# Fixed-Shape Ellipse by Three Points

Danilo Tedeschi
Dra. Marina Andretta

Universidade de São Paulo

25 de Outubro de 2019

# Introduction

The shape of an ellipse is given by its major-axis and minor-axis, $(a, b) \in \mathbb{R}^2$, with $a > b > 0$.

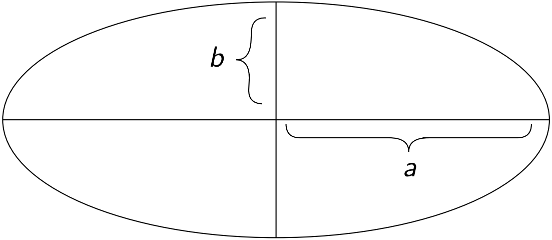


Figura: An ellipse with shape parameters $a$ and $b$.

# Introduction

Here, the shape will be fixed and the center and angle of rotation are free.

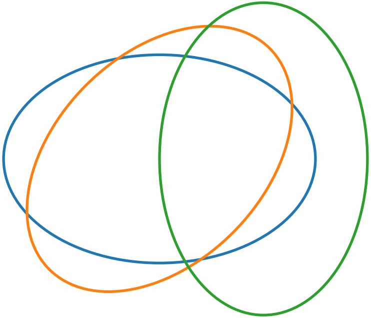

Figura: A fix-shape ellipse at different centers and with different angles of rotation.

Given three points $u, v, w \in \mathbb{R}^2$, and the shape $(a, b) \in \mathbb{R}^2$ of an ellipse:

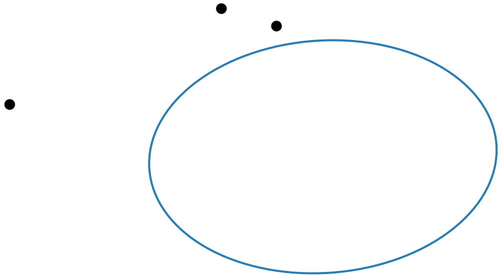

Figura: An instance of the problem.

A solution is given by the ellipse's center $q \in \mathbb{R}^2$ and the angle of rotation $\theta \in [0, \pi)$, such that $u, v, w$ lie on its border. **We want to find every solution!**

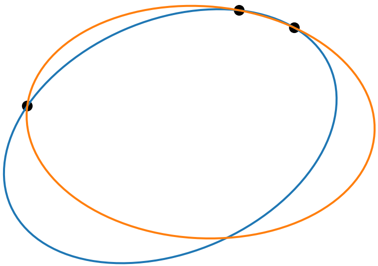

Figura: Every solution for the instance shown previously.

# Introduction

The equation of an ellipse is given by:

$$\frac{\left(\begin{bmatrix} x - q_x \\ y - q_y \end{bmatrix}^T \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}\right)^2}{a^2} + \frac{\left(\begin{bmatrix} x - q_x \\ q_y - y \end{bmatrix}^T \begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix}\right)^2}{b^2} = 1.$$

- Fixing the points $u, v, w$, we get 3 equations and 3 unknowns $(q_x, q_y, \theta)$.
- Finding every solution is difficult.

# Transforming the problem

Let's make the problem simpler by transforming it into a circle problem.
An ellipse with shape $(a, b)$ can be transformed into a circle of radius $b$ through scaling the $x$-axis by $\frac{b}{a}$:

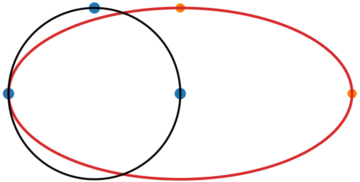

Figura: Turning an ellipse with shape $(a, b)$ into a circle of radius $b$.

# Transforming the problem

Let's rotate the points instead of rotating the ellipse:

Figura: Three points at their initial location.

# Transforming the problem

Firstly, we rotate leaving one point fixed at $(0, 0)$:

Figura: After rotation.

# Transforming the problem

Then, we scale by $\frac{b}{a}$ and check the radius of the circle:

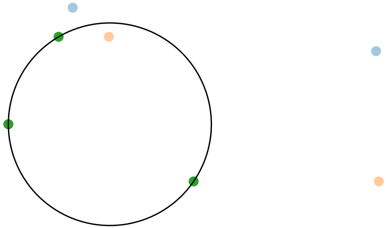

Figura: After scaling.

# Transforming the problem

If the radius is $b$, the angle is a solution:

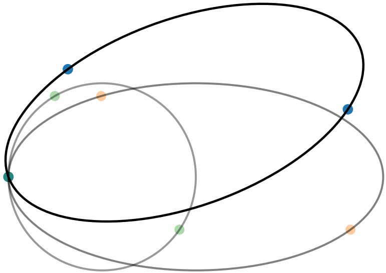

Figura: One solution for this instance.

# Transforming the problem

Formally, we can transform the problem by:

- Translate the points so $u = (0, 0)$.
- Rotate by $\theta$ and scale the $x$-axis by $\frac{b}{a}$.
- Find the $\theta$'s which produce a circle with radius $b$.

This transformation is expressed by:

$$\varphi(p, \theta) = \left[ \begin{array}{cc} \frac{b}{a} & 0 \\ 0 & 1 \end{array} \right] \left[ \begin{array}{cc} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{array} \right] \left[ \begin{array}{c} p_x \\ p_y \end{array} \right],$$

with $p = u, v, w$.

**This is an one variable problem on a closed interval!**

# Fixed-shape ellipse by three points

There is a known formula for the radius of a circumscribed circle:

$$R = \frac{\|\varphi(v,\theta)\|_2 \, \|\varphi(w,\theta)\|_2 \, \|\varphi(v,\theta) - \varphi(w,\theta)\|_2}{4A(\theta)}$$

- $R$ is the radius.
- $A(\theta)$ is the area of the triangle defined by the points $\varphi(u,\theta), \varphi(v,\theta), \varphi(w,\theta)$.

# Fixed-shape ellipse by three points

We define the function $\xi : [0, \pi) \mapsto \mathbb{R}$:

$$\xi(\theta) = 16b^2 A(\theta)^2 - \|\varphi(v,\theta)\|_2^2 \|\varphi(w,\theta)\|_2^2 \|\varphi(v,\theta) - \varphi(w,\theta)\|_2^2$$

The roots of $\xi$ are solutions of our problem.

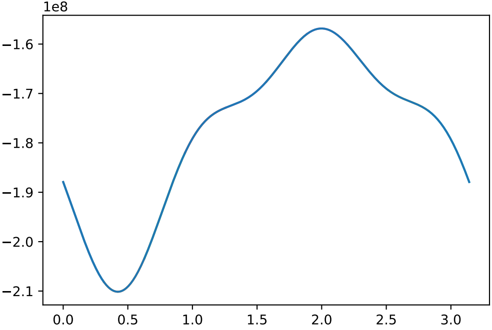


Figura: An example of $\xi$.

# Fixed-shape ellipse by three points

There is no clear pattern in $\xi$.

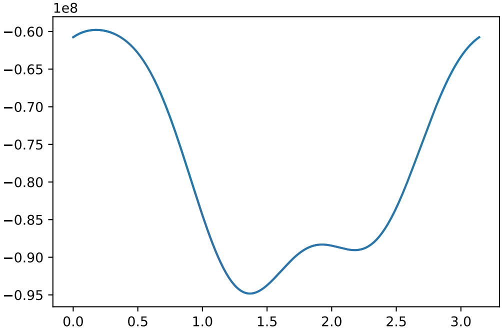


Figura: Another example of $\xi$.

# Fixed-shape ellipse by three points

An example with two roots.

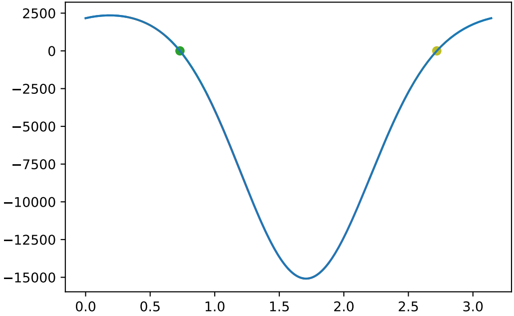


Figura: Yet another example of $\xi$.

# Chebyshev Interpolation

## Chebyshev Polynomial

$T_n : [-1, 1] \mapsto [-1, 1]$ is the $n$-degree Chebyshev polynomial:

$$T_n(\cos t) = \cos(nt)$$

Also, it can be defined recursively:

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_n(x) &= 2xT_{n-1}(x) - T_{n-2}(x) \end{aligned}$$

# Chebyshev Interpolation

Interpolation on the roots of $T_n$, also known as Chebyshev Nodes:

$$x_k = \cos\left(\pi\frac{2k-1}{2n}\right)$$

The interpolation of a function $f : [-1, 1] \mapsto \mathbb{R}$ can be written directly using Chebyshev polynomial as basis:

$$f(x) \approx \sum_{k=0}^{n} a_k T_k(x)$$

- A simple change of coordinates lets the interpolation to be done on any closed interval!
- This can be done in $\mathcal{O}(n^2)$.

# Chebyshev Interpolation

Why is it good?

- Numerically stable! Way better then polynomials in the power format.
- No Runge's Phenomenon, the interpolation converges to $f$.
  - $\mathcal{O}(n^{-m})$ if $f$ is $m$ times differentiable.
  - $\mathcal{O}(C^n)$, for $C < 1$, if $f$ is analytical in a neighborhood of $[-1, 1]$.
- Very used in practice: present in external libraries like NumPy for Python.

# Chebyshev Interpolation

Choosing the degree of the interpolation:

- There is no guaranteed way to choose it.
- A good rule is to examine the last coefficient.
- For a predefined $\epsilon$, choose $n$, such that:

$$|a_n| \leq \epsilon$$

- There are other ways like checking the error on a Lobatto grid.

# Chebyshev Interpolation

For $n = 32$, a precision of $10^{-10}$ is expected.

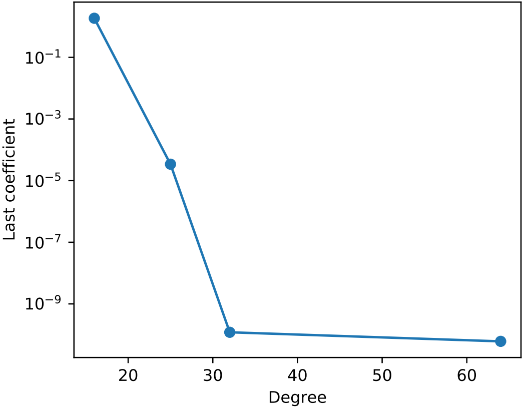


Figura: $|a_n|$ for the interpolation of $\xi$ for an instance.

The roots of a Chebyshev polynomial can be found though determining the eigenvalues of a Chebyshev companion matrix. This is it for $n = 5$:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ -\frac{a_0}{2a_5} & -\frac{a_1}{2a_5} & -\frac{a_2}{2a_5} & -\frac{a_3}{2a_5} & -\frac{a_4}{2a_5} \end{bmatrix}$$

- This matrix is a Hessenberg matrix.
- Its eigenvalues can be found by a QR algorithm in $\mathcal{O}(n^3)$.

# Chebyshev Interpolation

## Roots

The largest error on roots that were found for $n = 32$ is around $10^{-14}$ for an instance:

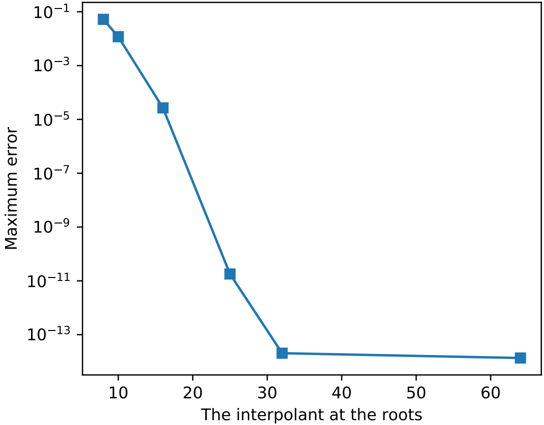


Figura: $|\xi(\hat{\theta})|$, where $\hat{\theta}$ is a root of $f_n$.

# Chebyshev Interpolation

## Roots

The experiments were made using Python with the NumPy library.
The running time is really low, even for $n = 64$.

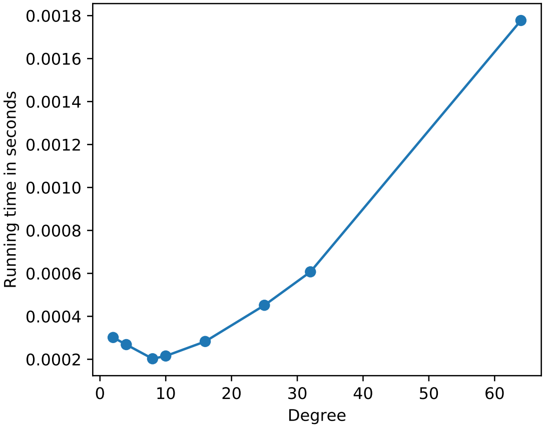


Figura: The running time to find the roots of $f_n$.

# Example

An example with 4 solutions.[?].

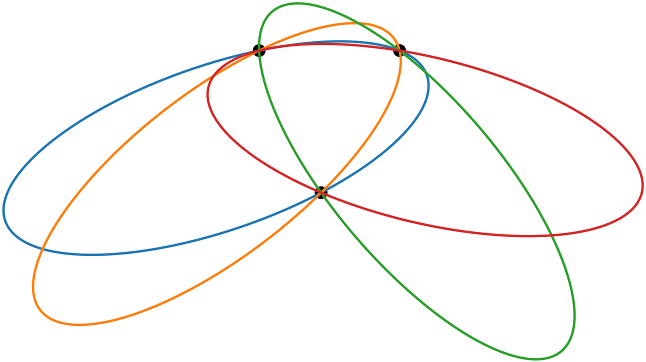

Figura: An example with 4 solutions.