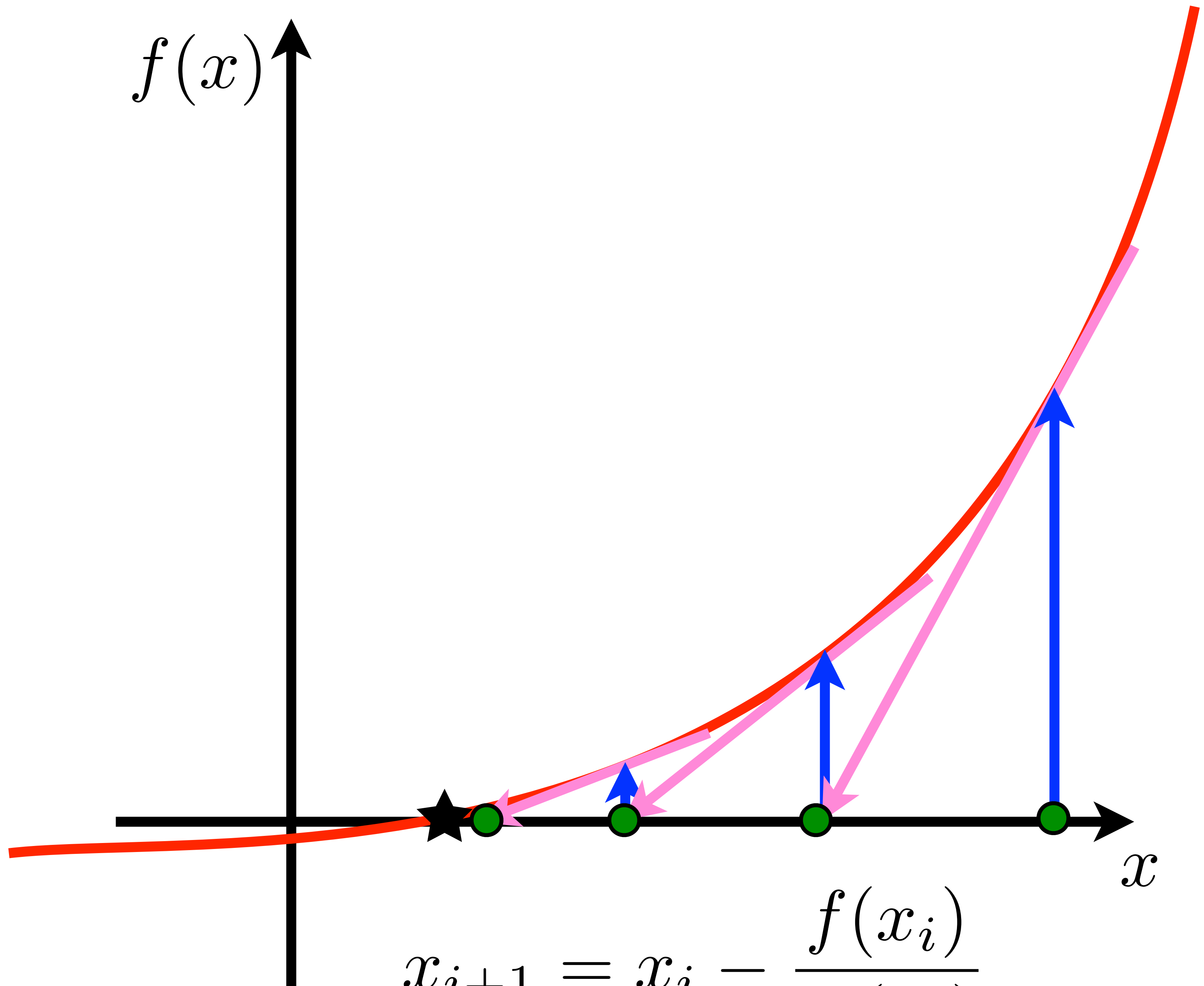
10.34: Numerical Methods Applied to Chemical Engineering

Lecture 8:
Quasi-Newton-Raphson methods

Recap

- Solutions of nonlinear equations
- The Newton-Raphson method

Recap



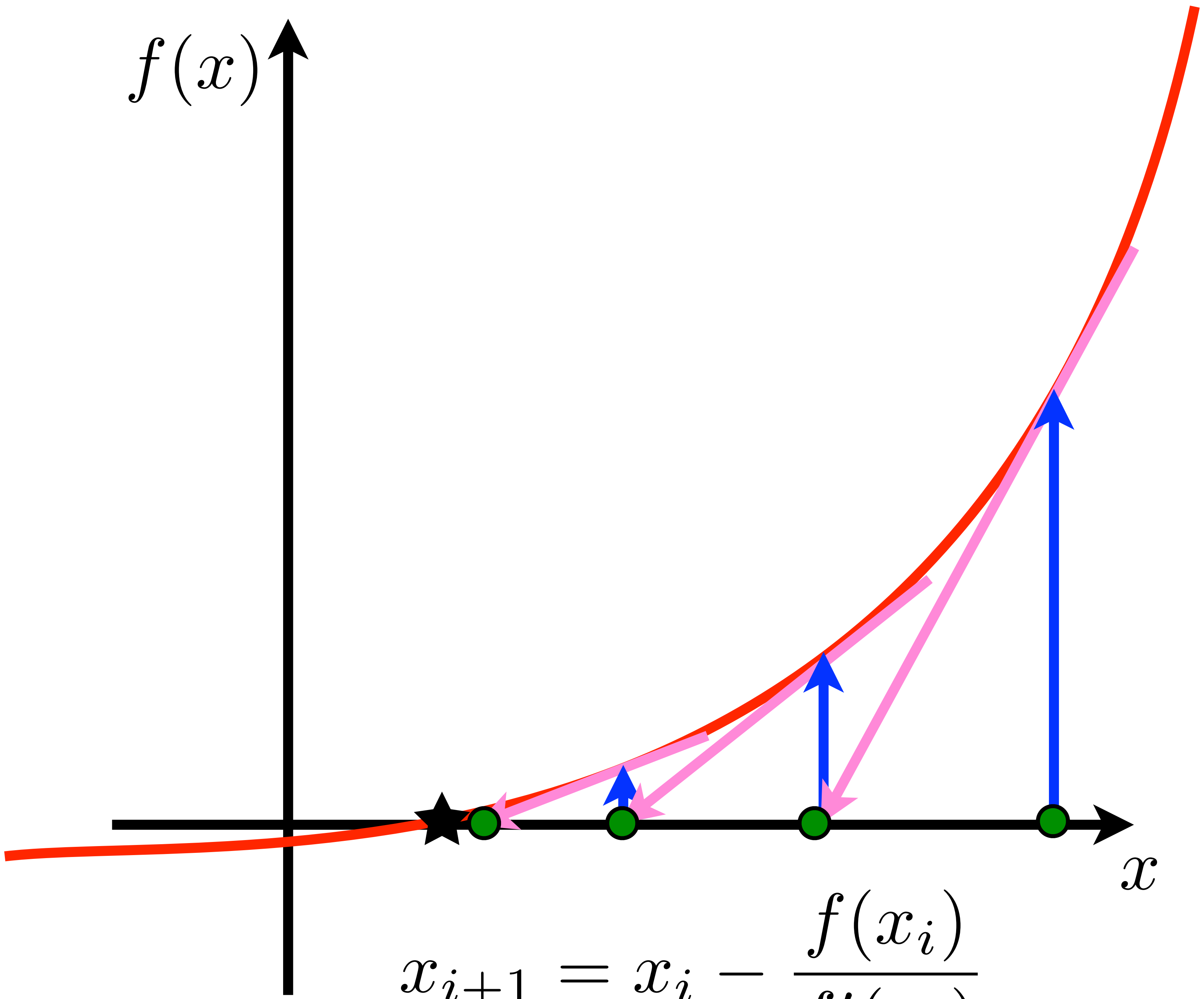
$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

Recap

- Derive the Babylonian method for finding square roots. Apply the Newton-Raphson method to find the roots of the equation:

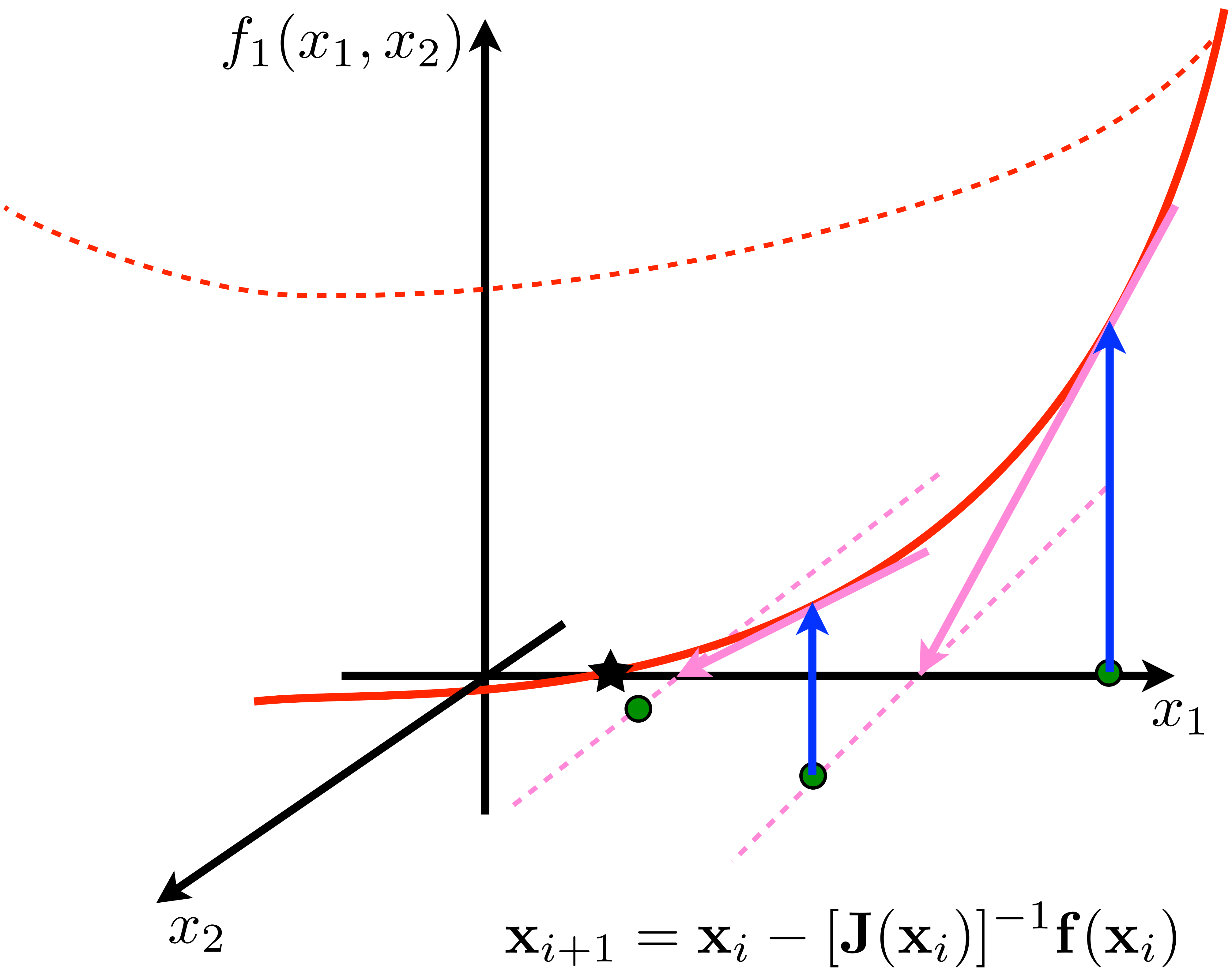
$$f(x) = x^2 - S$$

Recap



$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

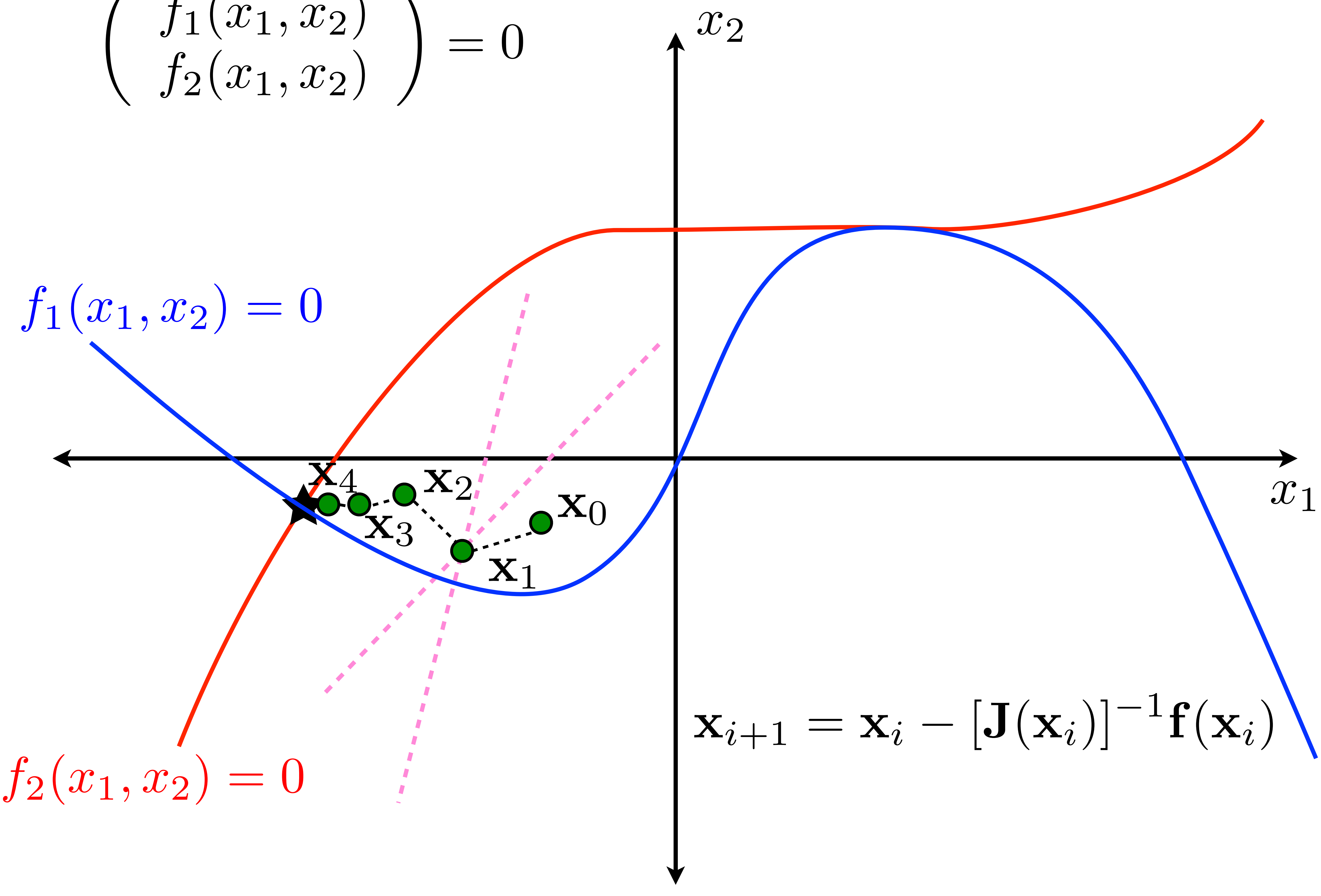
Recap



$$\mathbf{x}_{i+1} = \mathbf{x}_i - \left[\mathbf{J}(\mathbf{x}_i)\right]^{-1}\mathbf{f}(\mathbf{x}_i)$$

Recap

$$\begin{pmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{pmatrix} = 0$$



$$\mathbf{x}_{i+1} = \mathbf{x}_i - [\mathbf{J}(\mathbf{x}_i)]^{-1}\mathbf{f}(\mathbf{x}_i)$$

Convergence of NR Method

- The Newton-Raphson method converges quadratically.
- Proof for the 1-D case:
  - $$|x_{i+1} - x^*| = \left| x_i - \frac{f(x_i)}{f'(x_i)} - x^* \right|$$
  - Recall that:

$$f(x^*) = 0 = f(x_i) + f'(x_i)(x^* - x_i) + \frac{1}{2} f''(x_i)(x^* - x_i)^2 + \ldots$$

  - Therefore:

$$|x_{i+1} - x^*| = \left| \frac{1}{2} \frac{f''(x_i)}{f'(x_i)} (x_i - x^*)^2 \right| + O((x_i - x^*)^3)$$

  - When the Newton-Raphson method converges:

$$\lim_{i \to \infty} \frac{|x_{i+1} - x^*|}{|x_i - x^*|^2} \leq \left| \frac{1}{2} \frac{f''(x^*)}{f'(x^*)} \right|$$

- This holds as long as $f'(x^*) \neq 0$

Convergence of NR Method

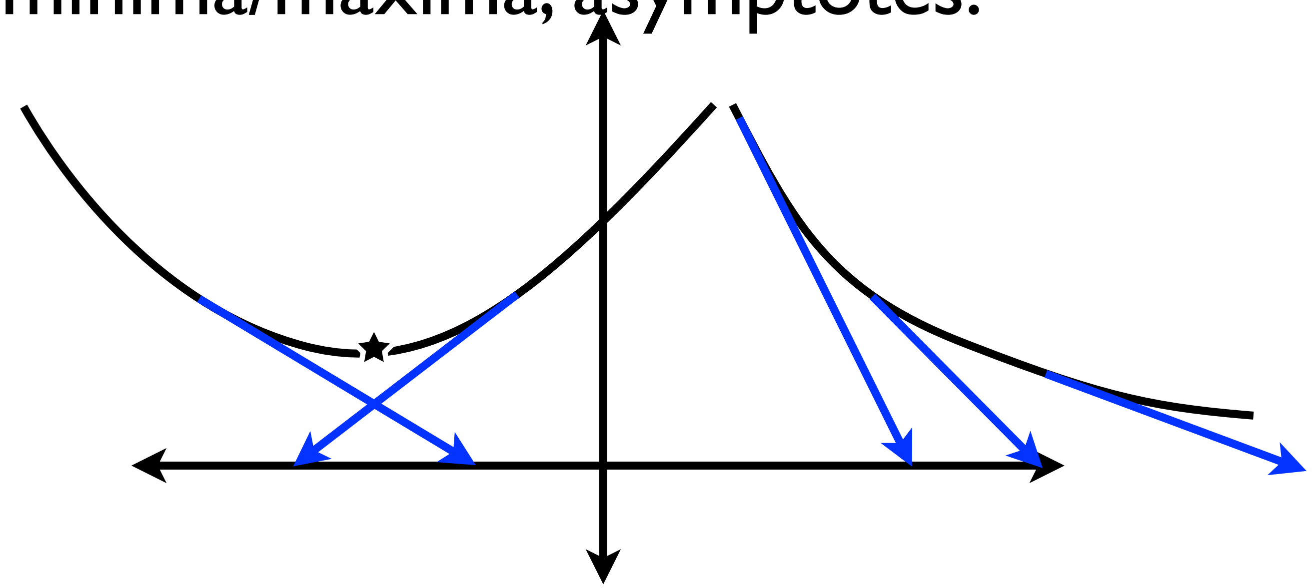
- The Newton-Raphson method converges quadratically:

$$\lim_{i\to\infty} \frac{\|\mathbf{x}_{i+1} - \mathbf{x}^*\|_p}{\|\mathbf{x}_{i+1} - \mathbf{x}^*\|_p^2} = C$$

  - as long as the Jacobian is not singular: $\det \mathbf{J}(\mathbf{x}^*) \neq 0$
- When the Jacobian is singular, linear convergence occurs.
- Notice that quadratic convergence is guaranteed only when the iterates are sufficiently close to the root.
  - Good initial guesses are essential to the success of the Newton-Raphson method. It is locally convergent!
  - Bad initial guesses can lead to a chaotic series of iterates which may or may not converge at all.

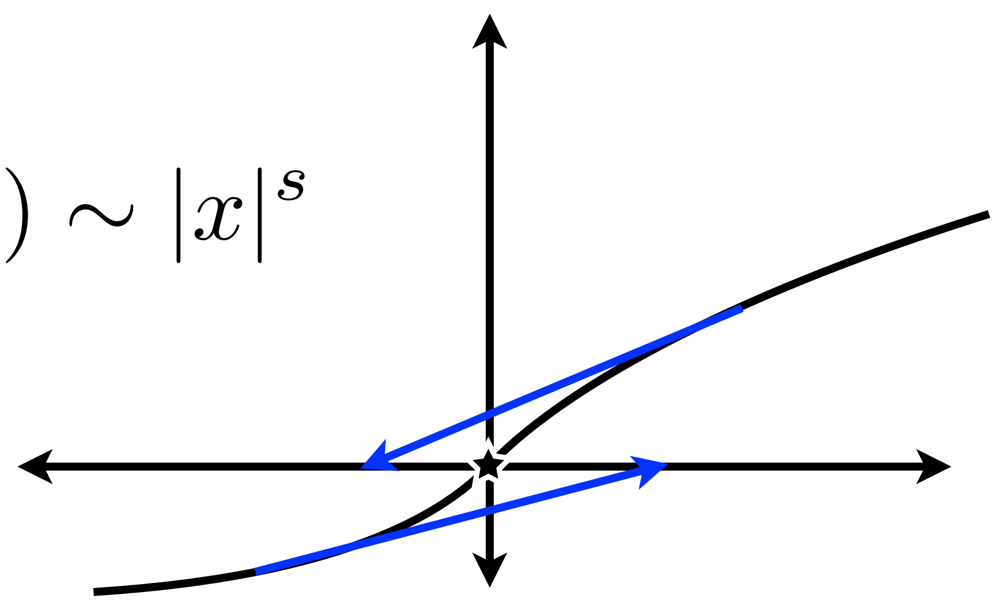
Failures of NR Method

- Example:
  - Local minima/maxima, asymptotes:



  - Overshoot:

$$f(x) \sim |x|^s$$



diverge

$$0 < s < 1/2$$

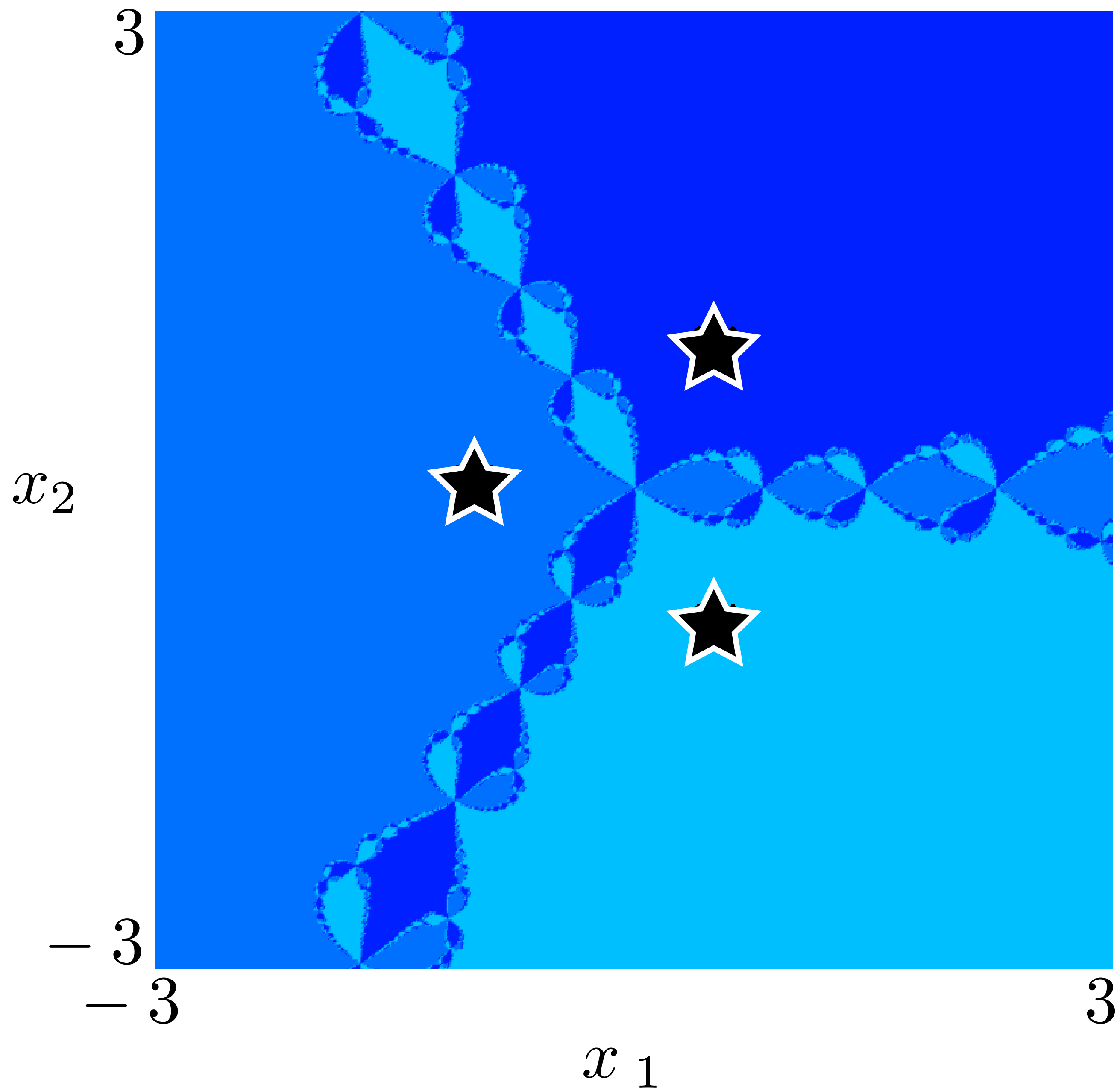
converge

$$1/2 < s < 1$$

Failures of NR Method

- Basins of attraction:

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} x_1^3 - 3x_1x_2^2 + 1 \\ x_2^3 - 3x_1x_2 \end{pmatrix}$$



Failures of NR Method

- Other problems with Newton-Raphson method:
  - The Jacobian may not be easy to calculate analytically.
    - What are possible sources for $\mathbf{f}(\mathbf{x})$?
  - Inverting the Jacobian many times may be too costly computationally.
    - What are some options for mitigating this?
  - The Newton-Raphson step may not converge to the nearest root to the initial guess.
    - overshoot/basins of attraction
- There are modifications to the Newton-Raphson method that can correct some of these issues.

Quasi-NR Methods

- There are modifications to the Newton-Raphson method that can correct some of these issues.
  - The penalty for modifying the Newton-Raphson method is a reduction in the convergence rate.
  - Newton-Raphson is based on a linear approximation of the function near the root. Quasi-NR methods reduce the accuracy of that approximation.
- Finite-difference approximation of Jacobian
- Broyden's method for approximating inverse Jacobian
- Damped NR-methods

Calculation of Jacobian

- Analytical calculation of the Jacobian requires an analytical formula for $\mathbf{f}(\mathbf{x})$.
  - For functions of a few dimensions, analytical calculations are easy.
  - For functions of many dimensions, this can be tedious at best and error prone at worst.
- Often, an analytical formulas for $\mathbf{f}(\mathbf{x})$or a few dimensions of $\mathbf{f}(\mathbf{x})$ are not available.
  - These function values might come from:
    - interpolation of data
    - results of simulations
- Is there an alternative way to compute the Jacobian?

Finite Differences

- Finite difference approximation of derivatives:

$$f'(x) = \frac{f(x+\epsilon) - f(x)}{\epsilon} + O(\epsilon)$$

- Accuracy depends on ϵ , but in a non-intuitive way
- Example: $f(x) = e^x$

$$f'(1) = e^1 \approx \frac{e^{1+\epsilon} - e^1}{\epsilon}$$

ϵ	$\lvert f'(1) - \exp(1) \rvert$	
10^{-3}	1.36×10^{-3}	truncation error in approximation of derivative
10^{-4}	1.36×10^{-4}	
10^{-5}	1.36×10^{-5}	
10^{-6}	1.36×10^{-6}	
10^{-7}	1.36×10^{-7}	
10^{-8}	5.10×10^{-8}	truncation error in calculation of difference
10^{-9}	2.28×10^{-7}	
10^{-10}	2.89×10^{-6}	

Finite Differences

- Finite difference approximation of derivatives:

$$f'(x) = \frac{f(x+\epsilon) \quad\quad f(x)}{\epsilon} + O(\epsilon)$$

- Accuracy depends on ϵ, but in a non-intuitive way
- Example: $f(x) = e^x$

$$f'(1) = e^1 \approx \frac{e^{1+\epsilon} - e^1}{\epsilon}$$

ϵ	$\lvert f'(1) - \exp(1) \rvert$
10^{-3}	1.36×10^{-3}
10^{-4}	1.36×10^{-4}
10^{-5}	1.36×10^{-5}
10^{-6}	1.36×10^{-6}
10^{-7}	1.36×10^{-7}
10^{-8}	5.10×10^{-8}
10^{-9}	2.28×10^{-7}
10^{-10}	2.89×10^{-6}

Error in finite difference is minimized when:

$$\epsilon \approx \sqrt{\epsilon_M}|x| \approx 10^{-8}|x|$$

FD Approximation of Jacobian

- The elements of the Jacobian are: $J_{ij}(\mathbf{x}) = \dfrac{\partial f_i}{\partial x_j}$
- These can be approximated by finite differences as:

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(\mathbf{x} + \epsilon \mathbf{e}_j) - f_i(\mathbf{x})}{\epsilon}$$

  - where $\mathbf{e}_j$ is a unit vector for which $\mathbf{x} \cdot \mathbf{e}_j = x_j$

- Equivalently, the columns of the Jacobian can be evaluated as:

$$\mathbf{J}_j^C = \frac{\mathbf{f}(\mathbf{x} + \epsilon \mathbf{e}_j) - \mathbf{f}(\mathbf{x})}{\epsilon}$$

- How many function evaluations does it take to calculate the Jacobian at a single point?
- How will approximation of the Jacobian affect convergence?

FD Approximation of Jacobian

- Example:
  - A MATLAB function that does the function evaluation:

```
function f = my_func( x )
    f = %Whatever this function does;
```

  - A MATLAB function that calculates the Jacobian

```
function J = my_jacobian( x )
    J = zeros( length( x ), length( x ) );
    for i = 1:length(x)
        dx = x;  eps = 10^-8 * x(i);
        dx( i ) = dx( i ) + eps;
        J( :, i ) = ( my_func( dx ) - my_func( x ) ) / eps;
    end;
```

Broyden's Method

- The Secant method is a special case of Newton-Raphson that uses a coarse approximation of the derivative:

$$x_{i+1} = x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}$$

- Can this be extended to many dimensions?
  - If I know $\mathbf{x}_i, \mathbf{x}_{i-1}, \ \mathbf{f}(\mathbf{x}_i), \ \mathbf{f}(\mathbf{x}_{i-1})$, can I approximate $\mathbf{J}(\mathbf{x}_i)$?
  - 1-D secant approximation:

$$f'(x_i)(x_i - x_{i-1}) = f(x_i) - f(x_{i-1})$$

  - N-D secant approximation:

$$\mathbf{J}(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{x}_{i-1}) = \mathbf{f}(\mathbf{x}_i) - \mathbf{f}(\mathbf{x}_{i-1})$$

Broyden's Method

- Underdetermined secant approximation for Jacobian:

$$\mathbf{J}(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{x}_{i-1}) = \mathbf{f}(\mathbf{x}_i) - \mathbf{f}(\mathbf{x}_{i-1})$$

- Newton's method for: $\mathbf{x}_i$

$$\mathbf{J}(\mathbf{x}_{i-1})(\mathbf{x}_i - \mathbf{x}_{i-1}) = -\mathbf{f}(\mathbf{x}_{i-1})$$

- Take the difference:

$$\left(\mathbf{J}(\mathbf{x}_i) - \mathbf{J}(\mathbf{x}_{i-1})\right)(\mathbf{x}_i - \mathbf{x}_{i-1}) = \mathbf{f}(\mathbf{x}_i)$$

  - Still underdetermined! One possible solution:
    - Let: $\mathbf{J}(\mathbf{x}_i) - \mathbf{J}(\mathbf{x}_{i-1}) = \dfrac{\mathbf{f}(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{x}_{i-1})^T}{\|\mathbf{x}_i - \mathbf{x}_{i-1}\|_2^2}$

- Iterative form for approximation of Jacobian:

$$\mathbf{J}(\mathbf{x}_i) = \mathbf{J}(\mathbf{x}_{i-1}) + \frac{\mathbf{f}(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{x}_{i-1})^T}{\|\mathbf{x}_i - \mathbf{x}_{i-1}\|_2^2}$$

Broyden's Method

- Rank-1 update approximation:

$$\mathbf{J}(\mathbf{x}_i) = \mathbf{J}(\mathbf{x}_{i\quad 1}) + \frac{\mathbf{f}(\mathbf{x}_i)(\mathbf{x}_i \quad \mathbf{x}_{i\quad 1})^T}{\|\mathbf{x}_i \quad \mathbf{x}_{i\quad 1}\|_2^2}$$

- Useful for calculating $\mathbf{J}(\mathbf{x}_i)^{-1}$ as well

- Sherman-Morrison formula :

$$\left(\mathbf{A} + \mathbf{u}\mathbf{v}^T\right)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^T\mathbf{A}^{-1}}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}}$$

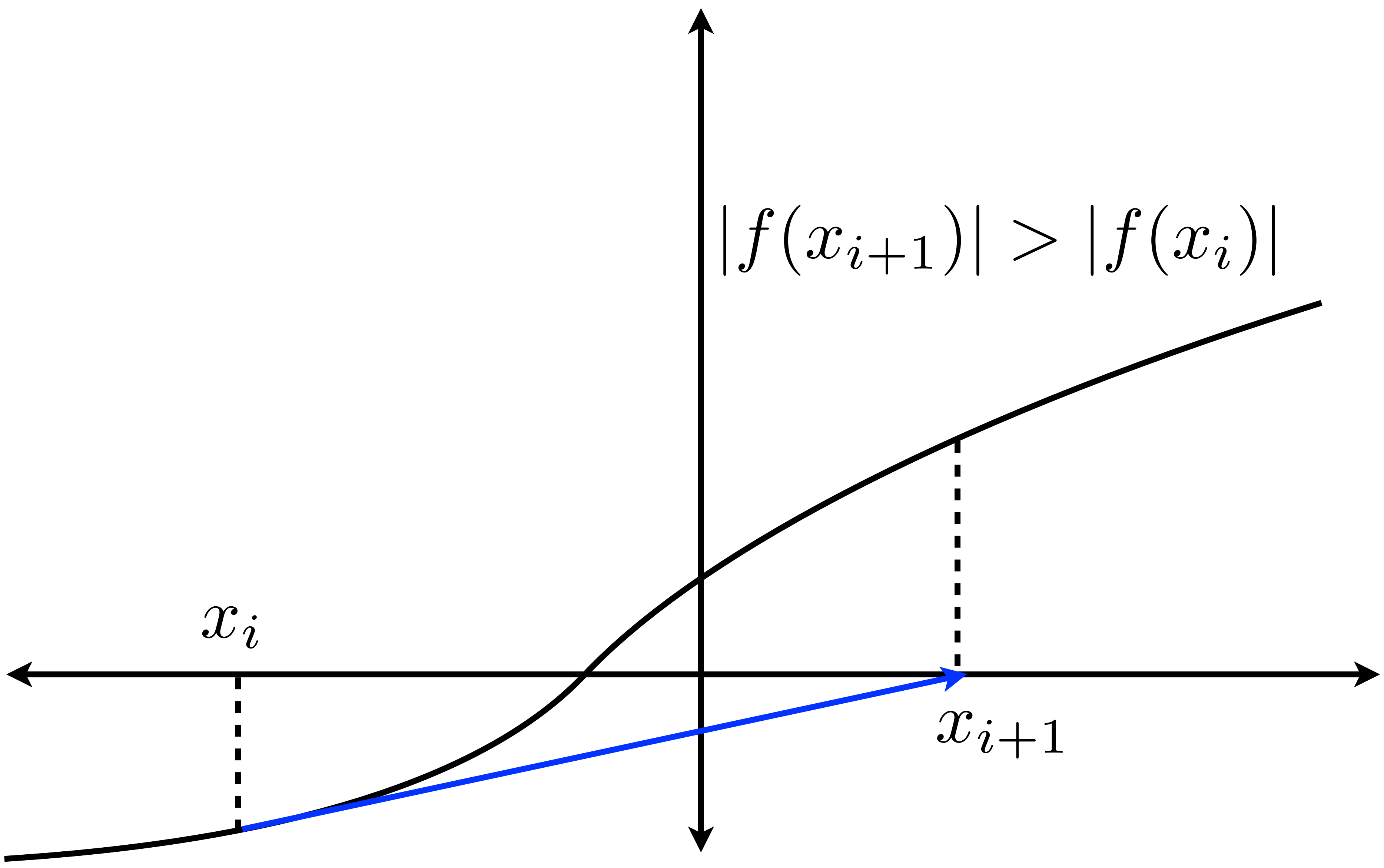
  - Applied to rank-1 update:

$$\mathbf{J}(\mathbf{x}_i)^{-1} = \mathbf{J}(\mathbf{x}_{i-1})^{-1} - \frac{\mathbf{J}(\mathbf{x}_{i-1})^{-1} f(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{x}_{i-1})^T \mathbf{J}(\mathbf{x}_{i-1})^{-1}}{\|\mathbf{x}_i - \mathbf{x}_{i-1}\|_2^2 + (\mathbf{x}_i - \mathbf{x}_{i-1})^T \mathbf{J}(\mathbf{x}_{i-1})^{-1}\mathbf{f}(\mathbf{x}_i)}$$

  - An iterative formula for the Jacobian inverse!

Damped NR Method

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$



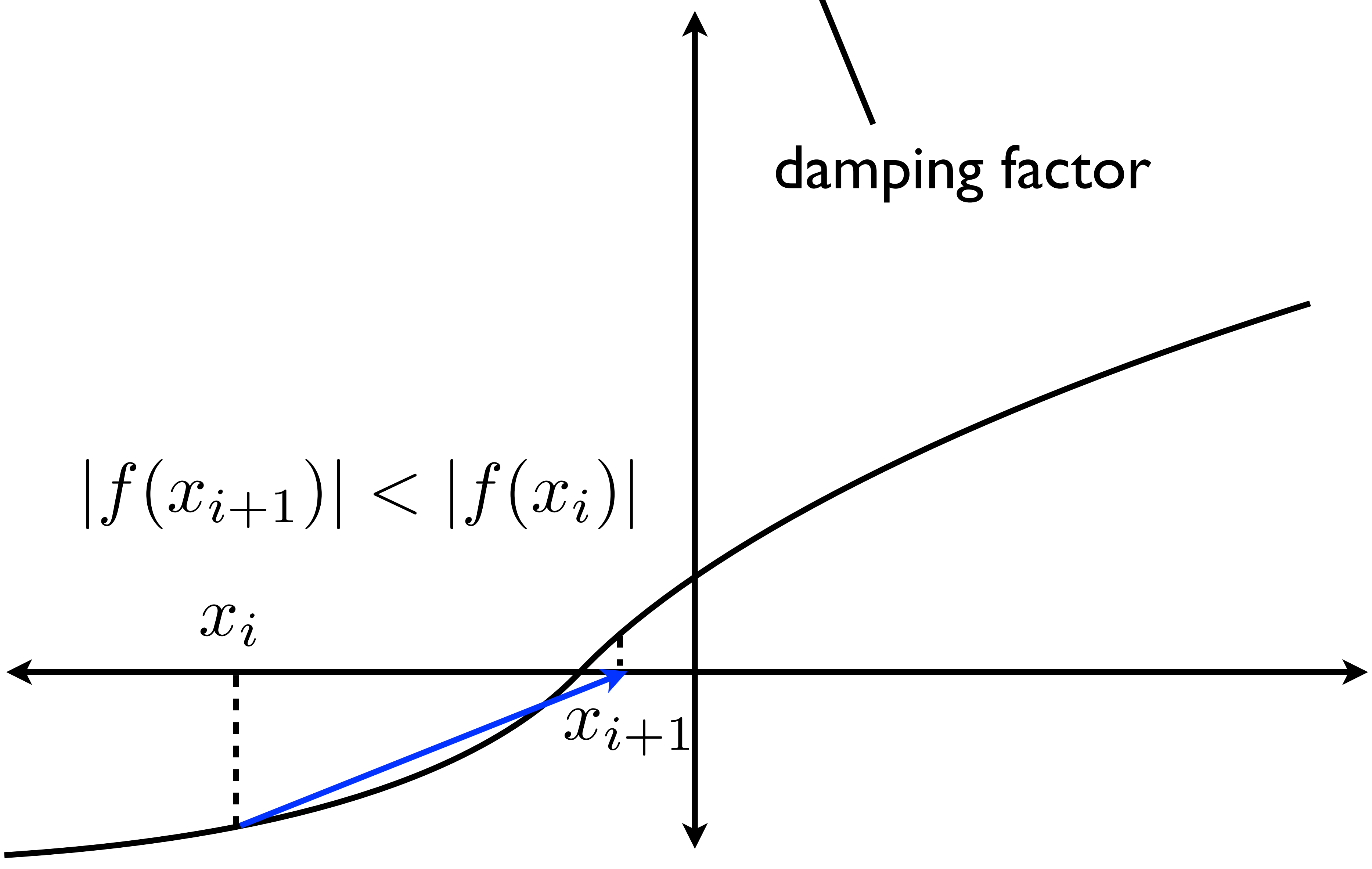
Damped NR Method

- The Newton-Raphson method converges quadratically but only near the root.
- Far from a root, the method gives an erratic response.
  - The direction of the NR step, $\mathbf{J}(\mathbf{x}_i)^{-1}\mathbf{f}(\mathbf{x}_i)$, is one that would reduce $\|\mathbf{f}(\mathbf{x}_{i+1})\|_p$
  - The magnitude of the NR step, $\|\mathbf{J}(\mathbf{x}_i)^{-1}\mathbf{f}(\mathbf{x}_i)\|_2$, can be so large that $\|\mathbf{f}(\mathbf{x}_{i+1})\|_p > \|\mathbf{f}(\mathbf{x}_i)\|_p$
  - Since the goal is to drive $\|\mathbf{f}(\mathbf{x}_{i+1})\|_p$ to zero, this is unacceptable.
- This behavior can be corrected by introducing an additional approximation to Newton-Raphson

Damped NR Method

$$x_{i+1} = x_i - \alpha \frac{f(x_i)}{f'(x_i)}$$

damping factor



$$|f(x_{i+1})| < |f(x_i)|$$

ideally $\alpha = \arg\min_{\alpha} \left| f\left(x_i - \alpha \frac{f(x_i)}{f'(x_i)} \right) \right|$

Damped NR Method

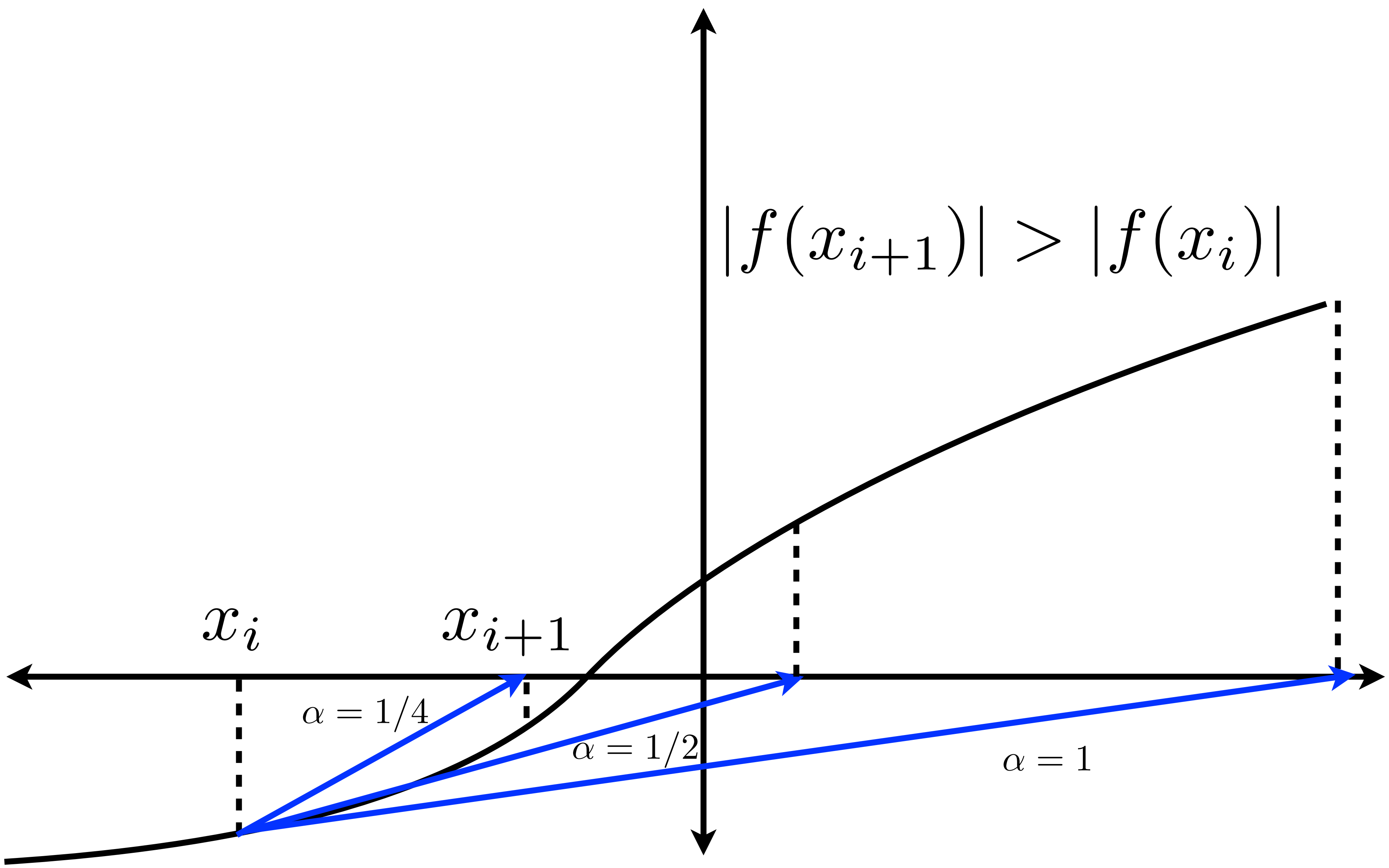
- In many dimensions:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \alpha \mathbf{J}(\mathbf{x}_i)^{-1}\mathbf{f}(\mathbf{x}_i)$$

  - where $\alpha = \arg \min_{0<\alpha\leq 1} \|\mathbf{f}\left(\mathbf{x}_i - \alpha \mathbf{J}(\mathbf{x}_i)^{-1}\mathbf{f}(\mathbf{x}_i)\right)\|_p$
- Finding the damping factor is as hard as finding the root.
- An approximate solution is to use a line search:
  - 1. Let $\alpha = 1$, this gives the full Newton-Raphson step
  - 2. Check whether $\|\mathbf{f}(\mathbf{x}_{i+1})\|_p < \|\mathbf{f}(\mathbf{x}_i)\|_p$
  - 3. If yes, accept $\mathbf{x}_{i+1}$ as the new iterate
  - 4. If no, replace α with $\alpha/2$ and repeat from 2

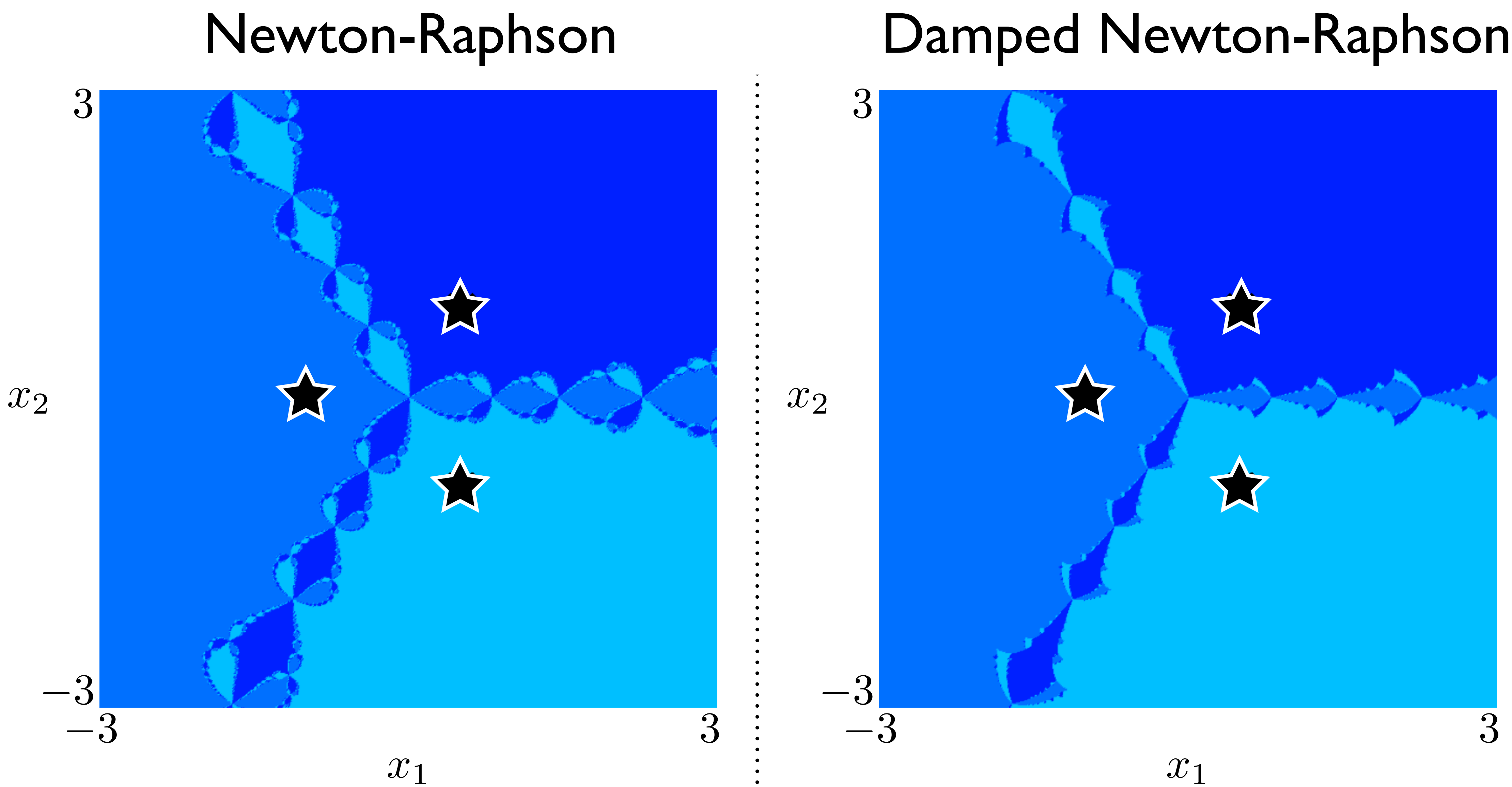
Damped NR Method

$$x_{i+1} = x_i - \alpha \frac{f(x_i)}{f'(x_i)}$$



Damped NR Method

- Basins of attraction:



Damped NR Method

- The damped Newton-Raphson method converges quadratically near a root because it behaves like the Newton-Raphson method.
- The damped Newton-Raphson method is globally convergent too (NR is locally convergent), but it converges to either:
  - roots
  - local minima/maxima
- Other modifications to Newton-Raphson are possible which can be used to improve reliability. We will see these in our study of optimization.