Bienvenidos

Integrantes

- Frank Salas
- Sebastian Postigo
- Luis Huachaca
- Angel Loayza

Serie de Fourier

- Definicion de serie de Fourier:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos\left(\frac{n\pi x}{L}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi x}{L}\right)$$

- Coeficientes:

$$a_n = \frac{1}{L}\int_{-L}^{L} f(x)\cos\left(\frac{n\pi x}{L}\right)dx$$

$$a_0 = \frac{1}{L}\int_{-L}^{L} f(x)dx$$

$$b_n = \frac{1}{L}\int_{-L}^{L} f(x)\sin\left(\frac{n\pi x}{L}\right)dx$$

$Sacando : a_0$

$$a_0 = \frac{1}{L}\int_{-L}^{L} f(x)dx$$

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} xdx$$

$$a_0 = \frac{1}{\pi}|\frac{x^2}{2}|_{-\pi}^{\pi}$$

$$a_0 = \frac{1}{\pi}(\frac{\pi^2}{2} - \frac{(-\pi)^2}{2})$$

$$a_0 = \frac{1}{\pi}(\frac{\pi^2}{2} - \frac{\pi^2}{2})$$

$$a_0 = 0$$

$$a_n = \frac{1}{L}\int_{-L}^{L} f(x)\cos\left(\frac{n\pi x}{L}\right)dx$$

Sacando : a_n

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x\cos\left(\frac{n\pi x}{\pi}\right)dx$$

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x\cos(nx)dx$$

$$a_n = \frac{1}{\pi}\left(\left(\frac{1}{n}x\sin(nx)\Big|_{-\pi}^{\pi}\right) - \left(\frac{1}{n}\int_{-\pi}^{\pi}\sin(nx)dx\right)\right)$$

$$a_n = \frac{1}{\pi}\left(\frac{1}{n}\pi\sin(n\pi) - \frac{1}{n}(-\pi)\sin(-n\pi) + \frac{1}{n^2}\cos(nx)\Big|_{-\pi}^{\pi}\right)$$

$$a_n = \frac{1}{\pi n^2}(\cos(n\pi) - \cos(-n\pi)) = \frac{1}{\pi n^2}(\cos(n\pi) - \cos(n\pi)) = 0$$

$$Sacando : a_n$$

$$b_n = \frac{1}{L}\int_{-L}^{L} f(x)\sin\left(\frac{n\pi x}{L}\right)dx$$

$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x\sin\left(\frac{n\pi x}{\pi}\right)dx$$

$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x\sin(nx)dx$$

$$b_n = \frac{1}{\pi}(-\frac{1}{n}x\cos(nx)|_{-\pi}^{\pi} + \frac{1}{n}\int_{-\pi}^{\pi}\cos(nx)dx)$$

$$b_n = \frac{1}{\pi}(-\frac{1}{n}\pi\cos(n\pi) + \frac{1}{n}(-\pi)\cos(-n\pi) + \frac{1}{n^2}\sin(nx)|_{\pi}^{\pi})$$

$$b_n = \frac{1}{\pi}(-\frac{2\pi}{n}\cos(n\pi) + \frac{1}{n^2}\sin(n\pi) - \frac{1}{n^2}\sin(-n\pi))$$

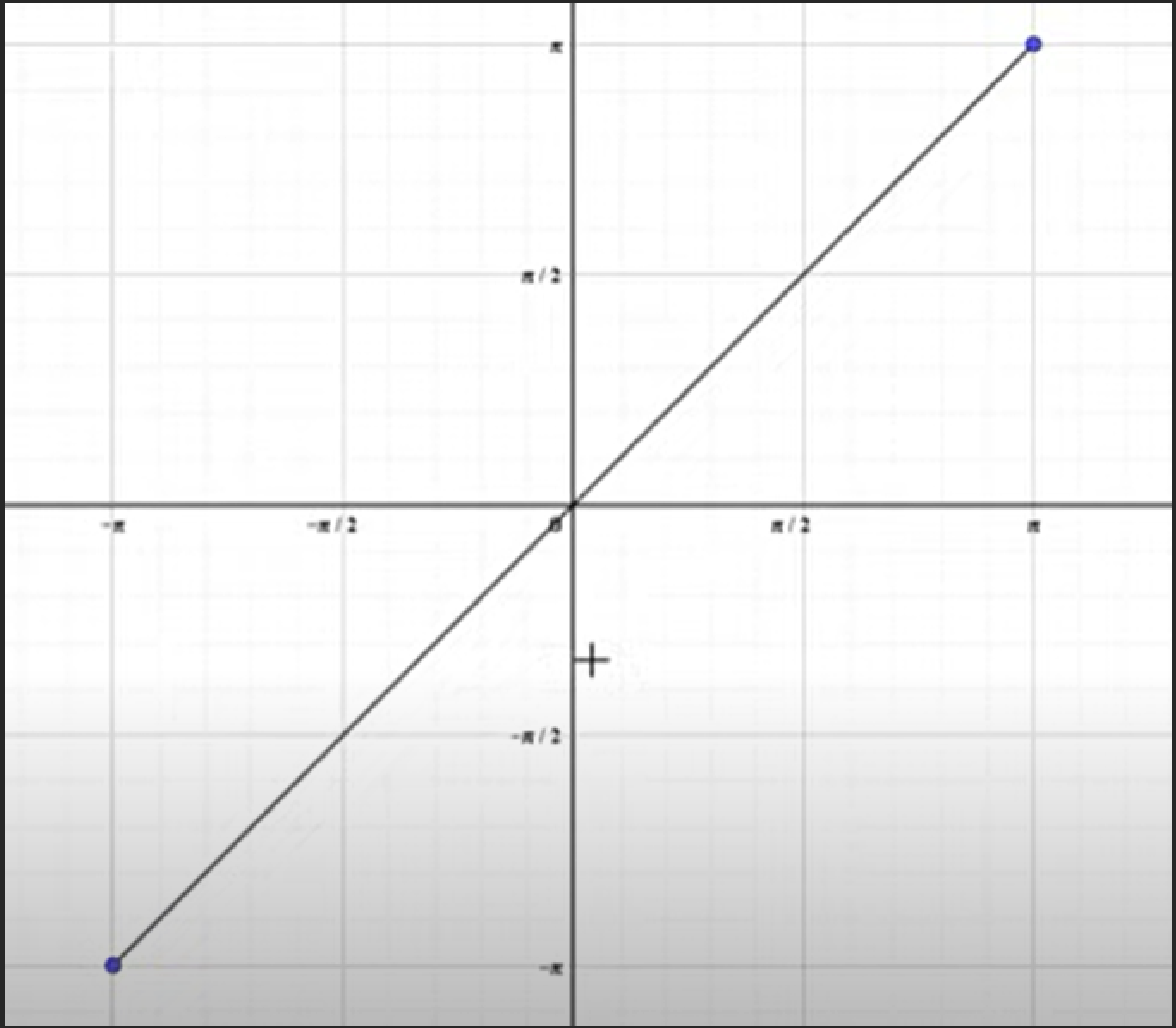
$$b_n = -\frac{2}{n}(-1)^n = \frac{2}{n}(-1)^{n+1}$$

Juntamos...

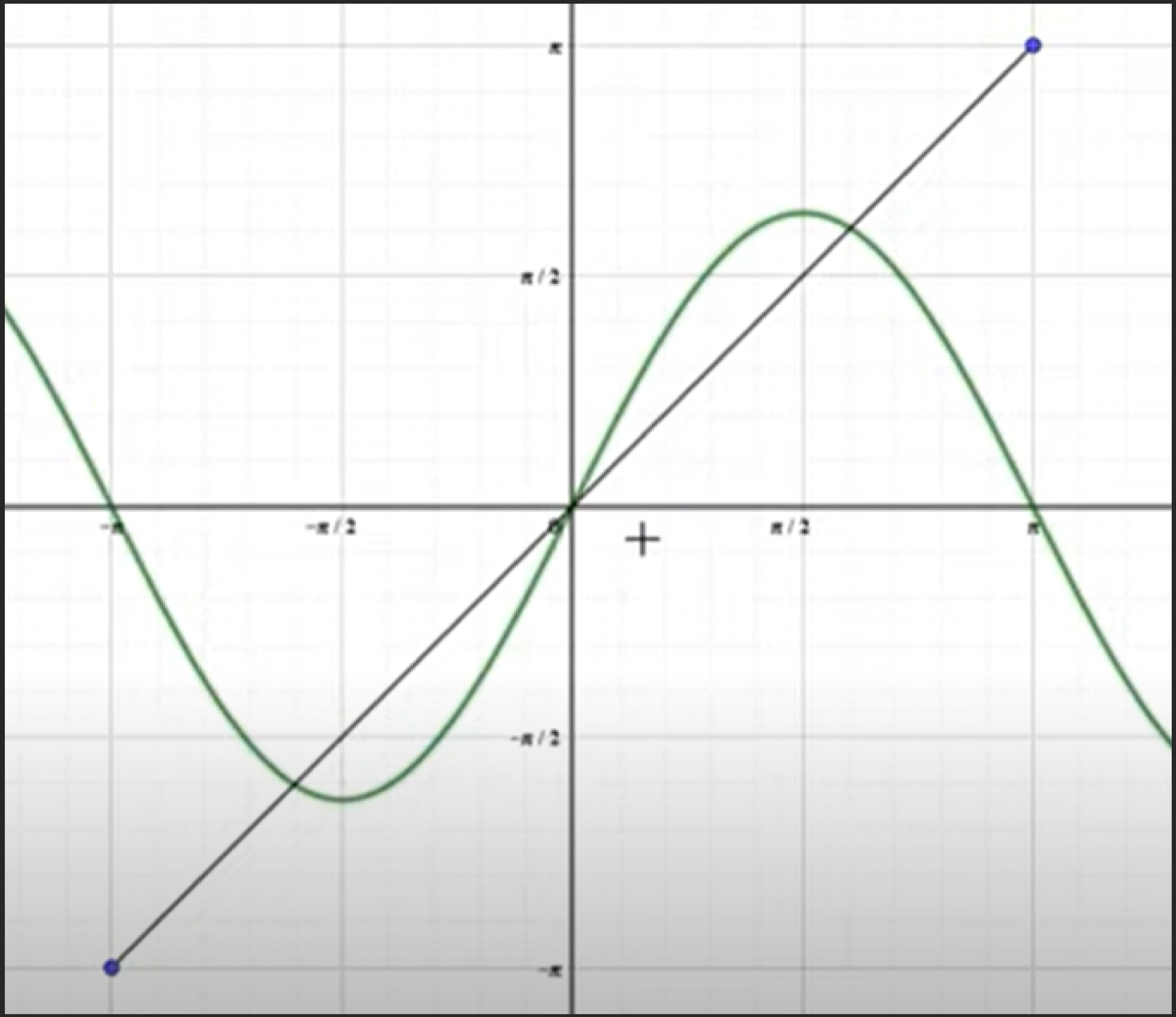
$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos\left(\frac{n\pi x}{L}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi x}{L}\right)$$

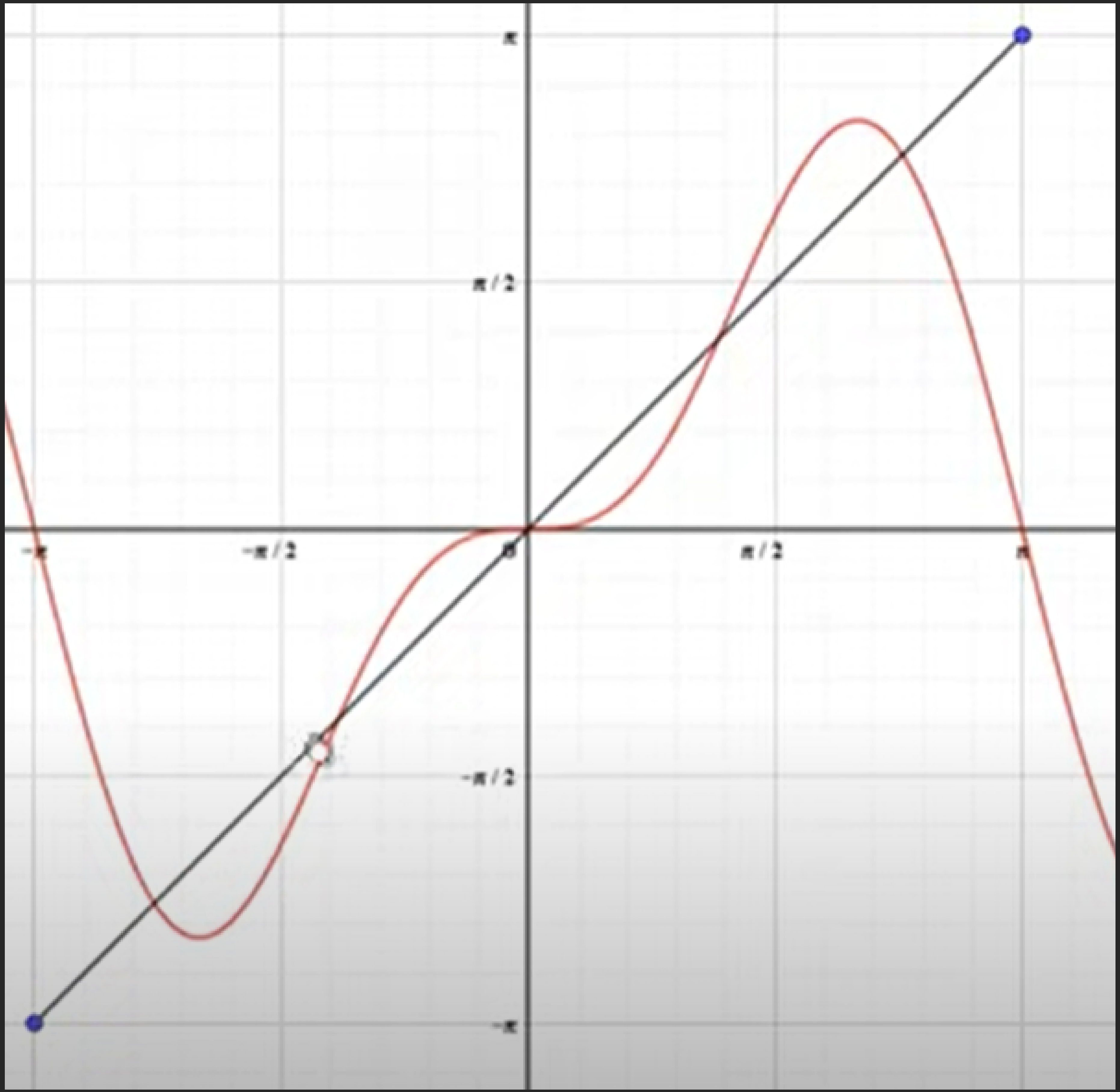
$$f(x) = \sum_{n=1}^{\infty} \frac{2}{n}(-1)^{n+1} sen(nx)$$

$$x = 2(\sin(x) - \frac{1}{2}\sin(2x) + \frac{1}{3}\sin(3x) - \frac{1}{4}\sin(4x) + \frac{1}{5}\sin(5x) - \ldots)$$

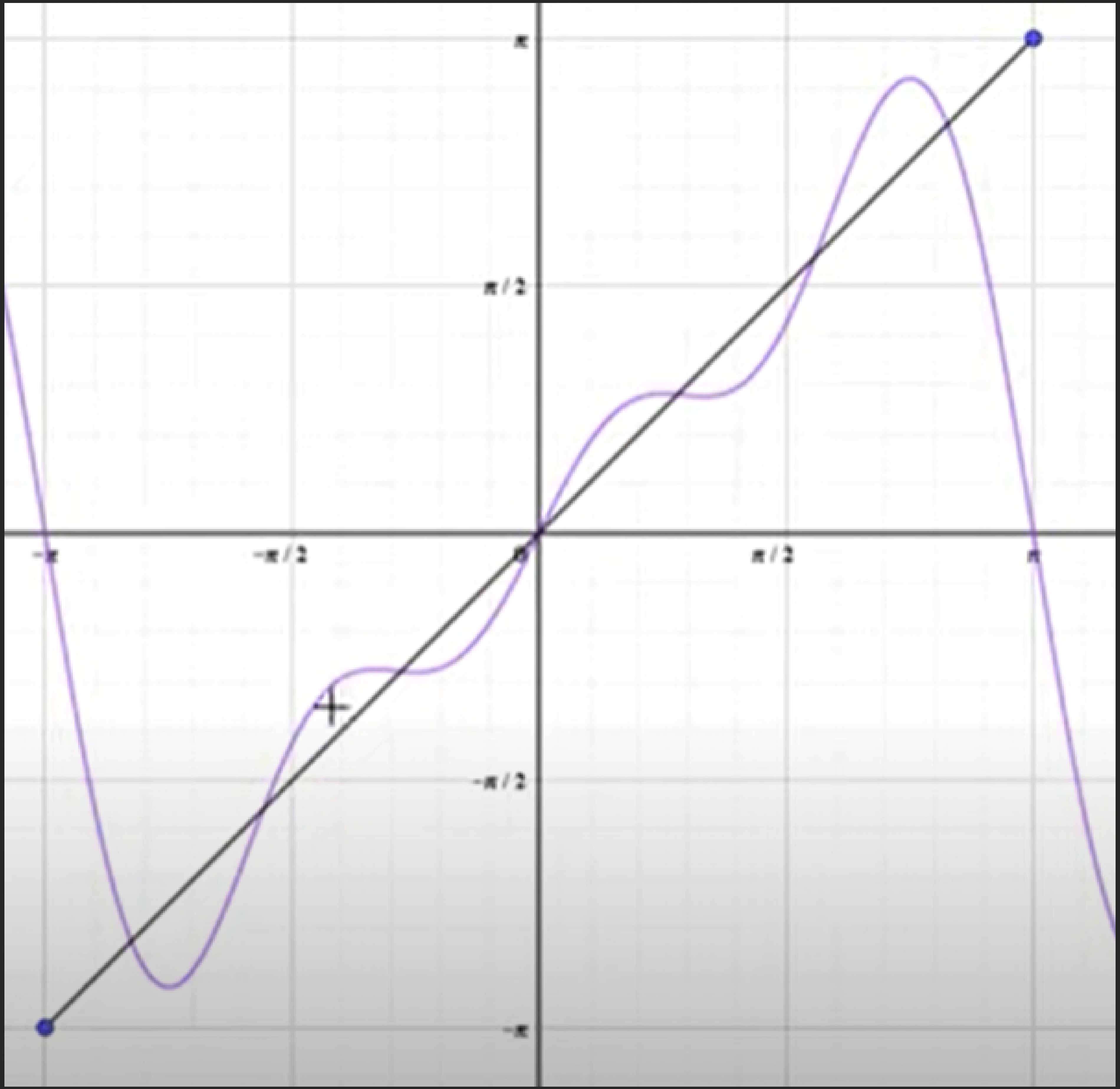


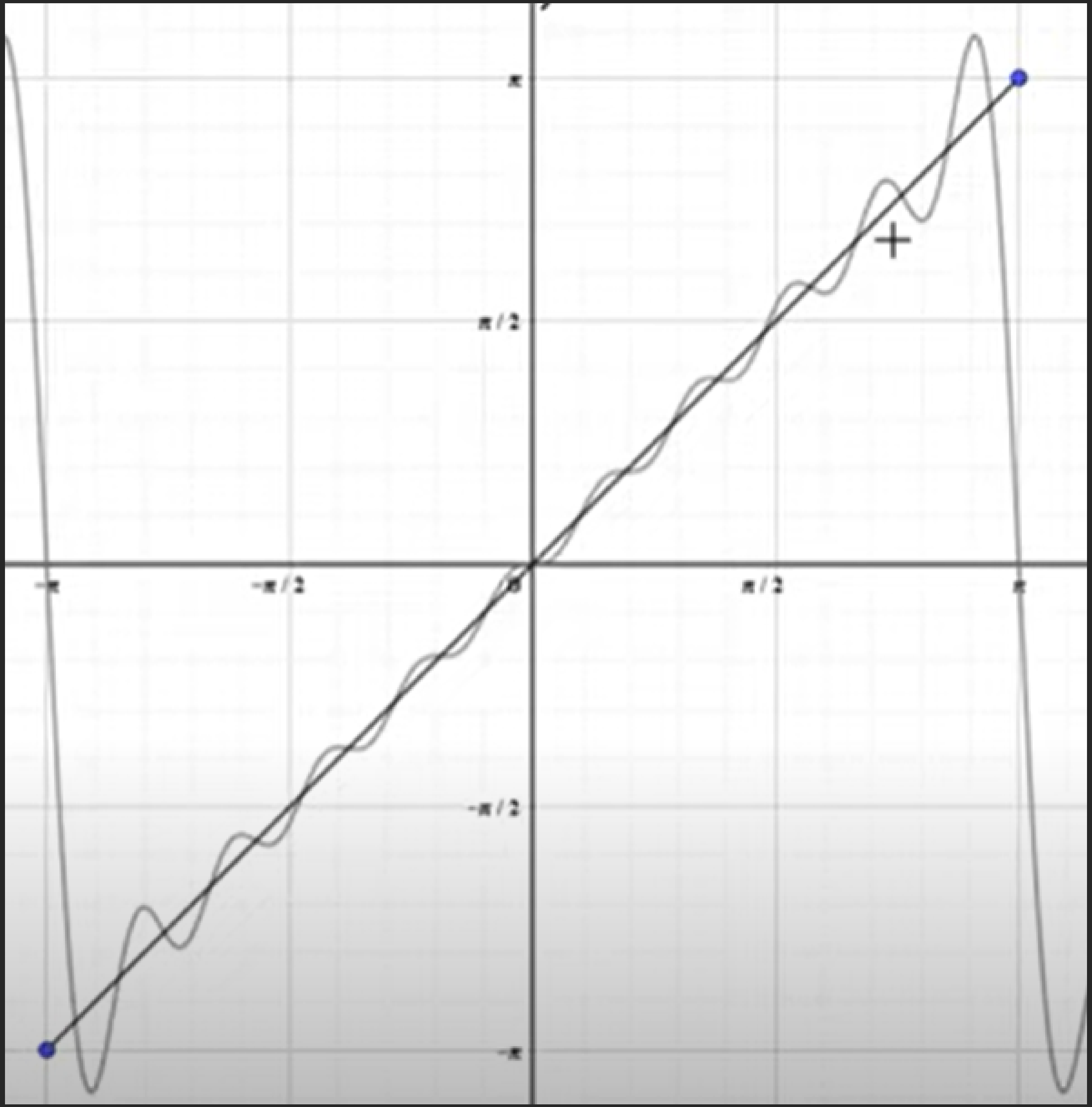
π
π / 2
-π
-π / 2
0
π / 2
π
-π / 2
-π


π
π / 2
-π / 2
-π
-π
-π / 2
0
π / 2
π

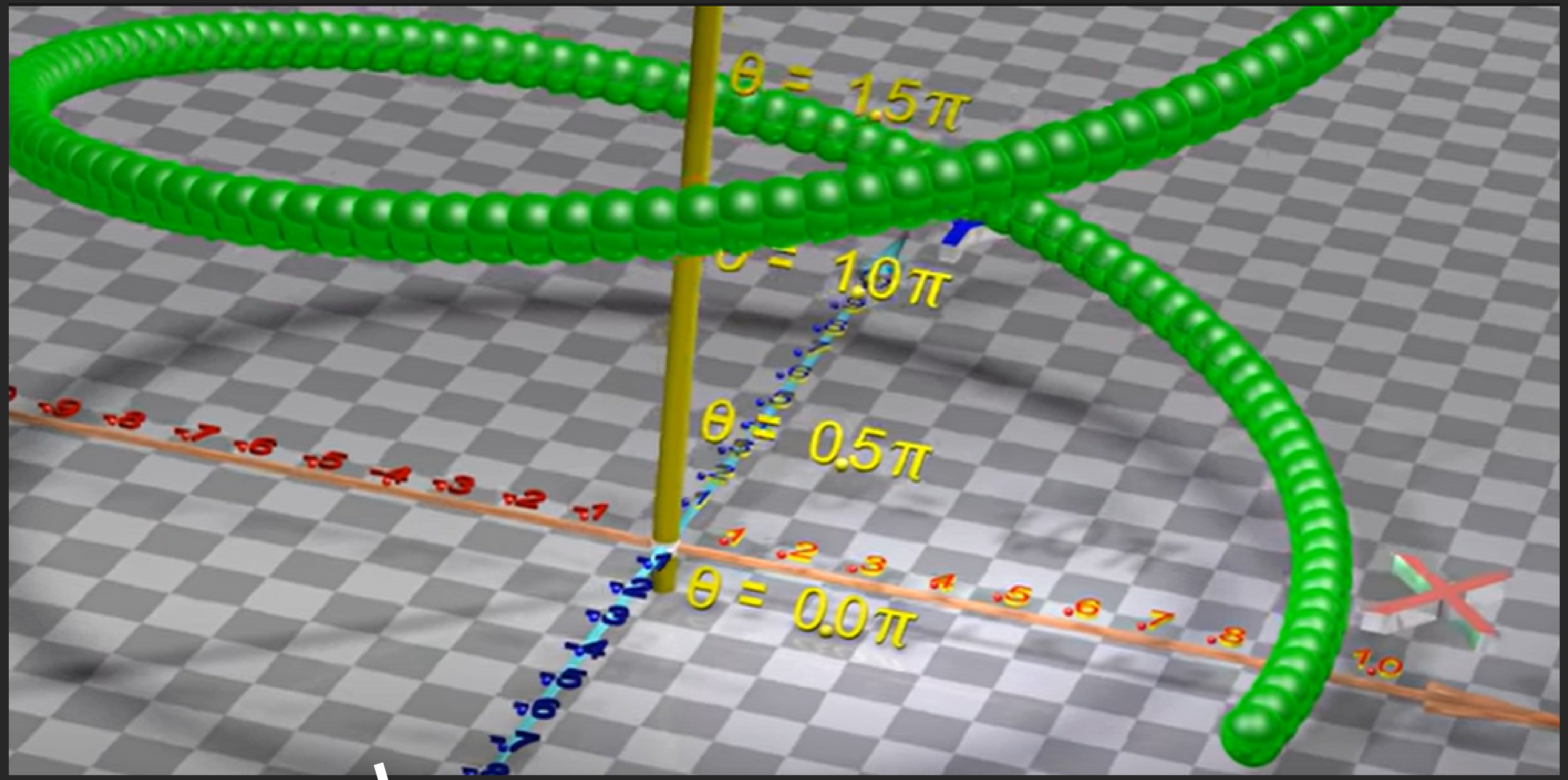


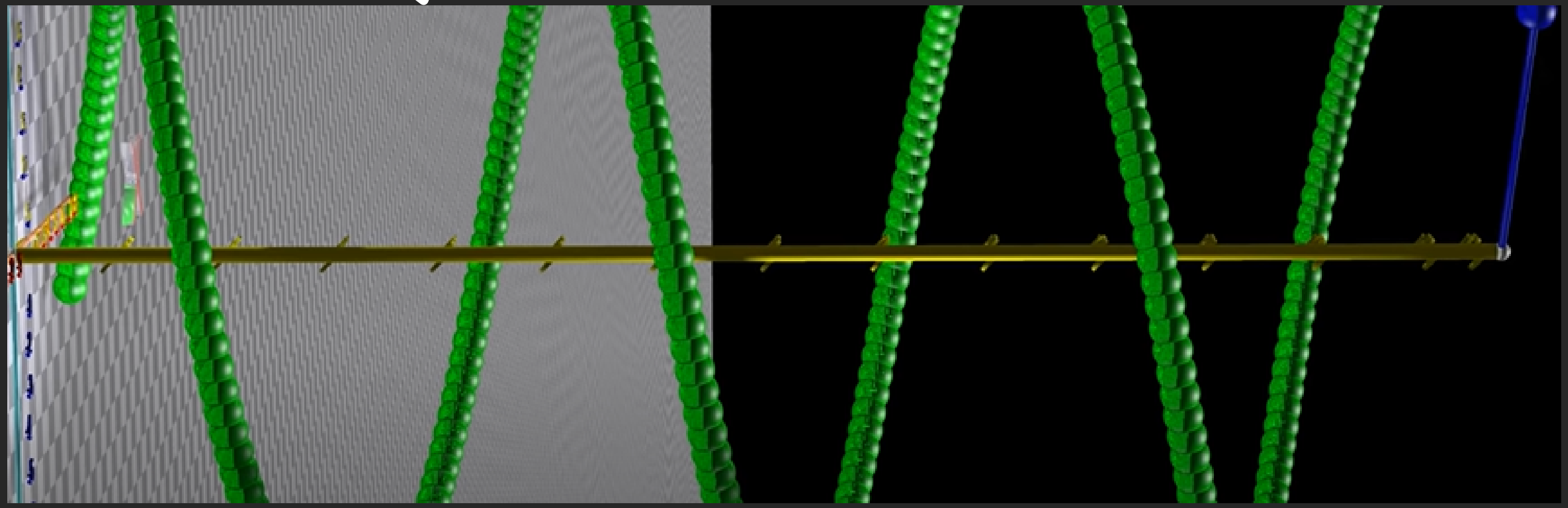
π
π / 2
−π
−π / 2
0
π / 2
π
−π / 2
−π

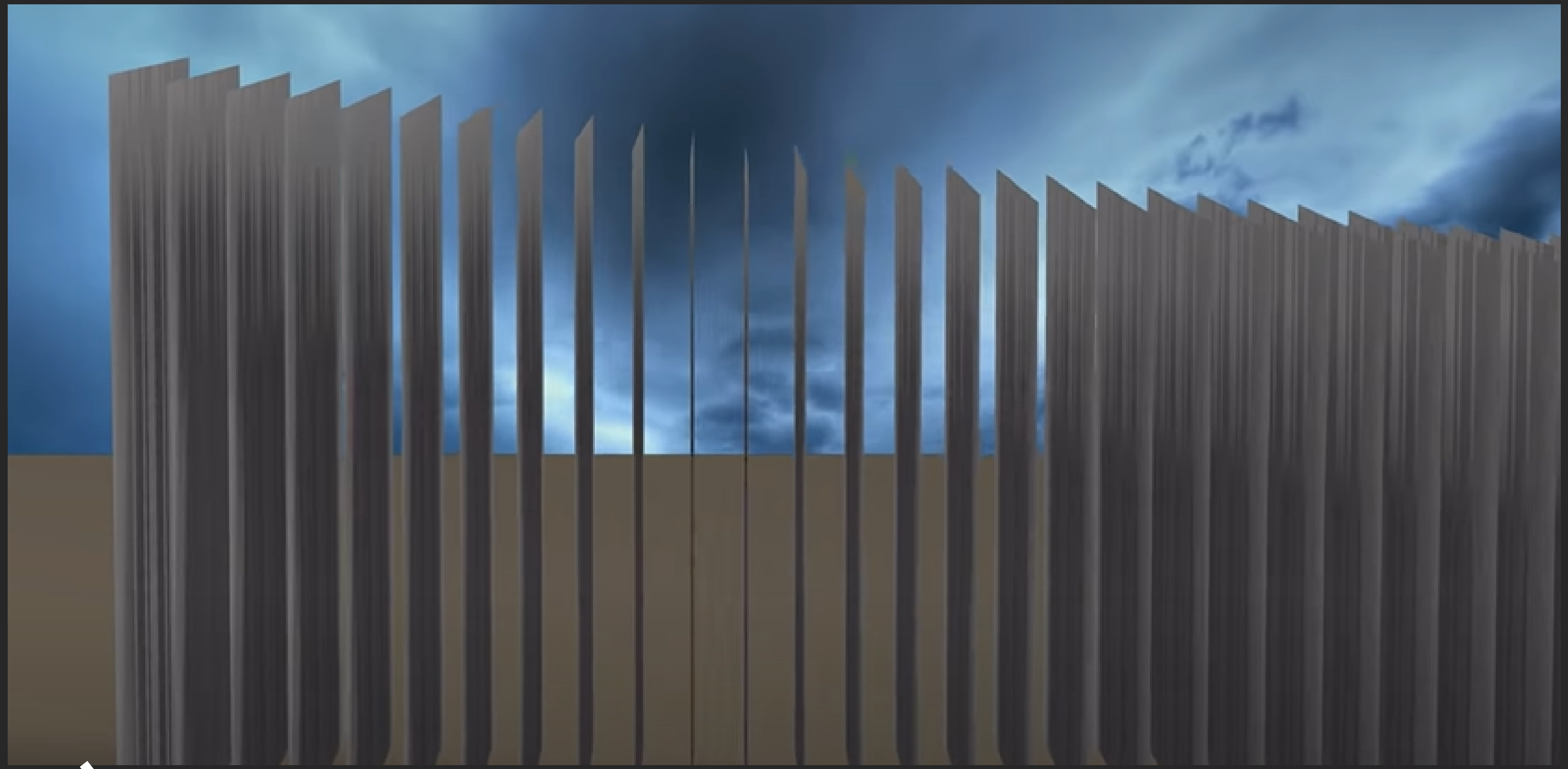

π
π / 2
-π
-π / 2
0
π / 2
π
-π / 2
-π

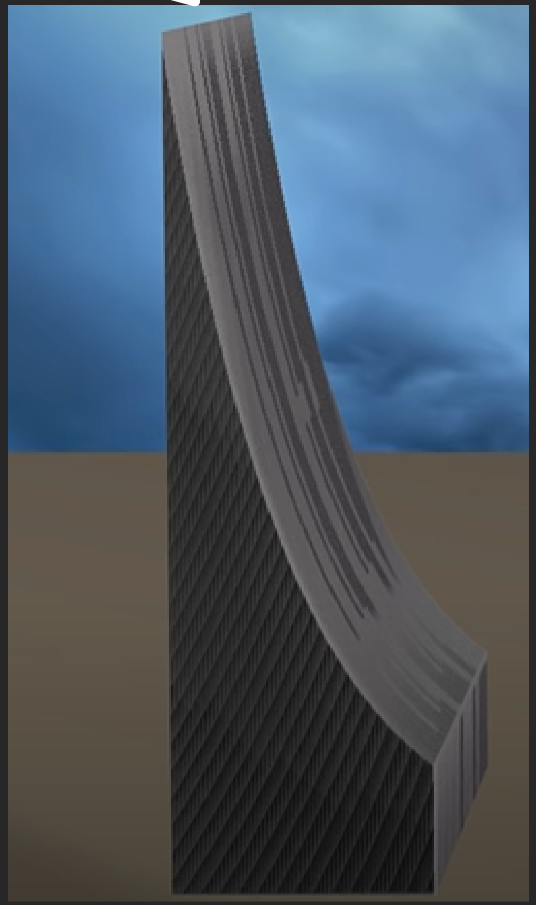


π
π / 2
−π / 2
−π
−π
−π / 2
0
π / 2
π

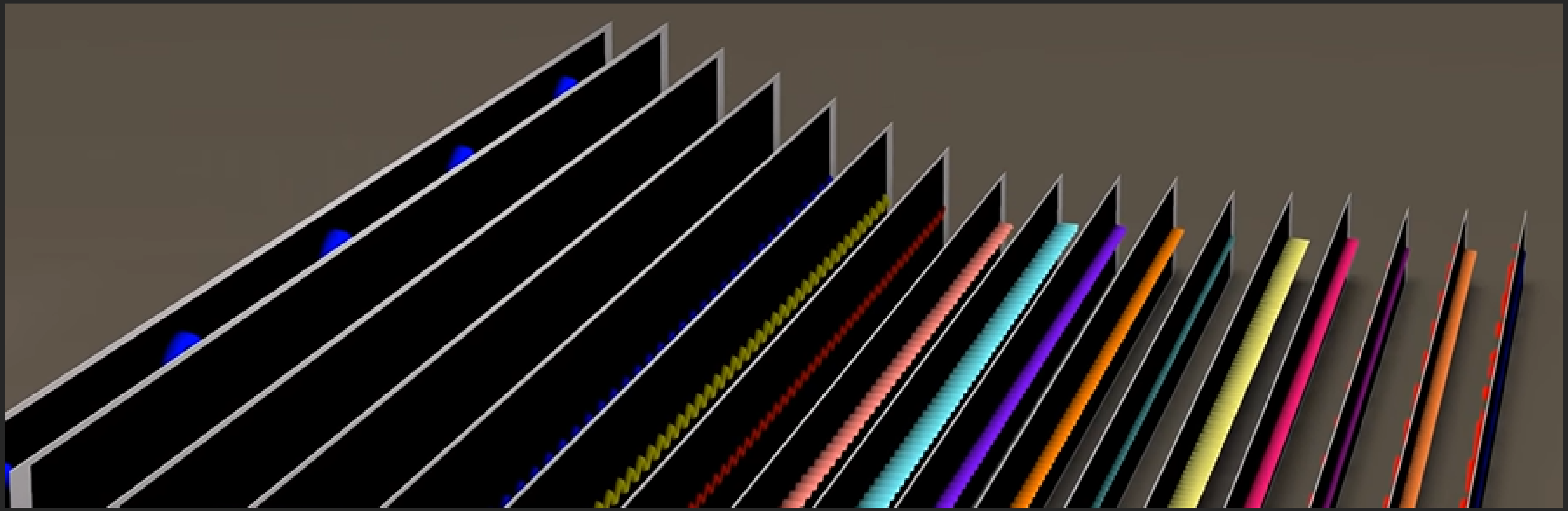






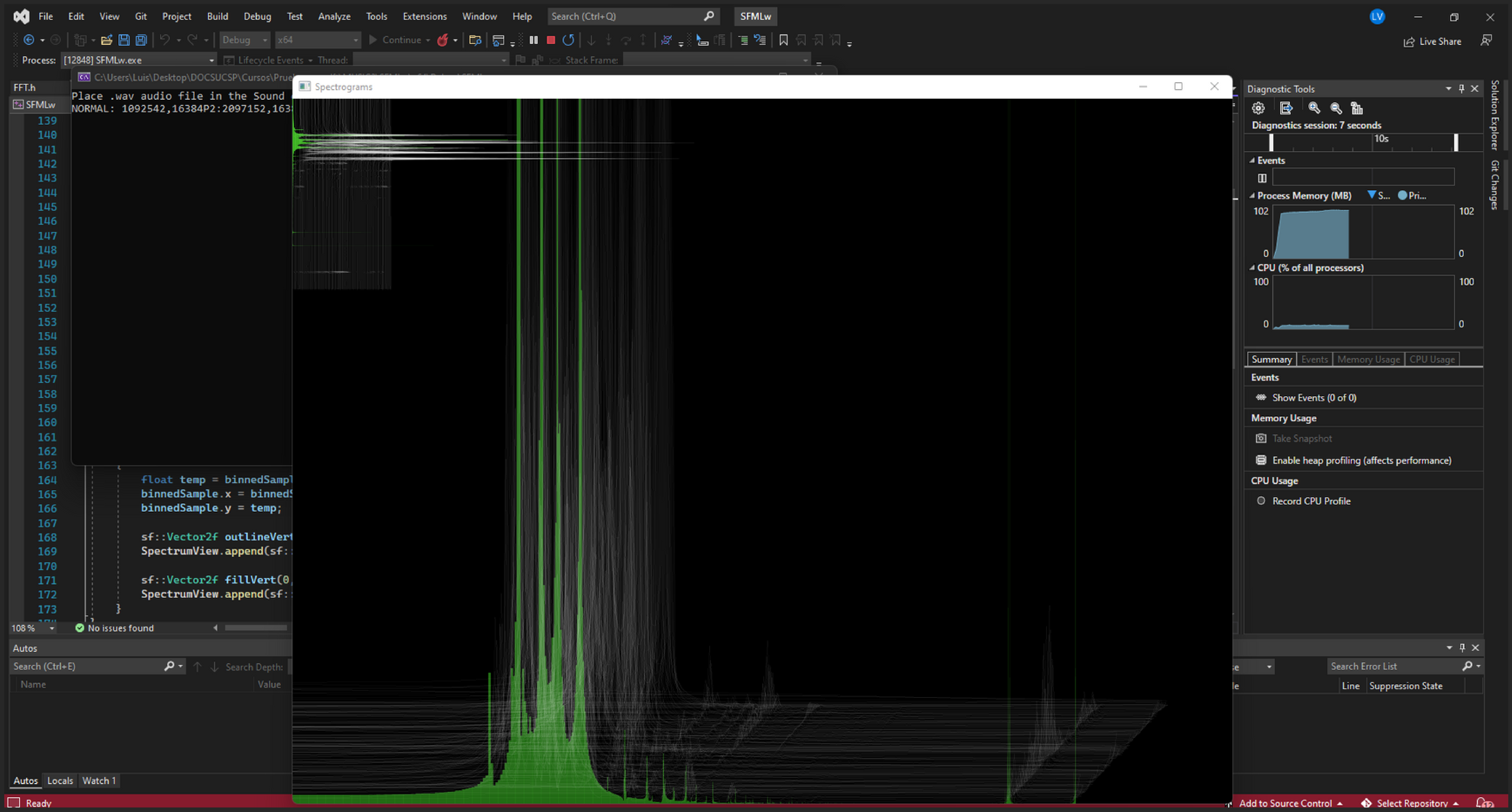


Spectrum con FFT



Entoonces -->

:)



File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
SFMLw
Live Share
Debug
x64
Continue
Process: [12848] SFMLw.exe
Lifecycle Events
Thread:
Stack Frame:
FFT.h
SFMLw
Spectrograms
Place .wav audio file in the Sound
NORMAL: 1092542,16384P2:2097152,163
float temp = binnedSampl
binnedSample.x = binned
binnedSample.y = temp;
sf::Vector2f outlineVert
SpectrumView.append(sf:
sf::Vector2f fillVert(0,
SpectrumView.append(sf:
}
108 %
No issues found
Autos
Search (Ctrl+E)
Search Depth:
Name
Value
Autos Locals Watch 1
Diagnostic Tools
Diagnostics session: 7 seconds
10s
Events
Process Memory (MB)
S...
Pri...
102
0
CPU (% of all processors)
100
0
Summary Events Memory Usage CPU Usage
Events
Show Events (0 of 0)
Memory Usage
Take Snapshot
Enable heap profiling (affects performance)
CPU Usage
Record CPU Profile
Search Error List
Line
Suppression State
Solution Explorer
Git Changes
Ready
Add to Source Control
Select Repository

Cooley Tukey

$$X_k = \sum_{m=0}^{N/2-1} x_{2m} e^{-\frac{2\pi i}{N}(2m)k} + \sum_{m=0}^{N/2-1} x_{2m+1} e^{-\frac{2\pi i}{N}(2m+1)k}$$

```
void CooleyTukeyFFT::FFT(SampleArray& values)
{
    const size_t N = values.size();
    if (N <= 1) return;

    SampleArray evens = values[std::slice(0, N / 2, 2)];
    SampleArray odds = values[std::slice(1, N / 2, 2)];

    FFT(evens);
    FFT(odds);

    for (size_t i = 0; i < N / 2; i++)
    {
        ComplexVal index = std::polar(1.0, -2 * PI * i / N) * odds[i];
        values[i] = evens[i] + index;
        values[i + N / 2] = evens[i] - index;
    }
}
```

$$X_k = \sum_{m=0}^{N/2-1} x_{2m} e^{-\frac{2\pi i}{N}(2m)k} + \sum_{m=0}^{N/2-1} x_{2m+1} e^{-\frac{2\pi i}{N}(2m+1)k}$$

$$X_k = \underbrace{\sum_{m=0}^{N/2-1} x_{2m} e^{-\frac{2\pi i}{N/2}mk}}_{\text{DFT of even-indexed part of } x_n} + e^{-\frac{2\pi i}{N}k} \underbrace{\sum_{m=0}^{N/2-1} x_{2m+1} e^{-\frac{2\pi i}{N/2}mk}}_{\text{DFT of odd-indexed part of } x_n} = E_k + e^{-\frac{2\pi i}{N}k} O_k$$

Decimation in time FFT

```
/*Aplicamos bit_reversal en nuestra data*/
int reversal = 0;
for (int i = 0; i < n; i++) {
    reversal = bit_reversal(i);
    if (i < reversal)
        swap(audio_data[i], audio_data[reversal]);
}
```

$$a = \Big\{ \big[(a_0, a_4), (a_2, a_6) \big], \big[(a_1, a_5), (a_3, a_7) \big] \Big\}$$

```
if (invert) {
    for (complex<double>& x : audio_data)
        x /= n;
}
```

```
for (int len = 2; len <= n; len <<= 1) {
    //Identidad de euler
    double ang = 2 * PI / len;
    if (!invert)
        ang = ang * -1;
    complex<double> euler_w(cos(ang), sin(ang));

    //Diagrama de la mariposa
    for (int i = 0; i < n; i += len) {
        complex<double> temp(1);
        for (int j = 0; j < len / 2; j++) {
            //y0=x0+x1*(raiz_de_unidad); y1=x0-x1*(raiz_de_unidad);
            complex<double> x0 = audio_data[i + j];
            complex<double> x1 = audio_data[i + j + len / 2] * temp;
            audio_data[i + j] = x0 + x1;
            audio_data[i + j + len / 2] = x0 - x1;
            temp *= euler_w;
        }
    }
}
```

$$y_0 = x_0 + x_1 \omega_n^k$$
$$y_1 = x_0 - x_1 \omega_n^k,$$

GRACIAS