# 1 Overview

The user draws a line on the screen, and we want to try and represent that line accurately using the fourier series. We have a formula for the fourier series to approximate any function, so we'll try to represent the position of the cursor as a function of time and fit a fourier series on that.

# 2 Approximating a function with the Fourier Series

For a target function $f(t)$, we approximate it as follows:

$$f(t) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty}(a_k \cos(kt) + b_k \sin(kt))$$

This has a period of $2\pi$ — the "slowest" term $k = 1$ has a period of $2\pi$, everything else aligns as multiples. Coefficients are

$$a_k = \frac{1}{\pi}\int_0^{2\pi} f(t)\cos(kt)dt$$
$$b_k = \frac{1}{\pi}\int_0^{2\pi} f(t)\sin(kt)dt$$

Numerically though, we can't exactly calculate the integrals, especially if we don't have $f$. This is where DFT comes in.

# 3 Finding coefficients with discrete samples (DFT)

Say we have $N$ evenly spaced samples $f_0, f_1, ..., f_{N-1}$ ($f_i = f(i \cdot \Delta t)$, and $\Delta t$ is our sampling rate for reading the cursor pos).
Approximating as a Riemann sum (use $\frac{i}{N}2\pi$ to lerp sample index into the $2\pi$ period):

$$\begin{aligned} a_k &\approx \frac{1}{\pi}\sum_{i=0}^{N-1} f_i \cos(k \cdot \frac{i}{N}2\pi) \cdot dt \\ &\approx \frac{1}{\pi}\sum_{i=0}^{N-1} f_i \cos(k \cdot \frac{i}{N}2\pi) \cdot \frac{2\pi}{N} \\ &\approx \frac{2}{N}\sum_{i=0}^{N-1} f_i \cos(k \cdot \frac{i}{N}2\pi) \end{aligned}$$

Same process for $b_k$.

$$b_k \approx \frac{2}{N}\sum_{i=0}^{N-1} f_i \sin(k \cdot \frac{i}{N}2\pi)$$

# 4 Applying this back to drawing

Say we collected $N$ samples with a time step of $\Delta t$.

- Our set of cursor coordinates collected are $\{(x_i, y_i) \mid i \in \mathbb{Z}; 0 \le i \le N-1\}$.
- $(x_i, y_i)$ collected at time $\Delta t \cdot i$.

For $M$ terms (user can set $M$), use

$$f(t) \approx \frac{a_0}{2} + \sum_{k=1}^{M}(a_k \cos(kt) + b_k \sin(kt))$$

and, given that this function's samples are $f_i$ (aka $x_i$ or $y_i$), its coefficients are

$$a_k = \frac{2}{N}\sum_{i=0}^{N-1} f_i \cos(k \cdot \frac{i}{N}2\pi)$$
$$b_k = \frac{2}{N}\sum_{i=0}^{N-1} f_i \sin(k \cdot \frac{i}{N}2\pi)$$

Do this for both $x$ and $y$ (ie $f \in \{x(t), y(t)\}$). Now you have an approximation of $x(t)$ and $y(t)$ from the samples $(x_0, y_0), ..., (x_{N-1}, y_{N-1})$ (with an important caveat that $t$ is compressed to the range $[0, 2\pi]$ instead of the original range in the approximated functions).