实编码的 GAs, 将在上述区域创建初始种群。表 6 中的结果显示, 两种 GAs 都不能找到最优的点 $(1, 1)$ ¹。在这个函数中, 向最优点的斜率非常小, 种群收敛到沿山脊的一个点。当 SBX 算子沿着连接两个亲本的线执行时 (类似于赖特线交叉 (Wright, 1991), 但用多项式概率分布选择了一个点), 对于 $n=1$ 到 3, 观察到接近 100% 的收敛。

函数 $-i11$ 是一个五维的阶梯式函数, 其中函数值随着变量的递增而单调不递减。最初的功能是设计为 $-5.12 < X_i < 5.12$ 为 $z=1$ 到 5。由于 SBX 操作符可能在这个范围之外创建子点, 所以我们通过取在 $0 <$ 范围内创建初始总体的函数的绝对值来修改原始函数 < 10.24 。这就导致了一个最小点

习=0。为了使比较公平, 二进制系统也在这个修改范围为 $0 <$ 的函数上运行 GAs < 10.24 。这两种 GAs 在这个功能上表现得同样好。很有趣

需要注意的是, 尽管函数在搜索空间的许多点是不连续的, 使用 SBX 的真实编码 GAs 能够找到真正的最优。这是因为函数向最优点单调地不递减, 而且因为 GAs 的种群方法不允许它们被困在任何平台上。

函数 $-iv$ 是一个具有零均值高斯噪声的 30 维单峰函数。确定性函数的最优点为 $=0$ ($z=1$ 到 30), 其函数值等于零。但由于高斯噪声的加入, 最优点和最优函数值发生了变化。为了确定收敛到最优, 我们假设一个目标函数值为 $/y=-3.0$ 。如果任何一个点的函数值小于 $/y$, 则终止模拟。所使用的终止裕度为 $6=0.1$ 。对于 $n=2$ 到 5, 使用 SBX 操作符的真实编码的 GAs 比二进制编码的 GAs 表现更好, 这可能是因为二进制气体的模拟中使用的种群大小是不够的。

函数 $-v$ 是一个 25 维的多模态函数, 有 25 个最小点, 其中只有一个是全局最小值。全局最小点的吸引范围非常窄; 因此, 经典的优化方法可能难以解决全局最优性的问题。仿真结果表明, 二进制编码的单点交叉算法在该函数上具有优越性。在这个问题中, 二进制编码气体的成功部分是由于在等间隔内最优点的有偏定位。当随机放置 25 个最优位置时, 二进制编码的 GAs 和真实编码的 GAs 的性能差异是 small — binary 编码的 GAs 仅在 64% 的模拟中成功, 而使用 SBX 的真实编码的 GAs 在模拟中有 54% 的成功。

从这些模拟中, 我们得出结论, 具有 SBX 的真实编码的气体在具有连续搜索空间的德容函数上比二进制编码的气体表现更好, 并且在离散的德容测试函数上与二进制编码的气体表现相似。为了进一步支持我们的结论, 我们将 SBX 算子应用于另外两个函数。

5.7 拉斯特里金[^]函数

最近被一些[^]研究人员使用 (穆伦拜因、施米施和出生, 1991 年; 戈登和惠特利, 1993)。该函数有 20 个变量, 并包含 10^{20} 局部最小点, 其中只有一个是全局最小点:

$$= 200 + f \text{ 或 } -10 \cos(2\pi t^{\wedge}), -5.12 <^{\wedge} < 5.12. \quad (25)$$

$i=1$

对于 $a1H=1, 2, \dots, 20$, 全局最小值点为 $=0$ 。此时, 函数值为零。当余弦项为局部极小值时, 另一个局部极小值出现。这个函数是一个困难的优化问题, 因为有 2 个²⁰ 不同的局部最优点 (围

绕着全局最优点），其函数值非常接近于全局最优点。我们设置了 $\delta=0.1$ 和目标函数值 $y=1.0$ 。当 20 个变量中至少有 19 个变量收敛到它们的全局最小值时，这就允许终止模拟。

首先应用所有变量子串长度的二进制编码 GAs。在所有的 10 个模拟中，二进制编码的 GAs 过早地收敛到一个错误的解决方案（表 7）。断程序

表 7：采用不同交叉算子的 GAs 对函数 f_u 的模拟结果。第五列中括号内的数字表示遗传算法根本不收敛到任何点的次数。

经营者	Pop 大小	n 或 ℓ	10 次运行的性能		平均。试验
			成功	Prem. 组。	
二进制	400	20	0	10	
		30	0	10	
波段应答	400	0	0	0 (10)	
		2	1	0	
		5	6	4	
		1	1	0	
		2	1	0	

在所有 10 个模拟中，最佳解的年龄函数值均为 11.32。使用 SBX 操作符的真实编码的 GAs 比二进制编码的 GAs 性能更好。在所有模拟中， $n=2$ 的平均函数值为 6.81， $n=10$ 的平均函数值为 3.77，也优于二进制编码的 GAs。实编码气体在这个问题中的成功是由于在 SBX 算子下的子点的控制分配。对于中等 n ，子点离父点不远。因此，搜索缓慢地向最佳点进行。这里值得一提的是，由于 20 个变量中每个变量的公差为 0.1，枚举搜索方法需要大约 $[(5.12 - (-5.12)) / 0.1]^{20}$ 或 1.48×10^{40} 函数评估。真实编码的气体只需要这个搜索空间的一小部分来定位一个接近全局的最小点。这些结果表明，利用 SBX 算子在求解多变量、多模态函数方面的有效性。

5.8 双变量阻塞函数

根据戈德堡¹（1991）的建议，设计了一个双变量阻塞函数。函数的平均值（超过 $a; i$ ）和全局切片（在研）绘制在图 8 中。全球范围内的

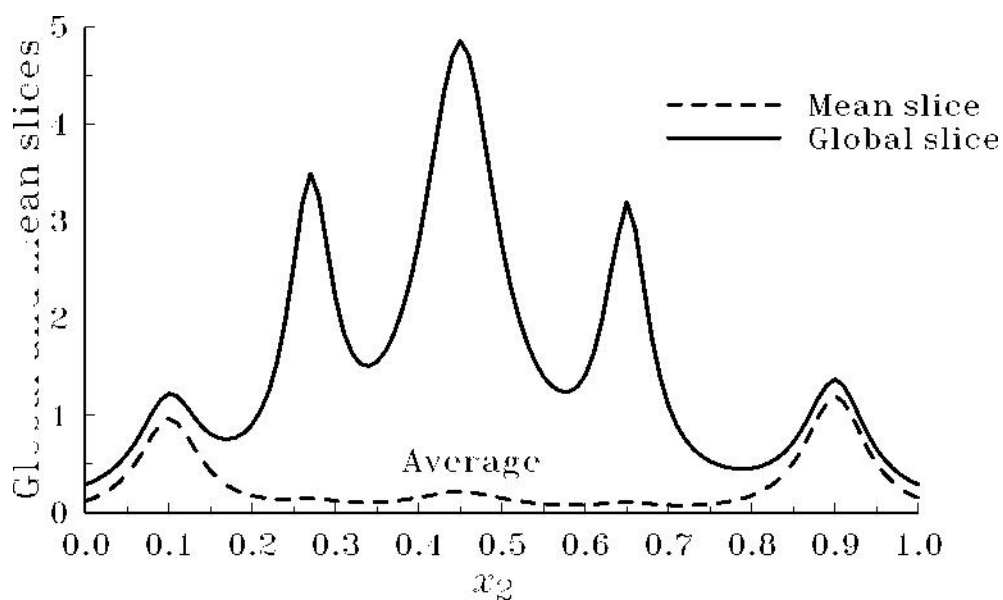


图 8: 显示了函数/12 的平均切片和全局切片（在夔处）。搜索空间上的平均函数值为 0.287。

1.1
7

最大点位于该点 $(0.4, 0.45)$ 处[†]. 两个当地的山丘位于全球最优状态的两侧。在全球切片（取自 $x \setminus = 0.4$ ）中，全局最大点被两个局部最大点包围，但平均切片（在 $x \setminus$ 值上的平均值）为

表 8：将不同交叉算子在函数/12 上的模拟结果制成表格。括号中的数量在第五列中表示气体收敛到全球盆地的次数，但没有收敛到全球最优点的次数。

50 次运行的性能					
经营者	Pop 大	n 或 ℓ	成功	Prem。组。	平均。试验
二进制	100	2	5	45 (2)	1,480
		3	1	35 (2)	1,753
		5	6	44 (3)	1,616
波段应答	100	0	4	8 (0)	2,323
		2	4	3 (0)	1,872
		5	4	10 (1)	1,677

主要是当地的山峰。有趣的是，沿着变量 x_i 的搜索是单峰的。因此，预计人口将收敛于 $x_i=0.4$ 。一旦大多数种群聚集在 $x_i=0.4$ 上或附近，搜索就会沿着全局切片进行。由于总体收敛到极端的局部峰值，经典算法很难找到全局点。其功能如下：

$$f_{12}(x_1,x_2)=-((x_1-0.4)^2+\frac{1}{2}(x_2-0.4)^2+\frac{1}{2}(x_1-x_2)^2) \tag{26}$$

这里，下面的参数	rs	习惯于				
	是	i				
		1	2	3	4	5
		0.0	0.0	0.0	0.0	0.0028
每		0.0	0.0	0.0	0.0	0.0010
象		0.1	0.	0.45	0.2	0.65
r		0	0	10	1	10

使用二进制编码和真实编码的气体的模拟结果如表 8 所示。虽然具有单点交叉的二进制编码 GAs 只收敛到全球最优点（精度为精度后三位 5 到 15 次），但在大约 80%的模拟中，它们收敛到全球盆地。使用 SBX 的真实编码的 GAs 也观察到类似的结果；在至少 80%的运行中，它们已经收敛到全局最大点。在所有成功的真实编码的 GAs 模拟中，可以观察到变量 x_i 首先收敛到真正的最优值，正如 Goldberg（1991）所预测的那样。随后，沿着全局切片继续搜索，位于两个极端最优处的种群逐渐改善，可以克服局部峰值的障碍，最终收敛到全局最优。当真实编码的 GAs 未能找到全局最优点时，就会首先收敛到变量。当这种情况发生时，种群就会在图 8 中平均切片图（虚线）上显示的任何两个峰值中发生变化。由于 x_i 上的函数是单峰的，最大值在 $x_i=0.4$ ，所以该解收敛到一个错误的解。

需要进一步的实验得出结论提出的算法将如何在高维阻塞函数，但仿真结果在上述双变量阻塞函数表明，尽管 SBX 操作员工作在实际参数值，其搜索能力解决函数类似于单点交叉。

6 个扩展

上述研究为进一步的研究开辟了许多途径。下面将讨论其中的一些问题：

1. 有界搜索空间,
2. 为收缩和扩展交叉点选择其他形式的概率分布,
3. 真实编码突变算子,
4. SBX 在实际优化问题中的应用,
5. 模拟真实编码气体的其他二进制交叉器,
6. 人口规模和收敛性证明,
7. 使用真实编码的气体进行多模态和多目标优化。

在本研究中, 我们允许儿童点位于负无穷到正无穷之间的任意地方。在某些问题中, 搜索空间可能是有界的。例如, 如果一个工程构件设计中的一个成员的长度是一个变量, 则该变量的负值就没有意义了。本研究使用的 SBX 算子并不总是保证孩子点为正, 即使父母双方都可能为正值。可以修改收缩和扩展的概率分布, 使在给定界外创建点的概率为零, 同时保持概率分布的相同性质。实现这一点的一种方法是, 对于任意两个给定的父点, 首先计算在给定边界内的子点的累积概率。如果这个概率是 a (比如说), 那么概率分布函数乘以一个因子 $1/a$, 因此在边界内的点的总体概率等于 1。

本文所使用的收缩交叉的概率分布是对扩散因子的多项式函数。此分布用于减少每个交叉操作中所需的计算量。也可以使用概率分布的其他函数形式, 但必须注意选择一个与图 4 中给出的形式相似的分布。在这方面, 均值的高斯分布为 1, 方差 (a^2) 选择在指定边界内最多 $\pm 3a$ 点也可以使用。

在本文中, 我们单独研究了交叉的影响, 保持突变概率为零。为了保持种群的多样性, 可以使用突变算符。在真实编码的 GAs 中, 突变操作可以通过选择当前点附近的一个相邻点来实现。这可以通过确定显式概率分布或采用其他基于人群的真实编码研究中使用的方法来实现 (Back, 霍夫迈斯特和 Schwefel, 1991)。

本研究表明, 对于具有连续搜索空间的问题和需要任意精度的问题, 真实编码的气体比二进制编码的气体更好。许多工程设计优化问题包含混合 variables — both 离散和连续。经典的优化方法在解决这类问题 (称为混合整数规划问题) 方面表现不佳。结合二进制编码 (对于离散变量) 和真实编码 (对于连续变量) 的 GAs 可以有效地解决这些问题。交叉操作可执行如下操作。首先选择一个变量来进行交叉。如果需要交叉一个离散变量, 则单点交叉或另一个二进制运算符只能用于表示该变量的位。但是如果需要交叉连续变量, 可以使用 SBX 运算符。这样, 生成的字符串也会成为一个有效的字符串。我们目前正在将该技术应用于桁架结构优化问题, 在桁架结构优化问题中, 同时实现了桁架拓扑 (由二元变量表示) 和构件的横截面 (连续变量) 的优化, 这是一种理想的方法, 但不能用经典的优化方法来实现。

这里开发的 SBX 算子是基于单点交叉中子串的概率分布。对于多点和均匀交叉, 也可以得到类似的概率分布 (扩散因子 0 的函数), 并可以设计其他类似的实编码交叉。概率分布类似于这里开发的分布是预期的, 调查它们是否会如此可能是有趣的。根据分析的结果, 一个交叉优于其

他交叉的问题可以得到解决，特别是在遗传算法应用于连续变量的背景下。

在本研究中，使用了合理的种群规模值。正如在其他地方所证明的那样（戈德堡、德布和克拉克，1992 年），适当的人口规模对于气体的成功工作很重要。该研究还提出了一种基于可检测函数值的差异与函数方差的比值的总体规模。我们可以尝试从类似的观点推导出真实编码气体的种群规模。在二进制编码的气体中，实现收敛证明的主要困难在于交叉算子的力学。虽然单点交叉的力学（或就此问题的任何其他二进制交叉算子）是简单的，但它们的数学建模变得困难。随机过程的任何收敛性证明都涉及到转移概率的计算。由于在 SBX 算子中，使用了任何给定父点的子点的直接概率分布，遗传算法的收敛证明可能变得更容易。

二元气体在许多单目标、单峰问题上的成功，使研究人员将遗传算法的适用性扩展到其他类型的优化 problems — multimodal 和多目标函数优化问题。为了同时找到多个最优解，使用共享函数创建人工适应度值，这些值用于繁殖算子中 (Deb, 1989; 戈德堡和理查森, 1987)。为了找到多目标函数优化的多个帕累托优化点，我们使用非支配的概念来创建人工适应度值，该值再次用于再现算子 (Srinivas 和 Deb, 在出版中)。由于只需要对复制算子进行修改才能解决上述问题，因此可以使用类似的复制技术和本研究中使用的 SBX 算子来研究实编码气体在定义在连续搜索空间上的多模态和多目标问题中的有效性。

7 结论

本文基于二进制编码气体中单点交叉的搜索特性，开发了一种实编码交叉算子。为了定义交叉算子的搜索能力，引入了一个扩展因子，即子点的绝对差与父点的绝对差的比值。然后，为两个给定的父点为单点交叉创建一个子点的概率。由于二进制编码 GAs 在离散搜索空间问题中的成功，开发了模拟二进制交叉 (SBX) 算子来解决具有连续搜索空间的问题。SBX 操作符具有与单点交叉相似的搜索能力。

在许多测试函数包括德容的五个测试函数，已经发现真实编码气体 SBX 运营商可以克服固有的困难在解决二进制编码气体连续搜索空间 problems — Hamming 悬崖问题，任意精度问题，和固定映射编码问题。在比较真实编码的气体与 SBX 操作符和二进制编码的气体与单点交叉操作符，已经观察到，前者的性能比后者在连续函数和前者的性能类似于后者在解决离散和困难的功能。与其他地方提出的另一个真实编码的交叉算符 (BLX-0.5) 相比，SBX 在困难的测试函数中表现得更好。人们还观察到，SBX 在最优点的边界不知道和存在多个最优，其中一个是全球的问题中特别有用。

使用 SBX 操作符的真实编码的 GAs 也被尝试来求解一个双变量阻塞函数（阻塞函数的概念是由戈德堡 (1991) 引入的）。阻塞函数对于实际编码的 GAs 来说，是困难的，因为局部最优点阻塞搜索过程，继续向全局最优点前进。对双变量阻塞函数的模拟结果表明，在大多数情况下，搜索过程按照戈德堡预测的那样进行。最重要的是，我们观察到具有 SBX 的真实编码 GAs 在克服局部峰的障碍并收敛于全球盆地方面与具有单点交叉的二进制编码 GAs 相似。然而，要断定具有 SBX 算符的实编码 GAs 是否能够克服高维阻塞函数中的局部障碍还为时过早。

这些结果令人鼓舞，并为进一步的研究提供了途径。因为 SBX 算子使用一个概率分布来选择

一个子点，真实编码的气体与

SBX 在实现气体的收敛性证明方面，比二进制编码的气体领先一步。利用本文中使用的儿童和父母点之间的直接概率关系，可以借鉴经典随机优化方法的线索来实现 GAs 的收敛性证明，或者经典优化方法与遗传算法之间更紧密的联系即将到来。

确认

这项工作由新德里科学技术部资助，拨款范围为 SR/SY/E-06/93。

参考文献

A 艺术, E. 和科斯特, J. (1989). 模拟退火和玻尔兹曼机: 一种组合优化和神经计算的随机方法。奇切斯特: 威利。

回来, 霍夫迈斯特和施韦费尔, 惠普。(1991)。对进化策略的调查。在 R. K. Belew 和 L. B. Booker (电子版) 中, 第四届遗传算法国际会议论文集。2-9)。

布克公司 (1993)。气体的重组分布 D. Whitley (Ed.), 《遗传算法的基础》(第 29-44 页)。

戴维斯出版社 (1991 年)。爬攀爬、表征偏差和测试套件设计。在 R. K. Belew, & L. B. Booker (修订版) 中, 第四届遗传算法国际会议论文集。18-23)。

德容, k. (1975). 一类遗传适应系统的行为分析 (博士论文, 密歇根大学, 安娜堡)。论文文摘国际, 56 (10), 5140B。(大学缩微胶片公司 No. 76-9381)

Deb, K. (1989). 多模态函数优化中的遗传算法, Master 论文 (TCGA Report No. 89002)。塔斯卡卢萨: 阿拉巴马大学。

埃谢尔曼, l. j. 和谢弗, j. d. (1993)。真实编码的遗传算法和区间图式。在 D. Whitley (版) 中, 遗传算法的基础 II (pp. 187-202)。

戈德堡, 法学博士 (1989)。搜索中的遗传算法, 优化和机器学习。阅读: 艾迪生-韦斯利。

戈德堡, 法学博士 (1991)。真实编码的遗传算法, 虚拟字母表, 和阻塞。复杂系统, 5 (2), 139-168。(也是 ILLI GAL 报告 90001)

戈德堡博士, 德布博士, 和克拉克博士, J. H. (1992)。遗传算法、噪声和种群规模。复杂系统 6, 333-362。

戈德堡, d. e., Deb. K., Kargupta, H, 和 Harik, G. (1993)。使用混乱的遗传算法对难题进行快速、准确的优化。在 S. Forrest (修订版) 中, 第五届遗传算法国际会议论文集。56-64)。

犹太老板 D. E., Korb, B., 和 Deb, K. (1989)。混乱的遗传算法: 动机, 分析, 和第一个结

果, 复杂系统⁵, 93-530。

戈德堡博士和理查森, J. (1987). 多模态函数优化的遗传算法。在 J. J. Grefenstette (电子版) 中, 第二次遗传算法国际会议论文集。41-49)。

戈登案和惠特利博士 (1993)。串行和并行遗传算法作为函数优化算法。在 S. Forrest (修订版) 中, 第五届遗传算法国际会议论文集。177-183)。

卡格普塔, 德布和戈德堡, d 博士 (1992)。命令遗传算法和欺骗。在 R. Manner 和 B. Manderick (eds.) 中, 《自然》的并行问题解决 (第 47-56 页)。

穆伦拜因, H., 施米施, M., 博恩, J. (1991)。并行遗传算法作为函数优化器。在 R. K. Belew 和 L. B. Booker (Eds.) 中。第四届遗传算法国际会议论文集。271-278)。

雷德克里夫, 新泽西州。 (1991)。形式分析和随机尊重重组。在 R. K. Belew 和 L. B. Booker (Eds.) 中。第四届遗传算法国际会议论文集。222-229)。

雷德克里夫, 新泽西州。 (1993)。基因集合重组。在 D. Whitley (d 版) 遗传算法的基础 I 页。203-219)。

雷臣伯格, 我。 (1973)。进化策略: 选择生物进化的技术系统和生物进化的原理。斯图加特。

斯里尼瓦斯, N. 和德布, K. (出版)。多目标函数优化使用非支配排序遗传算法, 进化计算^{MIT}出版社。

Syswerda, G. (1989)。遗传算法中的均匀交叉。在 J. D. Schaffer (Ed.) 一案中, 第三次遗传算法国际会议论文集。2-9)。

赖特, A. (1991)。真实参数优化的遗传算法。在 G. J. E. Rawlins (电子版) 中, 遗传算法的基础。205-220)。