

More precise determination of the bounding boxes of tikzpictures

Current status

TikZ determines the bounding box of (cubic) Bezier curves by establishing the smallest rectangle that contains the end point and the two control points of the curve (cf. Figure 1).

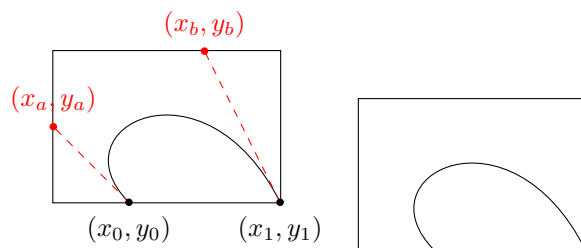


Figure 1: TikZ bounding boxes and an example.

This may lead to drastic overestimates of the bounding box (cf. Figure 2).

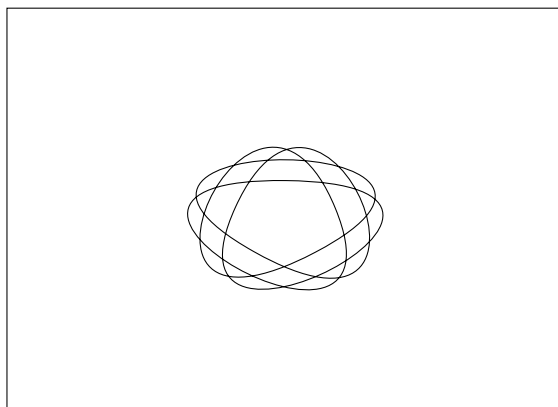


Figure 2: Example from <https://tex.stackexchange.com/q/43621/121799>.

Computing the bounding box

Establishing the precise bounding box has been discussed in various places, the following discussion uses in part the results from <https://pomax.github.io/bezierinfo/>. What is a cubic Bezier curve? A cubic Bezier curve running from (x_0, y_0) to (x_1, y_1) with control points (x_a, y_a) and (x_b, y_b) can be parametrized by

$$\gamma(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} t^3x_1 + 3t^2(1-t)x_b + (1-t)^3x_0 + 3t(1-t)^2x_a \\ t^3y_1 + 3t^2(1-t)y_b + (1-t)^3y_0 + 3t(1-t)^2y_a \end{pmatrix}, \quad (1)$$

where t runs from 0 to 1 (and $\gamma(0) = (x_0, y_0)$ and $\gamma(1) = (x_1, y_1)$). Surely, the bounding box has to contain (x_0, y_0) and (x_1, y_1) . If the functions $x(t)$ and $y(t)$ have extrema in the interval $[0, 1]$, then the bounding box will in general be larger than that. In order to determine the extrema of the curve, all we need to find the extrema of the functions $x(t)$ and $y(t)$ for $0 \leq t \leq 1$. That is, we need to find the solutions of the quadratic equations

$$\frac{dx}{dt}(t) = 0 \quad \text{and} \quad \frac{dy}{dt}(t) = 0. \quad (2)$$

Let's discuss x , y is analogous. If the discriminant

$$d := x_0x_1 - x_0x_b - x_1x_a + x_a^2 - x_ax_b + x_b^2 \quad (3)$$

is greater than 0, there are two solutions

$$t_{\pm} = \frac{x_0 - 2x_a + x_b \pm \sqrt{x_0x_1 - x_0x_b - x_1x_a + x_a^2 - x_ax_b + x_b^2}}{x_0 - x_1 - 3x_a + 3x_b}. \quad (4)$$

In this case, we need to make sure that the bounding box contains, say $(x(t_-), y_0)$ and $(x(t_+), y_0)$. If $d \leq 0$, the bounding box does not need to be increased in the x direction. As already mentioned, the analogous statements apply to $y(t)$.

It is rather straightforward to implement this procedure in TikZ. The relevant macros are `\pgf@lt@curveto` (and `\pgf@nlt@curveto`).¹ The macro `\pgf@lt@curveto` takes six arguments, which are x_a , y_a , x_b , y_b , x_1 and y_1 (in that order). x_0 and y_0 are stored in `\pgf@path@lastx` and `\pgf@path@lasty`, respectively.

Examples

¹Some care has to be taken when squaring lengths since they are all measured in points, and the square can easily become large and trigger a `dimension too large` error. When computing the discriminant d , I thus divided these distances by some taming factor that I took to be 32 for no special reasons. It might very well be that there are better taming factors, or that one needs to dial the taming factor as a function of the input values.

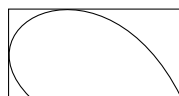


Figure 3: Tight bounding box for figure 1.

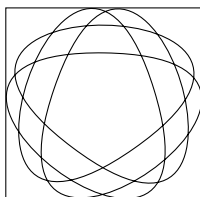


Figure 4: Tight bounding box for figure 2.