# 3 Solution of Nonlinear Equations

## 3.1 Bisection Method

The main idea is to make use of the Intermediate Value Theorem (IVT):

> For $f \in C[a,b]$ with $f(a)f(b) < 0$ there exists a number $c \in (a,b)$ such that $f(c) = 0$.

This leads to a simple **Algorithm:**

1. Take $c = \dfrac{a+b}{2}$.
2. If $f(a)f(c) < 0$ (i.e., root lies in $(a,c)$), let $a = a$, $b = c$.
   If $f(c)f(b) < 0$ (i.e., root lies in $(c,b)$), let $a = c$, $b = b$.
   If $f(c) = 0$ (i.e., root lies in $c$), stop.
3. Repeat

Convergence Analysis
We shall label the intervals used by the algorithm as

$$[a,b] = [a_0,b_0], [a_1,b_1], [a_2,b_2], \ldots$$

By construction

$$b_n - a_n = \frac{1}{2}(b_{n-1} - a_{n-1}), \qquad n \geq 1.$$

Thus, recursively,

$$b_n - a_n = \frac{1}{2^n}(b_0 - a_0), \qquad n \geq 1. \tag{1}$$

We also know $a_0 \leq a_1 \leq a_2 \leq \ldots \leq b$, and $b_0 \geq b_1 \geq b_2 \geq \ldots \geq a$, which shows us that the sequences $\{a_n\}$ and $\{b_n\}$ are both bounded and monotonic, and therefore **convergent**.

Using standard limit laws, equation (1) gives us

$$\lim_{n\to\infty} b_n - \lim_{n\to\infty} a_n = (b_0 - a_0) \lim_{n\to\infty} \frac{1}{2^n} = 0.$$

So we now also know that the sequences $\{a_n\}$ and $\{b_n\}$ have the **same** limits, i.e.,

$$\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n =: r. \tag{2}$$

It remains to be shown that this number $r$ is a root of the function $f$. From the bisection algorithm we know $f(a_n)f(b_n) < 0$. Or, taking limits

$$\lim_{n\to\infty} f(a_n)f(b_n) \leq 0.$$

Finally, using (2), we have

$$[f(r)]^2 \leq 0 \quad \Longrightarrow \quad f(r) = 0.$$

Summarizing, the bisection method always converges (provided the initial interval contains a root), and produces a root of $f$.

Errors

If the algorithm is stopped after $n$ iterations, then $r \in [a_n, b_n]$. Moreover, $c_n = \frac{a_n + b_n}{2}$ is an approximation to the exact root. Note that the error can be bounded by

$$\begin{aligned} |r - c_n| &\leq \frac{1}{2}(b_n - a_n) \\ &= \frac{1}{2^{n+1}}(b_0 - a_0). \end{aligned}$$

Therefore, the error after $n$ steps of the bisection method is guaranteed to satisfy

$$\boxed{|r - c_n| \leq \frac{1}{2^{n+1}}(b - a).} \tag{3}$$

**Note:** This bound is independent of the function $f$.

**Remark:** Recall that linear convergence requires

$$e_n \leq C e_{n-1}, \tag{4}$$

with some constant $C < 1$. Thus,

$$e_n \leq C^2 e_{n-2} \leq \ldots \leq C^n e_0 \tag{5}$$

is necessary (but not sufficient) for linear convergence.
Now, for the bisection method,

$$e_n = |r - c_n| \leq \frac{1}{2^{n+1}}(b - a),$$

and

$$e_0 = \frac{b - a}{2}.$$

Thus, condition (5) is satisfied, but we know from observation (e.g., in the Maple worksheet on convergence) that the bisection method **does not converge linearly**, i.e., condition (4) at each step is not satisfied.

**Remark:** The previous discussion may "explain" why so many textbooks wrongly attribute linear convergence to the bisection method.

**Remark:** The Maple worksheet `577_convergence.mws` contains some code that produces an animation of several steps of this iterative procedure.

Implementation of the Bisection Method

Some details to consider (or slight modifications of the basic algorithm):

1. Do not compute $c_n = \frac{a_n + b_n}{2}$. This formula may become unstable. It is more stable to use $c_n = a_n + \frac{b_n - a_n}{2}$, since here the second summand acts as a small correction to the first one.

2. When picking the "correct" sub-interval to continue with, don't use the test $f(a)f(c)\lesseqgtr 0$. Instead, use $\text{sign} f(a) \neq \text{sign} f(c)$. The multiplication is more expensive than a simple sign lookup (remember the standard scientific notation), and it can also produce over- or underflow.

3. Implement some kind of (practical) stopping criterion. **All** of the following three may be used:

   (a) Specify a maximum number of allowable iterations in the `for-loop` construction.

   (b) Check if the error is small enough. We know the bound (3), so check, e.g., if
   $$\frac{1}{2^{n+1}}(b-a) < \delta,$$
   where $\delta$ is some specified tolerance. This can also be used as a-priori estimator for the number of iterations you may need.

   (c) Check if $f(c)$ is close enough to zero, i.e., check if
   $$|f(c)| < \epsilon,$$
   where $\epsilon$ is another user-specified tolerance.

Note that the stopping criteria in 3 if used by themselves may fail (see the explanation and figure on p.77 of the textbook).

### 3.1.1 Modification of the Bisection Method: Regula Falsi

The following discussion cannot be found in our textbook.

The modification comes from taken $c_n$ not as the average of $a_n$ and $b_n$, but as the weighted average

$$c_n = \frac{|f(b_n)|}{|f(a_n)| + |f(b_n)|} a_n + \frac{|f(a_n)|}{|f(a_n)| + |f(b_n)|} b_n. \tag{6}$$

**<u>Note:</u>** We still have $f(a_n)f(b_n) < 0$ (by assumption), and therefore (6) is equivalent to

$$c_n = \frac{f(b_n)a_n - f(a_n)b_n}{f(b_n) - f(a_n)},$$

or

$$c_n = b_n - \frac{f(b_n)(b_n - a_n)}{f(b_n) - f(a_n)}.$$

Notice that this last formula contains the reciprocal of the slope of the secant line at $a_n$ and $b_n$, and the choice of $c_n$ can be illustrated by Figure 1.

We will come across another secant-based method later – the secant method.

We determine the new interval as for the bisection method, i.e.,

if $f(a)f(c) < 0$ (i.e., root lies in $(a,c)$), let $a = a$, $b = c$,

if $f(c)f(b) < 0$ (i.e., root lies in $(c,b)$), let $a = c$, $b = b$,

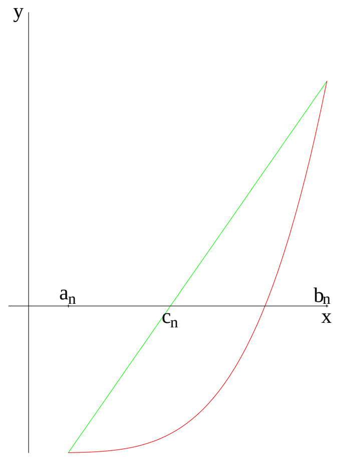

Figure 1: Choice of $c_n$ for *regula falsi*.

if $f(c) = 0$ (i.e., root lies in $c$), stop.

**Remark 1:** For concave functions one of the endpoints remains fixed. Thus, the interval $[a_n, b_n]$ does not get arbitrarily small.

**Remark 2:** It can be shown that the *regula falsi* converges linearly (see an example later on when we discuss general fixed point iteration).

### 3.2 Newton's Method

Let $r$ be such that $f(r) = 0$, and $x$ be an approximation of the root close to $r$, i.e,

$$x + h = r, \qquad h \text{ small.}$$

The quantity $h$ can be interpreted as the correction which needs to be added to $x$ to get the exact root $r$.

Recall Taylor's expansion:

$$f(r) = f(x + h) = f(x) + hf'(x) + \mathcal{O}(h^2),$$

or

$$f(r) \approx f(x) + hf'(x). \tag{7}$$

Now $r$ is a root of $f$, i.e., $f(r) = 0$, and so (7) can be restated as

$$0 \approx f(x) + hf'(x),$$

or

$$h \approx -\frac{f(x)}{f'(x)}. \tag{8}$$

Thus, using (8), an improved approximation to the root $r$ is

$$r = x + h \approx x - \frac{f(x)}{f'(x)}.$$

If we embed this into an iterative scheme and also provide an initial guess $x_0$, then we obtain

**Newton iteration:**

$$\boxed{\begin{array}{c} x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}, \qquad n \geq 0, \\ \text{with } x_0 \text{ as initial guess.} \end{array}} \tag{9}$$

**Graphical Interpretation:**

Consider the tangent line to the graph of $f$ at $x_n$

$$y - f(x_n) = f'(x_n)(x - x_n),$$

or

$$y = f(x_n) - (x - x_n)f'(x_n).$$

Now we intersect with the $x$-axis, i.e., set $y = 0$. This yields

$$0 = f(x_n) - (x - x_n)f'(x_n) \quad \Longleftrightarrow \quad x = x_n - \frac{f(x_n)}{f'(x_n)}.$$

The last formula coincides with the Newton formula (9), thus, in Newton's method, a new approximation to the root of $f$ is obtained by intersecting the tangent line to $f$ at a previous approximate root with the $x$-axis. Figure 2 illustrates this. The entire iterative procedure can also be viewed as an animation in the Maple worksheet `577_convergence.mws` on convergence.

**Convergence**

**Theorem 3.1** *If $f$ has a simple zero at $r$ and $f \in C^2[r-\delta, r+\delta]$ for a suitably small $\delta$, then Newton's method will converge to the root $r$ provided it is started with $x_0 \in [r-\delta, r+\delta]$. Moreover, convergence is quadratic, i.e., there exists a constant $C$ such that*

$$|r - x_{n+1}| \leq C|r - x_n|^2, \qquad n \geq 0.$$

**Proof:** We will use the notation $e_n = r - x_n$ for the error at step $n$. Then, following (9),

$$e_{n+1} = r - x_{n+1} = r - x_n + \frac{f(x_n)}{f'(x_n)}$$

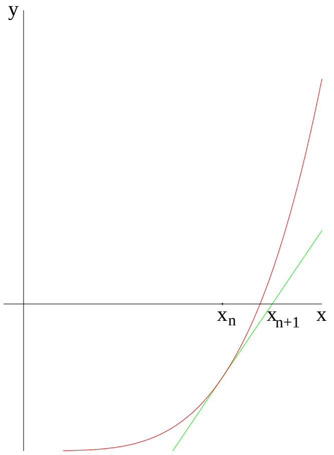

Figure 2: Graphical interpretation of Newton's method.

$$= \quad e_n + \frac{f(x_n)}{f'(x_n)} = \frac{e_n f'(x_n) + f(x_n)}{f'(x_n)}. \tag{10}$$

On the other hand, via Taylor expansion we know

$$0 = f(r) = f(x_n + e_n) = f(x_n) + e_n f'(x_n) + \frac{e_n^2}{2} f''(\xi_n),$$

with $\xi_n$ between $x_n$ and $x_n + e_n = r$. This immediately implies

$$e_n f'(x_n) + f(x_n) = -\frac{1}{2} e_n^2 f''(\xi_n). \tag{11}$$

By inserting (11) into (10) we get

$$e_{n+1} = -\frac{1}{2} \frac{f''(\xi_n)}{f'(x_n)} e_n^2. \tag{12}$$

Now, if the algorithm converges, then for $x_n$ and $\xi_n$ close to $r$ we have

$$|e_{n+1}| \approx \underbrace{\frac{1}{2} \frac{|f''(r)|}{|f'(r)|}}_{\text{const=:C}} e_n^2,$$

which establishes quadratic convergence.

Now we get to the rather technical part of verifying convergence. We begin by letting $\delta > 0$ and picking $x_0$ such that

$$|r - x_0| \leq \delta \quad \iff \quad |e_0| \leq \delta. \tag{13}$$

Then $\xi_0$ in (11) satisfies $|r-\xi_0| \le \delta$. Now consider (12) for $n=0$:

$$|e_1| = \frac{1}{2}\frac{|f''(\xi_0)|}{|f'(x_0)|}e_0^2$$

and define

$$c(\delta) := \frac{1}{2}\frac{\max_{|r-\xi_0|\le\delta}|f''(\xi_0)|}{\max_{|r-x_0|\le\delta}|f'(x_0)|}.$$

Then

$$|e_1| \le c(\delta)e_0^2 \le c(\delta)|e_0|\delta,$$

where we have used (13) for the second inequality.

Next, we define $\rho = \delta c(\delta)$ and, if necessary, go back and adjust $\delta$ such that $0 \le \rho < 1$. Note that this can be done since $c(\delta)$ approaches a constant as $\delta \to 0$. Thus,

$$|e_1| \le \rho|e_0| < |e_0|,$$

i.e., we have ensured that the error decreases. Finally, arguing recursively, we obtain

$$|e_{n+1}| \le \rho^n |e_0|,$$

and

$$\lim_{n\to\infty}|e_{n+1}| \le |e_0|\lim_{n\to\infty}\rho^n = 0.$$

♠

**Remarks:**

1. There are functions for which Newton's method converges for any initial guess:
   - If $f \in C^2(\mathbb{R})$, $f'(x) > 0$, $f''(x) > 0$, for all x, then Newton's method converges to the unique root for any $x_0$.
   - If $f(a)f(b) < 0$, $f'(x) \neq 0$, $x \in [a,b]$, $f''$ does not change sign on $[a,b]$, and $\left|\frac{f(a)}{f'(a)}\right|, \left|\frac{f(b)}{f'(b)}\right| < b-a$, then Newton's method converges to the unique root in $[a,b]$ for any $x_0 \in [a,b]$.

2. $|e_{n+1}| \approx \frac{1}{2}\frac{|f''(r)|}{|f'(r)|}e_n^2$ implies that the number of significant digits in the approximate root **doubles** from one iteration to the next. However, this only is true if we are close enough to the root.

   Therefore, one could possibly design a *hybrid method*: start with bisection to get reasonably close to the root, then use Newton for faster convergence.

3. The order of convergence is reduced at multiple roots, i.e., at a double root we have only linear convergence (see HW 14, but also not HW 19).

4. One disadvantage is the need for the derivative of $f$. This has to be provided by the user, and the code has to be able to handle this user-input. Possible alternatives to Newton's method are therefore the secant method (coming up next) or Steffensen's method (see HW 4).

**Note:** We will come back to the discussion of Newton's method for systems of nonlinear equations at the end of the semester.

### 3.3 The Secant Method

Recall the iteration formula in the Newton method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

It has two main drawbacks:

- it requires coding of the derivative (for every new function $f$),
- it requires evaluation of $f$ and $f'$ in every iteration.

One of the most straight-forward work-around that comes to mind is to approximate the continuous derivative $f'(x_n)$ by the difference quotient, i.e.,

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}.$$

Thus, we arrive at the
**Secant method:**

$$\boxed{\begin{array}{c} x_{n+1} = x_n - f(x_n)\dfrac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}, \qquad n \geq 1, \\ \text{with } x_0 \text{ and } x_1 \text{ as initial guesses.} \end{array}} \tag{14}$$

**Graphical Interpretation:**

Obviously, there will be an interpretation similar to that of Newton's method (with tangent lines replaced by secant lines). To see this, consider the secant to the graph of $f$ at the points $(x_{n-1}, f(x_{n-1}))$ and $(x_n, f(x_n))$:

$$y - f(x_n) = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}(x - x_n).$$

If we consider intersection of this line with the $x$-axis, i.e., set $y = 0$, we get

$$\frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}x_n - f(x_n) = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}x,$$

or, solving for $x$,

$$x = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}.$$

This latter formula is equivalent to the secant method (14). Thus, the next approximation to the root of $f$ using the secant method is obtained by intersecting the secant to two previous approximations with the $x$-axis. Figure 3 illustrates this fact. The entire iterative procedure can also be viewed as an animation in the Maple worksheet `577_convergence.mws` on convergence.

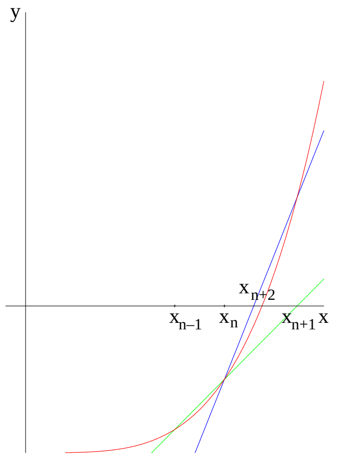

Figure 3: Graphical interpretation of two steps of the secant method.

**Convergence Analysis**

**Theorem 3.2** *If $f$ has a simple zero at $r$, $f \in C^2$, and $x_0$ and $x_1$ are close to $r$, then the secant method will converge to the root. Moreover,*

$$|r - x_{n+1}| \approx \frac{1}{2}\left|\frac{f''(r)}{f'(r)}\right|^{1/\alpha} |r - x_n|^{\alpha}, \qquad n \geq 0,$$

*where* $\alpha = \dfrac{1+\sqrt{5}}{2} \approx 1.618$ *(the golden ratio).*

**Proof:** We will use the notation $e_n = r - x_n$ for the error at step $n$. Then, using the basic iteration formula (14) for the secant method along with some simple algebra,

$$\begin{aligned}
e_{n+1} &= r - x_{n+1} = r - \left[x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}\right] \\
&= r - \frac{x_n f(x_n) - x_n f(x_{n-1}) - f(x_n)x_n + f(x_n)x_{n-1}}{f(x_n) - f(x_{n-1})} \\
&= \frac{rf(x_n) - rf(x_{n-1}) + x_n f(x_{n-1}) - f(x_n)x_{n-1}}{f(x_n) - f(x_{n-1})}.
\end{aligned}$$

Factoring, and then replacing $r - x_n$ by $e_n$ (and analogously for $n - 1$) we get

$$e_{n+1} = \frac{e_{n-1}f(x_n) - e_n f(x_{n-1})}{f(x_n) - f(x_{n-1})}.$$

Multiplying the right-hand side by $\frac{x_n - x_{n-1}}{x_n - x_{n-1}}$ we have

$$\begin{aligned} e_{n+1} &= \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} \frac{e_{n-1} f(x_n) - e_n f(x_{n-1})}{x_n - x_{n-1}} \\ &= \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} \frac{\frac{f(x_n)}{e_n} - \frac{f(x_{n-1})}{e_{n-1}}}{x_n - x_{n-1}} e_n e_{n-1}. \end{aligned} \tag{15}$$

Now we use Taylor's theorem in the form

$$f(x_n) = f(r - e_n) = \underbrace{f(r)}_{=0} - e_n f'(r) + \frac{1}{2} e_n^2 f''(r) + \mathcal{O}(e_n^3),$$

which implies

$$\frac{f(x_n)}{e_n} = f'(r) + \frac{1}{2} e_n f''(r) + \mathcal{O}(e_n^2). \tag{16}$$

Similarly,

$$\frac{f(x_{n-1})}{e_{n-1}} = f'(r) + \frac{1}{2} e_{n-1} f''(r) + \mathcal{O}(e_{n-1}^2). \tag{17}$$

Subtracting (17) from (16) we get

$$\frac{f(x_n)}{e_n} - \frac{f(x_{n-1})}{e_{n-1}} = \frac{1}{2}(e_n - e_{n-1}) f''(r) + \mathcal{O}(e_n^2) - \mathcal{O}(e_{n-1}^2),$$

or

$$\frac{f(x_n)}{e_n} - \frac{f(x_{n-1})}{e_{n-1}} \approx \frac{1}{2}(e_n - e_{n-1}) f''(r). \tag{18}$$

Also, $e_n = r - x_n$ and $e_{n-1} = r - x_{n-1}$, so that $e_n - e_{n-1} = -(x_n - x_{n-1})$, and (18) can be written as

$$\frac{f(x_n)}{e_n} - \frac{f(x_{n-1})}{e_{n-1}} \approx -\frac{1}{2}(x_n - x_{n-1}) f''(r). \tag{19}$$

Now we insert (19) into (15):

$$e_{n+1} \approx \underbrace{\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}}_{\approx \frac{1}{f'(r)} \text{ for } x_n,\, x_{n-1} \text{ close to } r} \left(-\frac{1}{2}\right) f''(r) e_n e_{n-1}.$$

Thus

$$|e_{n+1}| \approx \frac{1}{2} \left| \frac{f''(r)}{f'(r)} \right| |e_n e_{n-1}|$$

or

$$|e_{n+1}| \approx C |e_n e_{n-1}|. \tag{20}$$

This shows that the rate of convergence of the secant method is not quite quadratic. In order to establish the exact rate of convergence we assume

$$\lim_{n \to \infty} \frac{|e_{n+1}|}{|e_n|^\alpha} = A,$$

i.e., that $|e_{n+1}|$ and $|e_n|^\alpha$ grow asymptotically at the same rate. Our goal is to determine the constants $\alpha$ and $A$. We will use the following notation to denote asymptotically equal growth:

$$|e_{n+1}| \sim A|e_n|^\alpha. \tag{21}$$

(21) also implies

$$|e_n| \sim A|e_{n-1}|^\alpha$$

or

$$|e_{n-1}| \sim \left(\frac{|e_n|}{A}\right)^{1/\alpha}. \tag{22}$$

Now we insert (21) and (22) into (20):

$$A|e_n|^\alpha \sim C|e_n|\frac{|e_n|^{1/\alpha}}{A^{1/\alpha}} \quad\Longleftrightarrow\quad \underbrace{\frac{A^{1+1/\alpha}}{C}}_{\text{const}} \sim |e_n|^{1+1/\alpha-\alpha}. \tag{23}$$

To satisfy (23), i.e., in order for the right-hand side to behave like a constant, we must have

$$1+\frac{1}{\alpha}-\alpha=0.$$

But this is equivalent to

$$\alpha^2-\alpha-1=0 \tag{24}$$

or

$$\alpha=\frac{1\pm\sqrt{1+4}}{2}.$$

We take the positive solution, i.e., $\alpha=\dfrac{1+\sqrt{5}}{2}$. Finally, this choice of $\alpha$ implies that (23) reads as

$$\frac{A^{1+1/\alpha}}{C}\sim 1,$$

or

$$A \sim C^{\frac{1}{1+1/\alpha}} = C^{\frac{\alpha}{\alpha+1}} \overset{(24)}{=} C^{\frac{\alpha}{\alpha^2}} = C^{\frac{1}{\alpha}}.$$

Since $C=\dfrac{1}{2}\left|\dfrac{f''(r)}{f'(r)}\right|$, a look back at (21) finishes the proof. ♠

## Comparison of Root Finding Methods

In Table 1 all of the methods covered thus far (including in HW problems) are summarized. Their convergence order is listed, along with the number of initial points required, as well as the number of function evaluations per iteration. For the *regula falsi* and secant method straight-forward application of the iteration formula would imply 2 function evaluations per iteration, but by keeping the most recent value in memory, this effort can be reduced to one evaluation per iteration after the first one. Other special features not listed in the table are the positive fact that the bisection method **always converges** (provided the two initial points enclose a root), and the negative fact that Newton's method requires the coding of derivatives.

Taking into account both order of convergence and amount of work per iteration, a (more fair) comparison between Newton's method and the secant method should be

|  | order | # initial pts | # fct evals/it |
|---|---|---|---|
| bisection | not linear | 2 bracketing root | 1 |
| regula falsi | $\alpha = 1$ | 2 bracketing root | 1 (with memory) |
| Newton | $\alpha = 2$ | 1 close | 2 |
| Steffensen | $\alpha = 2$ | 1 close | 2 |
| secant | $\alpha = 1.618$ | 2 close | 1 (with memory) |

Table 1: Comparison of root finding methods.

based on the comparison of 2 steps of the secant method to a single Newton step. Thus, for the error for two secant steps we have

$$\begin{aligned} |e_{n+2}| &\approx A|e_{n+1}|^\alpha \approx A\left(A|e_n|^\alpha\right)^\alpha \\ &= A^{1+\alpha}|e_n|^{\alpha^2}. \end{aligned}$$

Now, from (24) we know $\alpha^2 = \alpha + 1 \approx 2.618$. Thus, the convergence rate for two steps of the secant method (which roughly require the same amount of work as one step of Newton's method) is **better than quadratic**.

**Remark:** There are generalizations of the secant method. Suppose we have $k+1$ approximations $x_n, x_{n-1}, \ldots, x_{n-k}$ to $r$. Then we can determine the interpolation polynomial $P$ of degree $k$ to $f$, i.e.,

$$P(x_{n-i}) = f(x_{n-i}), \qquad i = 0, \ldots, k.$$

The next approximation $x_{n+1}$ is the closest root of $P$ to $x_n$.

Note that for $k = 1$ this is just the secant method. For $k = 2$ this procedure results in what is called Müller's method. The cases $k \geq 3$ are rarely used since practical methods for finding roots of higher-degree polynomials are not available.

Polynomial interpolation will be studied in detail in Chapter 6 (most likely at the beginning of next semester).

Müller's method can also be used to find complex roots. It can be shown that Müller's method converges almost quadratically ($\alpha = 1.84$) with only one function evaluation per iteration. We do not give an explicit formula for Müller's method here. Figure 4 illustrates graphically one step of Müller's method.

## 3.4 Fixed Points and Functional Iteration

**Example:** Consider the equation $x^2 - x - 1 = 0$ whose solution is the golden ratio. Equivalent formulations include

$$\begin{aligned} x &= 1 + \frac{1}{x}, \\ x &= x^2 - 1, \\ x &= \pm\sqrt{1+x}. \end{aligned} \tag{25}$$

We use these various equations as iteration formulas to find the golden ratio in the Maple worksheet `577_FixedPoints.mws`. The output is reproduced in Table 2.

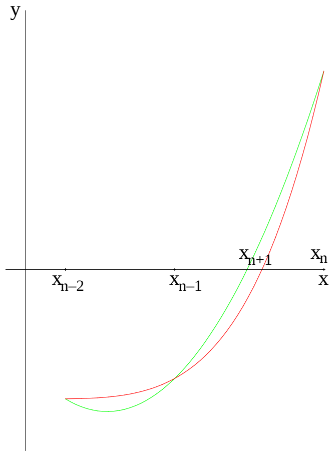

Figure 4: Graphical interpretation of Müller's method.

| $n$ | $x_{n+1} = 1 + \dfrac{1}{x_n}$ | $x_{n+1} = x_n^2 - 1$ | $x_{n+1} = \sqrt{1 + x_n}$ |
|---|---|---|---|
| 0 | 1.500000 | 3.000000 | 1.732051 |
| 1 | 1.666667 | 8.000000 | 1.652892 |
| 2 | 1.600000 | 63.00000 | 1.628770 |
| 3 | 1.625000 | 3968.000 | 1.621348 |
| 4 | 1.615385 | 1.574502e+07 | 1.619058 |
| 5 | 1.619048 | 2.479057e+14 | 1.618350 |
| 6 | 1.617647 | 6.145726e+28 | 1.618132 |
| 7 | 1.618182 | 3.776995e+57 | 1.618064 |
| 8 | 1.617978 | 1.426569e+115 | 1.618043 |
| 9 | 1.618056 | 2.035099e+230 | 1.618037 |
| 10 | 1.618026 | 4.141629e+460 | 1.618035 |
| 11 | 1.618037 | 1.715309e+921 | 1.618034 |
| 12 | 1.618033 | 2.942284e+1842 | 1.618034 |
| 13 | 1.618034 | 8.657038e+3684 | 1.618034 |
| 14 | 1.618034 | 7.494430e+7369 | 1.618034 |

Table 2: Behavior of three different iteration formulas (25) for the golden ratio, $x_0 = 2$.

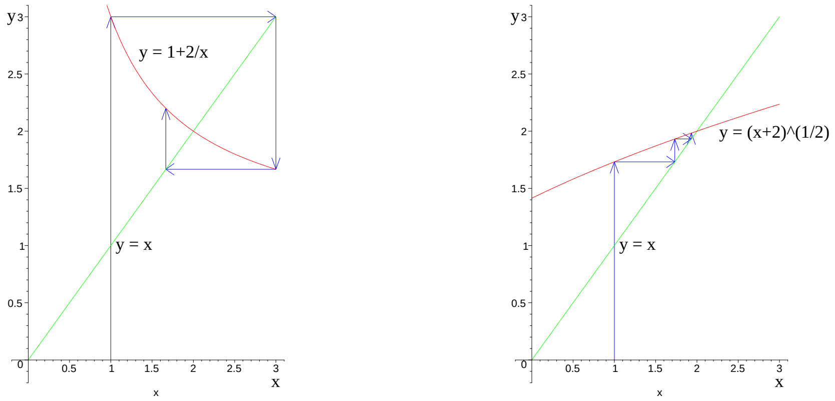

Figure 5: Graphical interpretation of fixed point iteration.

Obviously, one of the iterations does not converge, whereas the other two do. What is it that distinguishes these iteration formulas? We will answer this question below.

Other examples of so-called *fixed point iterations* that we have used earlier are:

$$\begin{aligned}
x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)} = F(x_n), && \text{Newton,} \\
x_{n+1} &= b - \frac{f(b)(b-x_n)}{f(b)-f(x_n)} = G(x_n), && \textit{regula falsi} \text{ for concave functions,} \\
x_{n+1} &= x_n - \frac{f(x_n)}{f(x_n+f(x_n))-f(x_n)} = H(x_n), && \text{Steffensen.}
\end{aligned}$$

In general, an algorithm for fixed point iteration is given by

**Functional (or Picard) Iteration:**

1. Let $x_0$ be an initial guess.
2. For $n$ from 0 to $N$ do

$$x_{n+1} = F(x_n).$$

We can illustrate the behavior of fixed point iteration graphically as in Figure 5. In that figure we have used the two iteration functions $F_1(x) = 1 + 2/x$ and $F_2(x) = \sqrt{x+2}$ which both lead to a solution of $x^2 - x - 2 = 0$. A vertical arrow corresponds to evaluation of the function $F$ at a point, and a horizontal arrow pointing to the line $y = x$ indicates that the result of the previous function evaluation is used as the argument for the next step.

If the fixed point algorithm is to be applied successfully, we need:

1. The iteration is well-defined, i.e., for a given $x_0$ the iterates $x_1, x_2, \ldots$ can be computed.

As a counterexample consider $F(x) = -\sqrt{1+x}$. Then $x_2 = -\sqrt{1 \underbrace{-\sqrt{1+x_0}}_{=x_1}}$, which is not well-defined for $x_0 > 0$.

2. The sequence converges, i.e., $\lim\limits_{n\to\infty} x_n = s$ exists.

3. $s$ is a fixed point of $F$, i.e., $F(s) = s$.

In order to provide a theorem that guarantees 1–3, we first define

**Definition 3.3** *A function (mapping) $F$ is called* contractive *if there exists a $\lambda < 1$ such that*

$$|F(x) - F(y)| \le \lambda |x-y| \tag{26}$$

*for all $x, y$ in the domain of $F$.*

**Remark:** (26) is also referred to as *Lipschitz continuity of $F$.*

**Lemma 3.4** *If $F$ is contractive on $[a,b]$ then $F$ is continuous on $[a,b]$.*

**Proof:** If $F$ is contractive, then by definition,

$$|F(x) - F(y)| \le \lambda |x-y| \qquad \text{for all } x, y \in [a,b],$$

where $\lambda < 1$. Now,

$$\lim_{y\to x} |F(x) - F(y)| \le \underbrace{\lambda}_{<1} \underbrace{\lim_{y\to x} |y-x|}_{=0} = 0.$$

So $F$ is continuous. ♠

**Note:** The converse is not true, i.e., continuity does not imply Lipschitz continuity.
**Ex.:** $F(x) = |x|$ is continuous on $[-1,1]$, but

$$|F(0) - F(1)| = ||0| - |1|| = 1$$

and $|0-1| = 1$. Therefore, no $\lambda < 1$ exists such that contraction property (26) is satisfied for $x = 0$ and $y = 1$.

A sufficient condition for $F$ to be a contraction has to be stronger. For example

**Lemma 3.5** *If $F$ is differentiable with $|F'(x)| \le \lambda < 1$ on $[a,b]$, then $F$ is a contraction.*

**Proof:** The contractive property $|F(x) - F(y)| \le \lambda |x-y|$ is equivalent to

$$\frac{|F(x) - F(y)|}{|x-y|} \le \lambda.$$

But, by the Mean Value Theorem, the left-hand side above is equal to $F'(\xi)$ for some $\xi$ between $x, y \in [a,b]$. Thus, if $F'$ satisfies the stated requirement (for all $x \in [a,b]$), then $F$ is a contraction. ♠

The central theorem of this section is

**Theorem 3.6** (**Contractive Mapping Theorem**) *Let $C$ be a closed subset of the real line and $F$ a contractive mapping of $C$ into itself. Then $F$ has a unique fixed point $s$. Moreover, $s = \lim_{n\to\infty} x_n$, where $x_{n+1} = F(x_n)$ and $x_0$ is any starting point in $C$.*

**Proof:** Consider

$$x_n = x_0 + (x_1 - x_0) + (x_2 - x_1) + \ldots + (x_n - x_{n-1}).$$

Thus, $\lim_{n\to\infty} x_n$ exists if and only if $\lim_{n\to\infty} \sum_{k=1}^{n} (x_k - x_{k-1}) = \sum_{n=1}^{\infty} (x_n - x_{n-1})$ exists.

To show that the infinite series converges, we show that it converges even absolutely, i.e., we show that

$$\sum_{n=1}^{\infty} |x_n - x_{n-1}| \quad \text{converges.}$$

To see this, consider (using functional iteration)

$$|x_n - x_{n-1}| = |F(x_{n-1}) - F(x_{n-2})|.$$

Using the contractivity of $F$ this can be bounded by $\lambda|x_{n-1} - x_{n-2}|$. Repeating this process recursively we arrive at

$$|x_n - x_{n-1}| \leq \lambda^{n-1} |x_1 - x_0|. \tag{27}$$

Thus,

$$\begin{aligned} \sum_{n=1}^{\infty} |x_n - x_{n-1}| &\leq \sum_{n=1}^{\infty} \lambda^{n-1} |x_1 - x_0| \\ &= |x_1 - x_0| \sum_{n=1}^{\infty} \lambda^{n-1} \\ &= |x_1 - x_0| \frac{1}{1-\lambda}, \end{aligned}$$

where we have used the formula for the sum of a geometric series in the last step. Obviously, the infinite series converges, and thus $\lim_{n\to\infty} x_n$ exists.

Now we define $s := \lim_{n\to\infty} x_n$ to be this limit. We can see that $s$ is a fixed point of $F$ since

$$F(s) = F(\lim_{n\to\infty} x_n) = \lim_{n\to\infty} F(x_n) = \lim_{n\to\infty} x_{n+1} = s,$$

where we have made use of the continuity of $F$ (cf. Lemma 3.4) and the process of functional iteration.

To see the uniqueness of $s$ we consider two (different) fixed points $s$ and $t$. Then

$$|s - t| = |F(s) - F(t)| \leq \underbrace{\lambda}_{<1} |s - t| < |s - t|.$$

Obviously, this has led to a contradiction (unless $s$ and $t$ are equal).

Finally, $s \in C$ since $s = \lim_{n\to\infty} x_n$ with $x_n \in C$ and $C$ a closed set. ♠

**Corollary 3.7** *If $F$ maps an interval $[a, b]$ into itself and $|F'(x)| \leq \lambda < 1$ for all $x \in [a, b]$, then $F$ has a unique fixed point $s$ in $[a, b]$ which is the limit of functional iteration $x_{n+1} = F(x_n)$.*

**Examples:** We now take another look at the examples given at the beginning of the section (25).

1. $F_1(x) = 1 + \frac{1}{x}$ with $F_1'(x) = -\frac{1}{x^2}$. So

$$|F_1'(x)| = \left|\frac{1}{x^2}\right| < 1 \qquad \text{for } x > 1.$$

2. $F_2(x) = x^2 - 1$ with $F_2'(x) = 2x$. So

$$|F_2'(x)| = |2x| < 1 \qquad \text{only for } x < 1/2.$$

3. $F_3(x) = \sqrt{1+x}$ with $F_3'(x) = \frac{1}{2\sqrt{1+x}}$. So

$$|F_3'(x)| = \left|\frac{1}{2\sqrt{1+x}}\right| < 1 \qquad \text{for } x > -\frac{3}{4}.$$

**Corollary 3.8** *If $F$ is as in the Contractive Mapping Theorem (or as in Cor. 3.7) then*

$$|s - x_n| \leq \frac{\lambda^n}{1-\lambda}|x_1 - x_0|, \quad n \geq 1.$$

**Proof:** Consider (27) with $n$ replaced by $n+1$, i.e.,

$$|x_{n+1} - x_n| \leq \lambda^n |x_1 - x_0|, \qquad n \geq 0.$$

Now, for any $m > n \geq 0$

$$\begin{aligned}
|x_m - x_n| &= |x_m - x_{m-1} + x_{m+1} - \ldots + x_{n+1} - x_n| \\
&\leq |x_m - x_{m-1}| + |x_{m-1} - x_{m-2}| + \ldots + |x_{n+1} - x_n| \\
&\leq \lambda^{m-1}|x_1 - x_0| + \lambda^{m-2}|x_1 - x_0| + \ldots + \lambda^n |x_1 - x_0| \\
&= \lambda^n \left(1 + \lambda + \lambda^2 + \ldots + \lambda^{m-n-1}\right)|x_1 - x_0|.
\end{aligned}$$

From above we know $\lim_{m\to\infty} x_m = s$, so

$$\begin{aligned}
|s - x_n| &= \lim_{m\to\infty} |x_m - x_n| \\
&\leq \lambda^n \sum_{k=0}^{\infty} \lambda^k |x_1 - x_0| \\
&= \frac{\lambda^n}{1-\lambda}|x_1 - x_0|.
\end{aligned}$$

♠

**Remark:** Corollary 3.8 relates the bound on $|F'(x)|$ to the rate of convergence of the fixed point iteration. In particular, $|e_n| = |s - x_n|$ will go to zero fast if $\lambda$ is small, i.e., $|F'(x)|$ is small.

In fact,

**Detailed Convergence Analysis**

Consider

$$\begin{aligned} e_{n+1} &= s - x_{n+1} \\ &= F(s) - F(x_n), \end{aligned}$$

where we have used the functional iteration along with the fact that $s$ is a fixed point of $F$. By the Mean Value Theorem the previous is equal to $F'(\xi_n)(s - x_n)$ for some $\xi_n$ between $s$ and $x_n$. Thus,

$$e_{n+1} = F'(\xi_n)e_n.$$

Taylor's Theorem implies

$$\begin{aligned} e_{n+1} &= s - x_{n+1} = F(s) - F(x_n) \\ &= F(s) - F(s - e_n) \\ &= F(s) - \left[F(s) - e_nF'(s) + \frac{1}{2}e_n^2F''(s) + \ldots\right]. \end{aligned}$$

In other words,

$$e_{n+1} = e_nF'(s) - \frac{1}{2}e_n^2F''(s) + \ldots + \frac{(-1)^k}{(k-1)!}e_n^{k-1}F^{(k-1)}(s) + \frac{(-1)^{k+1}}{k!}e_n^kF^{(k)}(\xi_n).$$

If $F'(s) = F''(s) = \ldots = F^{(k-1)}(s) = 0$ we get

$$|e_{n+1}| = \frac{|e_n^k|}{k!}|\underbrace{F^{(k)}(\xi_n)}_{\approx F^{(k)}(s) \text{ for convergent iteration}}|.$$

Equivalently, we can write

$$\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^k} = \frac{|F^{(k)}(s)|}{k!} = \text{const}.$$

Thus, $k$ gives us the **order of convergence** of the fixed point iteration defined by $F$.

This means that if we can find $k$ such that $F^{(k)}(s) \neq 0$ and $F^{(j)}(s) = 0$, $j < k$, then the iteration $x_{n+1} = F(x_n)$ converges with order $k$. For linear convergence we also require that $|F'(s)| < 1$.

**Example:** (Rate of convergence of *regula falsi*) Recall that the iteration is defined by

$$x_{n+1} = b - \frac{f(b)(b - x_n)}{f(b) - f(x_n)},$$

so

$$F(x) = b - \frac{f(b)(b - x)}{f(b) - f(x)}$$

and

$$F'(x) = -\frac{-f(b)\left(f(b)-f(x)\right)+f(b)(b-x)f'(x)}{\left(f(b)-f(x)\right)^2}.$$

Therefore,

$$F'(s) = \frac{f(b)\left(f(b)-f(s)\right)-f(b)(b-s)f'(s)}{\left(f(b)-f(s)\right)^2}. \tag{28}$$

Remember that we are finding the root of the function $f$, i.e., $f(s) = 0$. Therefore, (28) simplifies to

$$\begin{aligned} F'(s) &= \frac{\left(f(b)\right)^2 - f(b)(b-s)f'(s)}{\left(f(b)\right)^2} \\ &= 1 - f'(s)\frac{b-s}{f(b)} \\ &= 1 - f'(s)\frac{b-s}{f(b)-\underbrace{f(s)}_{=0}}. \end{aligned} \tag{29}$$

Note that $\dfrac{f(b)-f(s)}{b-s} = f'(\xi)$ for some $s < \xi < b$.

Furthermore, concavity of the function $f$ (which we assume for this form of the *regula falsi*) implies that $f'$ is increasing.

Therefore, $f'(\xi) > f'(s)$ and $\dfrac{f'(s)}{f'(\xi)} < 1$. Using this in (29) we see that $0 < F'(s) < 1$, and we have proven linear convergence.

**Aitken Acceleration**

Can be used to accelerate the convergence of any linearly convergent iteration scheme (not only fixed point iteration).

Recall the definition of linear convergence

$$\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|} = C, \quad 0 < C < 1,$$

i.e., for large $n$ we have

$$e_{n+1} \approx Ce_n. \tag{30}$$

Then we also have $e_{n+2} \approx Ce_{n+1}$ or

$$C \approx \frac{e_{n+2}}{e_{n+1}}. \tag{31}$$

Inserting (31) into (30) we get

$$e_{n+1}^2 \approx e_{n+2}e_n.$$

Next we replace $e_n$ by $s - x_n$ for all relevant values of $n$. This leads to

$$\begin{aligned} & (s-x_{n+1})^2 &\approx\;& (s-x_{n+2})(s-x_n) \\ \iff\quad & s^2 - 2sx_{n+1} + x_{n+1}^2 &\approx\;& s^2 - s(x_{n+2}+x_n) + x_{n+2}x_n \end{aligned}$$

or

$$s \approx \frac{x_{n+2}x_n - x_{n+1}^2}{x_{n+2} - 2x_{n+1} + x_n}. \tag{32}$$

We now introduce some notation commonly used in this context. Basic *forward differences* are defined as

$$\Delta x_n = x_{n+1} - x_n,$$

and higher-order forward differences are defined recursively, i.e.,

$$\Delta^2 x_n = \Delta(\Delta x_n) = \Delta(x_{n+1} - x_n) = x_{n+2} - x_{n+1} - (x_{n+1} - x_n) = x_{n+2} - 2x_{n+1} + x_n.$$

Thus (32) implies

$$s \approx \frac{x_{n+2}x_n - x_{n+1}^2}{\Delta^2 x_n} =: \hat{x}_{n+2}. \tag{33}$$

**<u>Claim:</u>** The $\{\hat{x}_n\}$ sequence converges faster than the $\{x_n\}$ sequence, i.e.,

$$\lim_{n\to\infty} \frac{\hat{e}_n}{e_n} = 0.$$

**Proof:** First we show that the errors satisfy (33), i.e.,

$$\hat{e}_{n+2} = \frac{e_{n+2}e_n - e_{n+1}^2}{\Delta^2 e_n}. \tag{34}$$

This is true since the denominator of the right-hand side yields

$$\begin{aligned} \Delta^2 e_n &= e_{n+2} - 2e_{n+1} + e_n \\ &= s - x_{n+2} - 2(s - x_{n+1}) + s - x_n \\ &= -\Delta^2 x_n, \end{aligned}$$

whereas for the numerator we get

$$\begin{aligned} e_{n+2}e_n - e_{n+1}^2 &= (s - x_{n+2})(s - x_n) - (s - x_{n+1})^2 \\ &= s^2 - s(x_{n+2} + x_n) + x_{n+2}x_n - s^2 + 2sx_{n+1} - x_{n+1}^2 \\ &= x_{n+2}x_n - x_{n+1}^2 - s\Delta^2 x_n \\ &\overset{(33)}{=} \hat{x}_{n+2}\Delta^2 x_n - s\Delta^2 x_n \\ &= -\hat{e}_{n+2}\Delta^2 x_n \end{aligned}$$

Combining the two partial results we get

$$\frac{e_{n+2}e_n - e_{n+1}^2}{\Delta^2 e_n} = \frac{-\hat{e}_{n+2}\Delta^2 x_n}{-\Delta^2 x_n} = \hat{e}_{n+2}.$$

This establishes (34). Now we compute $\lim_{n\to\infty} \frac{\hat{e}_{n+2}}{e_{n+2}}$. Since $\{x_n\}$ converges linearly we can write

$$e_{n+1} = (c + \delta_n)e_n, \quad \text{where } \delta_n \to 0 \text{ as } n \to \infty, \text{ and } 0 < c < 1. \tag{35}$$

Using (34) we get

$$\frac{\hat{e}_{n+2}}{e_{n+2}} = \frac{e_n - \frac{e_{n+1}^2}{e_{n+2}}}{\Delta^2 e_n} = \frac{e_n - \frac{e_{n+1}^2}{e_{n+2}}}{e_{n+2} - 2e_{n+1} + e_n}.$$

Factoring $e_n$ from the numerator and denominator the previous expression becomes

$$\frac{e_n\left(1 - \frac{e_{n+1}^2}{e_n e_{n+2}}\right)}{e_n\left(\frac{e_{n+2}}{e_n} - 2\frac{e_{n+1}}{e_n} + 1\right)},$$

or, using (35),

$$\frac{\hat{e}_{n+2}}{e_{n+2}} = \frac{1 - \frac{c+\delta_n}{c+\delta_{n+1}}}{\frac{e_{n+2}}{e_{n+1}}\frac{e_{n+1}}{e_n} - 2\frac{e_{n+1}}{e_n} + 1}.$$

Finally, we can also apply (35) to the denominator to arrive at

$$\frac{\hat{e}_{n+2}}{e_{n+2}} = \frac{1 - \frac{c+\delta_n}{c+\delta_{n+1}}}{(c+\delta_{n+1})(c+\delta_n) - 2(c+\delta_n) + 1},$$

so that, for $\delta_n \to 0$ we have

$$\frac{\hat{e}_{n+2}}{e_{n+2}} \to \frac{0}{c^2 - 2c + 1} = \frac{0}{(c-1)^2},$$

which verifies the claim (as $0 < c < 1$ by assumption). ♠

**Implementation of Aitken's Acceleration**

1. Rewrite (33) as

$$\hat{x}_{n+2} = x_{n+2} - \frac{(\Delta x_{n+1})^2}{\Delta^2 x_n} \qquad \text{(see homework).} \tag{36}$$

   This is more stable since an existing quantity is updated by a small correction.

2. Even with (36) the second difference $\Delta^2 x_n$ in the denominator can lead to cancellation of significant digits and blow-up.

3. Here is a practical algorithm (sometimes referred to as *Steffensen's method for functional iteration*):

```
Input iteration function $F$, initial guess $x_0$
for $k = 0, 1, 2, \ldots$ do
    $z_0 = x_k$
    % Perform 2 steps of regular fixed point iteration:
    $z_1 = F(z_0)$
    $z_2 = F(z_1)$
    % Aitken update
    $x_{k+1} = z_2 - \frac{(z_2 - z_1)^2}{z_2 - 2z_1 + z_0}$
    end do
```

You will implement this with $F(x) = \sqrt{10/(x+4)}$ and compare it's performance with regular functional iteration on the homework.

**Remarks:**

1. Since cancellation can occur when iterates are close, one needs to stop the algorithm in this case.
2. Aitken's acceleration applied to partial sums of convergent series yields the sum of the series. You will do as a homework problem:

   Show, if $\{x_n\}$ is the sequence of partial sums of the geometric series $\sum\limits_{k=0}^{\infty} ar^k$, $|r| < 1$, then $\hat{x}_2 = \dfrac{a}{1-r}$ (the sum of the series).
3. Aitken's acceleration may also yield the "sum" of divergent series!