On Space Filling Curves

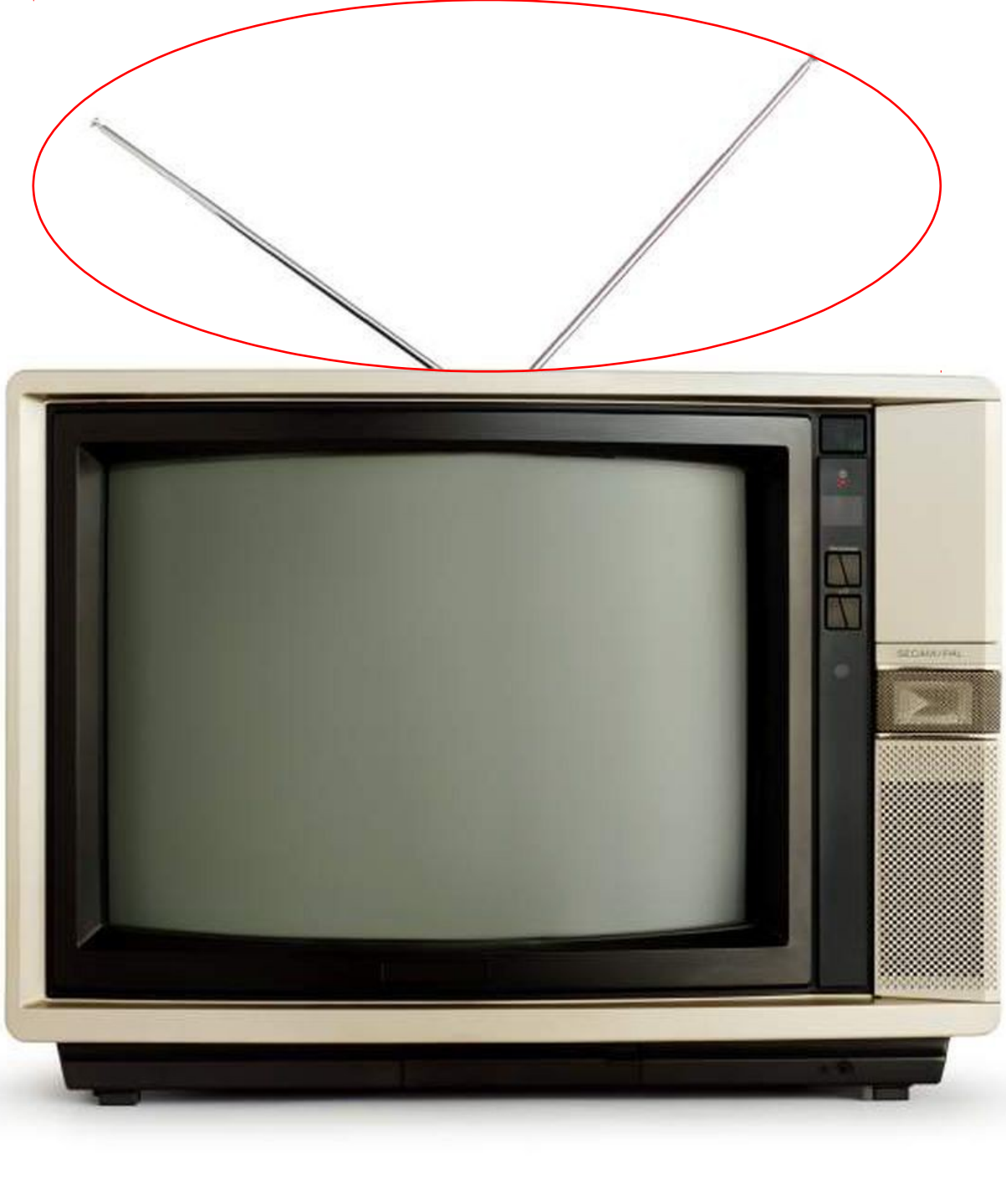
M. Berk Turgut

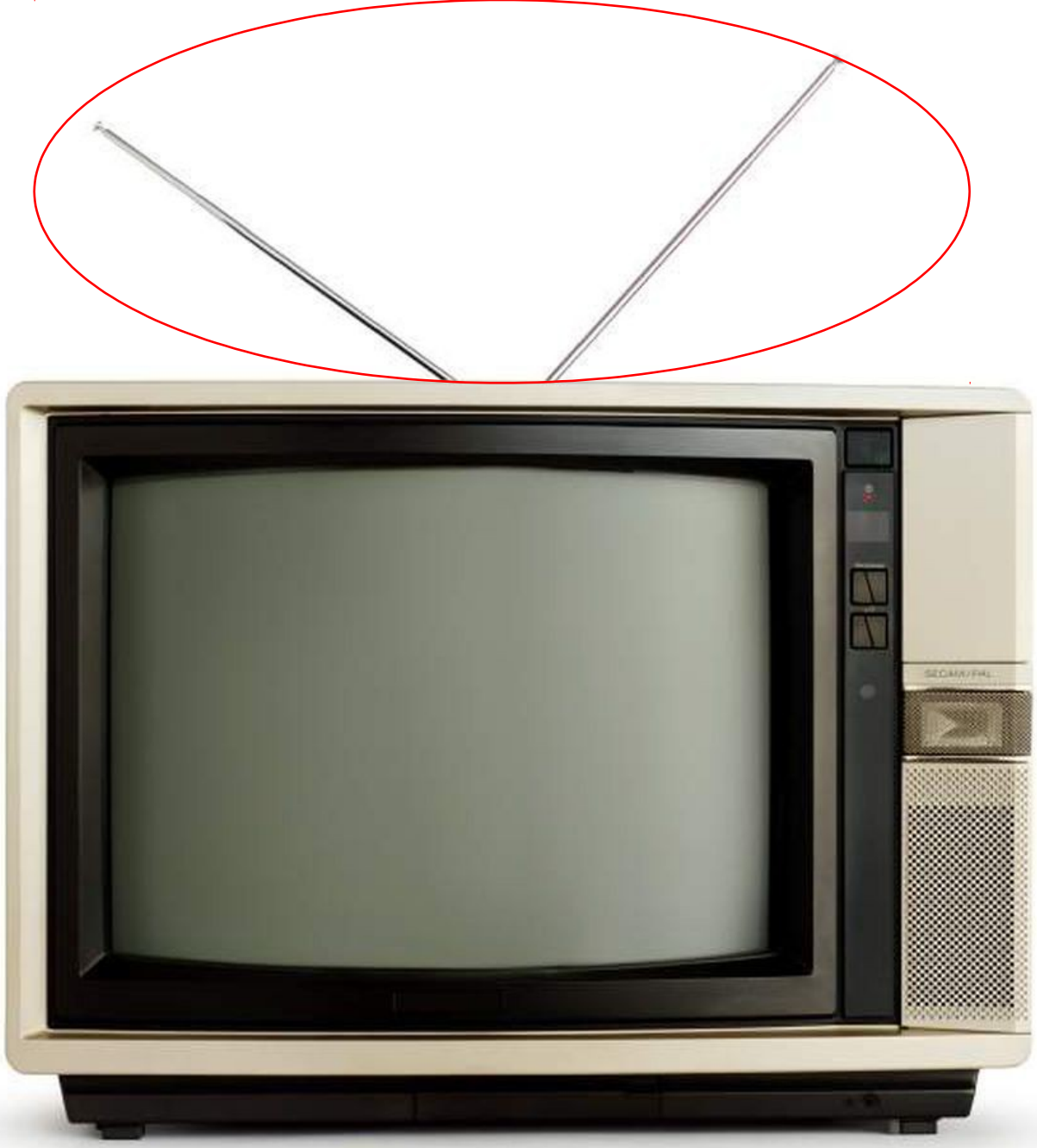
CmpE220 Section II

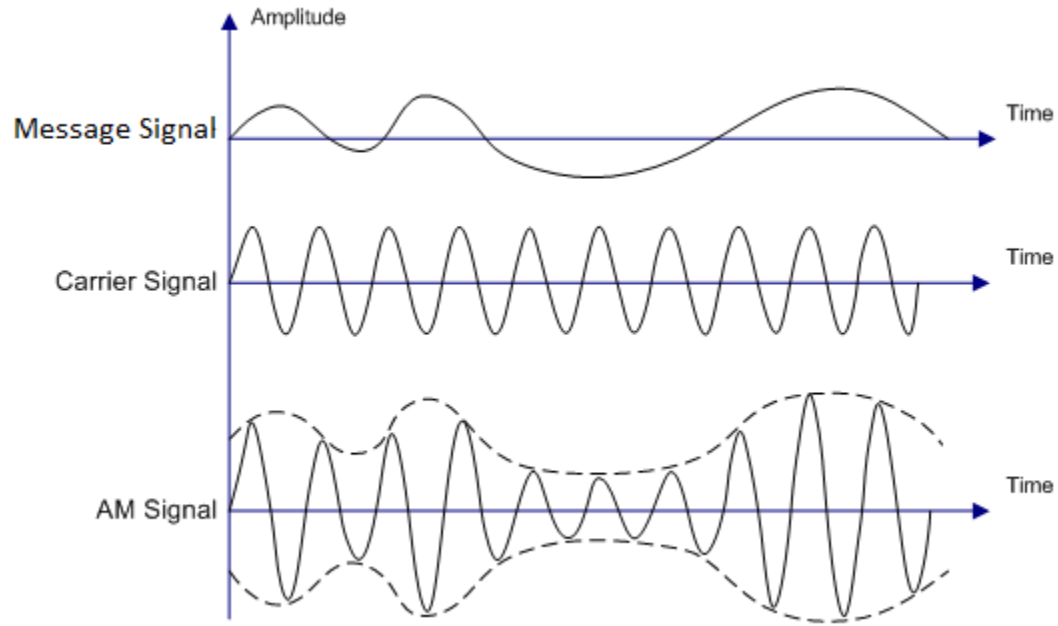
2020 Fall – 20.01.2021

$N > 2^N$?

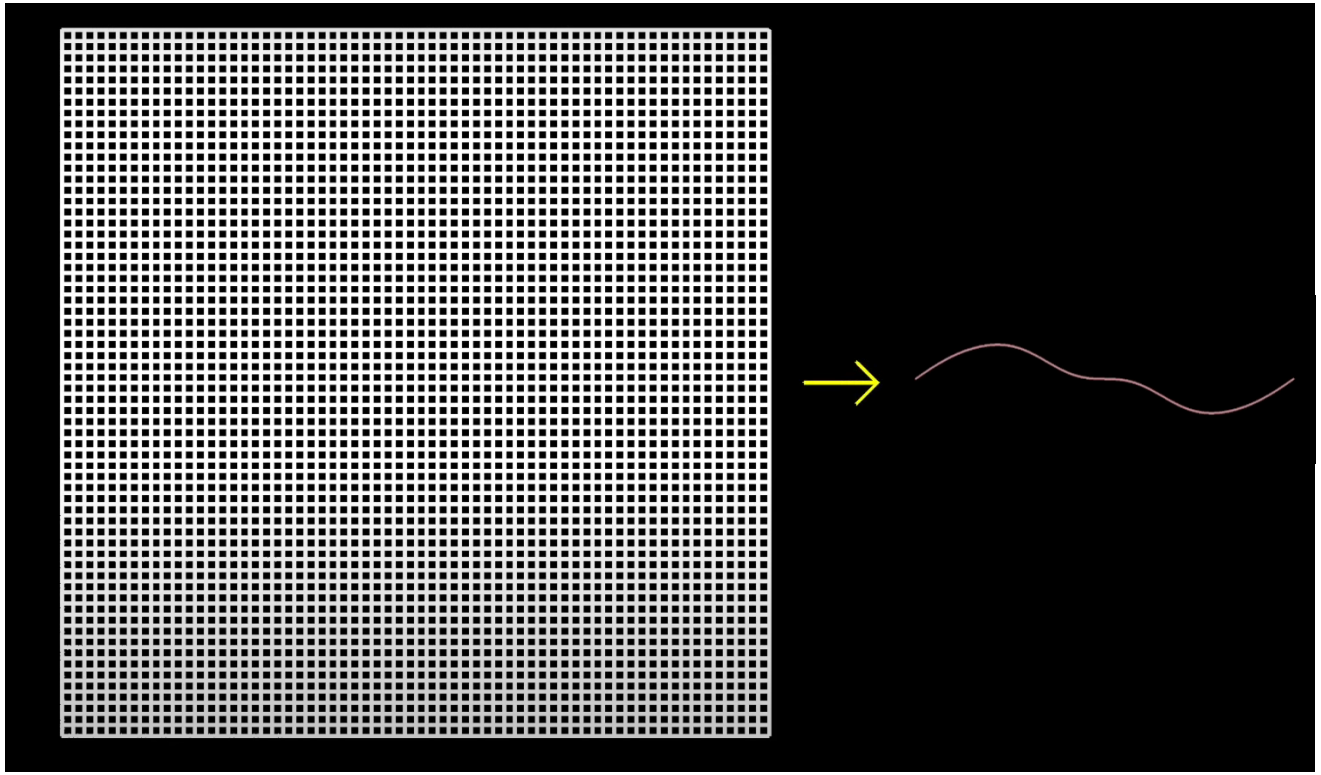
$N >$?



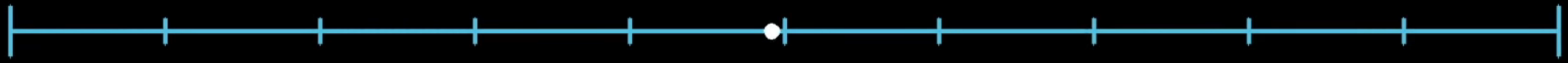



Amplitude
Message Signal
Time
Carrier Signal
Time
AM Signal
Time

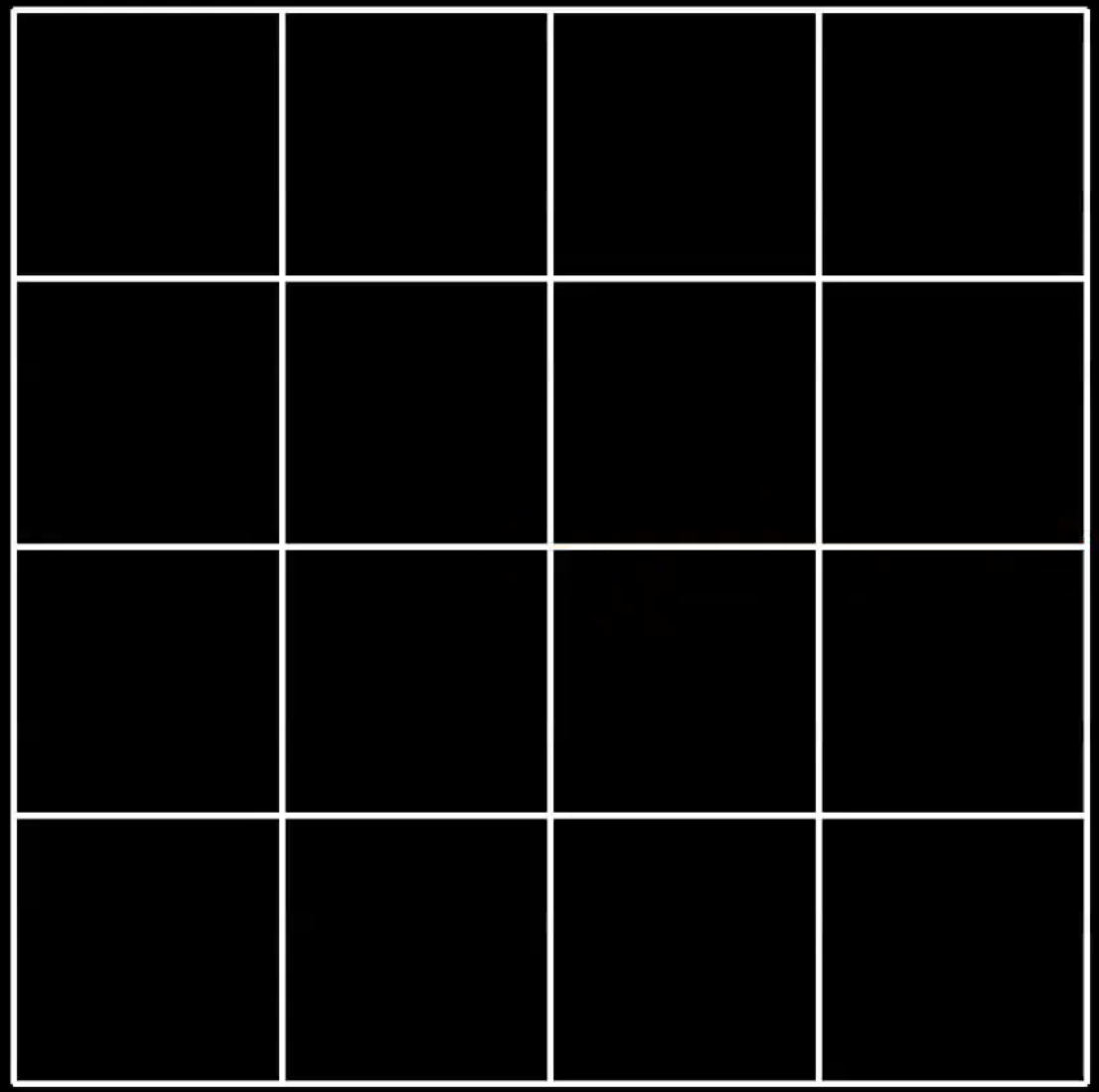
2D → 1D?



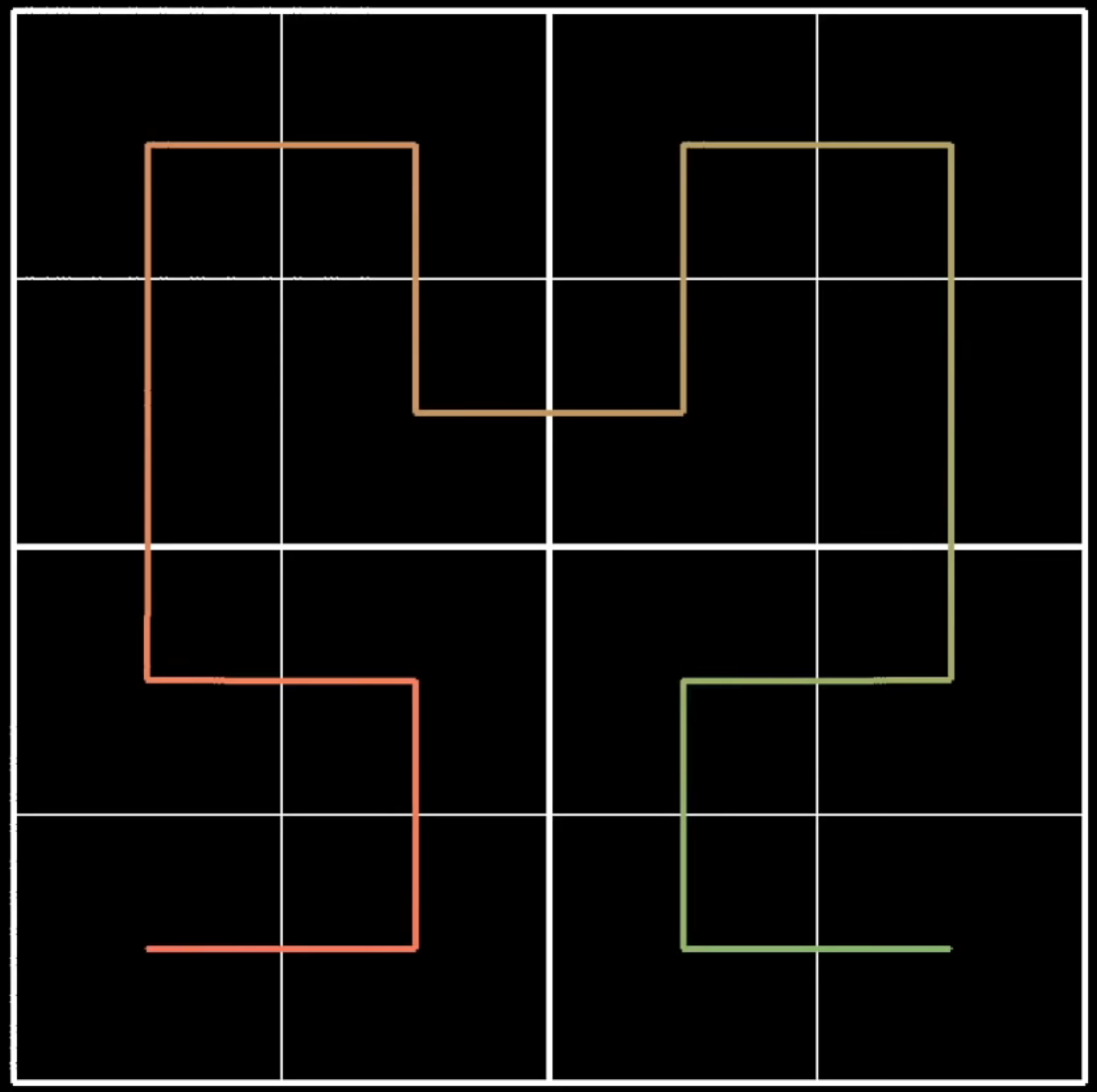
Credit: 3b1b

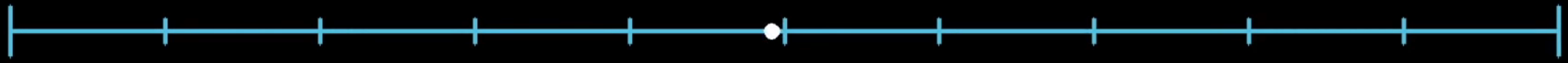


”Pseudo” – Hilbert Curve



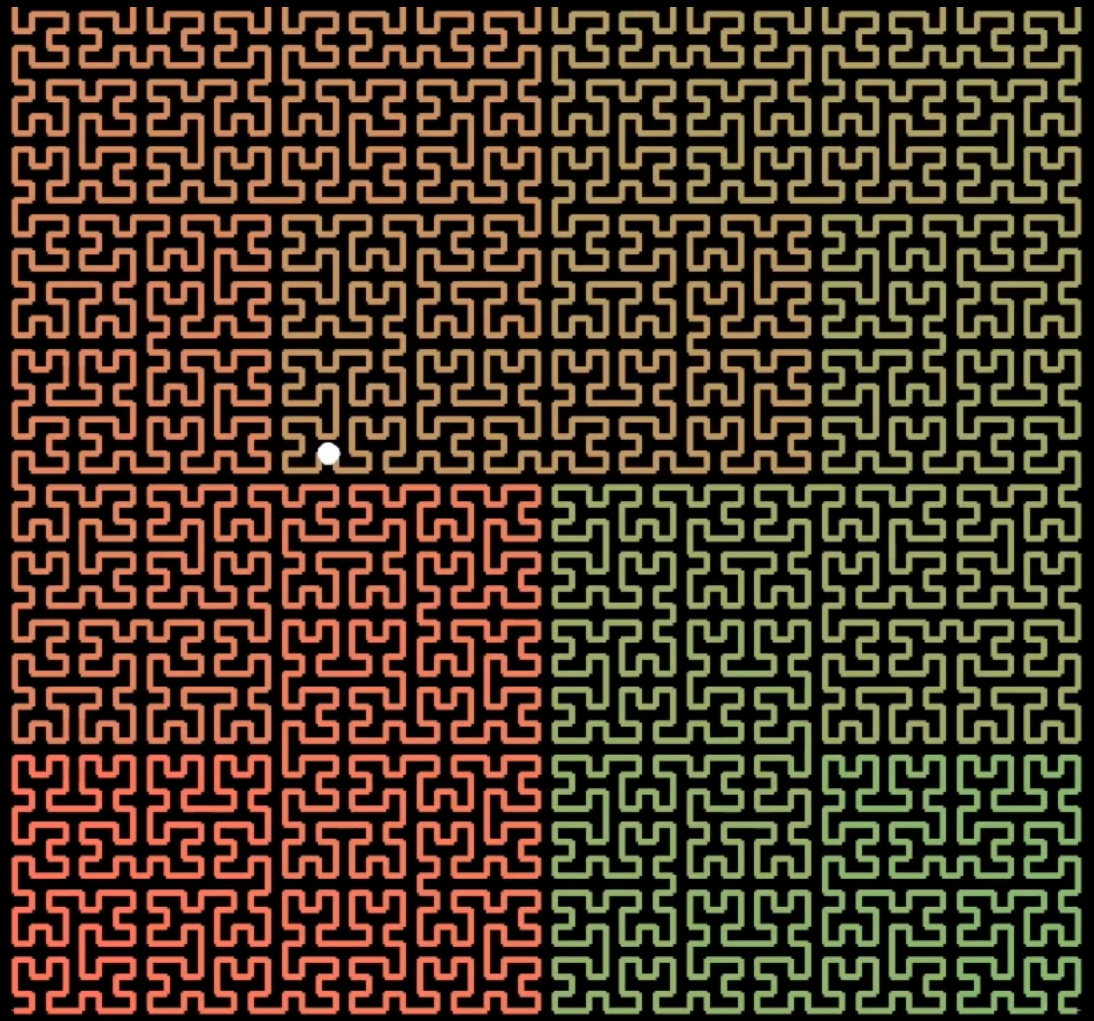
Order 2 Pseudo-Hilbert Curve





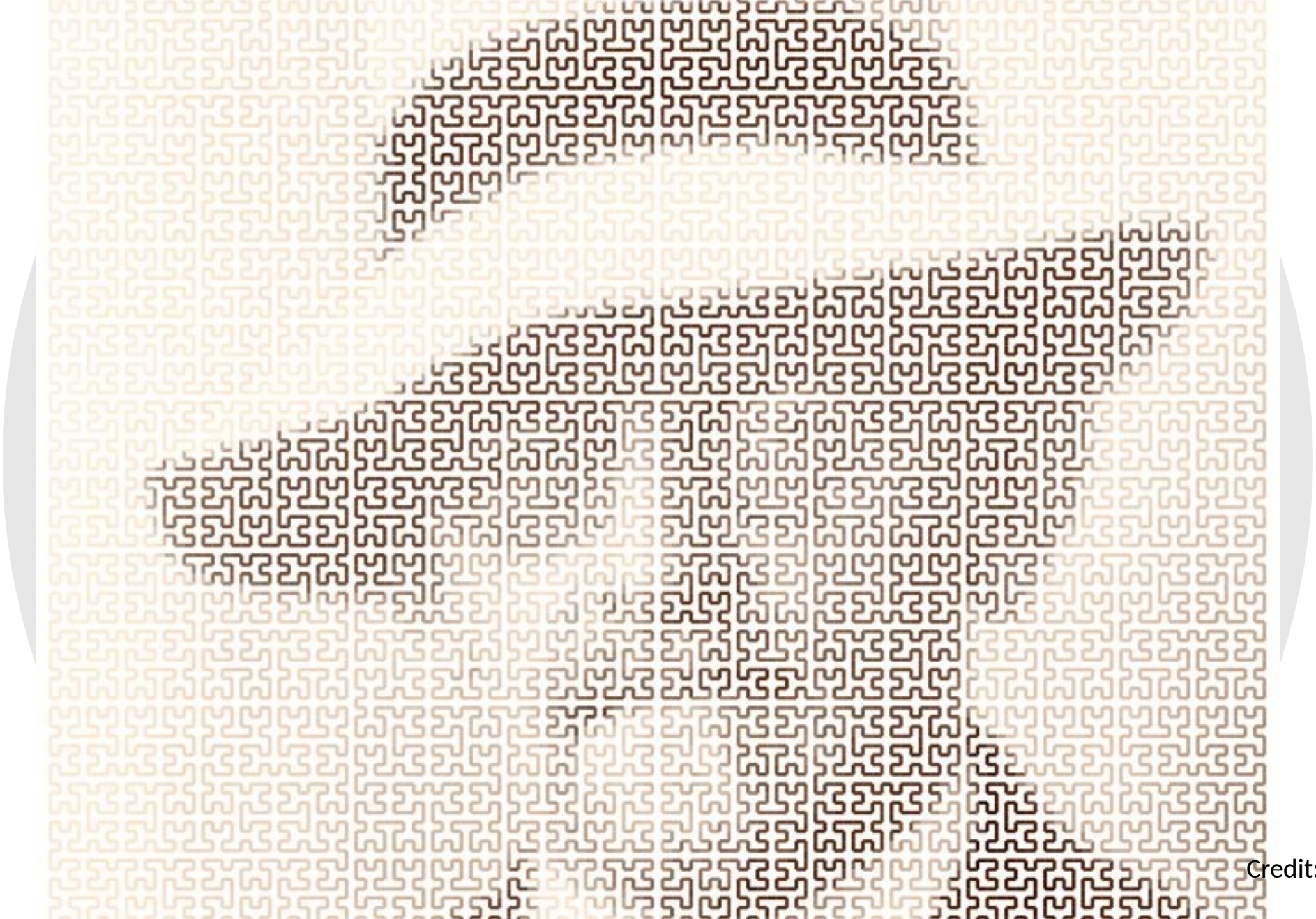
$\mathrm{PHC}_n(x)$ has a limit point as $n \to \infty$

for all x



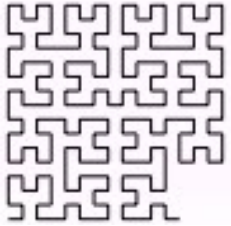
Credit: 3b1b

Credit: 3b1b



Credit: Stansy

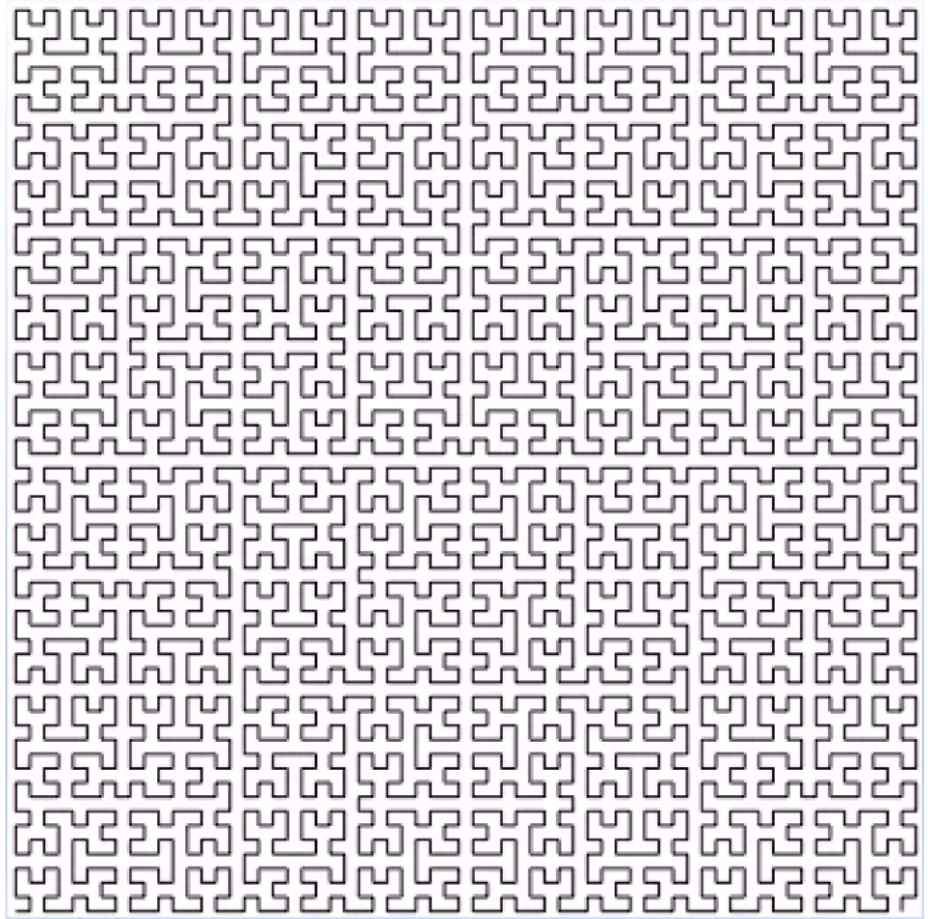
6th Order Pseudo-Hilbert Curve



```
int xy2d (int n, int x, int y) {
    int rx, ry, s, d=0;
    for (s=n/2; s>0; s/=2) {
        rx = (x & s) > 0;
        ry = (y & s) > 0;
        d += s * s * ((3 * rx) ^ ry);
        rot(n, &x, &y, rx, ry);
    }
    return d;
}

//convert d to (x,y)
void d2xy(int n, int d, int *x, int *y) {
    int rx, ry, s, t=d;
    *x = *y = 0;
    for (s=1; s<n; s*=2) {
        rx = 1 & (t/2);
        ry = 1 & (t ^ rx);
        rot(s, x, y, rx, ry);
        *x += s * rx;
        *y += s * ry;
        t /= 4;
    }
}

//rotate/flip a quadrant appropriately
void rot(int n, int *x, int *y, int rx, int ry) {
    if (ry == 0) {
        if (rx == 1) {
            *x = n-1 - *x;
            *y = n-1 - *y;
        }

        //Swap x and y
        int t  = *x;
        *x = *y;
        *y = t;
    }
```

Credit: CalmWords

6^{th} Order Pseudo-Hilbert Curve



```
int xy2d (int n, int x, int y) {
    int rx, ry, s, d=0;
    for (s=n/2; s>0; s/=2) {
        rx = (x & s) > 0;
        ry = (y & s) > 0;
        d += s * s * ((3 * rx) ^ ry);
        rot(n, &x, &y, rx, ry);
    }
    return d;
}

//convert d to (x,y)
void d2xy(int n, int d, int *x, int *y) {
    int rx, ry, s, t=d;
    *x = *y = 0;
    for (s=1; s<n; s*=2) {
        rx = 1 & (t/2);
        ry = 1 & (t ^ rx);
        rot(s, x, y, rx, ry);
        *x += s * rx;
        *y += s * ry;
        t /= 4;
    }
}

//rotate/flip a quadrant appropriately
void rot(int n, int *x, int *y, int rx, int ry) {
    if (ry == 0) {
        if (rx == 1) {
            *x = n-1 - *x;
            *y = n-1 - *y;
        }

        //Swap x and y
        int t  = *x;
        *x = *y;
        *y = t;
    }
```

Credit: CalmWords

$N > 2^N$?

$N <$?

References

- Is Infinite Maths Useful? – 3B1B: https://youtu.be/3s7h2MHQtxc
- Space Filling Curves – Numberphile: https://youtu.be/x-DgL49CFlM
- Hilbert Curve – Wikipedia
- Hilbert Curve – Wolfram Language Documentation
- Your Guide to Digital TV Conversion – Popular Science, Feb 2008

Contact Mail: mberkturgut[at]gmail.com