Computer Science 70: Discrete Mathematics

The Berlekamp-Welch Algorithm

In this note we give a simple example of the Berlekamp-Welch algorithm with $n = 3$ and $k = 1$.

Suppose we want to send the packets "1," "3," and "7." Then we interpolate to find the polynomial

$$P(X) = X^2 + X + 1,$$

which is the unique polynomial of degree 2 such that $P(0) = 1$, $P(1) = 3$, and $P(2) = 7$.

Now we transmit the $n + 2k = 5$ messages $P(0) = 1$, $P(1) = 3$, $P(2) = 7$, $P(3) = 13$, and $P(4) = 21$. Suppose $P(1)$ is corrupted, so the receiver receives 0 instead of 3 in that packet.

Let $E(X) = X - e$ be the *error-locator* polynomial—we don't know what e is yet—and let $R(X)$ be the polynomial whose values at $0, \ldots, 4$ are precisely the values we received over the channel. Then clearly

$$P(X)E(X) = R(X)E(X)$$

for $X = 0, 1, \ldots, 4$ (if the corruption occured at position i, then $E(i) = 0$, so equality trivially holds, and otherwise $P(i) = R(i)$). We don't know what P is (though we do know it is a degree 2 polynomial) and we don't know what E is either, but using the relationship above we can obtain a linear system whose solutions will be the coefficients of P and E.

Let

$$Q(X) = aX^3 + bX^2 + cX + d = P(X)E(X),$$

where a, b, c, d are unknown coefficients (which we will soon try to determine), so

$$aX^3 + bX^2 + cX + d = R(X)E(X) = R(X)(X - e),$$

which we can rewrite as

$$aX^3 + bX^2 + cX + d + R(X)e = R(X)X.$$

Now we substitute $X = 0$, $X = 1$, $\ldots$, $X = 4$ to get five linear equations (recall that $R(i)$ is the value we received for the fifth packet):

$$\begin{aligned} d + e &= 1 \\ a + b + c + d &= 0 \\ 8a + 4b + 2c + d + 7c &= 14 \\ 27a + 9b + 3c + d + 13e &= 39 \\ 64a + 16b + 4c + d + 21e &= 84. \end{aligned}$$

We then solve this linear system for a, b, c, d, e, and this gives us the polynomials $Q(X)$ and $E(X)$. We can then find $P(X)$ by computing the quotient $Q(X)/E(X)$, and from P we can obviously recover the original (uncorrupted) values.