

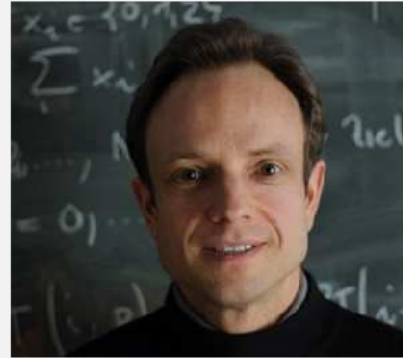
Course Video

Section 8.4.2.1: Newton Method in the Scalar Case

Prof. R. Hiptmair, SAM, ETH Zurich

Date: December 1, 2020

(C) Seminar für Angewandte Mathematik, ETH Zürich



Prerequisites.

- Differentiation: product rule and chain rule

Dependencies. Section 8.2.1, Section 8.2.2

Duration: 36 minutes

VIII Iterative Methods for Non-Linear Systems of Equations

8.4. Finding Zeros of Scalar Functions

Find $x^* \in I : f(x^*) = 0, f : I \subset \mathbb{R} \rightarrow \mathbb{R}$

8.4.2. Model Function Methods



Idea: Given recent iterates (approximate zeroes)

$$x^{(k)}, x^{(k-1)}, \dots, x^{(k-m+1)}, m \in \mathbb{N}$$

- 1 replace F with a k -dependent model function $\tilde{F}_k$

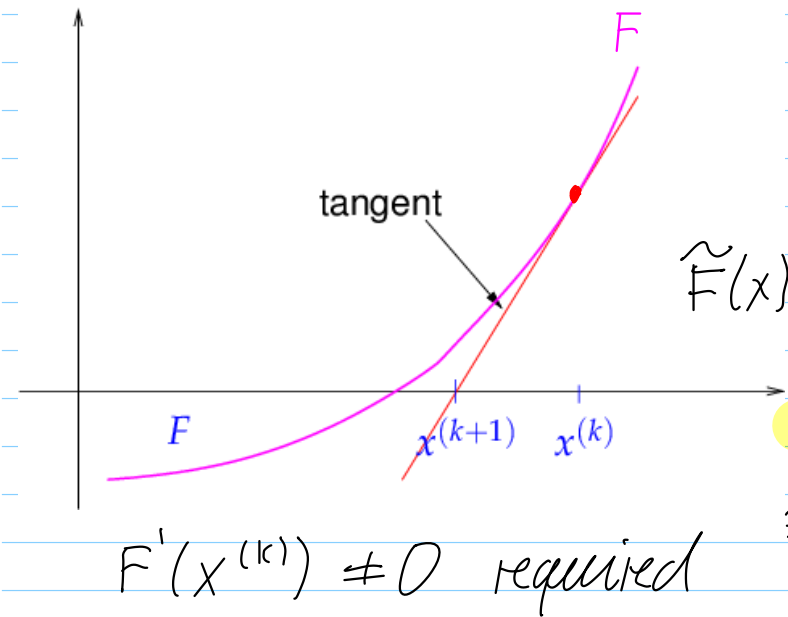
(based on function values $F(x^{(k)}), F(x^{(k-1)}), \dots, F(x^{(k-m+1)})$ and, possibly, derivative values $F'(x^{(k)}), F'(x^{(k-1)}), \dots, F'(x^{(k-m+1)})$)

- 2 $x^{(k+1)} := \text{zero of } \tilde{F}_k: \tilde{F}_k(x^{(k+1)}) = 0$

(has to be readily available $\leftrightarrow$ analytic formula)

Locally around $x^{(k)}$: $F(x) \approx F_k(x)$

8.4.2.1. Newton Method in the Scalar Case



Assume: $f \in C^1$

Model function $\tilde{F}$

$\hat{=}$ tangent at G_f in $(x^{(k)}, F(x^{(k)}))$

$$\tilde{F}(x) = F(x^{(k)}) + F'(x^{(k)})(x - x^{(k)})$$

$$x^{(k+1)} = x^{(k)} - \frac{F(x^{(k)})}{F'(x^{(k)})}$$

$\hat{=}$ Newton iteration
Newton correction

Ex. 8.4.2.3 (Square-root iteration as a Newton iteration)

$$F(x) = x^2 - a, \quad a > 0$$

$$F'(x) = 2x$$

$$\triangleright x^{(k+1)} = x^{(k)} - \frac{(x^{(k)})^2 - a}{2x^{(k)}} = \frac{1}{2} \left(x^{(k)} + \frac{a}{x^{(k)}} \right),$$

Observed: quadratic conv.

§ 8.4.2.5 (Convergence of Newton's method in 1D)

Newton iteration as Fixed-point iteration

$$x^{(k+1)} = x^{(k)} - \frac{F(x^{(k)})}{F'(x^{(k)})} = \phi(x^{(k)}), \quad \phi(x) = x - \frac{F(x)}{F'(x)}$$

C++11-code 8.4.2.2: Newton method in the scalar case $n = 1$

```

1 template <typename FuncType, typename DervType, typename Scalar>
2 Scalar newton1D(FuncType &&F, DervType &&DF,
3 const Scalar &x0, double rtol, double atol)
4 {
5     // initial guess
6     Scalar s, x = x0;
7     do {
8         s = F(x)/DF(x); // compute Newton correction
9         x -= s;          // compute next iterate
10    }
11    // correction based termination (relative and absolute)
12    while ((std::abs(s) > rtol*std::abs(x)) && (std::abs(s) > atol));
13    return (x);
14 }
```

Lemma 8.3.2.15. Higher order local convergence of fixed point iterations

If $\Phi : U \subset \mathbb{R} \mapsto \mathbb{R}$ is $m+1$ times continuously differentiable, $\Phi(x^*) = x^*$ for some x^* in the interior of U , and $\Phi^{(l)}(x^*) = 0$ for $l = 1, \dots, m$, $m \geq 1$, then the fixed point iteration (8.3.1.2) converges locally to x^* with order $\geq m+1$ ($\rightarrow$ Def. 8.2.2.10).

③

$$\phi'(x) = 1 - \frac{F'(x)^2 - F(x)F''(x)}{F'(x)^2} = \frac{F(x)F''(x)}{F'(x)^2}$$

$$\left. \begin{array}{l} F(x^*) = 0 \\ F'(x^*) \neq 0 \end{array} \right\} \Rightarrow \phi'(x^*) = 0$$

$$\bullet |x^{(k+1)} - x^*| = |\phi(x^{(k)}) - \phi(x^*)|$$

$$\text{Taylor for } \phi \in C^2: = |\phi'(x^*)| |x^{(k)} - x^*| + O(|x^{(k)} - x^*|^2)$$

Convergence of Newton's method in 1D

Newton's method locally quadratically converges ($\rightarrow$ Def. 8.2.2.10) to a zero x^* of F , if $F'(x^*) \neq 0$

Ex. 8.4.2.4: (Newton method in 1D)

$$F(x) = xe^x - 1 \Rightarrow F'(x) = e^x(1+x) \Rightarrow x^{(k+1)} = x^{(k)} - \frac{x^{(k)}e^{x^{(k)}} - 1}{e^{x^{(k)}}(1+x^{(k)})} = \frac{(x^{(k)})^2 + e^{-x^{(k)}}}{1+x^{(k)}}$$

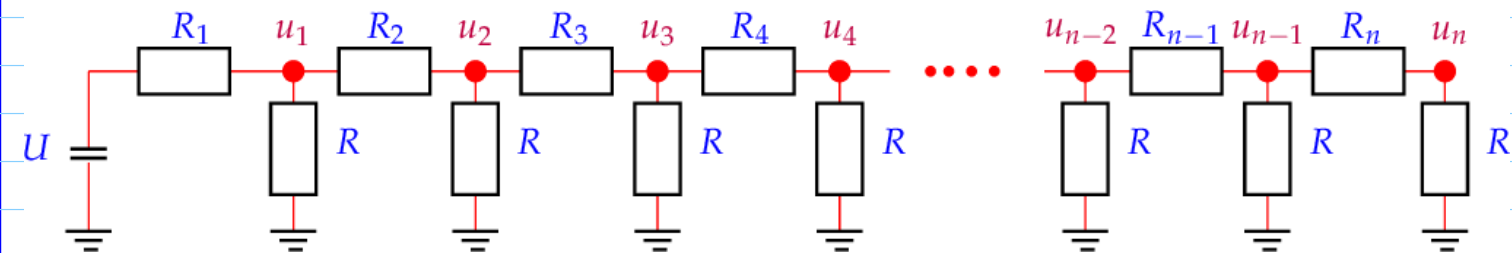
$$F(x) = x - e^{-x} \Rightarrow F'(x) = 1 + e^{-x} \Rightarrow x^{(k+1)} = x^{(k)} - \frac{x^{(k)} - e^{-x^{(k)}}}{1 + e^{-x^{(k)}}} = \frac{1 + x^{(k)}}{1 + e^{x^{(k)}}}$$

↑
same x^*

↑
different Newton iterations

Ex 8.4.2.9. (Implicit differentiation of F)

Linear circuit:



Find R such that potential $u_j = 1$, $j \in \{1, \dots, n\}$

Nodal analysis:

$$\text{Node 1: } \frac{1}{R_1}(u_1 - U) + \frac{1}{R}u_1 + \frac{1}{R_2}(u_1 - u_2) = 0,$$

$$\text{Node } j: \frac{1}{R_{j-1}}(u_j - u_{j-1}) + \frac{1}{R}u_j + \frac{1}{R_j}(u_j - u_{j+1}) = 0, \quad j = 2, \dots, n-1, \quad (8.4.2.1b)$$

$$\text{Node } n: \frac{1}{R_n}(u_n - u_{n-1}) + \frac{1}{R}u_n = 0.$$

$$\left(\mathbf{A} + \frac{1}{R} \mathbf{I} \right) \mathbf{u} = \mathbf{b}, \quad (8.4.2.1c)$$

$$\mathbf{A} = \begin{bmatrix} \frac{1}{R_1} + \frac{1}{R_2} & -\frac{1}{R_2} & & & \\ -\frac{1}{R_2} & \frac{1}{R_2} + \frac{1}{R_3} & -\frac{1}{R_3} & & \\ & -\frac{1}{R_3} & \frac{1}{R_3} + \frac{1}{R_4} & -\frac{1}{R_4} & \\ & & \ddots & \ddots & \ddots \\ & & & -\frac{1}{R_{n-2}} & \frac{1}{R_{n-2}} + \frac{1}{R_{n-1}} & -\frac{1}{R_{n-1}} \\ & & & & -\frac{1}{R_{n-1}} & \frac{1}{R_n} + \frac{1}{R_{n-1}} & -\frac{1}{R_n} \\ & & & & & -\frac{1}{R_n} & \frac{1}{R_n} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \frac{U}{R_1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

④ Find $x := 1/R > 0$:

$$F(x) = 0 \text{ with } F: \begin{cases} \mathbb{R} \rightarrow \mathbb{R} \\ x \mapsto \underbrace{\mathbf{w}^T (\mathbf{A} + x\mathbf{I})^{-1} \mathbf{b}}_{\underline{u}} - 1 \end{cases}, \quad (8.4.2.12)$$

$= \vec{e}_j$ (coordinate vector) $\rightarrow$

$$\begin{cases} F(x) := \underline{w}^T \underline{u}(x) - 1 \\ (\mathbf{A} + x\mathbf{I}) \underline{u}(x) = \underline{b} \end{cases}, \quad x > 0$$

$F'(x)$? Trick: *Implicit differentiation*

$$\frac{d}{dx}: (\mathbf{A} + x\mathbf{I}) \cdot \underline{u}(x) = \underline{b}$$

product rule $\mathbf{I} \cdot \underline{u}(x) + (\mathbf{A} + x\mathbf{I}) \underline{u}'(x) = 0$

$$\triangleright \underline{u}'(x) = -(\mathbf{A} + x\mathbf{I})^{-1} \underline{u}(x)$$

$$\triangleright F'(x) = \mathbf{w}^T \underline{u}'(x)$$

$\triangleright$ Newton iteration

2 LSE with tridiagonal matrices

$$\begin{aligned} x^{(k+1)} &= x^{(k)} + F'(x^{(k)})^{-1} F(x^{(k)}) \\ &= \frac{\mathbf{w}^T \underline{u}(x^{(k)}) - 1}{\mathbf{w}^T \underline{z}}, \quad \begin{aligned} (\mathbf{A} + x^{(k)}\mathbf{I}) \underline{u}(x^{(k)}) &= \underline{b}, \\ (\mathbf{A} + x^{(k)}\mathbf{I}) \underline{z} &= \underline{u}(x^{(k)}). \end{aligned} \end{aligned} \quad (8.4.2.14)$$

§ 8.4.2.18 (Preconditioning of Newton's method)

$$\text{If } F \approx \hat{F} \Rightarrow g(x) := \hat{F}^{-1}(F(x)) \approx x$$

$\uparrow$ Assume: $\hat{F}^{-1}$ can be computed easily

$$F(x) = 0 \Leftrightarrow G(x) := g(x) - \hat{F}^{-1}(0) = 0$$

G is "almost linear": expect large domain of quadratic convergence

$$(\hat{F}^{-1})' = \frac{1}{\hat{F}' \circ \hat{F}^{-1}} \Rightarrow g'(x) = \frac{F'(x)}{\hat{F}'(g(x))}$$

$\triangleright$ Newton it. for G :

$$x^{(k+1)} = x^{(k)} - \frac{G(x^{(k)}) \hat{F}'(g(x^{(k)}))}{F'(x^{(k)})}, \quad (8.4.2.19)$$

$$g(x^{(k)}) = \hat{F}^{-1}(F(x^{(k)})), \quad G(x^{(k)}) = g(x^{(k)}) - y_0, \quad y_0 := \hat{F}^{-1}(0).$$

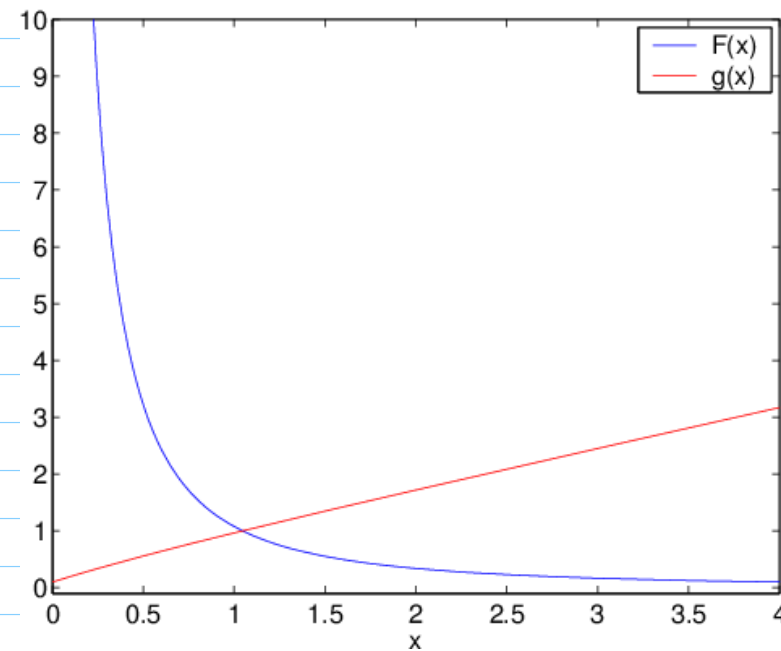
5

Ex. 8.4.2.20 (Preconditioned Newton method)

$$F(x) = \frac{1}{(x+1)^2} + \frac{1}{(x+0.1)^2} - 1, \quad x > 0$$

$$F(x) \approx \hat{F}(x) = \frac{1}{x^2} - 1$$

$$\hat{F}'(y) = \frac{1}{\sqrt{y+1}}$$

Convergence recorded for $x^{(0)} = 0$:

k	$x^{(k)}$	$F(x^{(k)})$	$x^{(k)} - x^{(k-1)}$	$x^{(k)} - x^*$
1	0.91312431341979	0.24747993091128	0.91312431341979	-0.13307818147469
2	1.04517022155323	0.00161402574513	0.13204590813344	-0.00103227334125
3	1.04620244004116	0.00000008565847	0.00103221848793	-0.00000005485332
4	1.04620249489448	0.00000000000000	0.00000005485332	-0.00000000000000

▷ Review questions 8.4.2.16

7