SOLVING NONLINEAR EQUATIONS

David F. Gleich · CS 314 · Purdue University

November 27, 2016

1 MOTIVATION

Consider launching a projectile from a cannon. The *artillery problem* is to determine where to aim the cannon such that the projectile launched at a fixed velocity will reach a target at a given distance. The height of the projectile, as a function of time t, follows the equation:

$$y''(t) = -g - ky'(t), \text{ where}$$
$$y'(0) = v_0 \sin\theta \quad \textit{initial velocity}$$
$$y(0) = 0 \quad \textit{initial position}.$$

In this case, θ is the angle of the cannon and v_0 is the total velocity of the projectile, so The coefficient g models the effect of gravity. The coefficient k models the air resistance, which is proportional to velocity. The solution of this differential equation can be found analytically:

$$y(t) = -\frac{1}{k}e^{-kt}v_0\sin\theta - \frac{g}{k}(t + \frac{1}{k}e^{-kt}) + \frac{1}{k}v_0\sin\theta + \frac{g}{k^2}.$$

The projectile hits the ground when $y(t) = 0$ and $t > 0$ (since it starts on the ground). Let t_{impact} be this time. The horizontal position satisfies the equation:

$$x''(t) = -kx'(t), x'(0) = v_0\cos\theta, x(0) = 0.$$

Notice that this has an identical form with $g = 0$, thus, we have:

$$x(t) = \left(\frac{1}{k} - \frac{1}{k}e^{-kt}\right)v_0\cos\theta.$$

The horizontal distance the projectile travels is:

$$x(t_{\text{impact}}).$$

The problem Given that we want $x(t_{\text{impact}}) = d$ (the distance to a target), v_0, k, g, how do we choose θ?

General form Note that $x(t_{\text{impact}})$ can be computed by a computer function that depends on d, v_0, k, g, θ. Let $f(\theta)$ be this function as θ varies. We want to find θ such that $f(\theta) = d$.

Consider the nonlinear ordinary differential equation:

$$y'(t) = (t+1)e^{-y}, y(0) = 0$$

If we apply the backwards Euler method with timestep $h = 0.1$, then we need to solve:

$$\frac{y(h) - y(0)}{h} = f(h, y(h)),$$

or, with all the variables substituted:

$$\frac{y(0.1) - 0}{0.1} = (0.1 + 1)e^{-y(0.1)}.$$

Let $y(0.1) = a$ be the unknown we are trying to solve for. Then:

$$ae^a = (0.1)(0.1 + 1).$$

The problem There is no analytic formula for solutions of such equations. By convention, solutions are referred to through the product-log function or Lambert's W function. We need a way to solve this function in order to use our backwards Euler method!

General form Again, note that this problem can be written

$$\text{find } a \text{ such that } f(a) = c.$$

2 NONLINEAR EQUATIONS

The general problem of solving nonlinear equations is to find x such that:

$$f(x) = 0.$$

The function f may be a scalar function $f : \mathbb{R} \mapsto \mathbb{R}$ or a multivariate function $f : \mathbb{R}^m \mapsto \mathbb{R}^n$. In this class, we'll only briefly mention the multivariate case.

Why is this form general? In both of the previous two examples, we wanted to solve $f(x) = c$ for some value of c. In the artillery problem, this was $f(\theta) = d$; and in the backward Euler example, this was $f(a) = (0.1)(0.1 + 1)$.

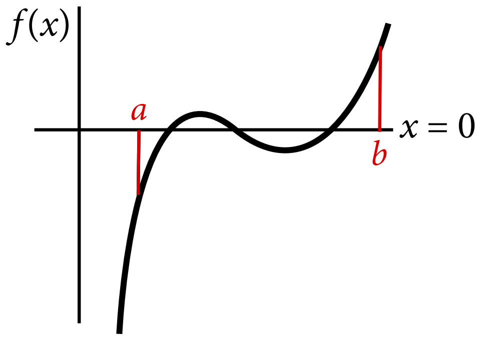
Quiz Why do you think we state the problem in the form $f(x) = 0$ instead of $f(x) = c$?

3 THE BISECTION ALGORITHM

Assumptions The function f is continuous, scalar ($f : \mathbb{R} \mapsto \mathbb{R}$).

In this case, then the function's behavior is not arbitrary. If f is continuous, and we have two values a and b of x such that $f(a)$ and $f(b)$ have different signs, then the function f must have a point where $f(x) = 0$ between a and b.[1]

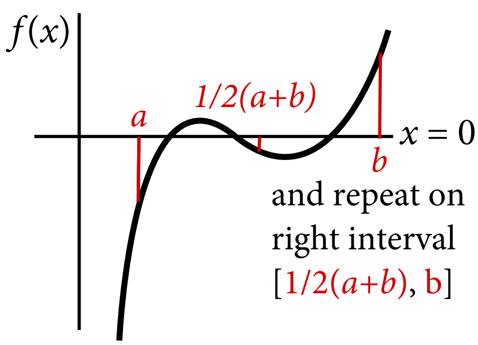
[1] This is a corollary of the intermediate value theorem if you like such things!



In this example, it has three such points! So these points need not be unique. But, if we evaluate $f((a+b)/2)$, then one of three things can happen:

1. $f((a+b)/2)$ has the same sign as $f(a)$
2. $f((a+b)/2)$ has the same sign as $f(b)$
3. $f((a+b)/2) = 0$.

If $f((a+b)/2)$ has the same sign as $f(a)$, then we *know* that $[(a+b)/2, b]$ is a *smaller* subinterval that must contain a value of x where $f(x) = 0$. Likewise, if $f((a+b)/2)$ has the same sign as $f(b)$, then we *know* that $[a, (a+b)/2]$ is a smaller subinterval that must contain a value of x where $f(x) = 0$. Finally, if $f((a+b)/2) = 0$, then we are done and can just return that value of x.



In this case $f((a+b)/2)$ has the same sign as $f(a)$.

```
function x0 = bisection(f, a, b, Δ)
% BISECTION Find a point where f(x) = 0 through bisection
% x0 = bisection(f, a, b, Δ) does an interval bisection search
% to find a region of size Δ that contains a zero of
% the function f, by default Δ = 2.2e-16, the machine eps.
fa = f(a); fb = f(b); assert(sign(fa*fb) ≤ 0); maxit = 52;
for i=1:maxit
    ab2 = 0.5*a + 0.5*b; fab2 = f(ab2); if abs(fab2) < eps, break; end
    if abs(b-a) ≤ Δ, break; end
    if sign(fab2*fb) ≤ 0, a = ab2; fa = fab2;
    else b = ab2; fb = fab2; end
end
x0 = ab2;
```

NEWTON'S METHOD

The bisection method has one key problem: it requires a and b such that $f(a)$ and $f(b)$ have different signs. So we could never use it to solve $f(x) = x^2 = 0$. Newton's method is an alternative based on the following picture:

In words, if we know the derivative at the current point x, then we can use a linear approximation based on the gradient to find a point y that should be closer to a zero of the function f. This gives rise to the *iteration*

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

We can also derive Newton's method from Taylor's theorem. Recall that for a twice continuously differentiable function f, that

$$f(x+h) = f(x) + hf'(x) + \tfrac{1}{2}h^2 f''(\xi).$$

Notation We often write the solution of $f(x) = 0$ as x^*. So $f(x^*) = 0$.

Notation Writing x_k means the kth choice of x in the sequence generated by our algorithm.

Notation The variable x_0 is usually the starting point of our iteration. In most computer implementations, the user needs to provide this choice.

Theorem 4.3.1 If x_0 is *sufficiently close* to x^*, then the sequence of iterates generated by Newton's method converges to a solution *quadratically fast*.

Interpretation For each root of the equation x^*, there is some region around x^* where if we start Newton's method, then it'll get to the solution x^*. The precise form of this region is really hard to state as it involves the term ξ from Taylor's theorem and a bunch of approximations on *continuous functions* that are easy to state and hard to quantify. The term *quadratically fast* means *really fast!*. Here's an example. If $x_k \to 0$, then a quadratic sequence looks like:

0.1 0.01 0.0001 0.00000001 0.0000000000000001 0.00000000000000000000000000000001

Note Since Newton's method may diverge, *if* you ever detect that the sign of the function changes as you are running Newton's method, *then* you can combine Newton and bisection in order to guarantee convergence.

4 Fun with Newton's method

There are many really interesting things you can do with Newton's method. Suppose someone gave you a calculator with a broken "division" key. If you are familiar with Newton's method, you can still compute a/b with ease!

Idea 1 The term a/b is the solution of $bx - a = 0$. So maybe we should apply Newton's method here, but that just leads to $x_{k+1} = x_k - \frac{bx_k - a}{b} = a/b$. While it's nice that this is correct, we can't evaluate this term without the division symbol.

If you had enough time, you'd all come up with this idea, I promise! Consider:

$$f(x) = \frac{1}{x} - b.$$

Then $f(x) = 0$ when $x = 1/b$; and so we can multiply the resulting answer by a. If we apply Newton's method, then

$$x_{k+1} = x_k - \frac{\frac{1}{x_k} - b}{-\frac{1}{x_k^2}} = x_k + x_k^2\left(\frac{1}{x_k} - b\right) = x_k(2 - bx_k).$$

To continue with this idea, we'd need show whether or not this idea converges for all values b or not. But this illustrates one of the most common uses of Newton's method! You can compute complicated functions through much more simple operations.

5 The secant method

One annoying feature of Newton's method is that we need a function to evaluate $f'(x)$ as well as $f(x)$. This requires more work for people to use (bad), and is often a source of bugs that lead to the method failing (double-bad!). In the secant method, we do the *lazy* thing and approximate $f'(x) \approx \frac{1}{h}(f(x+h) - f(x))$. But, we can be even smarter! Since we have a sequence of iterations x_k, just use the approximation

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

Thus, secant method is:

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}.$$

To start the Secant method, we need either, x_0 and x_1 *or* we can approximate $f'(x_0)$ via a finite difference formula for the first step. (Your book doesn't discuss this, but it's a reasonable idea.)

See Lemma 4.4.1 for a proof of convergence.

6 Fixed point methods

There is another class of methods to solve $f(x) = 0$, or non-linear equations in general. These are called *fixed point methods* and they study solutions of:

$$g(x) = x.$$

These *fixed points* come up so often that they get their own name. The PageRank method is an example of a linear fixed point equation (for a matrix!).

6.1 Fixed points are equivalent to nonlinear equations

If $g(x) = x$, then $f(x) = g(x) - x = 0$ is an equivalent problem.

If $f(x) = 0$, then $g(x) = f(x) + x = x$ is an equivalent problem. So is $g(x) = -f(x) + x$.

The fixed point statement implies an algorithm:

$$x_{k+1} = g(x_k).$$

See Theorem 4.5.2 for a convergence statement.