Space-filling Curve Algorithm for the Traveling Salesman Problem

author

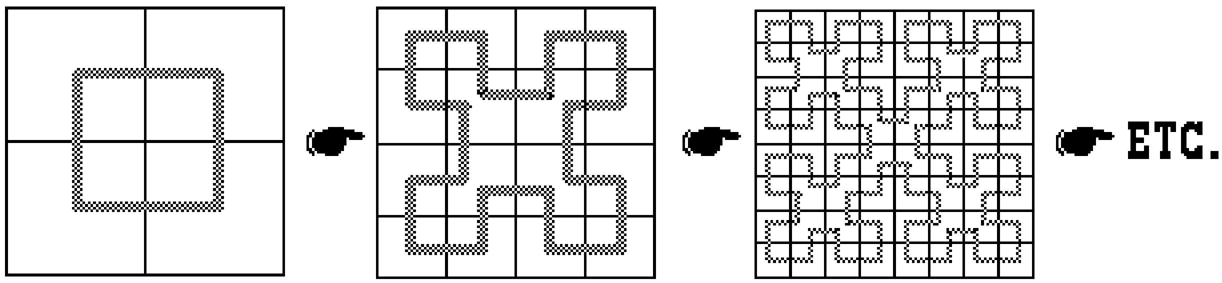
This Hypercard stack was prepared by:
Dennis L. Bricker,
Dept. of Industrial Engineering,
University of Iowa,
Iowa City, Iowa 52242
e-mail: dennis-bricker@uiowa.edu

Spacefilling Curve

- a continuous mapping of the unit interval [0,1] onto a unit hypercube (for example, a square)
- first introduced by Peano (1891), Hilbert (1891), and others as "topological monsters"
- tend to preserve "nearness" among points:
  - if 2 points are close on the curve, they are close in the square
  - if 2 points are close in the square, they are "likely" to be close on the curve.

Examples...

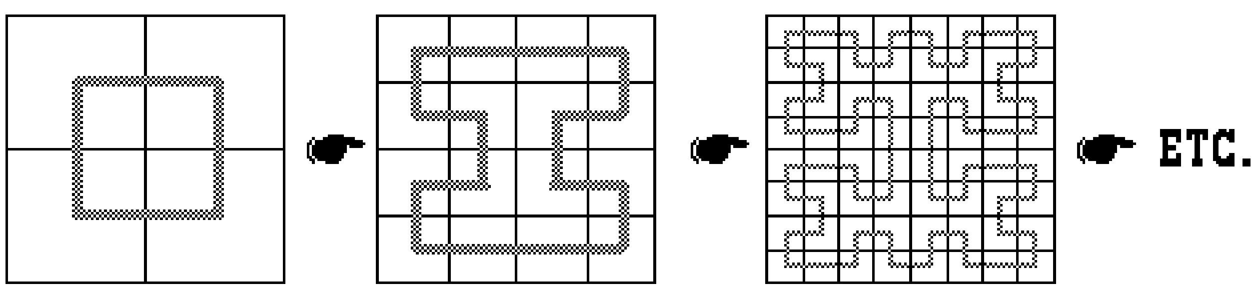
A:



Each curve is the limit of a sequence of recursive constructions.

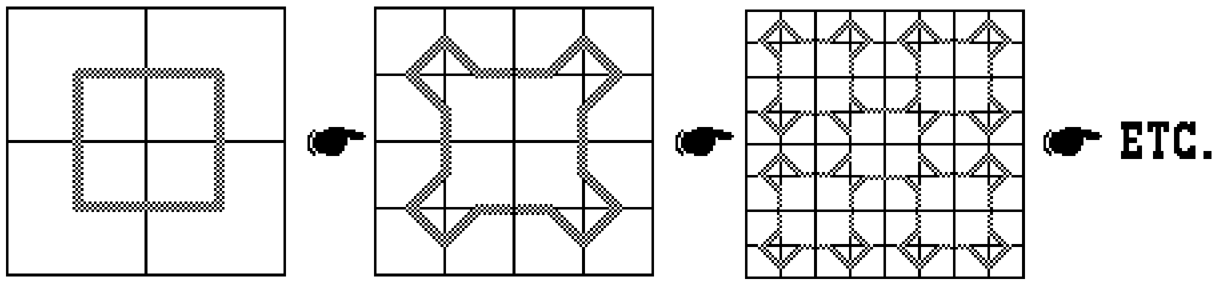
Examples...

B:



Each curve is the limit of a sequence of recursive constructions.

Examples...

C: 

Each curve is the limit of a sequence of recursive constructions.

"Generic" Spacefilling Curve Heuristic:

1. Transform the problem (routing, location, etc.) in the plane, via a spacefilling curve, to a problem on the unit interval.
2. Solve the (easier) problem on the unit interval.
3. Map the solution back to the problem in the plane.

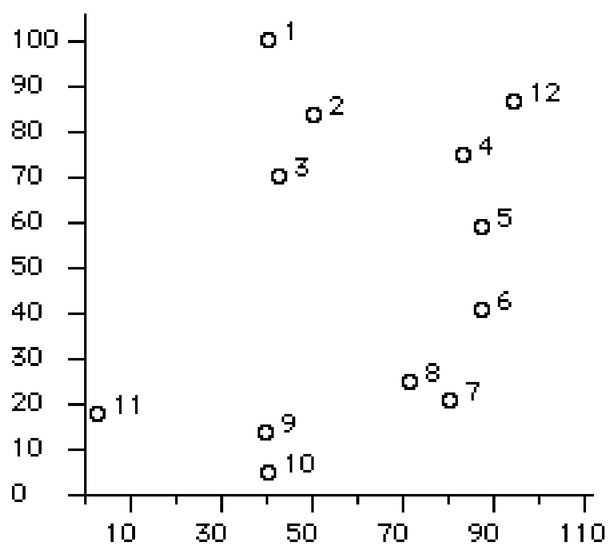
Pseudo-code to compute position on the space-filling curve (A) of a point (x,y) in the unit square:

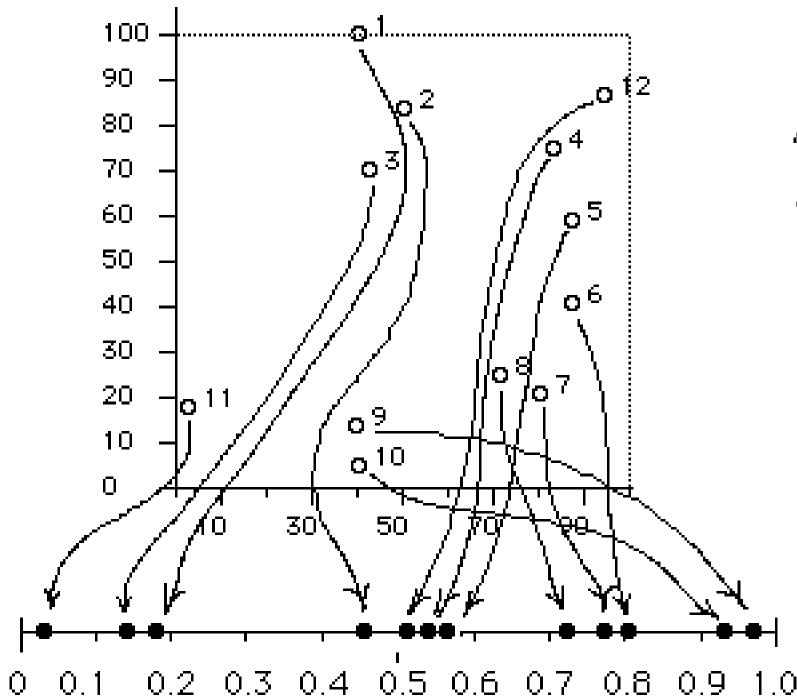
```
∇z←POSITION xy;x;y;Q;T;ij
 ⍝
 ⍝          Return position of (x,y) on a
 ⍝              space-filling curve
 ⍝           (cf. Bartholdi & Platzman)
 ⍝          (Recursively-defined function)
 →Compute IF eps<⌈/|xy-1 1       ⋄ z←0.5    ⋄ →End

Compute:x←xy[1] ⋄ y←xy[2]
ij←(1⌊⌊2×x),(1⌊⌊2×y)
⍝ Q is quadrant in which (x,y) lies
Q←¯1+((4 2⍴0 0 0 1 1 1 1 0)∧.=ij)⍳1
T←POSITION (2×|xy-0.5)
→L1 IF 1≠Q|2  ⋄  T←1-T
L1:z←(7÷8)+(Q+T)÷4  ⋄ z←z-⌊z
End:
∇
```

APL Code

Example

Random Symmetric TSP
(seed= 133398)





The square is mapped into the interval [0, 1]

Coordinate on Space-filling Curve

i	Xi	Yi	Position
1	41	100	0.18236921
2	51	84	0.45818231
3	43	70	0.13481542
4	84	75	0.52611921
5	88	59	0.55589939
6	88	41	0.80589939
7	81	21	0.77961296
8	72	25	0.71850459
9	40	14	0.96073832
10	41	5	0.93535641
11	3	18	0.02027783
12	95	87	0.51850270

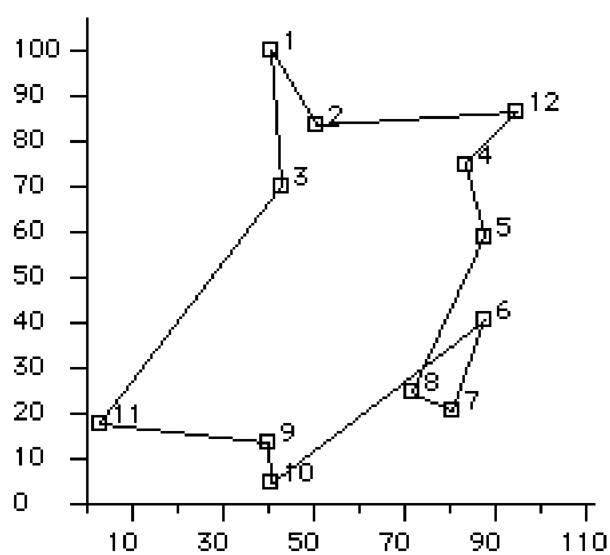
Sort the points:

i	Xi	Yi	Position
1	41	100	0.18236921
2	51	84	0.45818231
3	43	70	0.13481542
4	84	75	0.52611921
5	88	59	0.55589939
6	88	41	0.80589939
7	81	21	0.77961296
8	72	25	0.71850459
9	40	14	0.96073832
10	41	5	0.93535641
11	3	18	0.02027783
12	95	87	0.51850270

i	Position
11	0.02027783
3	0.13481542
1	0.18236921
2	0.45818231
12	0.51850270
4	0.52611921
5	0.55589939
8	0.71850459
7	0.77961296
6	0.80589939
10	0.93535641
9	0.96073832

This determines the order in which cities are visited...

Space-filling-curve tour: 1 2 12 4 5 8 7 6 10 9 11 3 1
with length 365



The space-filling curve has been suggested as a "low-technology" algorithm...

1) Find location on map and read (x,y) coordinates
2) Find (x,y) in table giving position
3) Prepare card with (x,y) and position, and insert into card file according to the position
4) The sorted cards give a tour.

"A Minimal Technology Routing System for Meals on Wheels", Interfaces, June 1983 (Volume 13, no. 2), by J. Bartholdi et al.