Chapter 3: Nonlinear equations in One Variable

Uri M. Ascher and Chen Greif
Department of Computer Science
The University of British Columbia
{ascher,greif}@cs.ubc.ca

Slides for the book
A First Course in Numerical Methods (published by SIAM, 2011)
`http://bookstore.siam.org/cs07/`

Some of the slides are from Peter Arbenz @ETH, Course: Numerical Methods in Computational Science and Engineering

Goals of this chapter

- To develop useful methods for a basic, simply stated problem, including such favourites as fixed point iteration and Newton's method;
- to develop and assess several algorithmic concepts that are prevalent throughout the field of numerical computing;
- to study basic algorithms for minimizing a function in one variable.

Why study a nonlinear problem before a linear one?
Several important methods and algorithm properties can be studied in a general context
A single linear equation is too easy to solve, multiple linear equations bring complications

The problem

Want to find solutions of the scalar nonlinear equation

$$\boxed{f(x) = 0} \quad \text{with continuous} \quad f : [a, b] \subset \mathbb{R} \mapsto \mathbb{R}$$

We denote a solution of the equation (called root, or zero) by x^*.

In contrast to scalar linear equations

$$ax - b = 0 \quad \underset{a \neq 0}{\Longrightarrow} \quad x^* = \frac{b}{a}$$

nonlinear equations have an undetermined number of zeros.

We denote the set of all continuous functions on the interval $[a, b]$ by $C[a, b]$.
So, above, we require $f \in C[a, b]$.

Examples

1. $f(x) = x - 1$ on $[a, b] = [0, 2]$.
2. $f(x) = \sin(x)$

   On $[a, b] = [\frac{\pi}{2}, \frac{3\pi}{2}]$ there is one root $x^* = \pi$.

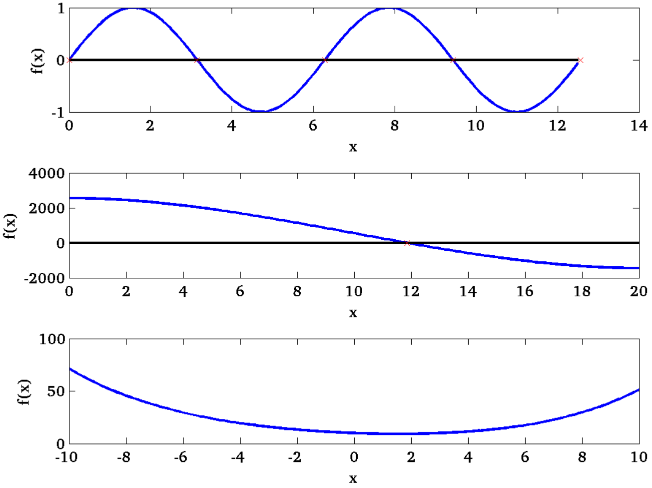
   On $[a, b] = [0, 4\pi]$ there are five roots, cf. Fig. on next page.
3. $f(x) = x^3 - 20x^2 + 2552$ on $[0, 20]$.
4. $f(x) = 10\cosh(x/4)$ on $-\infty < x < \infty$

   $\cosh(t) = \frac{1}{2}\left(e^t + e^{-t}\right)$

3. a cubic polynomial: in general with complex coefficients, has 3 complex roots. with real coefficients, when solution is restricted to be real, no general rule on how many (real) roots to expect

4. No roots

Examples (cont.)



Desirable algorithm properties

because analytical solutions are very rare (even for polynomials)

Generally for a nonlinear problem, must consider an iterative method: starting with initial iterate (guess) x_0, generate sequence of iterates $x_1,\ x_2,\ \ldots,\ x_k, \ldots$ that hopefully converge to a root x^*.

Desirable properties of a contemplated iterative method are:

- Efficient: requires a small number of function evaluations.
- Robust: fails rarely, if ever. Announces failure if it does fail.
- Requires a minimal amount of additional information such as the derivative of f.
- Requires f to satisfy only minimal smoothness properties.
- Generalizes easily and naturally to many equations in many unknowns.

Like many other wish-lists, this one is hard to fully satisfy...

Stopping an iterative procedure

In general, an iterative procedure does not find the solution but gets (arbitrarily) close.

Various criteria are used to check (almost) convergence: We terminate iterating after n iterations if:

$$
\begin{aligned}
|x_n - x_{n-1}| &< \texttt{atol}, \qquad \text{and/or} \\
|x_n - x_{n-1}| &< \texttt{rtol}\,|x_n|, \qquad \text{and/or} \\
|f(x_n)| &< \texttt{ftol},
\end{aligned}
$$

where atol, rtol, and ftol are user-specified constants.

Usually (but not always) the relative criterion is more robust than the absolute one.

A combination of the first two is

$$|x_n - x_{n-1}| < \texttt{tol}\,(1 + |x_n|)$$

Outline

- Bisection method
- Fixed point iteration
- Newton's method and variants
- Minimizing a function in one variable

Bisection

- Method for finding a root of scalar equation $f(x) = 0$ in an interval $[a, b]$.
- Assumption: $\boxed{f(a)f(b) < 0}$.

due Intermediate Value Theorem

- Since f is continuous there must be a zero $x^* \in [a, b]$.
- Compute midpoint m of the interval and check the value $f(m)$.
- Depending on the sign of $f(m)$, we can decide if $x^* \in [a, m]$ or $x^* \in [m, b]$.

  (Of course, if $f(m) = 0$ then we are done.)

Bisection method development

- Given $a < b$ such that $f(a) \cdot f(b) < 0$, there must be a root in $[a, b]$. Refer to $[a, b]$ as the uncertainty interval.
- So, at each iteration, evaluate $f(p)$ at $p = \frac{a+b}{2}$ and check the sign of $f(a) \cdot f(p)$.
  If positive, set $a \leftarrow p$, if negative set $b \leftarrow p$.
  Note: only one evaluation of the function f per iteration.
- This reduces the length of the uncertainty interval by factor 0.5 at each iteration. So, setting $x_n = p$, the error after n iterations satisfies $|x^* - x_n| \leq \frac{b-a}{2} \cdot 2^{-n}$.
- Stopping criterion: given (absolute) tolerance `atol`, require $\frac{b-a}{2} \cdot 2^{-n} \leq \texttt{atol}$.
- This allows a priori determination of the number of iterations n: unusual in algorithms for nonlinear problems.

Bisection method

- Simple
- Safe, robust
- Requires only that f be continuous
- Slow
- Hard to generalize to systems

bisect function

```
function [p,n] = bisect(func,a,b,fa,fb,atol)
if (a >= b) | (fa*fb >= 0) | (atol <= 0)
   disp('something wrong with the input:  quitting');
   p = NaN; n=NaN;
   return
end

n = ceil ( log2 (b-a) - log2 (2*atol));
for k=1:n
   p = (a+b)/2;
   fp = feval(func,p);
   if fa * fp < 0
     b = p;
     fb = fp;
   else
     a = p;
     fa = fp;
   end
end
p = (a+b)/2;
```

Outline

- Bisection method
- Fixed point iteration
- Newton's method and variants
- Minimizing a function in one variable

Fixed point iteration

The methods discussed now have direct extensions to more complicated problems, e.g., to systems of nonlinear equations and to more complex functional equations.
Problem $f(x) = 0$ can be rewritten as

$$\boxed{x = g(x).} \tag{$*$}$$

(There are many ways to do this.)

Given $(*)$ we are looking for a fixed point, i.e., a point x^* satisfying $g(x^*) = x^*$.

Algorithm: Fixed point iteration

Given a scalar function $f(x)$. Select a function $g(x)$ such that

$$f(x) = 0 \quad \Longleftrightarrow \quad g(x) = x.$$

Then:

1. Start from an initial guess x_0.
2. For $k = 0, 1, 2, \ldots$ set

$$\boxed{x_{k+1} = g(x_k), \qquad k = 0, 1, \ldots}$$

until x_{k+1} satisfies some termination criterion

Examples of fixed point iterations

Note: there are many ways to transform $f(\mathbf{x}) = 0$ into fixed point form! Not all of them "good" in terms of convergence.
Options for fixed point iterations for

$$f(x) = xe^x - 1, \qquad x \in [0, 1]$$

Different fixed point forms:

$$\begin{aligned} g_1(x) &= e^{-x}, \\ g_2(x) &= \frac{1+x}{1+e^x}, \\ g_3(x) &= x + 1 - xe^x. \end{aligned}$$

Examples of fixed point iterations (cont.)

k	$x_{k+1} := g_1(x_k)$	$x_{k+1} := g_2(x_k)$	$x_{k+1} := g_3(x_k)$
0	0.500000000000000	0.500000000000000	0.500000000000000
1	0.606530659712633	0.566311003197218	0.675639364649936
2	0.545239211892605	0.567143165034862	0.347812678511202
3	0.579703094878068	0.567143290409781	0.855321409174107
4	0.560064627938902	0.567143290409784	-0.156505955383169
5	0.571172148977215	0.567143290409784	0.977326422747719
6	0.564862946980323	0.567143290409784	-0.619764251895580
7	0.568438047570066	0.567143290409784	0.713713087416146
8	0.566409452746921	0.567143290409784	0.256626649129847
9	0.567559634262242	0.567143290409784	0.924920676910549
10	0.566907212935471	0.567143290409784	-0.407422405542253

Examples of fixed point iterations (cont.)

k	$\lvert x_k - x^* \rvert$	$\lvert x_k - x^* \rvert$	$\lvert x_k - x^* \rvert$
0	0.067143290409784	0.067143290409784	0.067143290409784
1	0.039387369302849	0.000832287212566	0.108496074240152
2	0.021904078517179	0.000000125374922	0.219330611898582
3	0.012559804468284	0.000000000000003	0.288178118764323
4	0.007078662470882	0.000000000000000	0.723649245792953
5	0.004028858567431	0.000000000000000	0.410183132337935
6	0.002280343429460	0.000000000000000	1.186907542305364
7	0.001294757160282	0.000000000000000	0.146569797006362
8	0.000733837662863	0.000000000000000	0.310516641279937
9	0.000416343852458	0.000000000000000	0.357777386500765
10	0.000236077474313	0.000000000000000	0.974565695952037

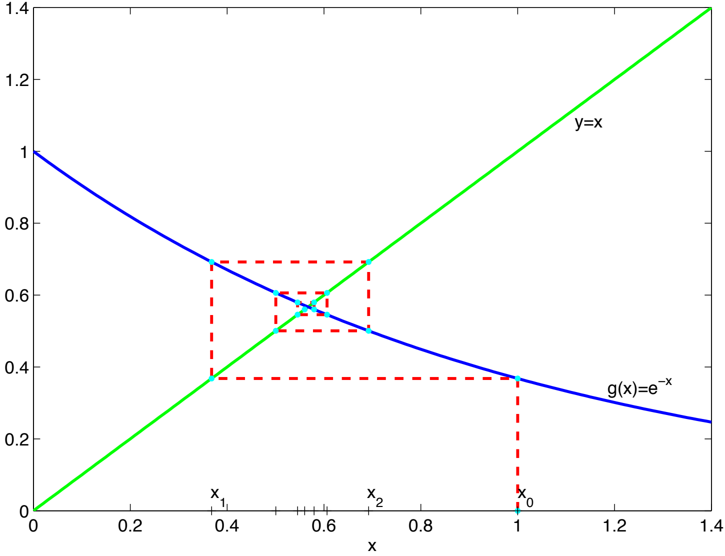
Choosing the function g

- Note: there are many possible choices g for the given f: this is a family of methods.
- Examples:

$$
\begin{aligned}
g(x) &= x - f(x), \\
g(x) &= x + 2f(x), \\
g(x) &= x - f(x)/f'(x) \quad (\text{assuming f}' \text{ exists and } \text{f}'(\text{x}) \neq 0).
\end{aligned}
$$

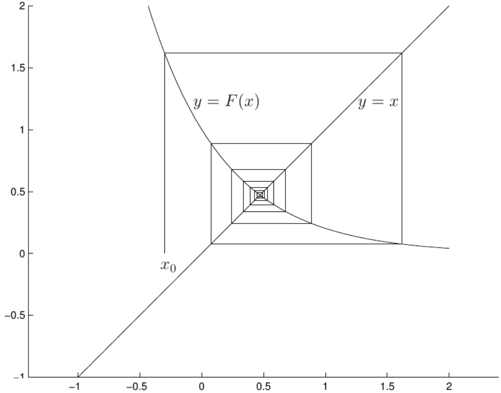
  The first two choices are simple, the last one has potential to yield fast convergence (we'll see later).
- Want resulting method to
  - be simple;
  - converge; and
  - do it rapidly.

Graphical illustration, $x = e^{-x}$, starting from $x_0 = 1$



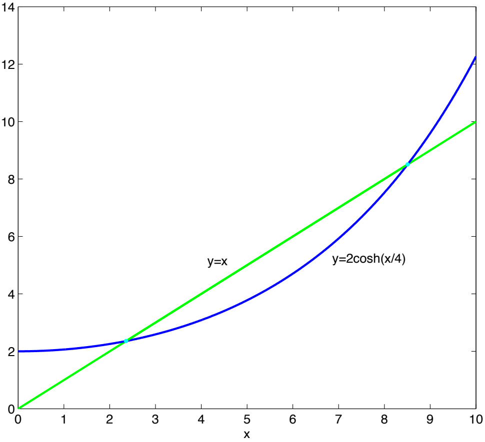
Geometric interpretation of fixed point iteration

x_0	start with x_0 on the x-axis
$F(x_0)$	go parallel to the y-axis to the graph of $F \equiv g$
$x_1 = F(x_0)$	move parallel to the x-axis to the graph $y = x$
$F(x_1)$	go parallel to the y-axis to the graph of F
etc.	



Example: cosh with two roots

$$f(x) = g(x) - x, \quad g(x) = 2\cosh(x/4)$$



Fixed point iteration with g

For tolerance 1.e-8:

- Starting at $x_0 = 2$ converge to x_1^* in 16 iterations.
- Starting at $x_0 = 4$ converge to x_1^* in 18 iterations.
- Starting at $x_0 = 8$ converge to x_1^* (even though x_2^* is closer to x_0).
- Starting at $x_0 = 10$ obtain **overflow** in 3 iterations.

Note: bisection yields both roots in 27 iterations.

Some questions regarding the fixed point iteration

Suppose that we have somehow determined the continuous function $g \in C[a, b]$. Now let us consider the fixed point iteration $x_{k+1} = g(x_k)$.

Obvious questions arise:

1. Is there a fixed point x^* in $[a, b]$?
2. If yes, is it unique?
3. Does the sequence of iterates converge to a root x^*?
4. If yes, how fast?
5. If not, does this mean that no root exists?

Fixed point theorem

If $g \in C[a, b]$ and $a \leq g(x) \leq b$ for all $x \in [a, b]$, then there is a fixed point x^* in the interval $[a, b]$.

If, in addition, the derivative g' exists and there is a constant $\rho < 1$ such that the derivative satisfies

$$|g'(x)| \leq \rho \qquad \forall x \in (a, b),$$

then the fixed point x^* is unique in this interval.

See Ascher-Greif book for proof.

(This answers questions 1 and 2.)

Convergence of the fixed point iteration

3. Does the sequence of iterates converge to a root x^*?

$$|x_{k+1} - x^*| = |g(x_k) - g(x^*)| = \underbrace{|g'(\xi)|}_{\text{Mean value theorem}} \cdot |x_k - x^*| \leq \rho |x_k - x^*|$$

with $\xi \in [x_k, x^*]$.

This is a contraction if the factor $\rho < 1$. Thus,

$$|x_{k+1} - x^*| \leq \rho |x_k - x^*| \leq \rho^2 |x_{k-1} - x^*| \leq \cdots \leq \rho^{k+1} |x_0 - x^*|.$$

Since $\rho < 1$ then $\rho^k \to 0$ as $k \to \infty$.

Convergence.

Order of convergence

The method is said to be

- linearly convergent if there is a constant $\rho < 1$ such that

$$|x_{k+1} - x^*| \leq \rho |x_k - x^*|, \qquad \text{for } k \text{ sufficiently large;}$$

- quadratically convergent if there is a constant M such that

$$|x_{k+1} - x^*| \leq M |x_k - x^*|^2, \qquad \text{for } k \text{ sufficiently large;}$$

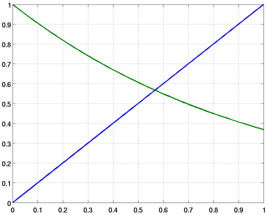
- superlinearly convergent if there is a sequence of constants $\rho_k \to 0$ such that

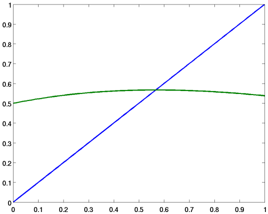
$$|x_{k+1} - x^*| \leq \rho_k |x_k - x^*| \qquad \text{for } k \text{ sufficiently large;}$$

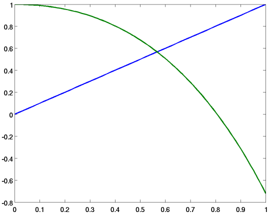
The quadratic case is superlinear with $\rho_k = M|x_k - x^*| \to 0$

Convergence of fixed point iterations in 1D

Vastly different behavior of different fixed point iterations:







g_1: linear convergence? g_2: quadratic convergence? g_3: no convergence?

dependent on behavior of |g'(x)|

For the conditions in the Fixed Point Theorem satisfied, the convergence is linear.

Rate of convergence

- Suppose we want $|x_k - x^*| \approx 0.1|x_0 - x^*|$.
- Since $|x_k - x^*| \le \rho^k |x_0 - x^*|$, want

$$\rho^k \approx 0.1,$$

i.e., $k \log_{10} \rho \approx -1$.

- Define the **rate of convergence** as

$$rate = -\log_{10} \rho.$$

- Then it takes about $k = \lceil 1/rate \rceil$ iterations to reduce the error by more than an order of magnitude.

Return to cosh example

- Bisection: $rate = -\log_{10} 0.5 \approx .3 \quad \Rightarrow \quad k = 4$.
- For the root x_1^* of fixed point example, $\rho \approx 0.31$ so

rho = |g'(x)|
0<rho<1

$$rate = -\log_{10} 0.31 \approx .5, \quad \Rightarrow \quad k = 2.$$

- For the root x_2^* of fixed point example, $\rho > 1$ so

$$rate = -\log_{10}(\rho) < 0, \quad \Rightarrow \quad \text{no convergence.}$$

Outline

- Bisection method
- Fixed point iteration
- Newton's method and variants
- Minimizing a function in one variable

Newton's method

This fundamentally important method is everything that bisection is not, and vice versa:

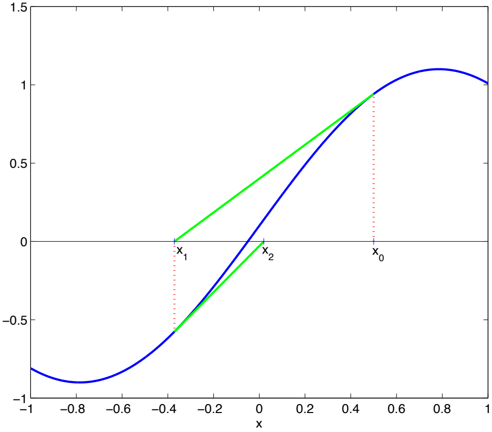
- Not so simple
- Not very safe or robust
- Requires more than continuity on f
- Fast
- Automatically generalizes to systems

In Newton's method the function f is linearized at some approximate value x_k ~~x*

Derivation

- By Taylor series,
$$f(x) = f(x_k) + f'(x_k)(x - x_k) + f''(\xi(x))(x - x_k)^2/2.$$
- So, for $x = x^*$
$$0 = f(x_k) + f'(x_k)(x^* - x_k) + \mathcal{O}\left((x^* - x_k)^2\right).$$
- The method is obtained by neglecting nonlinear term, defining $0 = f(x_k) + f'(x_k)(x_{k+1} - x_k)$, which gives the iteration step
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, 2, \ldots.$$

A geometric interpretation



Next iterate is x-intercept of the tangent line to f at current iterate.

Example: cosh with two roots

- The function
$$f(x) = 2\cosh(x/4) - x$$
has two solutions in the interval $[2, 10]$.
- Newton's iteration is
$$x_{k+1} = x_k - \frac{2\cosh(x_k/4) - x_k}{0.5\sinh(x_k/4) - 1}.$$
- For absolute tolerance 1.e-8:
  - Starting from $x_0 = 2$ requires 4 iterations to reach x_1^*.
  - Starting from $x_0 = 4$ requires 5 iterations to reach x_1^*.
  - Starting from $x_0 = 8$ requires 5 iterations to reach x_2^*.
  - Starting from $x_0 = 10$ requires 6 iterations to reach x_2^*.
- Tracing the iteration's progress:

k	0	1	2	3	4	5
$f(x_k)$	-4.76e-1	8.43e-2	1.56e-3	5.65e-7	7.28e-14	1.78e-15

- Note that the number of significant digits essentially doubles at each iteration (until the 5th, when roundoff error takes over).

Convergence theorem for Newton's method

If $f \in C^2[a,b]$ and there is a root x^* in $[a,b]$ such that $f(x^*) = 0,\ f'(x^*) \neq 0$, then there is a number δ such that, starting with x_0 from anywhere in the neighborhood $[x^* - \delta, x^* + \delta]$, Newton's method converges quadratically.

Idea of proof:

- Expand $f(x^*)$ in terms of a Taylor series about x_k;
- divide by $f'(x_k)$, rearrange, and replace $x_k - \frac{f(x)}{f'(x_k)}$ by x_{k+1};
- find the relation between $e_{k+1} = x_{k+1} - x^*$ and $e_k = x_k - x^*$.

Secant method

- One potential disadvantage of Newton's method is the need to *know and evaluate* the derivative of f.
- The secant method circumvents the need for explicitly evaluating this derivative.
- Observe that near the root (assuming convergence)

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}.$$

- So, define Secant iteration

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}, \quad k = 0, 1, 2, \ldots.$$

- Note the need for two initial starting iterates x_0 and x_1: a *two-step method*.

Example: cosh with two roots

$$f(x) = 2\cosh(x/4) - x.$$

Same absolute tolerance 1.e-8 and initial iterates as before:

- Starting from $x_0 = 2$ and $x_1 = 4$ requires 7 iterations to reach x_1^*.
- Starting from $x_0 = 10$ and $x_1 = 8$ requires 7 iterations to reach x_2^*.

k	0	1	2	3	4	5	6
$f(x_k)$	2.26	-4.76e-1	-1.64e-1	2.45e-2	-9.93e-4	-5.62e-6	1.30e-9

Observe superlinear convergence: much faster than bisection and simple fixed point iteration, yet not quite as fast as Newton's iteration.

both linearly converge

Recall:

Speed of convergence

A given method is said to be

- **linearly convergent** if there is a constant $\rho < 1$ such that

$$|x_{k+1} - x^*| \le \rho |x_k - x^*| \ ,$$

  for all k sufficiently large;

- **quadratically convergent** if there is a constant M such that

$$|x_{k+1} - x^*| \le M |x_k - x^*|^2 \ ,$$

  for all k sufficiently large;

- **superlinearly convergent** if there is a sequence of constants $\rho_k \to 0$ such that

$$|x_{k+1} - x^*| \le \rho_k |x_k - x^*|,$$

  for all k sufficiently large.

Newton's method as a fixed point iteration

- If $g'(x^*) \neq 0$ then fixed point iteration converges linearly, as discussed before, as $\rho > 0$.
- Newton's method can be written as a fixed point iteration with

$$g(x) = x - \frac{f(x)}{f'(x)}.$$

From this we get $g'(x^*) = 0$.

- In such a situation the fixed point iteration may converge faster than linearly: indeed, Newton's method converges quadratically under appropriate conditions.

Outline

- Bisection method
- Fixed point iteration
- Newton's method and variants
- Minimizing a function in one variable

Minimizing a function of one variable

- A major source of applications giving rise to root finding is optimization. a vast subject !
- One-variable version: find an argument $x = \hat{x}$ that minimizes a given objective function $\phi(x)$.
- Example from earlier: Find the minimum of the function

$$\phi(x) = 10\cosh(\frac{x}{4}) - x$$

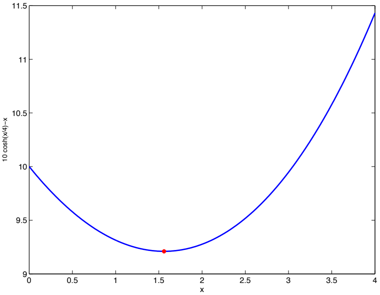
over the real line.
- Note: maximize function $\psi(x) \Leftrightarrow$ minimize $\phi(x) = -\psi(x)$.

Minimizing a function in one variable

- **Example**: find $x = x^*$ that minimizes

  $$\phi(x) = 10\cosh(x/4) - x.$$

- From the figure below, this function has no zeros but does appear to have one minimizer around $x = 1.6$.



Conditions for a minimum point

Assume that $\phi \in C^2[a, b]$. Denote

$$f(x) = \phi'(x).$$

An argument x^* satisfying $a < x^* < b$ is called a critical point if

$$f(x^*) = 0.$$

For parameter h small enough so that $x^* + h \in [a, b]$ we can expand in a Taylors series

$$\begin{aligned}\phi(x^* + h) &= \phi(x^*) + h\overset{=0}{\phi'(x^*)} + \frac{h^2}{2}\phi''(x^*) + \cdots \\ &= \phi(x^*) + \frac{h^2}{2}[\phi''(x^*) + \mathcal{O}(h)].\end{aligned}$$

Conditions for a minimum point (cont.)

Since $|h|$ can be taken arbitrarily small, it is now clear that at a critical point:

- If $\phi''(x^*) > 0$, then $\hat{x} = x^*$ is a local minimizer of $\phi(x)$. This means that ϕ attains a minimum at $\hat{x} = x^*$ in some neighborhood which includes x^*.
- If $\phi''(x^*) < 0$, then $\hat{x} = x^*$ is a local maximizer of $\phi(x)$. This means that ϕ attains a minimum at $\hat{x} = x^*$ in some neighborhood which includes x^*.
- If $\phi''(x^*) = 0$, then a further investigation at x^* is required.

Conditions for optimum and algorithm

- Necessary condition for an optimum:
  Suppose $\phi \in C^2$ and denote $f(x) = \phi'(x)$. Then a zero of f is a critical point of ϕ, i.e., where

  $$\phi'(x^*) = 0.$$

  To be a minimizer or a maximizer, it is necessary for x^* to be a critical point.
- Sufficient condition for an optimum:
  A critical point x^* is a minimizer if also $\phi''(x^*) > 0$.
- Hence, an algorithm for finding a minimizer is obtained by using one of the methods of this chapter for finding the roots of $\phi'(x)$, then checking for each such root x^* if also $\phi''(x^*) > 0$.

Example

To find a minimizer for

$$\phi(x) = 10\cosh(x/4) - x,$$

1. Calculate gradient

$$f(x) = \phi'(x) = 2.5\sinh(x/4) - 1$$

2. Find root of $\phi'(x) = 0$ using any of our methods, obtaining

$$x^* \approx 1.56014.$$

3. Second derivative

$$\phi''(x) = 2.5/4\cosh(x/4) > 0 \quad \text{for all } x,$$

so x^* is a minimizer.