COMPUTER SCIENCE 349A

Handout Number 14

ZEROS OF POLYNOMIALS USING NEWTON'S METHOD WITH HORNER'S ALGORITHM, AND POLYNOMIAL DEFLATION

Outline of a procedure to compute a zero of a polynomial $f(x)$ using Newton's method and Horner's algorithm:

Let x_0 be an initial approximation to a zero of $f(x)$
for $i = 1$ to imax
  use Horner's algorithm to evaluate $f(x_{i-1})$ and $f'(x_{i-1})$
  set $x_i \leftarrow x_{i-1} - \dfrac{f(x_{i-1})}{f'(x_{i-1})}$
  if $\left| 1 - x_{i-1} / x_i \right| < \varepsilon$ exit
end
output failed to converge in imax iterations

Polynomial Deflation. Suppose that the values $x_0, x_1, x_2, \ldots$ computed above converge in N iterations. Then x_N is the final computed approximation to some zero, say r_1, of $f(x)$. Now the final computation in the above procedure with Newton's method (after N iterations) is

$$x_N \leftarrow x_{N-1} - \frac{f(x_{N-1})}{f'(x_{N-1})} \quad .$$

If $b_n, b_{n-1}, \ldots, b_0$ are the values computed by Horner's algorithm to evaluate $f(x_{N-1})$ -- that is, in the last step of the above procedure (when $i = N$), then from page 2 of Handout Number 13 it follows that

(1) $$f(x) = (x - x_{N-1})Q(x) + b_0 ,$$

where

(2) $$Q(x) = b_1 + b_2 x + b_3 x^2 + \cdots + b_n x^{n-1} .$$

On letting $x = x_{N-1}$ in (1), we obtain

$$b_0 = f(x_{N-1}) \approx 0 \text{ since } x_{N-1} \approx x_N \approx \text{the zero } r_1 \text{ of } f(x).$$

Therefore, from (1),

$$f(x) \approx (x - x_{N-1})Q(x)$$

and consequently

$$Q(x) \approx \frac{f(x)}{x - x_{N-1}}.$$

That is, the polynomial $Q(x)$ defined in (2) above is the **deflated polynomial**; it is a polynomial of degree $n-1$ whose zeros are equal to those of $f(x)$, except for the zero at $x_{N-1} \approx r_1$. Note that the coefficients $b_1, b_2, \ldots, b_n$ of $Q(x)$ are determined from the last application (when $i = N$) of Horner's algorithm in the procedure at the beginning of this handout.

****

SUMMARY of the above procedure to compute an approximation x_N to some zero r_1 of a given polynomial $f(x)$:

Choose an initial approximation x_0.

$i=1$: use Horner's algorithm to evaluate $\{b_n, b_{n-1}, \ldots, b_0\}$ and $\{c_n, c_{n-1}, \ldots, c_1\}$,
so that $b_0 = f(x_0)$ and $c_1 = f'(x_0)$;
compute $x_1 \leftarrow x_0 - b_0 / c_1$. (Newton's method)

$i=2$: use Horner's algorithm to evaluate $\{b_n, b_{n-1}, \ldots, b_0\}$ and $\{c_n, c_{n-1}, \ldots c_1\}$,
so that $b_0 = f(x_1)$ and $c_1 = f'(x_1)$;
compute $x_2 \leftarrow x_1 - b_0 / c_1$. (Newton's method)

$\vdots$

$i = N$: use Horner's algorithm to evaluate $\{b_n, b_{n-1}, \ldots, b_0\}$ and $\{c_n, c_{n-1}, \ldots, c_1\}$,
so that $b_0 = f(x_{N-1})$ and $c_1 = f'(x_{N-1})$;
compute $x_N \leftarrow x_{N-1} - b_0 / c_1$. (Newton's method)

Assuming that the procedure has converged, that is, $\left|1 - \frac{x_{N-1}}{x_N}\right| < \varepsilon$, then x_N is taken as the computed approximation to a zero of $f(x)$, and the associated deflated polynomial is $Q(x) = b_1 + b_2 x + b_3 x^2 + \cdots + b_n x^{n-1}$, where the coefficients $\{b_i\}$ are those from the step $i = N$ above.

Example. An illustration of the application of Newton's method and Horner's algorithm to compute a zero of a polynomial $f(x) = x^4 - 0.2x^3 + 1.8x^2 - 0.6x - 3.6$ . .

With $x_0 = 2$, Horner's algorithm gives

$$\begin{aligned} b_4 &= 1 & c_4 &= 1 \\ b_3 &= 1.8 & c_3 &= 3.8 \\ b_2 &= 5.4 & c_2 &= 13 \\ b_1 &= 10.2 & c_1 &= 36.2 \\ b_0 &= 16.8 \end{aligned}$$

and Newton's method gives $x_1 = x_0 - \dfrac{f(x_0)}{f'(p_0)} = x_0 - \dfrac{b_0}{c_1} = 1.535912$

With $x_1 = 1.535912$, Horner's algorithm gives

$$\begin{aligned} b_4 &= 1 & c_4 &= 1 \\ b_3 &= 1.335912 & c_3 &= 2.871823 \\ b_2 &= 3.851842 & c_2 &= 8.262709 \\ b_1 &= 5.316089 & c_1 &= 18.006879 \\ b_0 &= 4.565043 \end{aligned}$$

and Newton's method gives $x_2 = x_1 - \dfrac{f(x_1)}{f'(x_1)} = x_1 - \dfrac{b_0}{c_1} = 1.282395$

With $x_2 = 1.282395$, Horner's algorithm gives

$$\begin{aligned} b_4 &= 1 & c_4 &= 1 \\ b_3 &= 1.082395 & c_3 &= 2.364790 \\ b_2 &= 3.188058 & c_2 &= 6.220653 \\ b_1 &= 3.488350 & c_1 &= 11.465684 \\ b_0 &= 0.873442 \end{aligned}$$

and Newton's method gives $x_3 = x_2 - \dfrac{f(x_2)}{f'(x_2)} = x_2 - \dfrac{b_0}{c_1} = 1.206216$

With $x_3 = 1.206216$, Horner's algorithm gives

$$
\begin{array}{ll}
b_4 = 1 & c_4 = 1 \\
b_3 = 1.006216 & c_3 = 2.212432 \\
b_2 = 3.013714 & c_2 = 5.682386 \\
b_1 = 3.035191 & c_1 = 9.889377 \\
b_0 = 0.0610965 &
\end{array}
$$

and Newton's method gives $x_4 = x_3 - \dfrac{f(x_3)}{f'(x_3)} = x_3 - \dfrac{b_0}{c_1} = 1.200038$

With $x_4 = 1.200038$, Horner's algorithm gives

$$
\begin{array}{ll}
b_4 = 1 & c_4 = 1 \\
b_3 = 1.000038 & c_3 = 2.200076 \\
b_2 = 3.000084 & c_2 = 5.640260 \\
b_1 = 3.000215 & c_1 = 9.768742 \\
b_0 = 0.000373183 &
\end{array}
$$

and Newton's method gives $x_5 = x_4 - \dfrac{f(x_4)}{f'(x_4)} = x_4 - \dfrac{b_0}{c_1} = 1.2000000015$

If the computations were terminated at this point, x_5 would be the computed approximation to the root $r_1 = 1.2$, and the corresponding **approximate deflated polynomial** would be

$$
\begin{aligned}
Q(x) &= b_1 + b_2 x + b_3 x^2 + b_4 x^3 \\
&= 3.000215 + 3.000084x + 1.000038x^2 + x^3
\end{aligned}
$$

Note that the exact deflated polynomial is $3 + 3x + x^2 + x^3$. Note also that the sequence of values $\{b_0\}$ converges to 0 as $\{x_i\}$ converges to a root r_1.

****

Note: if several zeros of $f(x)$ are approximated as above, and several deflations are carried out giving a sequence of deflated polynomials of degrees $n-1, n-2, n-3, \ldots$, then the successive computed zeros tend to become less and less accurate.

For example, consider from the above example, the computed approximation

$$\begin{aligned} Q(x) &= b_1 + b_2 x + b_3 x^2 + b_4 x^3 \\ &= 3.000215 + 3.000084x + 1.000038x^2 + x^3 \end{aligned}$$

to the deflated polynomial. The exact roots (to 8 significant digits) of $Q(x)$ are

$$-1.0000422 \text{ and } 0.21245115 \times 10^{-5} \pm 1.7320763i,$$

whereas the corresponding exact roots of $f(x)$ are

$$-1 \text{ and } \pm\sqrt{3}\,i \approx \pm 1.7320508i.$$

If a zero of $Q(x)$ is computed using Newton's method and Horner's algorithm, it will not give a very accurate approximation to a zero of $f(x)$ (because $Q(x)$ is not the exact deflated polynomial and because of truncation/roundoff errors in computing this zero). In order to improve its accuracy, use the technique of "**root polishing**"; this is mentioned on page 182 of the 7th ed. of the textbook (page 180 of the 6th ed.).

Root Polishing

Apply Newton's method to the approximate deflated polynomial $Q(x)$, giving a value $\hat{r}$.

The value $\hat{r}$ approximates some root r_2 of $f(x)$, but will not be fully accurate.

Use $\hat{r}$ as the initial approximation for Newton's method applied to $f(x)$. This will converge very quickly (1 or 2 iterations) to the fully accurate root r_2 (as $\hat{r}$ is very close to r_2).