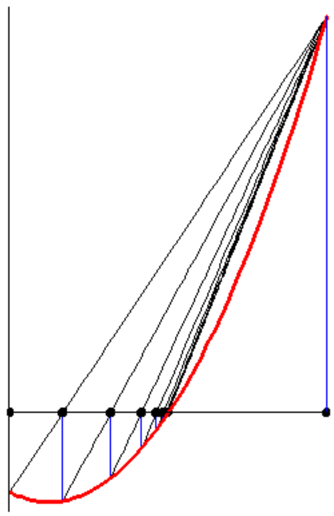
Math 4650 Midterm Solutions

1. Regula Falsi (false position) was proposed as a method that combines the best of two methods: bisection and the secant method.

   Explain, using words, diagrams, and/or formulas, why it's actually worse than either method in practice. (There are several correct answers.)

   Solution:

   The primary reason Regula Falsi is worse than bisection is that it generally does not bound the root effectively: in most cases (when the function does not have an inflection point at the root) only one endpoint actually gets close to the root, while the other endpoint remains fixed. Refer to the diagram for an example, where the left endpoint approaches the root, while the right endpoint never moves.



   For the same reason, Regula Falsi is generally slower than the secant method. The order of convergence of the secant method is approximately $\alpha \approx 1.62$, while Regula Falsi is in general only linear in convergence ($\alpha = 1$).

   The reason for the linear convergence rate is that once one of the endpoints stops moving (say for example the right endpoint b), all future iterations use the formula

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

   Writing $g(x) = b - \frac{x-b}{f(x)-f(b)} f(b)$, we compute that when $f(p) = 0$, we do indeed have $g(p) = p$, so p is a fixed point. However

$$g'(x) = \frac{f(b)\big(-f(x) + f(b) + xf'(x) - bf'(x)\big)}{\big(f(x) - f(b)\big)^2},$$

   so that at the fixed point p, we have

$$g'(p) = 1 + f'(p)\frac{p-b}{f(b)}.$$

This will generally be small but not zero (since $f'(p) \approx \frac{f(p)-f(b)}{p-b} = -\frac{f(b)}{p-b}$). Hence the method is linear once an endpoint stops moving.

2. Consider the sequence $p_n = \frac{1}{n-1}$ for $n \geq 2$.

   (a) Is p_n linearly convergent? If so, what is the asymptotic error constant?
   Solution: Since $p = \lim_{n\to\infty} p_n = 0$, we have

$$\lim_{n\to\infty} \frac{p_{n+1} - p}{p_n - p} = \lim_{n\to\infty} \frac{n}{n-1} = 1.$$

   Hence the method is linearly convergent, with asymptotic error constant $\lambda = 1$.

   (b) Use Aitken's method to get a new sequence $\hat{p}_n$, and fully simplify the formula for it. Does the new sequence converge substantially faster than the old one?
   Solution:
   Aitken's formula says

$$\begin{aligned}
\hat{p}_n &= \frac{p_{n+2}p_n - p_{n+1}^2}{p_{n+2} - 2p_{n+1} + p_n} \\
&= \frac{\frac{1}{n+1}\frac{1}{n-1} - \frac{1}{n^2}}{\frac{1}{n-1} - \frac{2}{n} + \frac{1}{n+1}} \\
&= \frac{\frac{n^2-(n^2-1)}{n^2(n^2-1)}}{\frac{n^2+n-2n^2+2+n^2-n}{n(n^2-1)}} \\
&= \frac{\frac{1}{n^2(n^2-1)}}{\frac{2}{n(n^2-1)}} \\
&= \frac{1}{2n}.
\end{aligned}$$

   It certainly does not converge substantially faster than the old sequence: the order and asymptotic error constant are both the same.

3. (a) Estimate $\int_0^1 x^3\,dx$ using the trapezoid rule with $n = 1$. What is the actual error?
   Solution:
   The trapezoid rule with $n = 1$ predicts $T = \frac{(1-0)}{2}\big(f(0) + f(1)\big) = 0.50$.
   The actual value is $\int_0^1 x^3\,dx = \frac{x^4}{4}\Big|_0^1 = 0.25$.
   So the actual error is 0.25.

   (b) What is the maximum error predicted by the general error formula?
   The general error formula is

$$E = \left|\frac{h^3}{12} f''(\xi)\right| = \frac{1}{2}|\xi| \leq 0.50.$$

   The actual error is within the general error bound.

4. Knowing that $f(0) = 2$, $f'(0) = 8$, $f(2) = 6$, and $f'(2) = 0$, estimate $f(1)$.

Solution: This looks like a job for Hermite polynomials.

The divided difference table looks as follows:

0	2			
0	2	8		
2	6	2	-3	
2	6	0	-1	1

Reading the coefficients off the diagonal, we get the Hermite polynomial

$$P(x) = 2 + 8x - 3x^2 + x^2(x - 2).$$

Therefore

$$f(1) \approx P(1) = 2 + 8 - 3 - 1 = 6.$$

5. Consider the following algorithm.

```
X = 1
Err = 1
WHILE Err>0.0001 DO
   Xnew = 3X - 1/X
   Err = ABS(Xnew-X)
   X = Xnew
END DO
PRINT X
```

(a) Go through the loop three times. What X comes out?

Solution:

Before the loop begins, $X = 1$.

After one iteration, $X = 3 - 1 = 2$.

After two iterations, $X = 6 - 1/2 = \frac{11}{2} = 5.5$.

After three iterations, $X = \frac{33}{2} - \frac{2}{11} \approx 16.3$.

(b) How many steps will this program take?

Solution:

The program will keep running in an infinite loop, since there is no way the error could ever be as small as demanded. In fact the X values effectively get multiplied by 3 at every step (as X starts to get large, the $1/X$ term has virtually no effect), so that X goes to infinity, as does the error term.

(c) What X do you think the programmer was trying to approximate?

Solution:

The programmer is iterating the function $g(x) = 3x - \frac{1}{x}$, so presumably the programmer was hoping to reach the fixed point of this function. That fixed point happens to satisfy $p = 3p - \frac{1}{p}$, so that $2p^2 = 1$, or $p = \pm\frac{1}{\sqrt{2}}$. Since the program starts at $X = 1$, presumably the programmer wanted to approximate $\frac{1}{\sqrt{2}}$.

(d) Why would this program not work as expected?

Solution:

The function g has $g'(x) = 3 + \frac{1}{x^2}$, so that at the fixed point $p = \frac{1}{\sqrt{2}}$, we have $g'(p) = 3 + 2 = 5$. Since $|g'(p)| > 1$, the fixed point is unstable, and no iteration program will converge to it (unless it started exactly on the fixed point already).

6. Suppose you estimate $f'(3)$ using the formula from class,

$$f'(3) \approx \frac{4f(3+h) - f(3+2h) - 3f(3)}{2h}.$$

When $h = 0.2$, you obtain $f'(3) \approx 4.8$.

When $h = 0.1$, you obtain $f'(3) \approx 4.5$

Use Richardson's extrapolation to obtain the best estimate of the true value of $f'(3)$.

Solution:

Writing $N(h) = [4f(3+h) - f(3+2h) - 3f(3)]/[2h]$, the error formula looks like

$$f'(3) = N(h) + \frac{h^2}{3} f'''(3) + \cdots .$$

Thus we have

$$f'(3) = N(0.2) + \frac{0.2^2}{3} f'''(3) + \cdots$$
$$f'(3) = N(0.1) + \frac{0.1^2}{3} f'''(3) + \cdots$$

Richardson's extrapolation technique suggests subtracting four times the second equation minus the first, to cancel out the unknown $f'''(3)$ term. Then we get

$$4f'(3) - f'(3) \approx 4N(0.1) - N(0.2) + O(0.1^3),$$

so that the best estimate for $f'(3)$ we can get with the given data is

$$f'(3) = \frac{4N(0.1) - N(0.2)}{3} = \frac{18 - 4.8}{3} = \frac{13.2}{3} = 4.4.$$

7. While driving from Boulder to Denver on Route 36 during rush hour to measure traffic congestion, a traffic engineer recorded the car's speed from the speedometer every five minutes and got the following values.

time (min)	0	5	10	15	20	25	30	35	40	45	50	55	60
speed (mph)	0	23	57	65	28	17	45	71	64	44	22	31	0

(a) If you were trying to determine the exact time at which the speed was lowest, what method would you use and why? (Don't actually *do* it, just describe the process in enough detail that anyone could program it and get the same answer.)

Solution:

Since we are dealing with experimentally-observed measurements with a low degree of precision, we should definitely *not* use an interpolating polynomial on this data. It will oscillate wildly (just like the duck from the homework problem), and so it will probably predict large drops where none exist.

Instead using a spline makes much more sense. Quadratic is unusable because there's no way to set the endpoint conditions, while linear is a bad idea since it would just predict the low of 17 miles per hour at 25 minutes. Instead we want to use cubic splines, since they will accurately reflect the shape of the graph.

The final question is what endpoint conditions to use. Although there are several logical possibilities, the one that makes most sense to me is to use a clamped spline with derivatives at the endpoints set to zero. This is because the derivative of the velocity is the acceleration, and since the driver will be accelerating and decelerating gradually (unless the car's tires are squealing out of the parking lot), they will start and end at zero.

So the algorithm would be to find the cubic spline with clamped boundary conditions through the given data, look at each cubic equation to find where it is minimized in each interval (we can do this exactly, since the derivative is a quadratic), and take the minimum over all intervals.

(b) If you were trying to determine where on Route 36 traffic was the worst, how would you get that position from this data? Why would you use that method? (Again, don't do it, just describe the process precisely.)

Solution: The position is the integral of the velocity, so we want to compute a numerical integral of the given data. The initial time is zero, of course, but the final time is most likely not one of the given points. (It is found from part (a).)

There are a couple of reasonable methods. One is to use the composite Simpson's rule on the data above up to the last data point before the minimizing time. Another, since we would have already found the spline coefficients in doing part (a), is to simply integrate each cubic spline, again up to the last data point before the minimizing time.

The final bit is integrating between the last data point (which is at, e.g., 20 minutes) and the actual minimizing time (which may be at, e.g., 23 minutes). To do this, I would use the formula for the cubic spline between 20 and 23, and integrate that cubic polynomial exactly from 20 to 23.

The reason for using one of these methods is that, since the data is experimentally measured, it makes a lot more sense to use a composite method than, say, a high-order Newton-Cotes method (for the same reason that it makes more sense to use splines than a high-order interpolating polynomial). Also, Simpson's method tends to be more accurate than either the midpoint or the trapezoid method. And since the graph keeps changing concavity, it doesn't seem likely that the midpoint or trapezoid rules would be useful, since they're mainly good if one wants upper- and lower-bounds for integrals of functions with no inflection points.