CADL2020

International Workshop on Computational Aspects of Deep Learning

ICPR 2020

Overview

Second Order Bifurcating Methodology for Neural Network Training and Topology Optimization

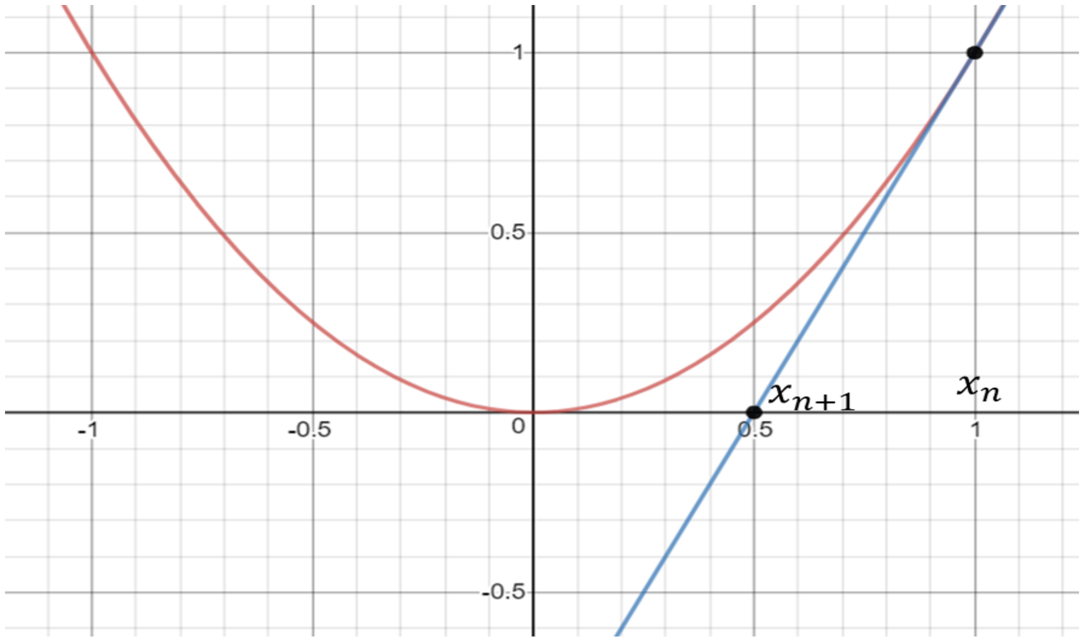
Presenter: Julio Zamora

intel

Authors:
Julio Zamora , Adan Cruz, Omesh Tickoo

Root finding

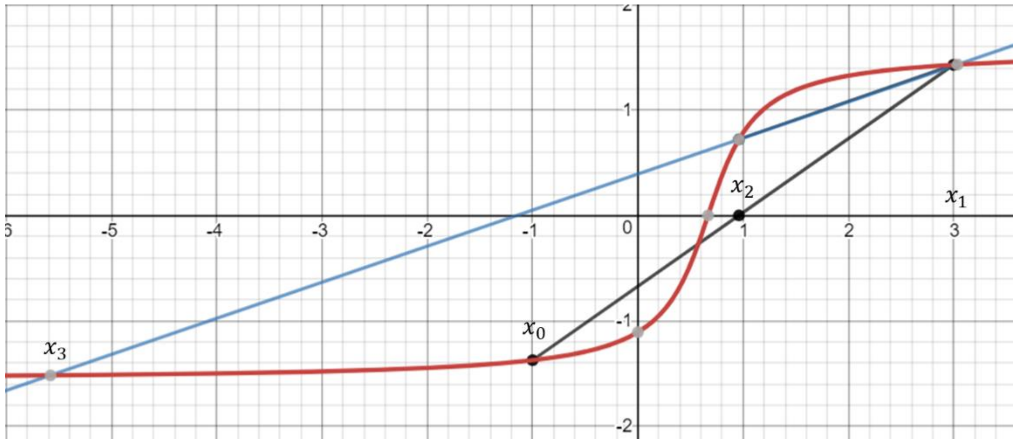
Newton-Raphson:
In this method the user selects an initial point x_0, the line L tangent of f at x_0 is computed, the root of L is used as next value x_1.



$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Root Finding

Secant:
Secant method is a variation of newton, but it iterates using the previous point. This method is cheaper computationally speaking.



$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n).$$

Root finding

Muller's Method (quadratic interpolation)

This method is similar to secant method, but it takes three points instead of two to create a parabola.

$$x_k = x_{k-1} - \frac{2f(x_{k-1})}{w \pm \sqrt{w^2 - 4f(x_{k-1})f[x_{k-1}, x_{k-2}, x_{k-3}]}}.$$

In this formula, the sign should be chosen such that the denominator is as large as possible in magnitude.

Inverse Quadratic Interpolation

Uses quadratic interpolation to approximate the inverse of the function f. Based on Lagrange's interpolation formula for *f-1fx* is used to derive the next candidate:

$$x_{n+1} = \frac{f_{n-1}f_n}{(f_{n-2} - f_{n-1})(f_{n-2} - f_n)}x_{n-2} + \frac{f_{n-2}f_n}{(f_{n-1} - f_{n-2})(f_{n-1} - f_n)}x_{n-1} + \frac{f_{n-2}f_{n-1}}{(f_n - f_{n-2})(f_n - f_{n-1})}x_n,$$

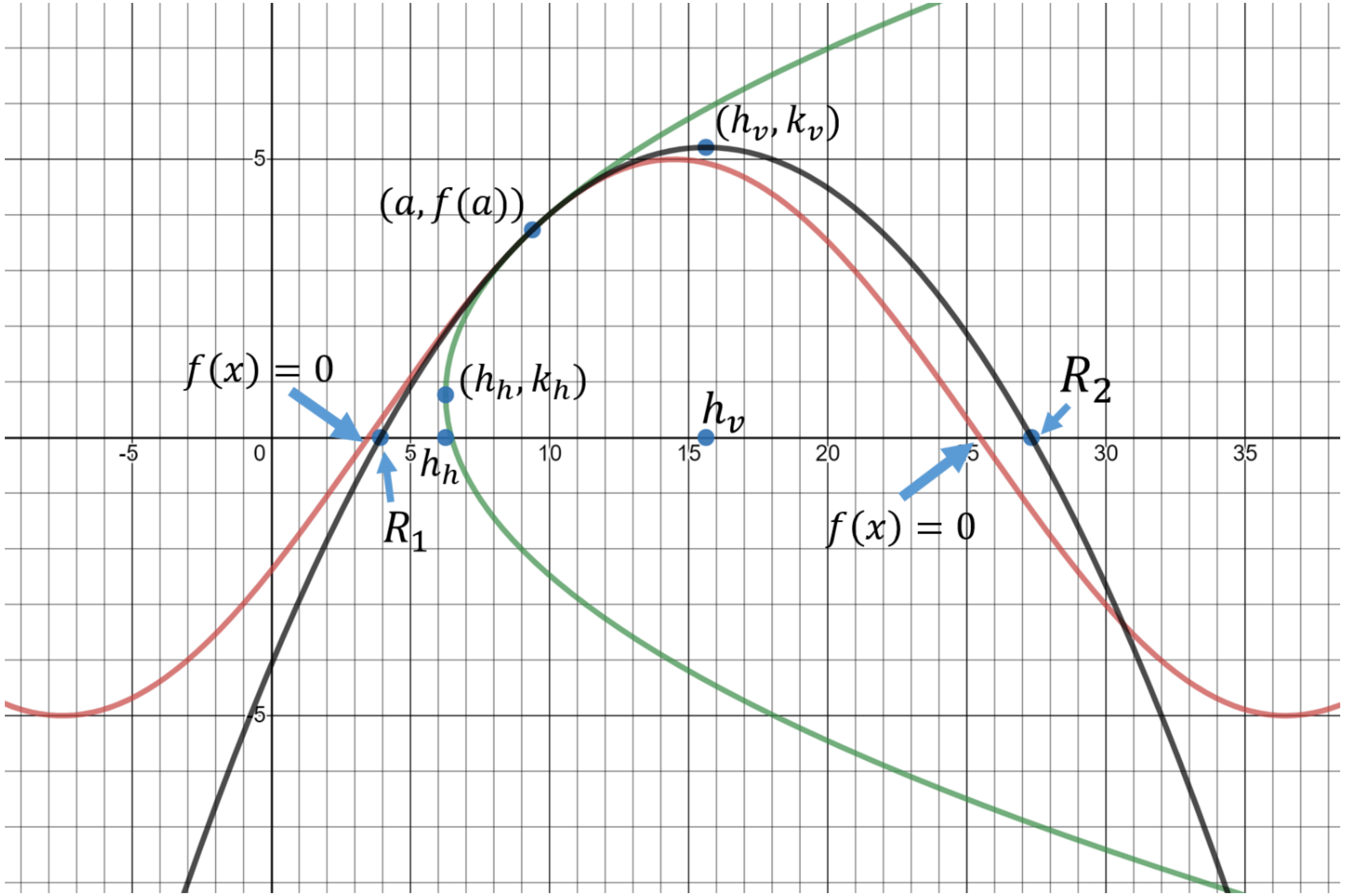
Bisections

It starts using an interval a0,b0 with fa0 and fb0 have opposite sign, to ensure you have at least one root.

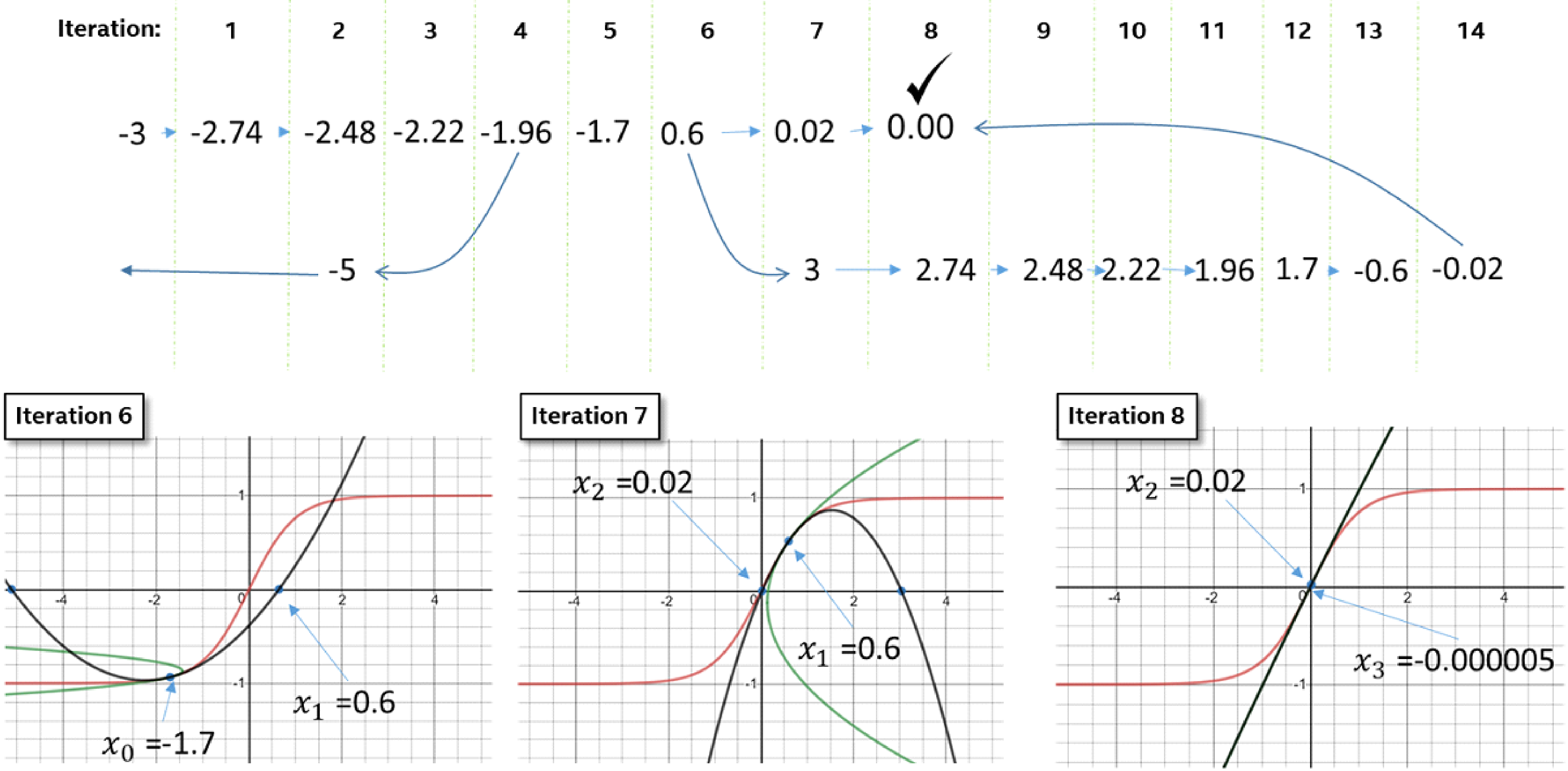
$$p_n = \frac{a_n + b_n}{2}, \quad a_{n+1} = \begin{cases} a_n & \text{si } f(a_n)\cdot f(p_n) < 0 \\ p_n & \text{si } f(a_n)\cdot f(p_n) > 0 \end{cases}, \quad b_{n+1} = \begin{cases} b_n & \text{si } f(b_n)\cdot f(p_n) < 0 \\ p_n & \text{si } f(b_n)\cdot f(p_n) > 0 \end{cases}$$

This method reduces the interval at each iteration until converges to the root.

Our method:Bifurcating

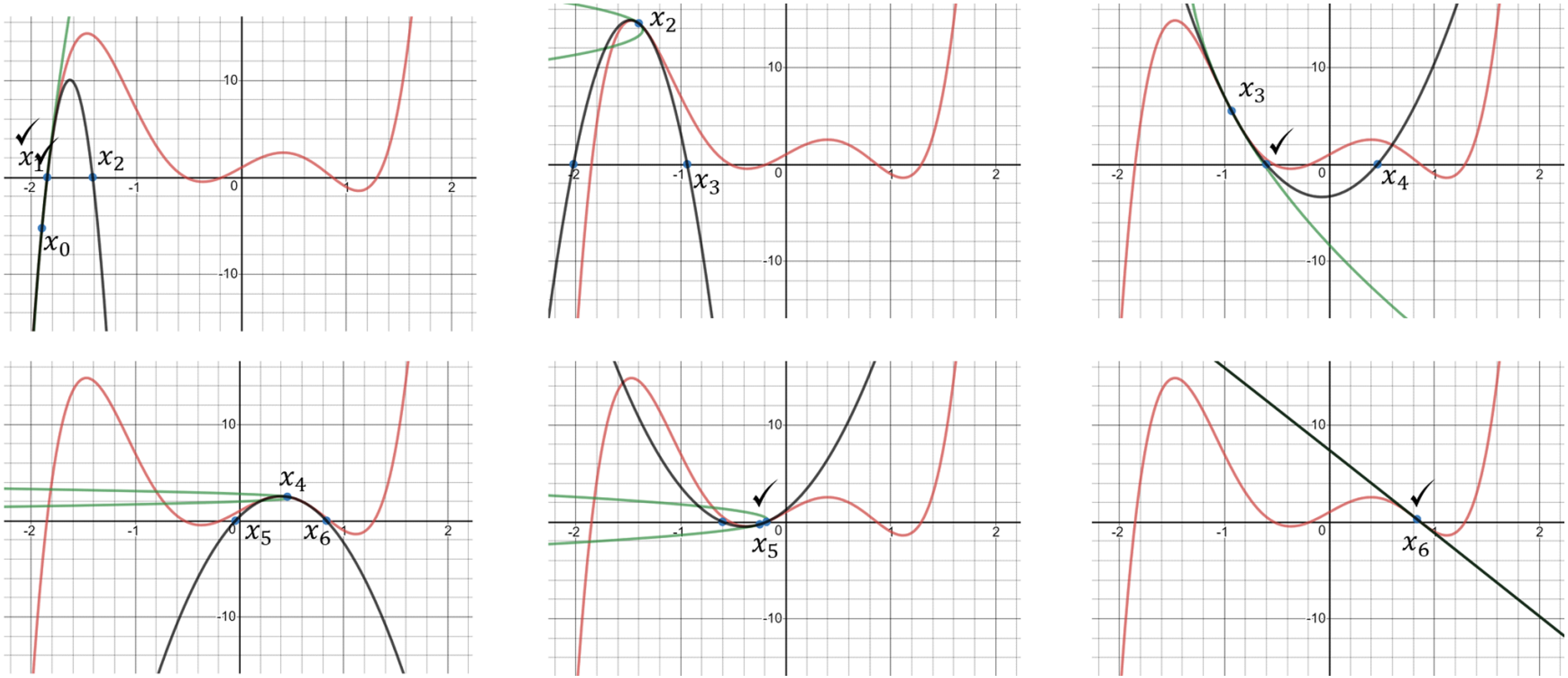


Example: Hyperbolic tangent

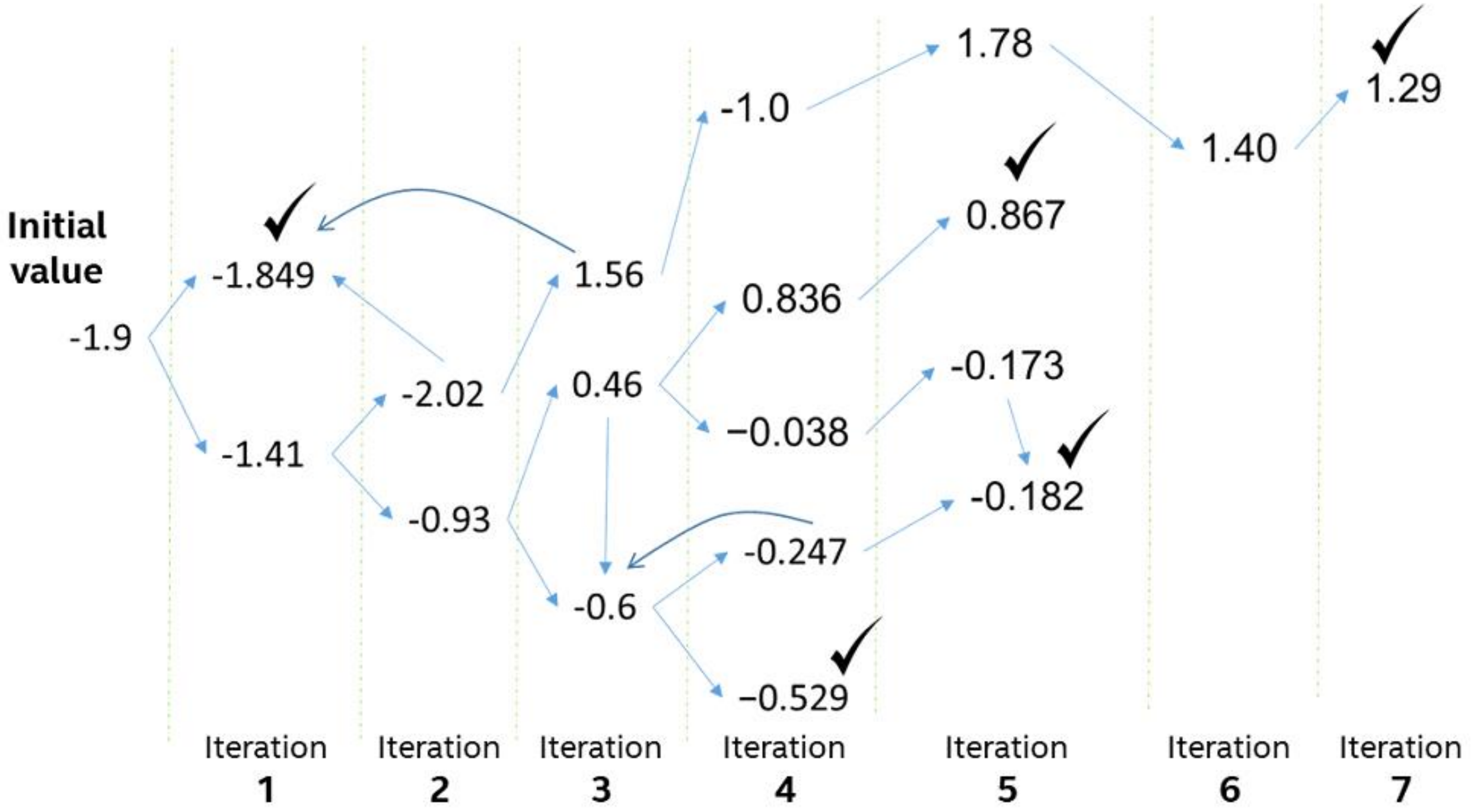


Example:

$$f(x) = 5x^5 + 2x^4 - 15x^3 + 6x + 1$$

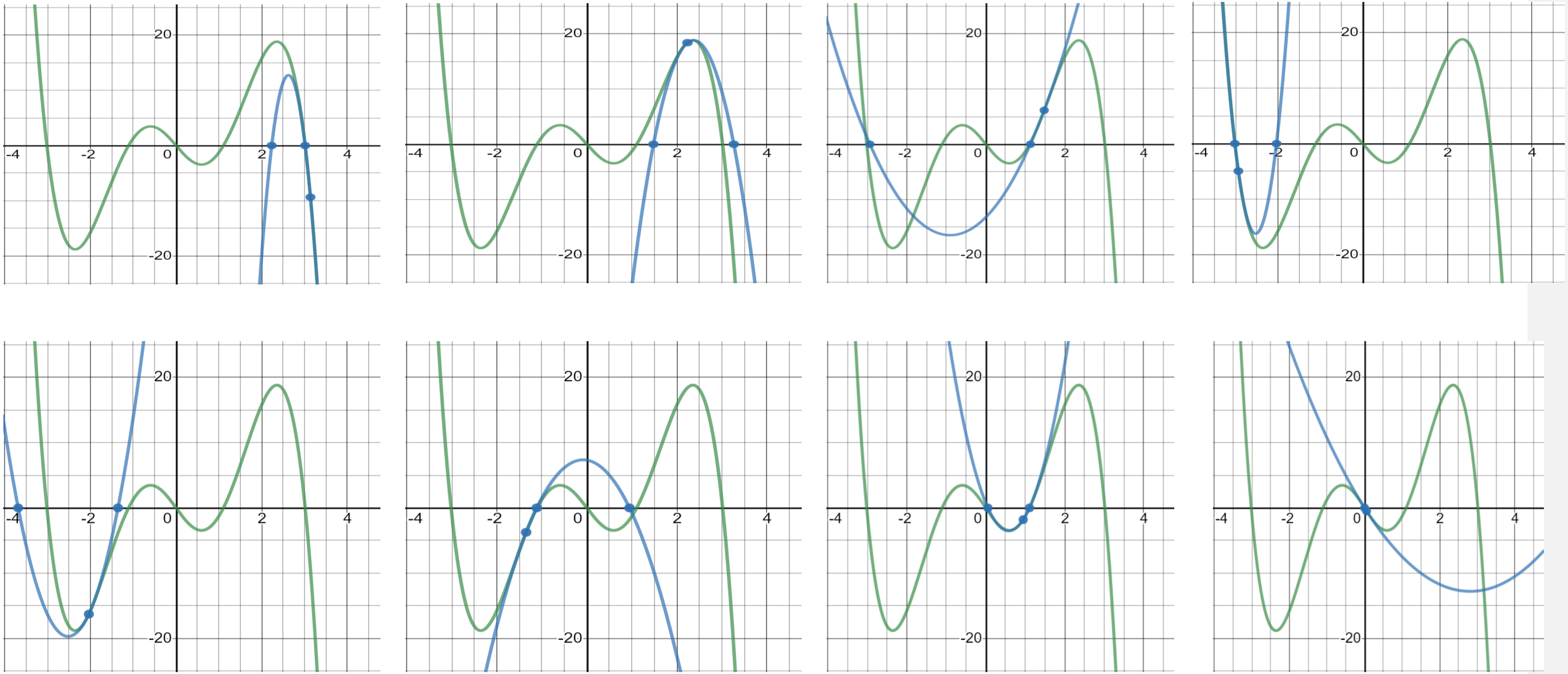


Example: $f(x) = 5x^5 + 2x^4 - 15x^3 + 6x + 1$



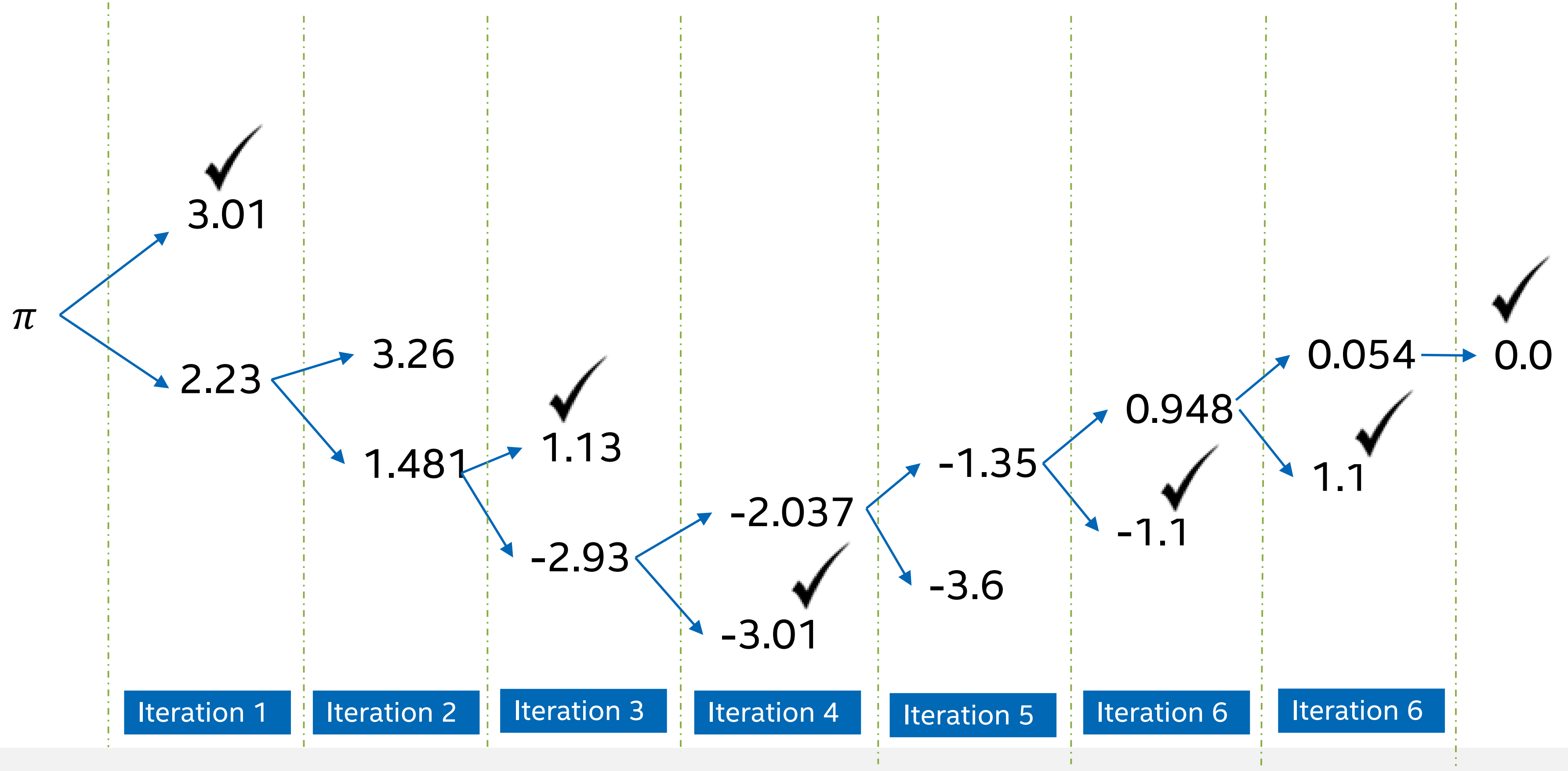
Example 2

$$f(x) = -3x^4\tanh(x) + 10x^3 - 9x$$

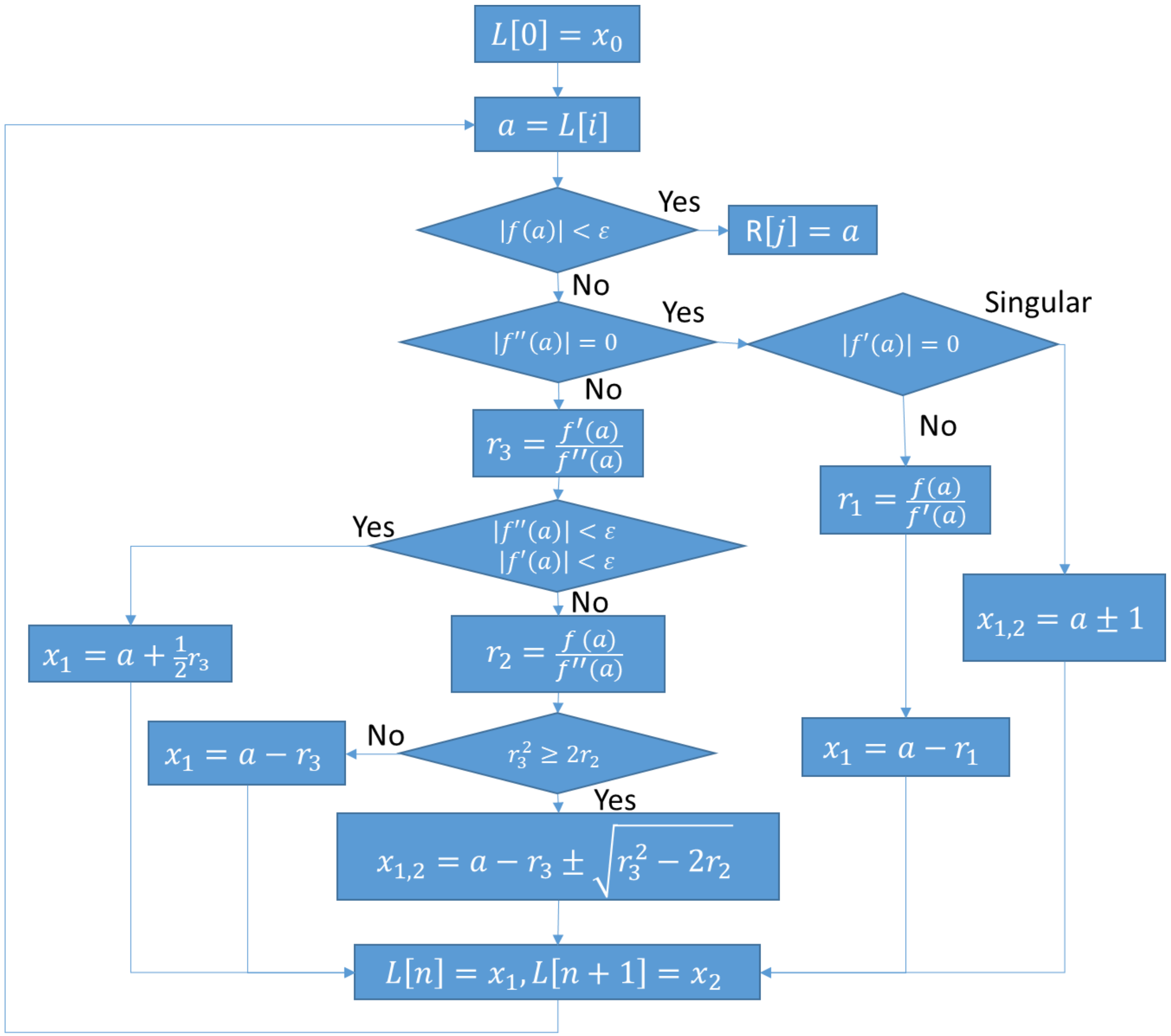


Example 2

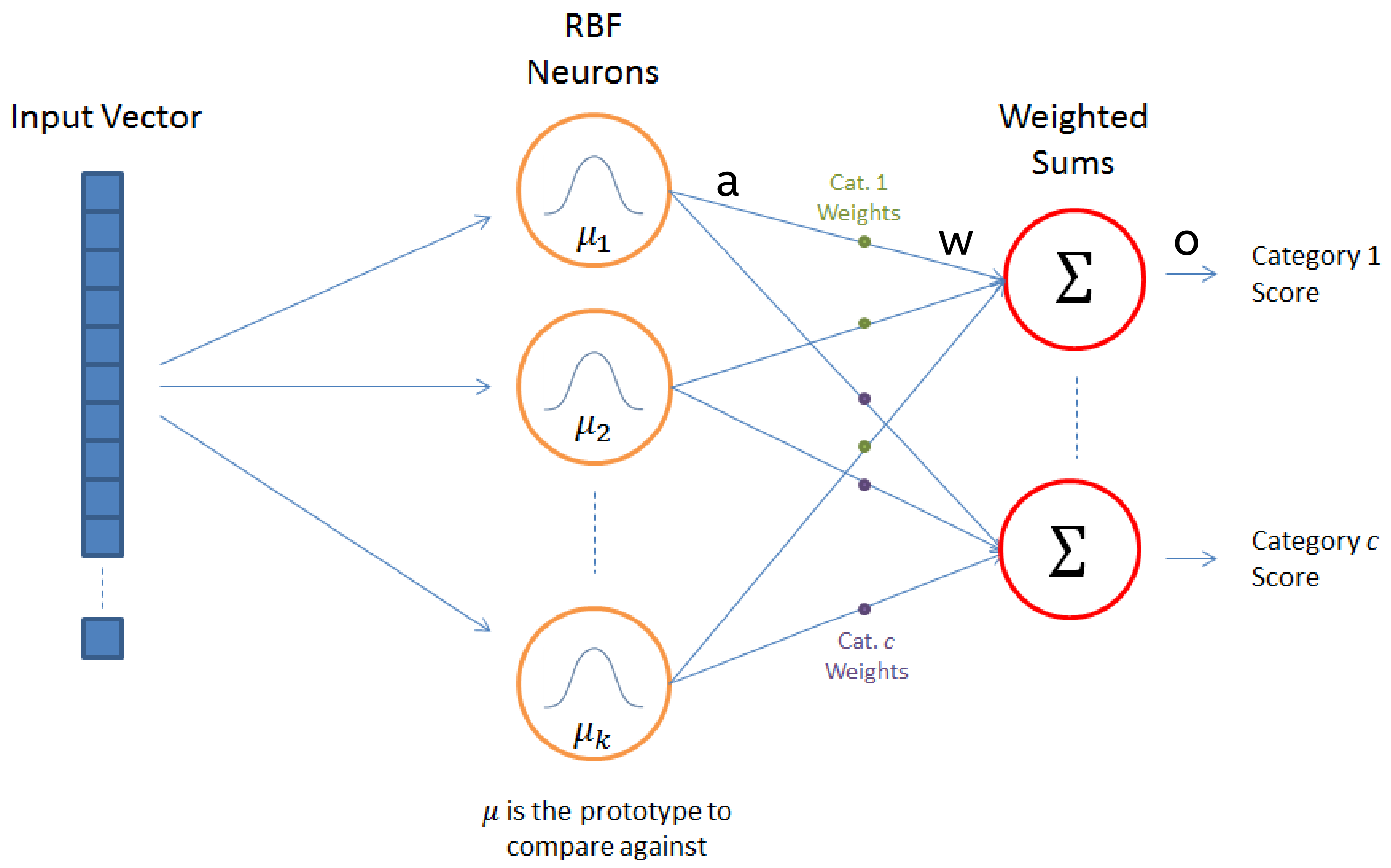
$$f(x) = -3x^4\tanh(x) + 10x^3 - 9x$$



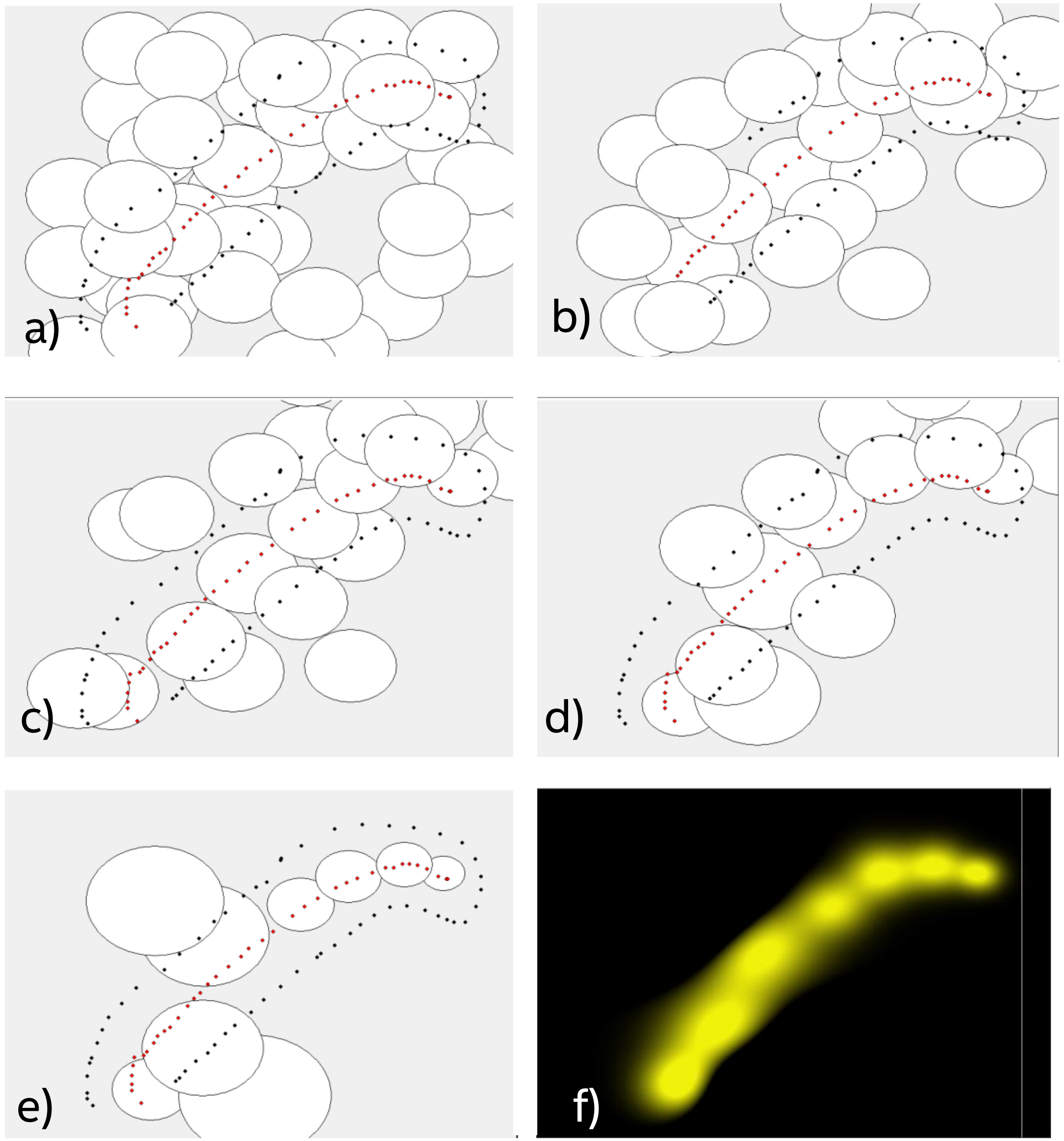
Root finding Algorithm



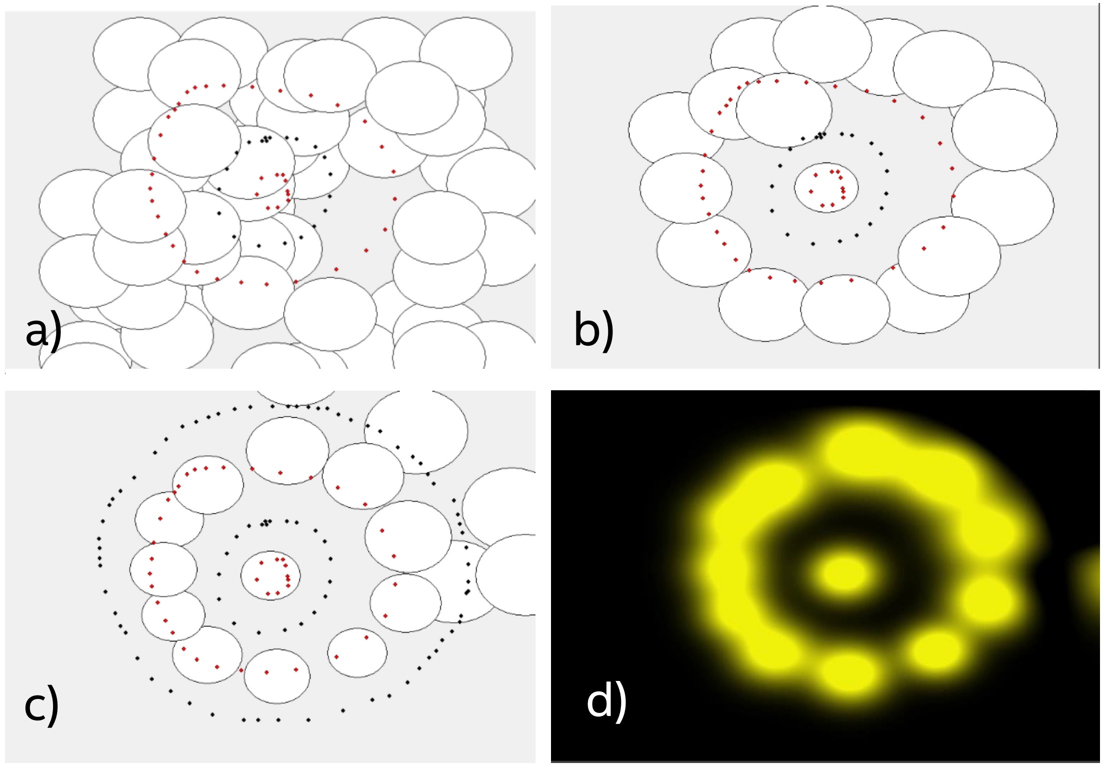
A Radial Basis Function NN



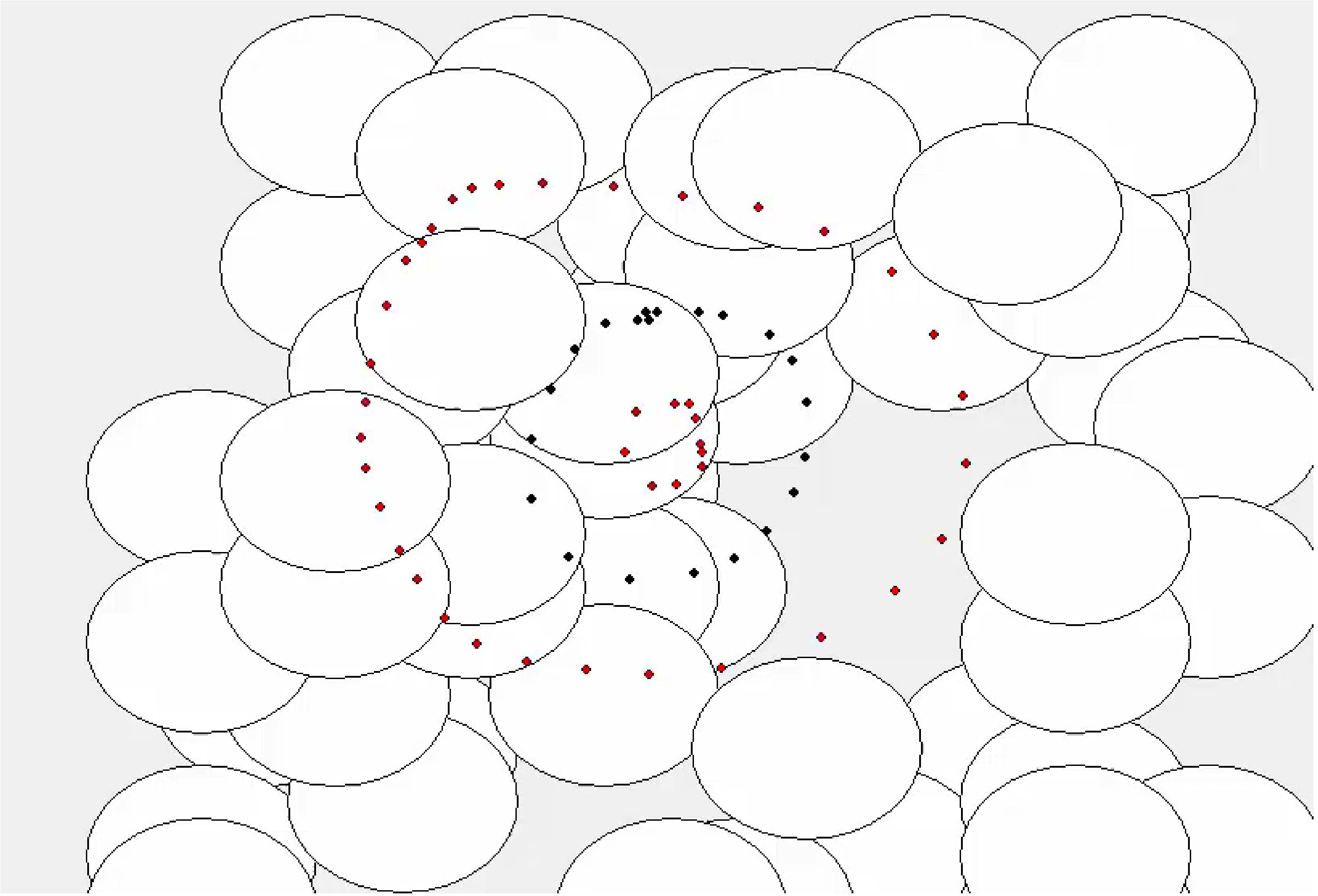
Topology Optimization



Topology Reduction

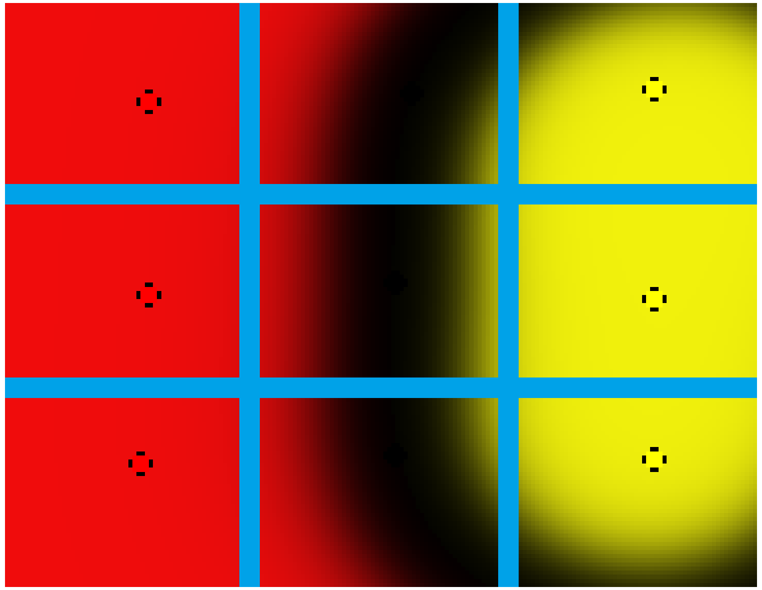


Toy scenario



Convolutional Kernel

3x3



The surface of the kernels can be now interpolated by a subset of neurons with less number of parameters

TABLE I
MNIST ACCURACY VS MEMORY

Neural Network	#Parameters	Accuracy
LeNet [14]	431K	99.4%
LetNet5 [15]	64K	99.24%
50-50-200-10NN [16]	226K	99.51%
Best Practices [17]	132.5K	99.5%
6-10-15-10	7.8K	99.31%
Our method	**5.4K**	99.20%