文章编号： 1003-7721(2003)02-0067-10

刘 健 收稿日期:2003-01-14

算法作曲及分层结构控制

内容提要：文章从关于算法作曲的基本概念及其可能的历史母体的论述开始，引出算法作曲中最重要的“分层结构控制”概念，并通过实例分析，归纳分层结构控制的基本特性。

关键词：算法作曲；分层结构；计算机音乐

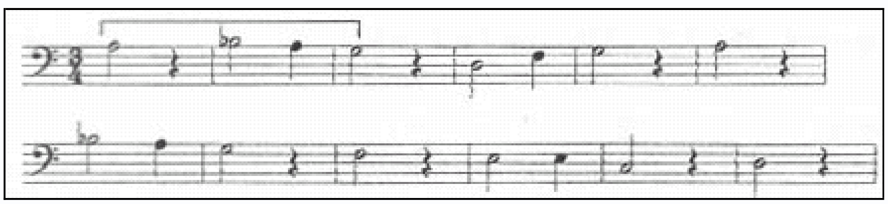
中图分类号：J614　　文章标识码：A

引　言

算法作曲(Algorithmic Composition)，又称为自动作曲(Automated Composition)，是指用某种逻辑过程来控制音乐的生成。人在算法作曲中扮演着“立法者”的角色，完成逻辑过程的设定后，尽可能少地干预音乐的生长，让计算机来控制完成音乐作品。当然，该过程的规则是由人来制定的，由此而生成的音乐作品的艺术趣味，取决于作曲家在创造“音乐模块”与制定规则时的想象力和控制力。

如果我们宽泛地认定算法作曲是某种模式化作曲概念在计算机上的延伸，那么，在计算机诞生以前，模式化作曲就已经是常用的音乐创作方法之一。早在14世纪，经文歌中男高音声部常使用的等节奏(isorhythmic)就具有模式化作曲的意义。见下例：

例1



前三小节构成的节奏模式始终贯穿旋律之中。3到5小节是节奏模型的重复，4到6小节是节奏模式的逆行，6到8小节是原型，9到11小节也是原型。

此外，我们熟悉的许多结构形式如卡农、赋格，和许多发展手法如倒影、逆行、模进等，也具有算法作曲的特征。

算法作曲的特征还体现在随机音乐的某些实现方式中。作曲家完成不同的音乐模块后，由演奏家掷骰子决定模块的演奏次序。最著名的例子是莫扎特的圆舞曲《音乐骰子游戏》，他创作了176小节音乐，然后将小节号排列为两个特别的矩阵图，用掷骰子的方式来决定演奏的次序，见下表：

	I	II	III	IV	V	VI	VII	VIII
2	93	22	141	41	105	122	11	30
3	32	6	128	63	146	46	134	81
4	69	95	158	13	153	55	110	24
5	40	17	113	85	161	2	159	100
6	148	74	163	45	80	97	36	107
7	104	157	27	167	154	68	118	91
8	152	60	171	53	99	133	21	127
9	119	84	114	50	140	86	169	94
10	98	142	42	156	75	129	62	123
11	3	87	165	61	135	47	147	33
12	54	130	10	103	28	37	106	5

	I	II	III	IV	V	VI	VII	VIII
2	70	121	26	9	112	49	109	14
3	117	39	126	56	174	18	116	83
4	66	139	15	132	73	58	145	79
5	90	176	7	34	67	160	52	170
6	25	143	64	125	76	136	1	93
7	138	71	150	29	101	162	23	151
8	16	155	57	175	43	168	89	172
9	120	88	48	166	51	115	72	111
10	65	77	19	82	137	38	149	8
11	102	4	31	164	144	59	173	78
12	35	20	108	92	12	124	44	131

约翰·凯奇在作品《Reunion》中，将演奏的选择交给两位下象棋的选手，他们在具有图像感应功能的棋盘上每走一步，都会触发不同的声音，这样，每盘棋都是这些声音材料的一次新的重组。此时，音乐生成的逻辑取决于棋盘上两军对抗的风云变幻。

在音乐思维中，模式化的过程是自然存在的。例如，在听音乐的时候，我们一方面沉浸在音响带给我们的新奇刺激之中，而另一方面，我们又不断地产生对音响的期待，构架对音乐过程的假设。期待与假设的基础就是所有储存在听众脑海中的已有模式。作曲家们早就了解，音乐过程中的某些要素是可以用符号系统的“逻辑表达式”来描述的。

虽然，当我们提及算法作曲时，总是和计算机音乐相联系。实际上，使用常规的作曲手段也能运用算法作曲的观念进行创作。不过，在计算机上运用这一方法更为简便、快捷。当我们将一定的算法输入计算机，它就能以此为依据，不断生成音乐数据，完成作品的创作。

一

近半个世纪以来，在计算机上实现算法作曲的途径非常多。但从根本上来看，所有的方法都能归于两种基本类型：1.用特定的算法控制声音产生与变化的过程，这种方法直接产生音响结果。2.用特定的算法控制音符的产生与变化，这种方法生成乐谱，然后利用MIDI设备或演奏家的演奏来获得音响结果。

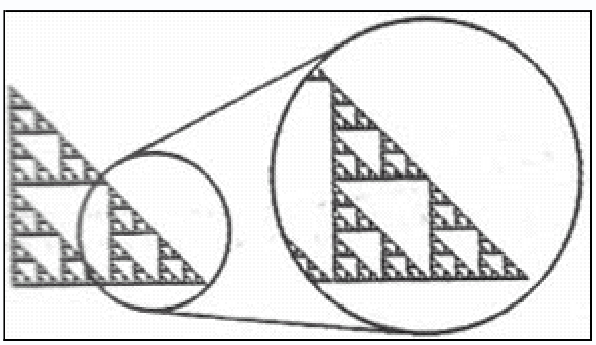
算法作曲所要面对的主要问题和传统作曲方式并无区别，都是要考虑如何构成音乐的核心材料，以及如何将核心材料变形发展。而此类问题的表达方式又各不一样：1.以多种音乐基本要素的综合形态为核心，首先构成短小的动机或主题，然后加以发展。2.以乐音关系为核心，用特定的音列或音集(包括特定的音程特性)作为核心材料(不一定构成性格鲜明的动机或主题)，控制乐音的运动。3.以发展逻辑为核心，任何局部都不具有核心的意义，只有通过音乐的整体发展过程才能揭示出来。第一和第二种方式在传统创作过程中常见，而在计算机音乐中，则第二和第三种方式更容易被计算机所“理解”。

在计算机上实现算法作曲，构成核心材料并不困难。第一，我们可以利用MIDI键盘，直接输入所需要的乐音数据。第二，我们也可以用随机数发生器任意生成数据，然后根据一定的模式，选择所需要的数据，构成核心音集、音序、音列或节奏模式。问题在于，我们如何让计算机控制音乐发展的过程。

构成多种多样的变化形态，是计算机的特长。在这方面，人都要自愧不如。在一定的范围内，计算机几乎可以穷尽所有可能的变化形态。如何恰当地选择变化形态，构成有意味、有审美价值的音乐过程，却是计算机的一大难题。虽然选择的困难对于任何从事音乐创作的人来说，也同样如影相随，难以摆脱。可人总能在一个模糊的范围内，把问题交给“感觉”去解决。计算机没有“感觉”，人所可能运用的某些控制因素，如情绪的、美学的、情节的等，都不可能作为计算机构建音乐结构的基础。它所能遵循的就是严密的逻辑与规律。分层结构就是算法作曲中常常遵循的一个结构原则。

二

分层结构又被称为“无限细节结构”。从图形来看，如果将分层结构图无限放大，我们可以不断看到新的分层结构图。一般来说，新一层的图形和上一层的图形相似，这就是所谓的整体与局部的自相似性。



上图就是分层结构图，右图是左图的某一局部的放大。如果将图无限放大，还可以看到新的细节。而普通的几何图形不具备这样的属性，不管如何放大，都不会出现新的细节。

有一种说法：“分层结构无处不在”。其实，在现实世界中，真正的分层结构并不多见。但是，的确有不少的物体在一定的层次内体现了分层结构原理。如树叶、晶体及其它由颗粒状物质构成的物体。

分层结构并不仅仅在视觉图形中存在，声音也同样具有这样的属性。根据傅立叶(Fourier)合成理论，所有复杂波

形都是由相似的简单波形叠加调制而成；或者可以表述为，声音是谐波的总和。在此理论基础上，任何复杂的声音也是由多层的细节构成(虽然不是无限的)，这些细节(谐波)也有着自相似性。最具有分层结构特征的声音被称为“比例噪声”(scaling noise)，典型代表是白噪声。正因为白噪声几乎覆盖了人的听域的所有频带，它也就具有最多的细节层次。由此可知，如果是单一频率构成的波形，就没有可分的层次，也就不具备分层结构的特性。

分层结构的特性还可以从空间域延伸到时间域，对于不同时间呈现的事件而言，任何事件都可以看作是前一事件的逻辑结果，这是概率论中马尔可夫理论的核心，也正是音乐艺术的一种特性。在时间域中，细节的呈现不是通过图形的放大，而是通过时间的延续。如果以音高参数为例，任何一个乐音的出现，都和其余所有的乐音一样，依赖于一个相同的比例关系。

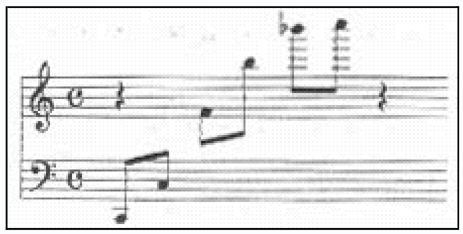
分层结构在形态上可能会显得非常复杂，如噪声的波形。但是，正因为它总是由简单细节所构成，构成分层结构的数学原理却非常简单。不同层次的细节可以依靠简单的比例关系来维系相互间的联系。用相同原则的递归来控制音乐要素是分层结构控制的主要特征，最常用的数学表达式是迭代函数：

从以上表达式可以看出，新的X(Xn+1)值是前一X(Xn)乘以一定的比例数的结果。假如我们规定X =0.1，a=2，然后不断将新的X值带入表达式，我们可以得到这样一系列数值：

X1=0.1，X2=0.18，X3=0.295，X4=0.416，X5=0.486，X6=0.5，X7=0.5

我们可以发现，这是个逐渐增加的数列，最后停在0.5上不再变化。我们称0.5为收敛点。它体现了分层结构控制具有的收敛性。

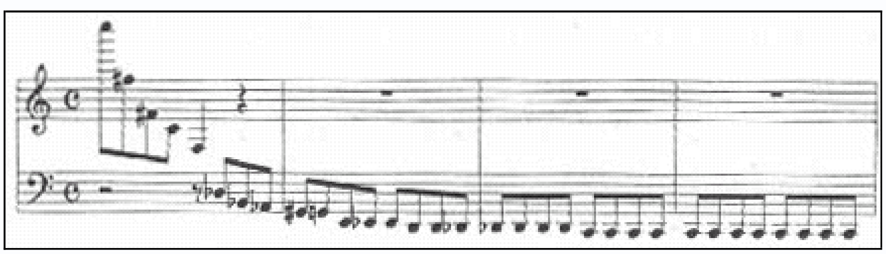
将它们转化为相应的音高：



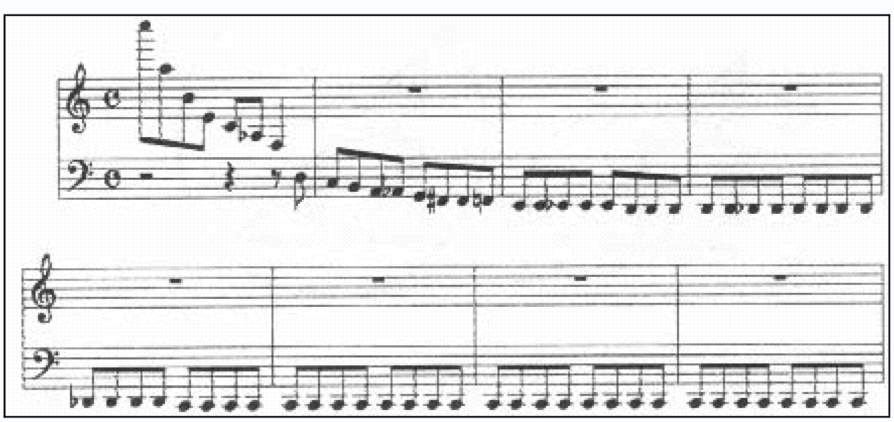
上例中，当乐音进行到达高音C时(也就是函数的收敛点)，便不会再有任何变化。

下例也是单一收敛的例子：

例3a



例3b



从以上例子可以发现，凡属单点收敛，其收敛音和起始音相同，都象是传统调性概念中的“中心音”。

有时，函数也许不会收敛到某一点，而是停留在某几个点，或一个特定的区域里。例如，当我们将表达式中的a值设定为3，得到的结果如下：

例4

为了节约篇幅，将长达150小节的变化过程作了删节。中间的休止小节就是删节的部分。从上例可以看出，上下两个声部逐渐靠近，最后停留在bD与D两个音上，构成双重收敛点。

还有四重收敛的例子：

例5

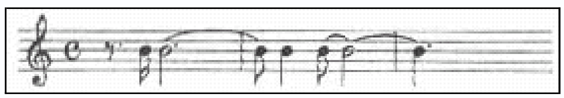
产生上例的迭代函数表达式中的a值为3.5。对于多重收敛点，我们可以称其为收敛区域。也许该称这种收敛现象为“多调性”。

分层结构也可用来构成节奏模型，如果用构成例5的表达式来控制节奏，我们可以得到下例：

例6

从21小节开始，节奏变化固定在四种时值组合的节奏模式之中：

例7



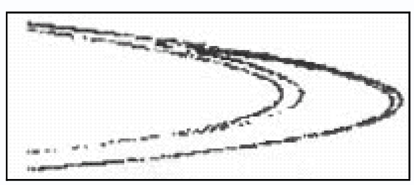
这也是四重收敛的结果。从一种看来非常随机的节奏组合开始，逐渐形成固定时值的节奏模型，这样的节奏处理手法，似乎在传统的作曲法中并不常见。

不同的表达式可以得到不同的收敛点或是收敛区域，不同的收敛区域构成不同的图形，以不同图形为基础来生成音乐成了算法作曲的一大特点。见以下表达式：

$$x_{n+1} = y_{n+1} - Ax_n^2 \qquad y_{n+1} = Bx_n$$

以上表达式可以构成的图形是：

图2



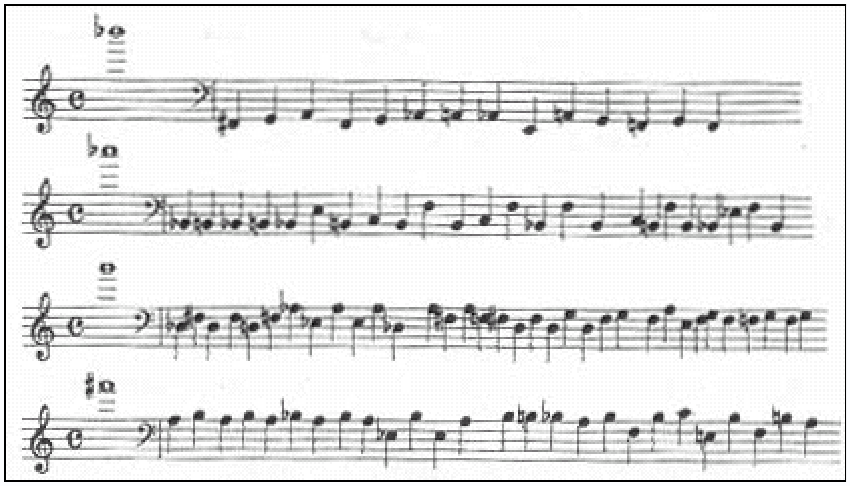
如果将以上表达式得到的函数值转化为乐音数据，我们可以得到下谱：

例8

要观察乐音变化的规律，音乐必须有一定的长度，因为篇幅所限，这里只能展示开始的18小节。如果观察较长篇幅的音乐(如在200小节范围内)，我们可以发现所有乐音都存在于一定的循环圈中，体现了分层结构控制的循环性。

例如，我们观察所有在高音bB、bA、G、#F以后发声的音，可以得到以下循环：

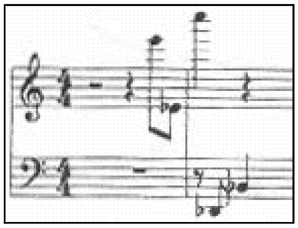
例9



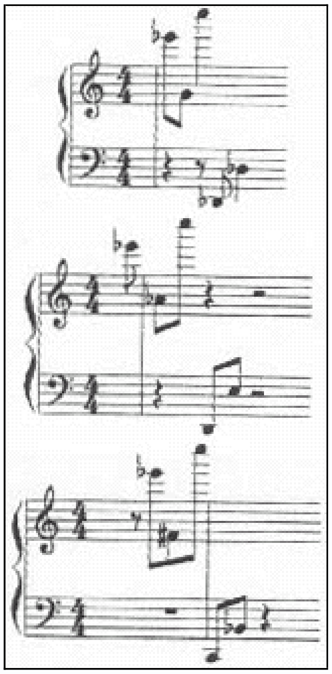
当然，我们还可列出和所有高音伴随的乐音，得出不同的循环体。值得注意的是，随着坐标音(高音)的降低，循环体的音区却不断提高。这种逐渐靠近的趋势，使我们想到和图2所呈现图形的某一部分有着内在的联系。

在乐谱中我们还可以观察到很多有趣的规律，这里就不一一列出。以任何一个乐音为坐标，都可以找到许多围绕着该坐标音形成的规律，当然也包括分层结构控制的自相似性。以上面的表达式生成的乐谱为例，以高音C为坐标，提取几个围绕高音C形成的片断，就可以看到这种相似性。见下例：

例10



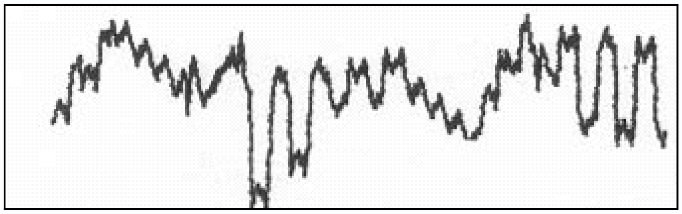
虽然每次出现的乐音构成都不一样，但旋律线条的轮廓是相似的。如果音乐进行的篇幅足够长，乐音构成也会出现循环。虽然在音乐过程中，这些细节属于“同一层次”，实际上，后者都是前者乘以一定的比例数的产物。



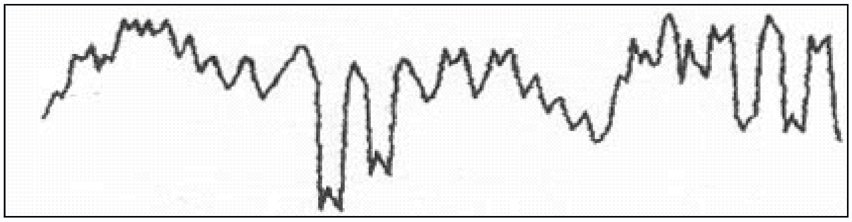
有趣的是，我们在古典音乐中也能很容易找到自相似性的例子。以巴赫的《C大调二部创意曲》(BWV772)为例，如果我们将主要旋律进行转化为图形，然后相隔一定的乐音删去一些音符，所获得的图形依然和原始图形相似，见下图（以下图形仅是乐曲的一部分）：

图3

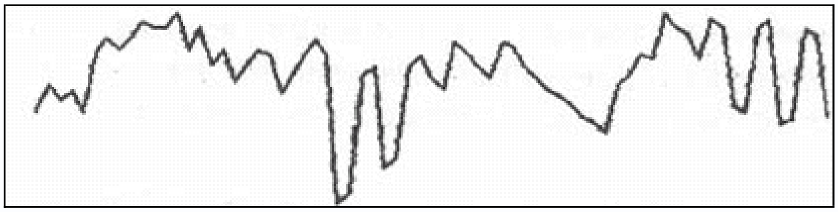
以十六分音符为基本单位(原型)：



以八分音符为基本单位：

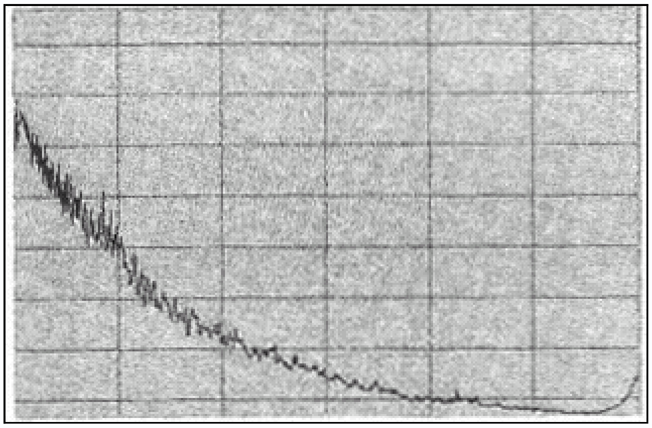


以四分音符为基本单位：

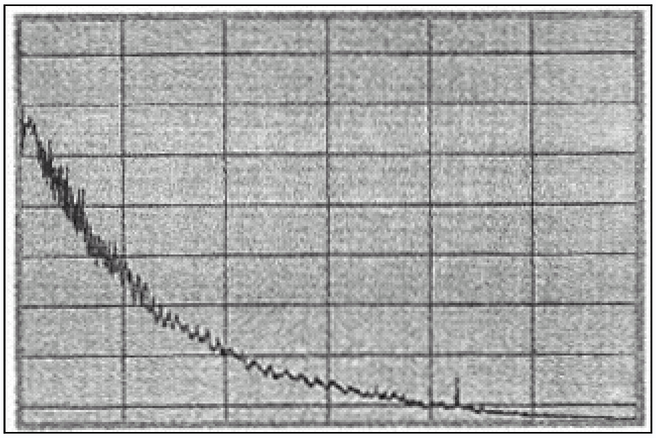


分层结构控制的另一重要特性是1/f noise，也就是比例噪声特性。在音乐中，比例噪声特性首先体现在声能频谱中。正如本文前面曾提到的，根据傅立叶合成理论，所有复杂的声音都是由不同频率的简单波形相互调制而成，而且，声音的声能是频率的函数，P(f)=1/f，频率越高，声能越低。例如，当我们将不同的音乐片断输入计算机，可以通过频谱分析程序得到以下的频谱图：

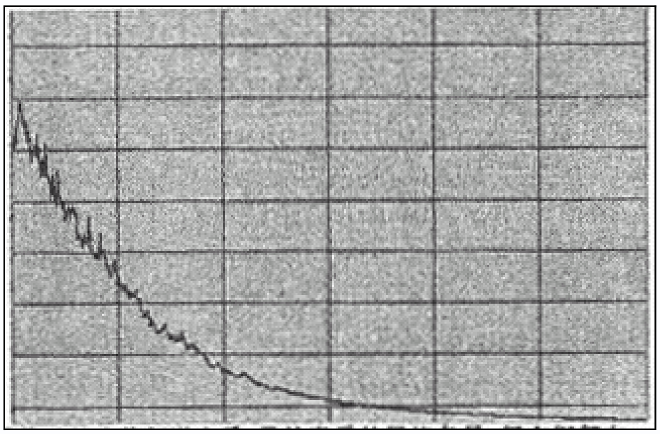
1.李盖蒂《大气》频谱：



2.德彪西管弦乐《吉格舞曲》频谱：



3.朱践耳《第四交响曲》频谱：



从以上三个不同类型音乐作品的声能频谱来看，虽然音乐的风格各异，但它们都在一定程度上符合1/f特性。同样的特性还可以在人的说话声、电压的波动规律、尼罗河的洪水泛滥规律中找到，体现了音乐与自然的某种内在联系。

1/f特性不但在声能频谱中存在，还体现在以分层结构规律生成的旋律之中。假如我们将1/f中的f认定为旋律进行

中相邻两音的音程值，而将能量值等同为音程值在作品中出现的可能性，那麽，旋律进行的音程变化概率与音程值成反比。音程值越大，出现的可能性就越小。下例就是计算机生成的具有1/f特性的旋律片断(没有特别的节奏处理)：

例11

统计以上片断的所有旋律音程，可以得到以下数据：二度进行：38次；三度进行：33次；四度进行：22次；五度进行：16次；六度进行：8次；其余进行：0次。

更令人惊奇的是，分层结构的1/f特性居然还体现在完全是人为结果的旋律进行中。以巴赫的《C大调二部创意曲》(BWV 772)为例，将前六小节的主要旋律线条作为分析样本，我们可以得到以下数据：二度进行：48次；三度进行：22次；四、五度进行：7次；六度进行：1次；其它进行：0次(以上计算不包括波音式的装饰进行)。

选取的分析样本越长，就越会贴近1/f 特性。

在声能频谱中，1/f 特性的基础是所有谐波之间的调制关系；而在旋律进行中，1/f特性的基础是：任何一个新出现的乐音都和先于它出现的所有乐音有着内在的联系。以下是笔者用Basic语言编写的1/f算子程序，该程序所生成的乐音数据都有着相互间的联系：

```
Cls
n = 12
Dim values(12) As Integer
seq_length = 2 ^ n
radom_max = 1 / n
previous_seq_index = seq_length - 1
seq_index = 0
Do While seq_index < seq_length
  output_value = 0
  i = 0
  Do While i < n
  a = i - seq_index
  b = i - previous_seq_index
  If a <> b Then
   values(i) = Rnd(radom_max)
   output_value = output_value + values(i)
  i = i + 1
  End If
      Loop
  previous_seq_index = seq_index
  seq_index = seq_index + 1
  Print output_value
  Loop
  End
```

(为了少谈音乐以外的问题，以及避免冗长的程序过程说明，本文就不详细分析程序的执行过程)

相对于另外两种有代表性的比例噪声特性：白噪声与勃朗宁噪声，1/f特性既不像白噪声特性那样处于完全随机状

态，而使旋律毫无规律可言；又不象勃朗宁噪声特性那样具有太强的可预见性，缺乏自由的妙味。1/f特性恰如其分的将逻辑性与自由度融为一体。在听觉实验中，将符合三种不同特性的旋律分别播放，听众普遍觉得符合1/f特性的旋律最能被人接受。

结　语

音乐之美在于有意味的乐音组织形式。“组织”的意味既在于能被人认知的逻辑，也在于能使人惊喜的意外。可以用多年前十分流行的比喻来形容，二者如同“鱼水关系”，逻辑似水，意外似鱼。组织逻辑虽然可以独立存在，可如果无意外之鱼，逻辑之水也就没有艺术的生气。水之欲可假计算机之手，鱼之欲则要借助于我们的艺术感觉。

音乐之妙在于其是自然与人性的完美结合。音程、和声、调式调性、节奏、音色等音乐要素自身的和谐与冲突、平衡与张力，都或多或少地可以找到客观物理和主观心理双方面的理由。音乐中所有现象存在，似乎都有着自然与人性的双重理由。对算法作曲与分层结构控制的讨论似乎也使我们看到了音乐中自然与人性相连的另一桥梁。计算机控制原理和数理逻辑，有时也会与人的思维过程在冥冥中“幽会”。也许因为计算机控制原理与数理逻辑本身既是自然的属性，也是人类思维的结晶。

由此看来，既然逻辑和意外不会对立、自然与人性不可对立，计算机与人更不该对立。不妨让二者融为一体，共同创造“美”“妙”音乐。

责任编辑　章　滨

参考文献：

作者简介：刘 健(1954~)，男，文学硕士，武汉音乐学院作曲与音乐音响导演系教授(武汉 430060).