

格子复杂性和符号序列的细粒化^{*}

柯大观[†] 张 宏 童勤业

(浙江大学生物医学工程系 杭州 310027)

(2004 年 2 月 17 日收到, 2004 年 5 月 24 日收到修改稿)

提出一种新的有限长一维符号序列的复杂性度量——格子复杂性, 建立在 Lempel-Ziv 复杂性和一维迭代映射系统的符号动力学基础上. 同时提出了符号序列的细粒化方法, 可与格子复杂性以及 Lempel-Ziv 复杂性结合. 新度量在细粒化指数较小时与 Lempel-Ziv 复杂性基本一致, 在细粒化指数增大时显示出截然不同的特性. 以 Logistic 映射为对象的计算实验表明, 格子复杂性对混沌区的边缘最敏感. 最后还讨论了上述复杂性度量的其他一些重要性质.

关键词: 混沌, 复杂性度量, 格子复杂性, 细粒化

PACC: 0545

1. 引 言

关于复杂性现在仍然是国内外一大研究热点. 但是, 在芬芳芜杂的文献当中, 始终没有人能够回答这样一个基本问题: 复杂性是什么? 其根本原因还是在于不存在复杂性的绝对度量. 而众多的研究者也几乎一致同意——找到这样的度量是不可能的.

目前为止, 复杂性的“定义”种类繁多, 各有其适用范围和局限. 既然具体的“定义”无助于得到关于复杂性的本质的认识, 不妨考察事物由简单到复杂的途径, 以得到启发.

郝柏林曾总结了这样几种复杂现象产生的途径^[1]:

- 1) 重复使用简单的规则, 可能形成极为复杂的行为或图形.
- 2) 把物理过程从高维空间投影到低维, 会使它们看起来更复杂.
- 3) 错误的参考系可能带来不必要的复杂化.

与此同时, 对复杂性的刻画经常伴随着对系统的粗粒化和符号化. 符号的描述需区分不同的层次. 比如 d, e, s, b, t 六个小写字母代表六种夸克, p, n, e 代表质子、中子和电子, 它们又组成各种基本元素. 构成 DNA 的核苷酸分子只用 a, t, c, g 四种符号代

表, 地球上所有生物 DNA 都由这四种核苷酸编码而成的. 这类层次问题也可被视为尺度问题. 一个序列在一种尺度下被认为是复杂的, 在另一种尺度下却可能被视为简单, 甚至仅用一个符号代替. 也就是说, 复杂可能是相对的、有层次的.

人们自然会有这样的疑问: 有无可能在复杂性度量方法中反映出这种相对性和层次性?

只限于讨论单个符号序列的复杂性. 自从 Kolmogorov 提出算法复杂性^[2]概念以来, 已有了不少可实现的算法. 其中被认为最精致的是 Lempel-Ziv 复杂性^[3], 甚至有许多文献将其与 Kolmogorov 复杂性等同起来, 并在物理上的时空模式^[4]、医学上的脑电识别^[5]以及语音识别^[6]等众多领域得到应用. 然而, 正如 Lempel 和 Ziv 以及其他学者所指出的那样^[7, 8], Lempel-Ziv 复杂性仅仅用复制和添加两种简单的操作代替了 Kolmogorov 复杂性定义中的通用图灵机, 即便对 Kolmogorov 复杂性定义而言也不可避免的有它的缺陷.

首先, Lempel-Ziv 复杂性衡量的是随机性, 而复杂性与随机性是有区别的. Grassberger 曾以三张图片直观地表示出这种区别^[9]. 一般认为理想的复杂性与随机性关系应如图 1 所示.

其次, Kolmogorov 复杂性以生成符号序列的最短的通用图灵机程序长度为复杂性的度量. Lempel-

^{*} 国家自然科学基金(批准号 30170267)和国家重大基础研究前期研究专项(批准号 2002CCA01800)资助的课题.

[†] E-mail: kedaguan@sohu.com

Ziv 复杂性作为简化的模型,对于由迭代产生的混沌伪随机序列视为复杂,但显然产生这些序列的程序其实相当短,并不能认为其 Kolmogorov 复杂性高.

最后,Lempel-Ziv 复杂性将一些人为构造的有明显规律的序列如布里渊序列也视为复杂,也是一个明显的缺陷.

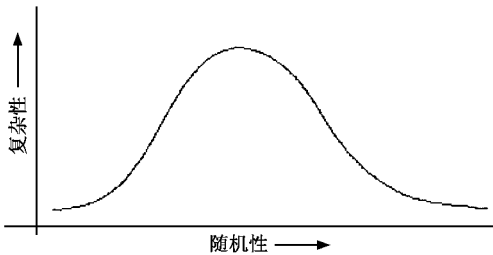


图 1 理想的复杂性与随机性关系

显然,一个既具有 Lempel-Ziv 复杂性优点又弥补其缺陷的新复杂性算法将可能带来复杂性研究的进步.一个很自然的问题是:能否通过设计一个比 Lempel 和 Ziv 定义的计算模型更复杂的机器来达到

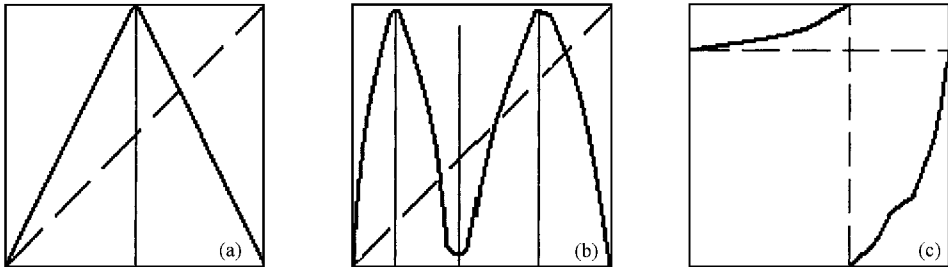


图 2 (a)单峰映射.(b)多临界点映射.(c)圆映射

我们知道非线性系统轨道内的任一点,按回复性的强弱,可分为不动点、周期点、回复点、非游荡点和游荡点^[13].对于某种程度的粗粒化过程而言,只考虑按有无回复性划分为两类——回复点和非回复点.现在,假定每个符号只表示这样一个点的邻域.给定任意有限长符号序列,其中特定长度的子序列总是能找到一个自映射迭代系统的轨道与之对应.这样的子序列将被视为简单,称为迭代序列.它包括出现回复点的周期性序列(和周期序列有区别,同一周期内符号不重复)和全部由非回复点构成的混沌序列.

我们结合迭代系统符号的动力学和 Lempel 和 Ziv 的计算模型,设计出一种新的复杂性算法——格子复杂性.给定符号序列的复杂性由该序列的格子数目决定.

作为一个借用的名词,格子在这里有特殊的

这个目的?

为 Anderson 所垢病的企图建立包罗一切的框架的努力^[10]不是本文的目的.但本文将力图对前述问题做出解答.

2. 格子复杂性与 Lempel-Ziv 复杂性

2.1. 格子复杂性

已知许多自映射迭代系统,如图 2 所示的线段自映射和圆映射系统,其动力学性态复杂,很容易就达到混沌态,其轨道可在系统区间内处处稠密.同时,这些自映射系统的轨道又总是可以通过对系统区间的粗粒化得到相对应的符号序列^[11,12].这种粗粒化过程通常以 i 个临界点为界,将系统区间划分为 $i + 1$ 个子区间,赋以不同的符号.这不可避免的会丢失许多细节.但如果粗粒化程度不断降低,每个符号代表的子区间就会不断缩小.

性质:

- 1) 格子在起首位置包含一个迭代序列;
 - 2) 格子具有记忆功能,可以记住之前的所有连续操作并简单重复,也就是 Lempel 和 Ziv 定义的递归复制的功能;
 - 3) 格子之间由一个添加操作生成的符号隔开.
- 以下介绍具体算法.

设序列为 $s_1 s_2 \dots s_n$,用 $lc(n)$ 表示格子复杂性.现考虑中间步骤,设已生成前缀 $s_1 s_2 \dots s_{r-1}$, $r < n$,并且下一个符号 s_r 是用添加操作完成的,记为

$$s_1 s_2 \dots s_n \rightarrow s_1 s_2 \dots s_{r-1} s_r \vee s_{r+1} \dots$$

这里 s_r 后面的符号“ \vee ”反映了其生成过程,也说明 s_r 是两个格子之间的分隔符号.现在给出进一步的法则.

第 1 步 先令 $Q = s_{r+1}$,这时 Q 只有一个符号,

不做任何操作.继续令 $Q = s_{r+1} s_{r+2}$,用 x 表示 Q 的最后一个字母,此时 $x = s_{r+2}$,观察 x 是否等于 $Q\pi$ 中的任意一个符号(π 表示将前面的符号串最后一个符号去掉,这里有 $Q\pi = s_{r+1}$).若否,继续令 $Q = s_{r+1} s_{r+2} s_{r+3}$,此时 $x = s_{r+3}$,观察 x 是否等于 $Q\pi$ 中的任意一个符号,若否,则令

$$Q = s_{r+1} s_{r+2} s_{r+3} s_{r+4},$$

依次类推,直到 x 可由 $Q\pi$ 中的一个符号复制为止,转入下一步.

第 2 步 设当前的 $x = s_{r+j}$, $Q\pi$ 中的相等符号为 s_{r+i} ($0 < i < j$),令 $P = s_{r+i+1}$, $R = s_{r+j+1}$,观察 P 是否等于 R ,若是,继续令

$$P = s_{r+i+1} s_{r+i+2}, R = s_{r+j+1} s_{r+j+2},$$

观察此时 P 是否等于 R ,若是,再令

$$P = s_{r+i+1} s_{r+i+2} s_{r+i+3},$$

$$R = s_{r+j+1} s_{r+j+2} s_{r+j+3},$$

依次类推,直到 $P \neq R$ 为止.显然,此时 $QR\pi$ 是一个迭代序列,进入下一步.

第 3 步 重新令 $Q = QR$,观察 Q 能否由 $sQ\pi$ 的某个子串复制得到,其中 $s = s_1 s_2 \dots s_r$,若能,设此时 Q 的最后一个字母为 s_{r+k} ,则继续令

$$Q = s_{r+1} s_{r+2} \dots s_{r+k} s_{r+k+1},$$

再观察 Q 能否由 $sQ\pi$ 中的子串复制得到,若能,则再考虑

$$Q = s_{r+1} s_{r+2} \dots s_{r+k} s_{r+k+1} s_{r+k+2},$$

依次类推.这样,或者当 Q 包含了给定符号序列的最后一个符号 s_n 时则分析结束,或者对某个 Q 不能由 $sQ\pi$ 的子串复制得到,这时就采取添加操作,添加 Q 的最后一个符号,并在其后面加上符号“ \vee ”表示这是格子之间的分隔符,这样又回到第 1 步开始前的情况.

可以看出,添加操作将整个符号序列划分为一定数目的格子.如果序列的最后一个符号是通过添加操作完成的,就说明最后一个格子是完整的,那么符号“ \vee ”的数目就等于格子数;如果序列的最后一个符号后面没有符号“ \vee ”,则说明最后一个格子是不完整的,但我们仍视之为是一个格子,格子数就等于符号“ \vee ”的数目加 1.

这个算法的意义在于,任何迭代序列代表的不是混沌(对限定符号集而言就是指符号不重复)轨道就是周期性回复轨道.对同一个映射,同一点(由同一符号代表)不可能既属于非回复点(产生混沌轨道

踪迹),又属于回复点(产生周期轨道踪迹).所以,一旦非周期性序列出现了回复点,就认为迭代序列结束了.随后或者是另一个迭代序列开始(也就是另一个格子出现),或者发现刚结束的迭代序列加上一些后续的符号都可由简单复制产生.

2.2. 格子复杂性与 Lempel-Ziv 复杂性的关系

观察上述格子复杂性算法,我们就会发现,如果将前两步去掉,直接令 $Q = s_{r+1}$ 而执行第 3 步,那就是 Lempel-Ziv 复杂性.所以格子复杂性与 Lempel-Ziv 复杂性的区别就在于每个格子在起始位置会有一个迭代序列,只有“跳过”迭代序列才开始检查整个格子能否复制.

现在,用 $c(n)$ 来代表 Lempel-Ziv 复杂性,以符号“ \cdot ”表示 Lempel-Ziv 复杂性计算中的添加操作,我们来考察一下 $lc(n)$ 和 $c(n)$ 之间的关系.

举例说明,设有二进制符号序列

$$s = 10011000011100000111101101,$$

其 Lempel-Ziv 复杂性和格子复杂性计算结果如下:

Lempel-Ziv 复杂性:

$$1 \cdot 0 \cdot 01 \cdot 1000 \cdot 0111 \cdot 00000 \cdot 1111 \cdot 01101 \cdot$$

格子复杂性:

$$1001 \vee 100001 \vee 1100000 \vee 11110 \vee 1101 \vee.$$

$$\text{复杂性 } c(n) = 8, lc(n) = 5.$$

简要介绍前几位的计算过程. Lempel-Ziv 复杂性计算中,前两位 10 作为新符号从空串开始添加了两次.随后的第 3 位 0 可以复制得到,而 01 不能从 100 中复制,所以采取添加操作.后面搜索到 1000 子串时又不能由复制得到,继续添加.随后的序列依次类推.

格子复杂性计算时,将前两位视为迭代序列的一部分,由于符号各不相同,可能是个混沌序列,不做操作.随后第 3 位又是 0,说明此迭代序列最终是个周期为 1 的周期性序列(不动点序列),第 4 位为 1,表明迭代序列结束,这一位只能采用添加操作,从而划分出第一个格子.随后 10000 是下一个不动点迭代序列,下一位的 1 表明迭代序列结束.此时检查发现 100001 不能由复制产生,所以这一位是添加操作的,第二个格子结束.依次类推.

Lempel-Ziv 复杂性的添加操作通常相当于划分格子.只在两种情况下两者不等价:

1) 出现一个没有重复符号的子序列,如上例中的前两位;

2)在添加操作之后出现同一个符号连续重复的子序列,其长度超过前面已出现过的同类子序列,如上例中的第八位。

显然,对非空符号序列 Lempel-Ziv 复杂性 $c(n)$ 和格子复杂性 $lc(n)$ 之间存在如下关系:

$$1 \leq lc(n) \leq c(n). \quad (1)$$

设符号集(也称字母表) S 有 α 个符号,第一种情况引起的两种复杂性之差值为 $\Delta_1(n)$,第二种情况引起的差值为 $\Delta_2(n)$,则有

$$c(n) - lc(n) = \Delta_1(n) + \Delta_2(n). \quad (2)$$

第一种情况下,在一个格子内,最大可能存在 α 次 Lempel-Ziv 定义下的添加操作.但是,这样的格子是有限的.因为 α 个符号的排列方式最多为 $\alpha!$,所以有

$$\lim_{n \rightarrow \infty} \Delta_1(n) \leq \alpha!. \quad (3)$$

第二种情况,既已发现 $\Delta_1(n)$ 与 α 有关,不妨先假定 n 为充分大的常数,考察 $\Delta_2(\alpha)$.

假定所有符号以相同的概率出现,则所有长度为 n 的序列出现的概率都为 α^{-n} .在这种情况下, Lempel 和 Ziv 证明^[7],几乎所有的序列复杂性都接近上界 $\frac{n}{\log_\alpha n}$,是复杂的。

现设 m 为一个有限正整数, $m < n$. 设 k 为给定序列中第 1 次出现的全部由相同符号组成的子序列的字长.显然,出现这样的子序列的概率为 $\alpha \cdot \alpha^{-k}$. 随后只有在其他符号作为间隔的情况下,又出现字长为 $k+1$ 以上的同一个符号连续重复的子序列(其概率最多为 $\alpha^{-(k+1)}$)时,才第 1 次出现第二种情况,此时有 $\Delta_2(\alpha) = 1$. 随后如果再有字长为 $k+2$ 的同类子序列出现(其概率为 $\alpha^{-(k+2)}$),才会使 $\Delta_2(\alpha) = 2$. k 最小只能等于 2,所以有

$$\begin{aligned} P[\Delta_2(\alpha) \geq 1] &< \alpha \cdot \alpha^{-k} \cdot \alpha^{-(k+1)} \\ &= \alpha^{-2k} \leq \alpha^{-4}, \\ P[\Delta_2(\alpha) \geq 2] &< \alpha^{1-k-(k+1)-(k+2)} \\ &= \alpha^{-3k-2} \leq \alpha^{-8}, \\ &\dots\dots \\ P[\Delta_2(\alpha) \geq m] &< \alpha^{1-k-(k+1)-(k+2)-\dots-(k+m)} \\ &\leq \alpha^{1-2-(2+1)-(2+2)-\dots-(2+m)} \\ &= \alpha^{1-\frac{1}{2}(m+1)(m+4)}, \end{aligned} \quad (4)$$

可得 $\lim_{\alpha \rightarrow \infty} P[\Delta_2(\alpha) \geq m] = 0 \quad (m = 1, 2, \dots, n).$ (5) 说明随着符号集测度趋向无穷, $\Delta_2(\alpha)$ 依概率

收敛于 0. 其实由(4)式也可看出,即便 α 数值上很小,大多数长为 n 的符号串,其 $\Delta_2(\alpha)$ 也不超过一个有限的正整数 m . 而对于少数例外,由于其具有的规律性,正是我们希望格子复杂性与 Lempel-Ziv 复杂性相区别的。

结合两种情况,则有

$$c(\alpha, m) - lc(\alpha, m) \leq \alpha! + m. \quad (6)$$

在 α 数值固定时, $\alpha! + m$ 一般来说是一个有限的常数.这符合 Kolmogorov 复杂性意义下不同图灵机之间差别不超过一个有限常数的性质.在此情况下, $n \rightarrow \infty$ 时同样有 $lc(n) \sim \frac{n}{\log_\alpha n}$, 并有归一化度量

$$U_n = \frac{lc(n) \log_\alpha n}{hn}, \quad (7)$$

其中 h 为信息熵。

但这个归一化度量对格子复杂性来说将在 α 增大时没有太大意义,仅供参考。

2.3. 混沌轨道踪迹的复杂性

我们已经证明 Lempel-Ziv 复杂性和格子复杂性之间的差别与符号集 S 的测度 α 有关.当用这两种复杂性计算符号序列时,它们的数值本身与 α 显然也有密切关系.所以,这两种复杂性的更恰当的表达方式应该是 $c(\alpha, m)$ 和 $lc(\alpha, m)$.

现在用这两种复杂性来考察如图 2 所示的各种自映射系统的混沌轨道.这里的混沌轨道是指在 Devaney 意义下在系统区间内处处稠密的非周期轨道^[14].

以图 2(a)为例,当按符号动力学的通常作法把系统迭代区间简单划分为左右两部分,即 $\alpha = 2$, 这时粗粒化程度最高. n 充分大时, $\Delta_1(2, n) \leq 2!$, 这可以说是微不足道的.但当对系统的粗粒化程度不断降低,比如 α 呈指数增长时,两者的差别 $\alpha!$ 就不断增大.若 α 增长速度为 $O(\alpha^2)$, $\Delta_1(\alpha, n)$ 的增长速度为 $O((\alpha^2)! - \alpha!)$. 当 α 趋向与无穷大时,每个符号表示的子区间趋向于无穷小,这个系统中的所有非回复点必定被包含在符号子区间范围内,由于混沌轨道的稠密性和非回复性,混沌轨道的踪迹与轨道本身趋于重合.所以,在 $\alpha \sim n$ 时,对于格子复杂性有

$$\lim_{\alpha \rightarrow \infty} \lim_{n \rightarrow \infty} c(\alpha, m) = 1. \quad (8)$$

上述命题对所有的周期轨道和准周期轨道都成

立,但所要求的条件比混沌轨道宽松.随后我们将在第 5 节给出更严格的证明.其他相关命题的证明也将在第 5 节给出.

不管怎样,我们已经得到了想要的结果.对格子复杂性而言,某种意义上混沌轨道和周期轨道一样简单.但对于 Lempel-Ziv 复杂性,由于每出现一个新符号就要进行一次添加操作,混沌轨道踪迹在 $\alpha \sim n$ 和 $\alpha \rightarrow \infty$ 时有 $C(\alpha, n) \sim \alpha$,显然只能是复杂的.

3. 复杂性的细粒化度量方法

格子复杂性与 Lempel-Ziv 复杂性之间的差异只有在符号集测度 α 增大的时候才能体现.这就不免牵涉到的对系统区间如何划分的问题.任何系统区间的分划,常用的无非是同质分划(homogenous partition)和生成分划(generating partition)两种^[15],可以用 P^H 和 P^G 来表示.前者将整个区间平均分成 N^H 个子区间,后者则根据系统的动态特性,使划分的 N^G 个子区间的边界总是映射到自身.图 3 直观的表示了两种分划的差异.

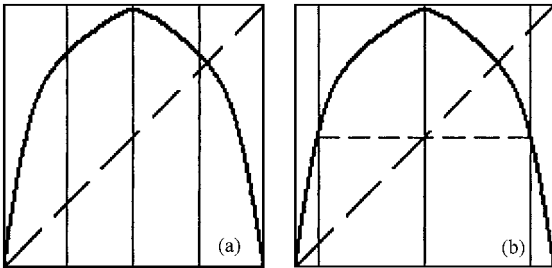


图 3 (a)同质分划.(b)生成分划

不论何种分划,对每个子区间赋以一个符号,在整个区间取定一点 x 为初始点,经过迭代就得到一条轨道,进而有一个符号序列 $s(x)$.我们就可以用复杂性度量来研究这些符号序列的复杂性.对区间划分的越细,字母表中的符号数目越多,格子复杂性就越显示出不同于 Lempel-Ziv 复杂性的特性.

显然,对杂乱的时间序列数据(如脑电)可用同质分划,对已知系统则作生成分划,都可不断细化来研究其复杂性.但对于已由高度粗粒化得到的一串符号序列又该怎么办?

设存在符号序列 $s = s_1 s_2 \dots s_n$,字母表 S 大小为 α .现在,若以字长为 r 的子列为新符号,采用移位方式搜索这个序列,将得到长度为 $n - r + 1$ 的新序列 s_r ,所有可能的符号用字母表 S^r 表示,其符号数

为 α^r .称 r 是以 α 为底的符号序列的细粒化指数.

符号动力学给出了轨道点、字母和逆函数的对应关系. r 就对应于生成分划中的细化指数(order of refinement).一次移位就相当于映射函数的一次迭代.

称最初的符号集 S 为基字母表.在细粒化指数增大的情况下,所有可能出现的字母数量呈指数增长,有限长度的序列不可能出现全部符号.所以在可能的字母表 S^r 和实际出现的符号组成的新字母表 S_r^* 之间存在映射 f ,新字母表大小则用 β 表示.即

$$f : S^r \rightarrow S_r^*, \alpha^r \rightarrow \beta.$$

实际应用中,任何先验知识可在映射 f 中得到体现.比如氨基酸的分类等等.

以上提供了一个在一定粗粒化基础上变尺度度量时间序列和符号序列复杂性的方法,从而解答了一些文献中关于粗粒化的疑问^[16,17].许多文献认为 Lempel-Ziv 复杂性只适用于二进制符号串是个误解.事实上,上述的细粒化方法对格子复杂性和 Lempel-Ziv 复杂性都适用,基字母表的测度 α 也不必然为 2.对归一化度量,对数以 β 为底.

基字母表测度 α 已知的情况下,对长度为 n 的序列, Lempel-Ziv 复杂性和格子复杂性也可采用细粒化指数为参数表示为 $C(r, n)$ 和 $l(r, n)$.本文随后的内容都将沿用这种表达.

4. 一维混沌系统中的应用

已知 Logistic 映射

$$x_{n+1} = 1 - ux_n^2, u \in [0, 2],$$

从 $[-1, 1]$ 之间的某一点作为初始点开始反复迭代,对不同的 u 将得到各种复杂度的序列.当参数增加到 $u = 1.40115515\dots$ 除少数窗口外进入混沌区.

以下都是在迭代 100 次之后取 $n = 8204$ 步的符号序列求得的复杂性图示($\Delta u = 0.01$).

图 4 表明在 $r = 1$ 时格子复杂性和 Lempel-Ziv 复杂性基本相同.

图 5 和图 6 表明了 Lempel-Ziv 复杂性和格子复杂性之间的差异随细粒化指数增大而放大.但有意思的是,两者又有共同的特征,都在 r 增大到一定程度时达到稳定的数值,都是随机性越小的序列到达稳定区域越缓慢.在全部达到稳定数值之前,两种复杂性数值按随机性大小排序相反.

使一串符号序列 s 的 Lempel-Ziv 复杂性和格子

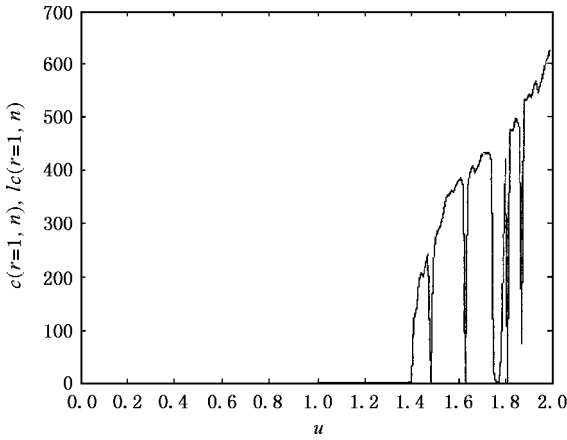


图 4 细化化指数为 1 时格子复杂性和 Lempel-Ziv 复杂性

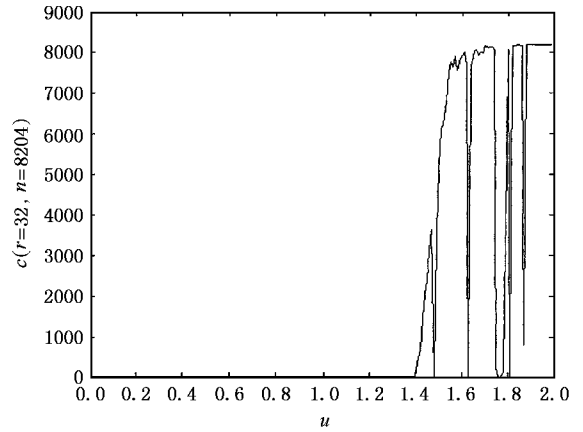


图 7 r = 29 时 Logistic 映射的 Lempel-Ziv 复杂性

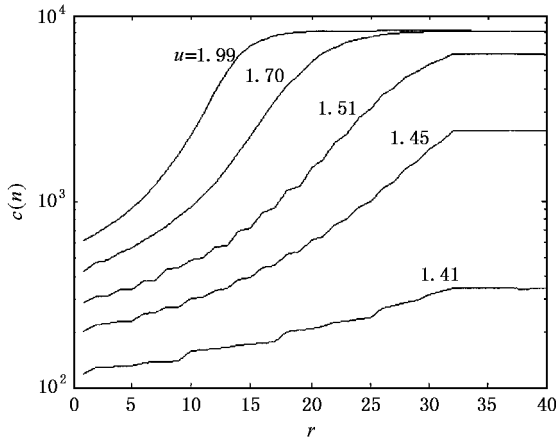


图 5 Lempel-Ziv 复杂性与细化化指数的关系

指数 $r^* = 29$ 此时 Lempel-Ziv 复杂性和格子复杂性如图 7 和图 8.

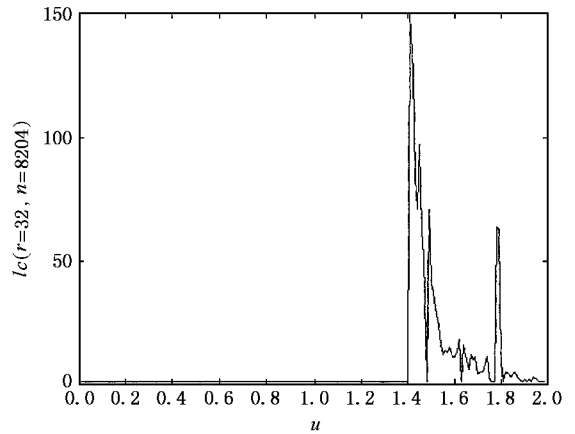


图 8 r = 29 时 Logistic 映射的格子复杂性

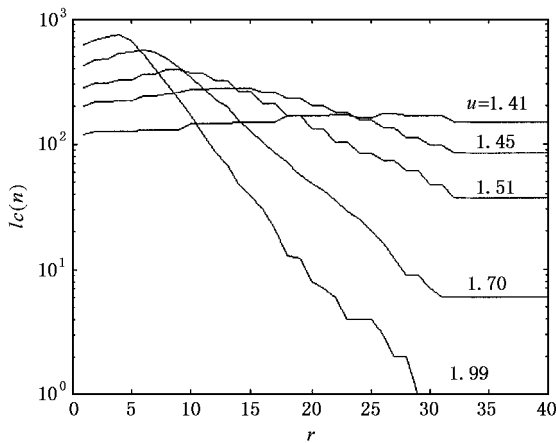


图 6 格子复杂性与细化化指数的关系

图 7 表明, 细化化指数增大使大多数序列的 Lempel-Ziv 复杂性普遍升高. 而由图 8 可明显发现, 细化化指数增大到一定程度后, 格子复杂性只在 Logistic 映射刚刚由倍周期分岔进入混沌区的边缘时达到峰值, 其后在周期窗口由危机进入混沌时达到其他较低的峰值. 随机性较大时降到和周期序列相等的 1. 也就是说, r 增大, 有更多的序列被 Lempel-Ziv 复杂性视为复杂, 而更多的序列被格子复杂性视为简单. 需要指出, 选择部分序列进入稳定区后的数值有利于确定基点, 非稳定区的特性对不同的应用场合提供了选择, 因而也具有重要价值.

复杂性达到稳定数值的最小细化化指数, 我们称之为该序列的临界指数, 用 r^* 表示. 我们将发现, 随机性越小, 临界指数越大. 选择图 5 图 6 中最小的临界

5. 相关命题的证明及讨论

除了第 3 节提到的性质以外, 关于两种复杂性

的细粒化度量方法,还有一些重要的命题.

设长度为 n 的符号序列 s , 基字母表 S 中有 α 个符号.

命题 1 若 s 是周期序列, 最小周期为 m ($m < n$), 则有

$$c(m, n) = m, \text{ 且 } l(c, n) = 1. \quad (9)$$

这个命题是显然的, 证明从略.

命题 2 若 s 为所有符号等概率分布的完全随机序列, 当细粒化指数 $r \propto n$ 时, 有

$$\lim_{n \rightarrow \infty} \lim_{r \rightarrow \infty} P[l(c, n) = 1] = 1, \quad (10)$$

同时有

$$\lim_{n \rightarrow \infty} \lim_{r \rightarrow \infty} P[c(r, n) = n - r + 1] = 1. \quad (11)$$

证明 由已知条件, 所有 r 子列出现的概率为 α^{-r} . 出现两 r 子列相等的概率为 $\alpha^{-r} \cdot \alpha^{-r} = \alpha^{-2r}$, 此时共有 $n - r + 1$ 个子列, 所以

$$P[l(c, n) > 1] \leq (n - r) \cdot \alpha^{-2r}.$$

考虑条件 $r \propto n$, 即有

$$\lim_{n \rightarrow \infty} \lim_{r \rightarrow \infty} P[l(c, n) > 1] = 0.$$

因为 $l(c, n) \geq 1$, 所以(10)式成立. 这说明所有 $n - r + 1$ 个子列依概率都不相等, 因而(11)式也成立. 命题得证.

特别的, 对命题 2 有如下推论.

推论 1 当 $n = r + c$ 时, c 为常数. 有 $\lim_{n \rightarrow \infty} \lim_{r \rightarrow \infty} P[c(r, n) = c + 1] = 1$ 成立.

推论 2 当 $n = c$ 时, $r \leq n$, 有 $\lim_{r \rightarrow c} P[c(r, n) = c - r + 1] = 1$ 成立.

上述命题说明, 对于格子复杂性, 周期和随机序列可视为简单. 只有既存在结构又具有随机性的序列才被视为复杂. 推论则说明对完全随机序列, 细粒化过程进入稳定区时 $c(r, n)$ 和 r 呈线性关系的特性. n 为常数时, $c(r, n)$ 随 r 增大线性单调下降. 最极端的情况 $r = n$ 时, $c(r, n) = l(c, n) = 1$, 此时整个序列为一个符号, 不论以格子复杂性还是 Lempel-Ziv 复杂性衡量都是最简单的.

一串长为 $\alpha^k + k - 1$ 的符号序列, 其字母表中有 α 个符号, 如果在其长为 k 的子序列中出现所有的 α^k 种组合, 即每种组合恰好出现一次, 则此序列是布里渊序列. 已被证明布里渊序列的 Lempel-Ziv 复杂性很高^[7], 但对格子复杂性有如下命题.

命题 3 若序列 s 为布里渊序列, 长度 $n = \alpha^k + k - 1$, 则此时有

$$r^* = k, \text{ 且 } l(k, n) = 1. \quad (12)$$

证明 由格子复杂性和布里特渊序列的定义可

知, 细粒化指数 $r = k$ 时, 该序列整体为一个迭代序列, 只有一个格子. $l(k, n) = 1$ 成立.

现证明 k 为临界指数. 显然只要 $r \geq k$ 时 $l(c, n)$ 数值不变, 而 $r < k$ 时数值有明显变化, k 就一定是临界指数.

1) $r > k$ 时, s 中所有长度为 r 的子列不可能有重复, 否则必定有长度为 k 的子列也重复, 这与布里特渊序列定义矛盾. 所以 $l(c, n) = l(k, n) = 1$, 即格子复杂性的数值在 $r \geq k$ 时是稳定的.

2) $r = k - 1$ 时, 所有长度为 $k - 1$ 的子列共有 α^{k-1} 种组合, 则序列 s 中必有 α 个 $k - 1$ 子列重复, 说明不会是混沌序列. s 也不是周期性序列, 否则 α 个 $k - 1$ 子列的后一位字母必定相同, 使长度为 k 的子列也重复, 与定义矛盾. 所以 $l(k - 1, n) \geq \alpha > 1$. 说明格子复杂性数值不稳定.

综上所述, 临界指数 $r^* = k$. 命题得证.

由命题 3 可知, 对于长度为 8204 的布里特渊序列, 临界指数为 13. 和上一节 Logistic 映射的结果相比, 在长度相同的非周期序列中布里特渊序列的临界指数是最小的. 这提供我们应用复杂性度量时一个有益的参考指标. 也就是说, 在寻找临界指数时不需要由 1 开始搜索.

需要指出, 临界指数并不总是如命题 3 那样明显, 也不必然是整数. 它和符号序列的长度以及基字母表有关. 改变基字母表测度和序列长度, 或者在基字母表不变时改变对应的系统区间的分划方式, 临界指数不排除出现分数的可能. 可以将临界指数理解为粗粒化基础上时间序列的特殊“分维”. 想象一个多维空间, 每一维坐标上被划分为与基字母表对应的子区间, 多维坐标上的子区间组合成多维“盒子”. 临界指数就是使序列落入每个多维“盒子”次数最少时的维数中最小的一个.

回顾格子复杂性的定义, 实际上建立在一个很强的假定上, 每个符号只代表一个回复点或者非回复点. 同时算法中存在统计的规则: 子序列中每个符号只出现一次, 就被视为混沌序列. 那么, 如果假定的条件不那么强, 统计规则不那么严格会怎样? 比如我们规定迭代序列除了前述定义以外每个符号出现 N ($1 < N \leq n/\alpha$) 次也视为混沌序列(可称为弱定义的格子复杂性), 省略繁琐的推导, 实际上都有类似于本节的命题成立(命题 2 需稍作修改), 而对实际应用中的数据, 临界指数会相对变小.

另外, 在细粒化符号序列时采用了移位方式, 对

应于映射系统的一次迭代,并最大可能的提取了子序列.如果采用多步的子转移方式,就对应映射函数的多步迭代,对研究混沌系统依然很有意义,其复杂性度量结果将出现非常有趣的变化.限于篇幅,将另文讨论,这里不再赘述.

6. 结 论

格子复杂性既保留了 Lempel-Ziv 复杂性的优点又弥补其缺陷.而对符号序列复杂性的细粒化度量方法则犹如复杂性度量上的“显微镜”,细粒化指数就如“放大倍数”.在不同的尺度上度量复杂性,反映了复杂性的层次性和相对性.有意思的是,格子复杂性和 Lempel-Ziv 复杂性在细粒化指数较小时基本等价,当细粒化指数增大时表现出迥然不同的特性,显

示出两者之间既联系又对立的辩证关系,像一枚硬币的两面,构成矛盾的统一体.

本文的复杂性度量方法,原则上适用于一切可被符号化的研究领域.符号化在小到量子^[18],大至传统的天文地理^[19]均已实现.对于已经应用了 Lempel-Ziv 复杂性或其他复杂性算法的领域,包括脑电、心电、伪随机序列^[20]等都具参考价值.

然而最重要的是,本文宣扬和支持了这样一种观念:在不存在复杂性的绝对度量的情况下,大多数的序列既是复杂的又是简单的.但在一定条件下(细粒化指数依次进入平稳区时)部分序列(尤其是混沌区边缘的序列)的格子复杂性具有客观的差异,这可能反映了复杂性的某种最本质的属性——“产生于秩序与混沌的边缘”^[21].

- [1] Hao B L 2001 *Physics* **30** 466 (in Chinese) [郝柏林 2001 物理 **30** 466]
- [2] Kolmogorov A N 1965 *Info. Trans.* **1** 3
- [3] Remo B and Antonio P 1997 *Complexity: hierarchical structures and scaling in physics* (Cambridge : Cambridge University Press) 233—235
- [4] Kaspar F and Schuster H G 1987 *Phys. Rev. A* **36** 842
- [5] Li L Y, Tong Q Y 2000 *Acta Electr. Sin.* **28** 97 (in Chinese) [李凌云、童勤业 2000 电子学报 **28** 97]
- [6] Li M D, Zhang H and Tong Q Y 2003 *Space Medicine&Medical Engineering* **16** 215 (in Chinese) [李敏丹、张宏、童勤业 2003 航天医学与医学工程 **16** 215]
- [7] Lempel A and Ziv. 1976 *IEEE Trans. Inform. Theor.* **IT-22** 75
- [8] Xie H M 1994 *Complexity and Dynamical System* (Shanghai : Shanghai Scientific and Technological Education Publishing House) (in Chinese) [谢惠民 1994 复杂性与动力系统(上海:上海科技教育出版社)]
- [9] Grassberger P 1986 *Inter. J. Theor. Phys.* **25** 907
- [10] Anderson P W 1991 *Physics Today* **7** 9
- [11] Zheng W M and Hao B L 1994 *Applied Symbolic Dynamics* (Shanghai : Shanghai Scientific and Technological Education Publishing House) (in Chinese) [郑伟谋、郝柏林 1994 实用符号动力学(上海:上海科技教育出版社)]
- [12] Zhang Z J and Chen S G 1989 *Acta Phys. Sin.* **38** 1 (in Chinese) [张张建、陈式刚 1989 物理学报 **38** 1]
- [13] Zhou Z L 1997 *Symbolic Dynamics* (Shanghai : Shanghai Scientific and Technological Education Publishing House) (in Chinese) [周作领 1997 符号动力系统(上海:上海科技教育出版社)]
- [14] Devaney R L 1989 *An Introduction to Chaotic Dynamical Systems*. Second Edition. (Addison-Wesley, Redwood City, Calif.)
- [15] Wackerbauer. R *et al* 1994 *Chaos, Solitons & Fractals* **4** 133
- [16] Meng X *et al* 2000 *Acta Biophysica Sin.* **16** 701 (in Chinese) [孟欣等 2000 生物物理学报 **16** 701]
- [17] Shen E H *et al* 2000 *Acta Biophysica Sin.* **16** 707 (in Chinese) [沈恩华等 2000 生物物理学报 **16** 707]
- [18] Boqomolny E B 1992 *Nonlinearity* **5** 805
- [19] Manson S M 2001 *Geoforum* **32** 405
- [20] Cai J P, Li Z, Song W T 2003 *Acta Phys. Sin.* **52** 1873 (in Chinese) [蔡觉平、李赞、宋文涛 2003 物理学报 **52** 1873]
- [21] Waldrop M M 1992 *Complexity: The Emerging Science at the Edge of Order and Chaos* (New York : Simon and Shuster)

Lattice complexity and fine-graining of symbolic sequence^{*}

Ke Da-Guan[†] Zhang Hong Tong Qin-Ye

(*Department of Biomedical Engineering , Zhejiang University , Hangzhou 310027 ,China*)

(Received 17 February 2004 ; revised manuscript received 24 May 2004)

Abstract

A new measure of complexity for finite symbol sequences , named as lattice complexity , is presented , based on Lempel-Ziv complexity and the symbolic dynamics of one-dimensional iterated maps system. To make lattice complexity distinguished from Lempel-Ziv measure , an approach called fine-graining method is also proposed. When fine-graining order is small enough , the two measures are almost equal. When fine-graining order goes to large , the differentiation between them becomes apparent. Applying these measures to studies of logistic map , we find those be regarded as complex sequences by lattice complexity are clearly generated at the edge of the chaotic region. The derived properties of the measures are also discussed.

Keywords : chaos , complexity measure , lattice complexity , fine-graining method

PACC : 0545

^{*} Project supported by the National Natural Science Foundation of China(Grant No.30170267) and the National Pre-project for Key Basic Research Special Foundation of China(Grant No. 2002CCA01800).

[†] E-mail :kedaguan@sohu.com