

# 基于活动标架对 Hilbert 曲线的研究

于延华, 刘 玲, 杨 云  
(东北大学 理学院, 辽宁 沈阳 110819)

**摘 要:** 现有刻画三维 Hilbert 曲线的算法大多是从始点到终点递归地计算节点坐标, 针对此类算法迭代次数较多的问题, 提出一种刻画三维 Hilbert 曲线的新算法. 借助于构造活动标架, 得到刚体运动下的不变量, 即离散曲率挠率. 考虑到活动标架, 曲线节点将被重新编码. 并建立曲线弯曲点位置编号与其对应的曲率挠率数对的映射, 编写相应算法使其对任意编号  $n$ , 能够输出该编号对应弯曲点的曲率挠率数对且画出弯曲点图象结构. 相比于基于 Matlab 生成 Hilbert 曲线的算法 Hilbert3( $n$ ), 该算法不局限于曲线的阶数、不依赖相邻阶曲线节点坐标之间的迭代. 实验结果表明此算法更加高效.

**关 键 词:** 活动标架; Hilbert 曲线; 离散曲率; 离散挠率; 迭代

**中图分类号:** O 158      **文献标志码:** A      **文章编号:** 1005-3026(2019)07-1061-04

## Study of Hilbert Curve Based on Moving Frame

YU Yan-hua, LIU Ling, YANG Yun

(School of Sciences, Northeastern University, Shenyang 110819, China. Corresponding author: YANG Yun, E-mail: freeuse\_st@126.com)

**Abstract:** Most algorithms for describing three-dimensional Hilbert curves calculate node coordinates from the start point to the end point recursively. Directing at the multiple iteration, a new algorithm was brought forth. By means of constructing moving frame, the invariants under rigid body motion are obtained, that is, discrete curvature and torsion. Considering moving frame, the nodes are recoded. Establishing a map between the inflection point location number and the discrete curvature and torsion of the inflection point, based on that, writing the corresponding algorithm to make it for any number  $n$ , the pairs of curvature torsion and the image structure corresponding to the inflection points can be output. Compared to the algorithm Hilbert3( $n$ ), the proposed algorithm is not limited to the order of the curve and does not depend on the iteration between the coordinates. Experimental results show that the algorithm is more efficient.

**Key words:** moving frame; Hilbert curve; discrete curvature; discrete torsion; iteration

Hilbert 曲线源于经典的 Peano 曲线族, 是所有能够填满二维或更高维区域的离散分形<sup>[1]</sup>曲线总称. Hilbert 曲线在图象存储和检索、空间数据库索引<sup>[2]</sup>等领域得到了广泛的应用. 因此空间 Hilbert 曲线的画法已经得到越来越多的关注.

已有刻画三维 Hilbert 曲线的算法一般是通过不断 8 等分一个正方体区域, 从始点到终点递归地计算画线的位置的过程, 因此方向是这些运算要考虑的问题. 由于这些算法大都是对曲线的每条线段逐渐细分做递归运算, 当迭代次数较大时计算非常耗时, 因此有必要提出一个有效减少

迭代次数的算法.

考虑到活动标架<sup>[3-6]</sup>, 将重新编码三维 Hilbert 曲线的节点, 建立曲线弯曲点位置号码与基于活动标架得到的弯曲点曲率挠率数对的映射<sup>[7]</sup>, 不必考虑曲线方向. 通过编写相应的程序, 对任意一个实数  $n$ , 能够输出对应弯曲点的曲率挠率数对并且画出相应弯曲点图象结构.

## 1 Hilbert 曲线的性质

Hilbert 曲线是离散曲线的一种, 有二维三维

甚至更高维的情形,本文只考虑三维. 三维 Hilbert 曲线能够不自交地充满正方体. 经典描述如下:首先将正方体 8 等分,求出各个小正方体的中心,并将它们按照某种顺序依次连接,得到图 1a 所示折线;其次将各个小正方体再细分为 8 个相同的小正方体,并连接各个小正方体的中心,得到图 1b;按照这种方法不断细分下去,并按一定规则一一连接,就可得到图 1 的 Hilbert 曲线.

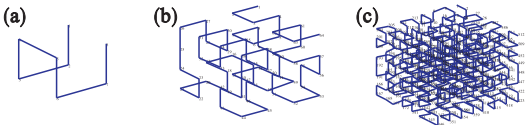


图 1 Hilbert 曲线  
Fig. 1 Hilbert curve

(a)—1 阶 Hilbert 曲线; (b)—2 阶 Hilbert 曲线;  
(c)— $n$  阶 Hilbert 曲线.

第  $n$  阶 Hilbert 曲线与第  $n+1$  阶 Hilbert 曲线是仿射相等的<sup>[8]</sup>. 考虑以 Hilbert 曲线边缘切向量<sup>[9]</sup> $t_{k-2}, t_{k-1}$ 以及  $t_{k-2} \times t_{k-1}$ 为活动标架. 通常地,第  $n$  阶曲线有  $8^n$  个节点,本文中仅使用曲线的弯曲点,即出现 3 个点或 3 个以上点共线的情况,仅标记直线的第一个点和最后一个节点使用,如图 2 所示第 24, 25, 26 点,将去掉第 25 点,将第 26 点标记为 25,按照如上所述的标记方法.



图 2 共线点  
Fig. 2 Collinear point

**定理 1** 第  $n$  阶与第  $n+1$  阶 Hilbert 曲线弯曲点总数  $N(n)$  及  $N(n+1)$  满足:

$$N(n+1) = \begin{cases} 8N(n) - 2, & n = 3k - 2; \\ 8N(n) - 4, & n = 3k - 1; \\ 8N(n) - 8, & n = 3k. \end{cases} \quad (1)$$

$N(1) = 8$ , 经过简单的计算,可得第  $n$  阶 Hilbert 曲线弯曲点总数为

$$N(n) = \begin{cases} 8^n - \frac{168 \times (8^{n-1} - 1)}{511}, & n = 3k - 2; \\ 62 \times 8^{n-2} - \frac{322 \times (8^{n-2} - 1)}{511}, & n = 3k - 1, \quad k \in \mathbf{Z}^+; \\ 492 \times 8^{n-3} - \frac{532 \times (8^{n-3} - 1)}{511}, & n = 3k. \end{cases} \quad (2)$$

## 2 基于活动标架计算离散曲率挠率及其迭代特点

经过重新编码,对于三维 Hilbert 曲线,由于切向量满足  $t_{k-2} \perp t_{k-1} \quad \forall k \in \mathbf{Z}^+, \text{ 且 } k \geq 3$ . 以  $t_{k-2}, t_{k-1}$  及  $t_{k-2} \times t_{k-1}$  为活动标架,有

$$t_k = \kappa_k t_{k-2} + \kappa_k t_{k-1} + \tau_k t_{k-2} \times t_{k-1}.$$

经过简单的推导,有

$$\begin{aligned} \kappa_k &= \frac{-[t_{k-2} \times t_{k-1}, t_{k-1}, t_k]}{[t_{k-2}, t_{k-1}, t_{k-2} \times t_{k-1}]}; \\ \kappa_k &= \frac{[t_{k-2} \times t_{k-1}, t_k, t_{k-2}]}{[t_{k-2} \times t_{k-1}, t_{k-1}, t_{k-2}]}; \\ \tau_k &= \frac{[t_{k-1}, t_k, t_{k-2}]}{[t_{k-2} \times t_{k-1}, t_{k-2}, t_{k-1}]} \end{aligned} \quad (3)$$

第  $3k-1$  阶 Hilbert 曲线 7 组连接点中的第 1, 2, 7 组连接点<sup>[10]</sup>如图 3 所示. 其中图 3a, 3b, 3c 的位置向量分别为  $r_{8^{3k-1} - \frac{168(8^{3k-3}-1)}{511}}, r_{2(8^{3k-1} - \frac{168(8^{3k-3}-1)}{511})}, r_{7(8^{3k-1} - \frac{168(8^{3k-3}-1)}{511}) - 2}$ .

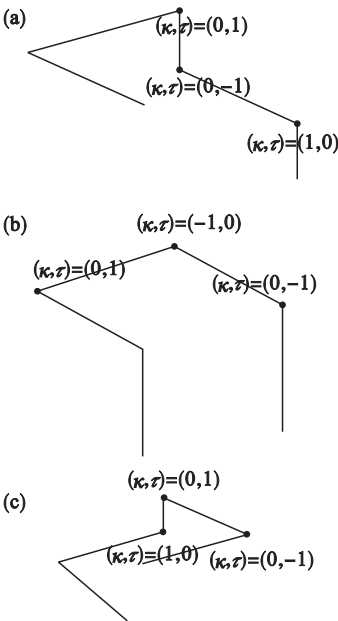


图 3 从第  $3k-2$  阶迭代到  $3k-1$  阶的部分连接点  
Fig. 3 Part of joints of iterative process from the  $3k-2$ nd to the  $3k-1$  step

(a)—第 1 组连接点; (b)—第 2 组连接点;  
(c)—第 7 组连接点.

同理从第  $3k-1$  阶迭代到第  $3k$  阶及从第  $3k$  阶迭代到  $3k+1$  阶分别产生 7 组类似的连接点.

**定理 2** 空间 Hilbert 曲线在每个弯曲点的离散曲率和挠率满足

$$(\kappa, \tau) \in \{(-2, 0), (-1, 0), (0, -1), (0, 1), (1, 0), (2, 0), \left(0, \frac{1}{2}\right), \left(-\frac{1}{2}, 0\right), (-2, 0), \left(0, -\frac{1}{2}\right), \left(\frac{1}{2}, 0\right), (0, 2), (-3, 0), \left(-\frac{1}{3}, 0\right)\},$$

$$\kappa=0$$

记

$$A = \{(0, -1), (-1, 0), (0, 1)\},$$

$$B_2 = \{(-1, 0), (0, 1), (0, -1), (1, 0), (-1, 0)\},$$

$$C_2 = \{(-1, 0), (0, 1), (-1, 0), (0, -1), (-1, 0)\},$$

$$D_2 = \{(-1, 0), (2, 0), \left(0, \frac{1}{2}\right), \left(-\frac{1}{2}, 0\right)\},$$

$$E_2 = \{(-2, 0), \left(0, -\frac{1}{2}\right), \left(\frac{1}{2}, 0\right), (-1, 0)\},$$

$$F_2 = \{(-1, 0), (1, 0), (0, 1), (0, -1), (-1, 0)\},$$

$$C_3 = \{(-1, 0), (1, 0), (-1, 0), (1, 0), (-1, 0)\},$$

$$D_3 = \{(-2, 0), (-1, 0), \left(0, -\frac{1}{2}\right), (-1, 0)\},$$

$$E_3 = \{(-1, 0), (0, 2), (-1, 0), \left(-\frac{1}{2}, 0\right)\},$$

$$C_4 = \{(-3, 0), (-1, 0), \left(-\frac{1}{3}, 0\right)\}.$$

可以用下面的方式展示 Hilbert 曲线曲率挠率序列的迭代结果:

$$n=1 \text{ 时, } (-1, 0)A(-1, 0);$$

$$n=2 \text{ 时, } (-1, 0)AB_2AC_2AD_2AC_2AE_2AC_2AF_2A(-1, 0),$$

如果用  $K_2$  表示第一个  $(-1, 0)$  和最后一个  $(-1, 0)$  的中间曲率挠率数对序列, 即

$$K_2 = AB_2AC_2AD_2AC_2AE_2AC_2AF_2A.$$

从而

$$K_3 = (-1, 0)K_2D_2K_2C_3K_2D_3K_2C_3K_2E_3K_2C_3K_2E_2K_2(-1, 0).$$

很容易得到如下的图式规律:

$$K_{3k-1} = (-1, 0)K_{3k-2}B_2K_{3k-2}C_2K_{3k-2}D_2K_{3k-2}C_2K_{3k-2}E_2K_{3k-2}C_2K_{3k-2}F_2K_{3k-2}(-1, 0),$$

$$K_{3k} = (-1, 0)K_{3k-1}D_2K_{3k-1}C_3K_{3k-1}D_3K_{3k-1}C_3K_{3k-1}E_3K_{3k-1}C_3K_{3k-1}F_3K_{3k-1}(-1, 0),$$

$$K_{3k+1} = (-1, 0)K_{3k}D_3K_{3k}C_4K_{3k}B_2K_{3k}C_4K_{3k}F_2K_{3k}C_4K_{3k}D_2K_{3k}(-1, 0).$$

曲率挠率数对构成的字母序列的长度满足

$$y(n) = 16 \times 8^{n-2} + 1. \quad (4)$$

### 3 弯曲点曲率挠率位置特点

首先计算  $3k-1$  阶曲线弯曲点的离散曲率和挠率, 显然由  $3k-2$  阶曲线可以得到以第 3 个点为起点以第  $8^{3k-3} - \frac{168 \times (8^{3k-3} - 1)}{511} - 1$  个点为终点的部分 Hilbert 曲线的每个弯曲点的曲率和挠率数对

$$(\kappa(3), \tau(3)), (\kappa(4), \tau(4)), \dots$$

$$\left(\kappa\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right), \tau\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right)\right).$$

如图 3 所示, 已经得到第  $3k-1$  阶的 Hilbert 曲线连接点的曲率挠率数对, 且  $3k-1$  阶曲线连接点以外的其他弯曲点满足如下等式:

$$\left(\kappa\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} - 5\right), \tau\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} - 5\right)\right) = (-1, 0);$$

$$\left(\kappa\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} - 4\right), \tau\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} - 4\right)\right) = (0, -1);$$

...

$$\left(\kappa\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} + 2\right), \tau\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} + 2\right)\right) = (1, 0);$$

$$\left(\kappa\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} + l\right), \tau\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511} + l\right)\right) = (\kappa(l), \tau(l));$$

$$\left(\kappa\left(2\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right)\right), \tau\left(2\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right)\right)\right) = (0, 1);$$

$$\left(\kappa\left(2\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right) + 1\right), \tau\left(2\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right) + 1\right)\right) = (-1, 0);$$

$$\left(\kappa\left(2\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right) + 2\right), \tau\left(2\left(8^{3k-2} - \frac{168 \times (8^{3k-3} - 1)}{511}\right) + 2\right)\right) = (0, -1);$$

$$\left(\kappa\left(2\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)+l\right),\tau\left(2\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)+l\right)\right)=$$
$$(\kappa(l),\tau(l));$$
$$\dots$$
$$\left(\kappa\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-2\right),\tau\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-2\right)\right)=(1,0);$$
$$\left(\kappa\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-1\right),\tau\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-1\right)\right)=(0,1);$$
$$\left(\kappa\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)\right),\tau\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)\right)\right)=(0,-1);$$
$$\left(\kappa\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-2+l\right),\tau\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-2+l\right)\right)=$$
$$(\kappa(l),\tau(l)),$$

其中, $l=3,4,\cdots,8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}-1$ .

因此容易通过  $3k-2$  阶曲线的弯曲点的曲率  
挠率得到  $3k-1$  阶曲线全部弯曲点的曲率挠率。  
类似地用上述方法,可以分别得到第  $3k$  阶和  
第  $3k+1$  阶的曲率挠率数对。

### 4 算法分析

Hilbert3( $n$ ) 是计算  $n$  阶 Hilbert 曲线上所有  
点三维坐标的经典算法,在此基础上编写计算曲  
线任意弯曲点曲率挠率的算法,并将此算法与本  
文算法进行比较,见表 1。

表 1 算法计算时间比较				
Table 1 Comparison of calculation time of different algorithm				
算法	$n$			
	100	1 000	10 000	100 000
本文算法	2. 15	2. 13	3. 94	4. 00
Hilbert3( $n$ )	0. 81	1. 25	11. 41	519. 56

算法速度提高的主要原因是基于三维 Hilbert  
曲线弯曲点离散曲率挠率分布特点有效减少了迭  
代次数。以  $k=1,l=3$  为例,由

$$\left(\kappa\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-2+l\right),\tau\left(7\left(8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}\right)-2+l\right)\right)=$$
$$(\kappa(l),\tau(l)),l=3,4,\cdots,8^{3k-2}-\frac{168\times(8^{3k-3}-1)}{511}-1.$$

可知, $(\kappa(57),\tau(57))$  的值可以通过直接  
 $(\kappa(3),\tau(3))$  获得,而不必通过计算前面 57 个  
弯曲点的三维坐标来获得 $(\kappa(57),\tau(57))$ ,随着  
 $k$  的增大,减少的迭代次数是呈指数增长的,因此  
有效提高了刻画 Hilbert 曲线任意弯曲点曲率挠  
率的效率。

### 5 结 语

针对刻画三维 Hilbert 曲线这个传统问题进  
行研究,给出了一套对于任意实数  $n$ ,快速输出  
Hilbert 曲线第  $n$  个弯曲点的曲率挠率数对,并画  
出其相应的图象结构的算法。基于对三维 Hilbert  
的深入分析,在自相似特征中基于活动标架归纳  
出 Hilbert 曲线弯曲点的曲率挠率数对分布特点,  
最后提出了基于该思想的输出 Hilbert 曲线任意  
弯曲点曲率挠率及该点处图象结构的算法。

### 参考文献:

[1] Olver P J. Modern developments in the theory applications of moving frames[EB/OL]. (2015-03-06). [https://www.lms.ac.uk/sites/lms.ac.uk/files/olver\\_l1\\_final\\_151212.pdf](https://www.lms.ac.uk/sites/lms.ac.uk/files/olver_l1_final_151212.pdf).

[2] Yang Y, Yu Y H. The moving frame on the fractal curves[EB/OL]. (2016-12-16)[2018-05-03]. <https://arxiv.org/abs/1612.05341v1>.

[3] Yang Y, Yu Y H. Moving frame and integrable system of the discrete centroaffine curves in  $R^3$ [EB/OL]. (2016-11-27)[2018-05-03]. <https://arxiv.org/abs/1601.06530v2>.

[4] Mansfield E, MariBeffa G, Wang J P. Discrete moving frames and discrete integrable systems[J]. *Foundations of Computational Mathematics*, 2013, 13: 545-582.

[5] Hu N. Centroaffine space curves with constant curvatures and homogeneous surfaces[J]. *Journal of Geometry*, 2011, 102: 103-114.

[6] 文志英. 分形几何理论与应用[M]. 杭州: 浙江科学技术出版社, 1998: 1-31.  
(Wen Zhi-ying. Theory and application of fractal geometry[M]. Hangzhou: Zhejiang Science and Technology Press, 1998: 1-31.)

[7] Christian B, Stefan B, Daniel A K. Searching in high dimensional space index structures for improving the performance of multimedia databases[J]. *ACM Computing Surveys(CSUR)*, 2001, 33(3): 322-373.

[8] 梅向明, 刘增贤, 王汇淳, 等. 高等几何[M]. 北京: 高等教育出版社, 2008: 1-85.  
(Mei Xiang-ming, Liu Zeng-xian, Wang Hui-chuan, et al. Advanced geometry[M]. Beijing: Higher Education Press, 2008: 1-85.)

[9] Yang Y. The frenet-serret formula of a discrete centroaffine curve[EB/OL]. (2016-11-27)[2018-05-15]. <https://arxiv.org/abs/1601.06530v1>.

[10] Olver P J. Joint invariant signatures[J]. *Foundations of Computational Mathematics*, 2001, 1(1): 3-68.