LAK-ComputerMathematik

Projekt **GH-13**

Bestimmung aller Nullstellen

Stand: 19. Oktober 2026, 13:36

Konsultationen zum Projekt: nach Terminvereinbarung (E-mail)

Für eine, im Intervall $[a, b]$ stetige, Funktion $f(x)$ sind **alle reellen** Nullstellen im angegebenen Intervall (numerisch) zu bestimmen und in den Funktionsgraphen einzuzeichnen.

Beispiele für zu benutzende Funktionen:

- $p(x) := \cos(x) \cdot \left(x^4 - \frac{x^3}{2} + \frac{x^2}{2} - \frac{x}{2} - \frac{1}{2}\right)$ im Intervall $[-2.5, 2]$.
  Ohne den Kosinusfaktor ergeben sich die reellen Nullstellen $x = -0.5$ und $x = 1$.
- $q(x) := \sin(1/x)$ im Intervall $[\frac{1}{1000}, 1]$.
  Wieviele Nullstellen gibt es im Intervall (analytisch bestimmbar!)?
- $s(x) := \ln(1 + x^2) - \exp(-x) * \cos(x) + x^2 * \sin(x) - \ln(\sqrt{1010})$ im Intervall $[-5, 7]$.
  Es gibt 5 Nullstellen, eine ist $x = -2.222939$.
- Fakultativ: Ermöglichen Sie eine Eingabe der Funktion als String von der Tastatur (`input`[1]), mit anschließender Umwandlung in eine symbolische Funktion (`str2sym`[2]).

Wie bestimme ich alle Nullstellen im Interval $[a, b]$?

Es gibt keinen allgemeingültigen Algorithmus zur globalen Bestimmung aller Nullstellen einer allgemeinen Funktion $f(x)$ im Interval $[a, b]$ $(a < b)$.

Eine Lösungsidee (brute force):

(a) Wenn sich für eine stetige Funktion $f(x)$ in einem (Teil-)Intervall $[x_i, x_{i+1}] \subset [a, b]$ die Vorzeichen von $f(x_i)$ und $f(x_{i+1})$ unterscheiden, dann muß mindestens eine Nullstelle im Intervall zu finden sein.
Dann kann man, z.B., mit `solve` oder `vpasolve` oder Bisektion, eine (oder mehrere) dieser Nullstellen bestimmen.

(b) $\Longrightarrow$ Unterteilung von $[a, b]$ in n (gleichgroße) Teilintervalle und Bestimmung der Nullstelle(n) im Teilintervall.
Achten Sie darauf, daß die Lösung(en) von `vpasolve` im Teilintervall liegt (auch: Intervallangabe als optionaler Parameter möglich).

(c) Erhöhung der Intervallanzahl n ($*4$ oder $*10$) und wie unter (b) verfahren.

(d) Die Schritte (c) und (b) solange wiederholen bis sich die Anzahl der gefundenen Nullstellen nicht mehr ändert.

Auch dieser Algorithmus garantiert nicht, daß alle Nullstellen gefunden werde, er ist aber eine brauchbare Heuristik.
Bei mehr Information über die Funktion $f(x)$ und Ausnutzung dieser vorhandenen Strukturen kann man bessere Algorithmen zur Nullstellenbestimmung ableiten.

Hinweise: `syms`, `int`, `solve`, `vpasolve`, `real`, `vpa`, `double`, `matlabFunction`, `max`, `min`, `plot`, `fplot`, `fill`, `linspace`, `legend`, `title`, `input`, `str2sym`

---

[1] `https://de.mathworks.com/help/matlab/ref/input.html`
[2] `https://de.mathworks.com/help/symbolic/str2sym.html`