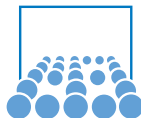


# Algorithms of Scientific Computing

## Space-Filling Curves and their Applications in Scientific Computing II

Tobias Neckel, Dirk Pflüger

Summer Term 2010



# Space-filling Curves – Algorithms

**Traversal** of  $h$ -indexed objects:

- given a set of objects with “positions”  $p_i \in \mathcal{Q}$
- traverse all objects, such that  $\bar{h}^{-1}(p_{i_0}) < \bar{h}^{-1}(p_{i_1}) < \dots$

**Compute mapping:**

- for a given index  $t \in \mathcal{I}$ , compute the image  $h(t)$

**Compute the index** of a given point:

- given  $p \in \mathcal{Q}$ , find a parameter  $t$ , such that  $h(t) = p$
- problem: inverse of  $h$  is not unique ( $h$  not bijective!)
- define a “technically unique” inverse mapping  $\bar{h}^{-1}$

# Arithmetic Formulation of the Hilbert Curve

## Idea:

- interval sequence within the parameter interval  $\mathcal{I}$  corresponds to a **quaternary representation**; e.g.:

$$\left[\frac{1}{4}, \frac{2}{4}\right] = [0_4.1, 0_4.2], \quad \left[\frac{3}{4}, 1\right] = [0_4.3, 1_4.0]$$

- self-similarity**: every subsquare of the target domain contains a scaled, translated, and rotated/reflected Hilbert curve.

⇒ **Construction** of the arithmetic representation:

- find quaternary representation of the parameter
- use quaternary coefficients to determine the required sequence of operations

# Arithmetic Formulation of the Hilbert Curve (2)

## Reversive approach:

$$h(0_4.q_1q_2q_3q_4\dots) = H_{q_1} \circ h(0_4.q_2q_3q_4\dots)$$

- $\tilde{t} = 0_4.q_2q_3q_4\dots$  is the relative parameter in the subinterval  $[0_4.q_1, 0_4.(q_1 + 1)]$
- $h(\tilde{t}) = h(0_4.q_2q_3q_4\dots)$  is the relative position of the curve point in the subsquare
- $H_{q_1}$  transforms  $h(\tilde{t})$  to its correct position in the unit square:
  - rotation
  - translation
- expanding the recursion equation leads to:

$$h(0_4.q_1q_2q_3q_4\dots) = H_{q_1} \circ H_{q_2} \circ H_{q_3} \circ H_{q_4} \circ \dots$$

## Arithmetic Formulation of the Hilbert Curve (3)

If  $t$  is given in quaternary digits, i.e.  $t = 0_4.q_1q_2q_3q_4\dots$ , then  $h(t)$  may be represented as

$$h(0_4.q_1q_2q_3q_4\dots) = H_{q_1} \circ H_{q_2} \circ H_{q_3} \circ H_{q_4} \circ \dots \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

using the following operators:

$$H_0 := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} \frac{1}{2}y \\ \frac{1}{2}x \end{pmatrix} \quad H_1 := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} \frac{1}{2}x \\ \frac{1}{2}y + \frac{1}{2} \end{pmatrix}$$

$$H_2 := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} \frac{1}{2}x + \frac{1}{2} \\ \frac{1}{2}y + \frac{1}{2} \end{pmatrix} \quad H_3 := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} -\frac{1}{2}y + 1 \\ -\frac{1}{2}x + \frac{1}{2} \end{pmatrix}$$

# Matrix Form of the Operators $H_0, \dots, H_3$

In matrix notation, the operators  $H_0, \dots, H_3$  are:

$$H_0 := \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad H_1 := \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix}$$

$$H_2 := \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \quad H_3 := \begin{pmatrix} 0 & -\frac{1}{2} \\ -\frac{1}{2} & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix}$$

Governing operations:

- scale with factor  $\frac{1}{2}$
- translate start of the curve, e.g.  $+ \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix}$
- reflect at  $x$  and  $y$  axis (for  $H_3$ )

# A First Comment Concerning Uniqueness

## Question:

Are the values  $h(t)$  independent of the choice of quaternary representation of  $t$  concerning trailing zeros:

$$h(0_4.q_1 \dots q_n) = h(0_4.q_1 \dots q_n000\dots),$$

## Outline of the proof:

1. compute the limit  $\lim_{n \rightarrow \infty} H_0^n$ , or  $\lim_{n \rightarrow \infty} H_0^n \begin{pmatrix} x \\ y \end{pmatrix}$ ;

$$\text{Result: } \lim_{n \rightarrow \infty} H_0^n \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

2. show:  $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$  is a fixpoint of  $H_0$ , i. e.  $H_0 \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ .

$\Rightarrow$  independence of trailing zeros, as  $H_{q_n}$  is applied to the fixpoint!

# A Second Comment Concerning Uniqueness

## Question:

Are the values  $h(t)$  independent of the choice of quaternary representation of  $t$ , as in:

$$h(0_4.q_1 \dots q_n) = h(0_4.q_1 \dots q_{n-1}(q_n - 1)333\dots), \quad q_n \neq 0$$

(if  $q_n = 0$ , then consider  $0_4.q_1 \dots q_n = 0_4.q_1 \dots q_{n-1}$ )

## Outline of the proof:

1. compute the limits  $\lim_{n \rightarrow \infty} H_0^n$  and  $\lim_{n \rightarrow \infty} H_3^n$ .
2. for  $q_n = 1, 2, 3$ , show that

$$H_{q_n} \circ \lim_{n \rightarrow \infty} H_0^n = H_{q_{n-1}} \circ \lim_{n \rightarrow \infty} H_3^n$$



# Algorithm to Compute the Hilbert Mapping

**Task:** given a parameter  $t$ , find  $h(t) = (x, y) \in \mathcal{Q}$

**Most important subtasks:**

1. compute quaternary digits – use multiply by 4:

$$4 \cdot 0_4.q_1q_2q_3q_4 \dots = (q_1.q_2q_3q_4 \dots)_4$$

and cut off the integer part

2. apply operators  $H_q$  in the correct sequence – use recursion:

$$h(0_4.q_1q_2q_3q_4 \dots) = H_{q_1} \circ H_{q_2} \circ H_{q_3} \circ H_{q_4} \circ \dots \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

3. stop recursion, when a given tolerance is reached  
 $\Rightarrow$  track size of interval or set number of digits

# Computing the Inverse Mapping

**Task:** find a parameter  $t$ , such that  $h(t) = (x, y)$  for a given  $(x, y) \in \mathcal{Q}$

**Problem:**  $h$  not bijective; hence,  $t$  is not unique

⇒ a strict inverse mapping  $h^{-1}$  does not exist

⇒ instead, compute a “technically unique” inverse  $\bar{h}^{-1}$

**Recursive Idea:**

- determine the subsquare that contains  $(x, y)$
- transform (using the inverse operations of  $H_0, \dots, H_3$ ) the point  $(x, y)$  into the original domain  $\rightarrow (\tilde{x}, \tilde{y})$
- recursively compute a parameter  $\tilde{t}$  that is mapped to  $(\tilde{x}, \tilde{y})$
- depending on the subsquare, compute  $t$  from  $\tilde{t}$

## Inverse Operators of $H_0, \dots, H_3$

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} = H_0 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \frac{1}{2}y \\ \frac{1}{2}x \end{pmatrix} \Rightarrow \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2\tilde{y} \\ 2\tilde{x} \end{pmatrix}$$

By similar computations:

$$H_0^{-1} := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} 2y \\ 2x \end{pmatrix} \quad H_1^{-1} := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} 2x \\ 2y - 1 \end{pmatrix}$$

$$H_2^{-1} := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} 2x - 1 \\ 2y - 1 \end{pmatrix} \quad H_3^{-1} := \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} -2y + 1 \\ -2x + 2 \end{pmatrix}$$

# Algorithm to Compute the Inverse Mapping

$$\bar{h}^{-1} := \text{proc}(x, y)$$

- (1) determine the subsquare  $q \in \{0, \dots, 3\}$  by checking  $x \lessgtr \frac{1}{2}$  and  $y \lessgtr \frac{1}{2}$ :

1	2
0	3

(treat cases  $x, y = \frac{1}{2}$  in a unique way: either  $<$  or  $>$   
 $\Rightarrow$  *technically unique inverse*)

- (2) set  $(\tilde{x}, \tilde{y}) := H_q^{-1}(x, y)$   
 (3) recursively compute  $\tilde{t} := \bar{h}^{-1}(\tilde{x}, \tilde{y})$   
 (4) return  $t := \frac{1}{4}(q + \tilde{t})$  as value

(stopping criterion still to be added)