Halley's Iteration

Halley's method provides an infinite number of higher-order generalizations of Newton's method for finding a root of a single nonlinear equation. The method requires analytical and numerical computation of higher-order derivatives of the function in question. The general algorithm for any fixed value of $n = 0, 1, \ldots$ is to iterate for $k = 0, 1, 2, \ldots$

$$x_{k+1} = x_k + (n+1) \frac{\left(\dfrac{1}{f(x_k)}\right)^{(n)}}{\left(\dfrac{1}{f(x_k)}\right)^{(n+1)}},$$

starting with an initial guess x_0. The superscript in the above formula denotes the order of the derivative.

The case $n = 0$ gives Newton's iteration because

$$\left(\frac{1}{f}\right)' = -\frac{f'}{f^2},$$

and therefore the algorithm is:

$$x_{k+1} = x_k + \frac{\left(\dfrac{1}{f(x_k)}\right)}{\left(\dfrac{-f'(x_k)}{f^2(x_k)}\right)} = x_k - \frac{f(x_k)}{f'(x_k)}.$$

This iteration converges quadratically, and so, roughly speaking the number of correct digits obtained doubles with every iteration.

For $n = 1$, since

$$\left(\frac{1}{f}\right)'' = -\frac{-f^2 f'' + 2f(f')^2}{f^4} = \frac{2(f')^2 - ff''}{f^3}$$

the algorithm is:

$$x_{k+1} = x_k - \frac{2f(x_k)f'(x_k)}{2(f'(x_k))^2 - f(x_k)f''(x_k)}.$$

One can show that this iteration has cubic convergence. This means that the number of correct digits obtained roughly triples with every iteration.

One can in principle continue in this fashion to obtain iterative formulas that exhibit quartic and higher convergences. However the iteration formula then involves higher order derivatives of $f(x)$ and thus becomes more and more complicated and requires more and more time to compute per step. The optimal value of n is difficult to find, but must depend on how easy it is to compute the function and its various derivatives. On machines where the machine ϵ is about 10^{-16}, it does not make too much sense to use algorithms that have convergence rates higher than 3 or 4 because one very quickly encounters the problem with limited precision.