

# Root findings : equations $f(x) = 0$

- Polynomial equations : [✖ Bairstow's method](#) is an efficient algorithm for finding the roots of a real polynomial of arbitrary degree
  - [Polynomials in NumPy](#)
  - [polynomial module](#), including `polyroots(c)` to compute the roots of a polynomial.
- [✖ Bisection method](#) (dichotomy) : very simple and robust method, but relatively slow. It assumes continuity of the function, and obtain one roots. The algorithm is based on a [✖ loop invariant](#) property : an interval  $[a, b]$  is said to bracket a root if  $f(a)$  and  $f(b)$  have opposite signs.
- [✖ Secant method](#) (retains the last two computed points)
- [✖ Regula falsi](#) (retains the points which preserve bracketing)
- [✖ Newton-Raphson method](#)

From:

<https://dvillers.umons.ac.be/wiki/> - **Didier Villers, UMONS - wiki**

Permanent link:

[https://dvillers.umons.ac.be/wiki/teaching:methcalchim:root-finding\\_algorithm?rev=1539934671](https://dvillers.umons.ac.be/wiki/teaching:methcalchim:root-finding_algorithm?rev=1539934671)

Last update: **2018/10/19 09:37**

