

Polynomial GCDs by Linear Algebra

Barry Dayton

Northeastern Illinois University

www.neiu.edu/~bhdayton

March 2004

This is a brief exposition on how to find the greatest common divisor of two univariate polynomials using linear algebra rather than the Euclidean Algorithm. This is based on the work of my colleague Z. Zeng. Here coefficients will be from the field \mathbb{F} which will typically be the rationals, \mathbb{Q} , the reals, \mathbb{R} , or possibly the complex numbers, \mathbb{C} .

Convolution Matrices The set $\mathbf{P}_n = \{a_0 + a_1x + a_2x^2 + \cdots + a_nx^n \mid a_j \in \mathbb{F}\}$ of polynomials of degree n or less forms a vector space. For a given polynomial $p(x) = p_0 + p_1x + \cdots + p_mx^m$ of degree m , or less, multiplication of polynomials in \mathbf{P}_n by $p(x)$ gives a linear transformation $\mathbf{P}_n \rightarrow \mathbf{P}_{m+n}$ because of the distributive law $p(f + g) = pf + pg$ and the commutative law $p(rf) = r(pf)$ for constants r .

If we take as an ordered basis for \mathbf{P}_n the list $1, x, x^2, \dots, x^n$, and likewise for \mathbf{P}_{m+n} the *convolution matrix* $C_n(p)$ is the $m + n + 1 \times n + 1$ matrix of the transformation $f \mapsto pf$ with respect to these bases. For example, if $p(x) = p_0 + p_1x + p_2x^2$ and $n = 3$ then $C_n(p)$ is given by

$$C_3(p) = \begin{bmatrix} p_0 & 0 & 0 & 0 \\ p_1 & p_0 & 0 & 0 \\ p_2 & p_1 & p_0 & 0 \\ 0 & p_2 & p_1 & p_0 \\ 0 & 0 & p_2 & p_1 \\ 0 & 0 & 0 & p_2 \end{bmatrix}$$

Note that the rows correspond to the monomials $1, x, x^2, x^3, x^4, x^5$ respectively and the first column thus represents p , the second xp , the third x^2p and the fourth x^3p . If $f = a_0 + a_1x + a_2x^2 + a_3x^3$ then multiplying pf is just adding $a_0p + a_1xp + a_2x^2p + a_3x^3p$ which means taking the appropriate linear combination of columns, that is forming the matrix product

$$\begin{bmatrix} p_0 & 0 & 0 & 0 \\ p_1 & p_0 & 0 & 0 \\ p_2 & p_1 & p_0 & 0 \\ 0 & p_2 & p_1 & p_0 \\ 0 & 0 & p_2 & p_1 \\ 0 & 0 & 0 & p_2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} p_0a_0 \\ p_1a_0 + p_0a_1 \\ p_2a_0 + p_1a_1 + p_0a_2 \\ p_2a_1 + p_1a_2 + p_0a_3 \\ p_2a_2 + p_1a_3 \\ p_2a_3 \end{bmatrix}$$

the result of which is the vector $\text{Coeff}(f, m + n)$ of the coefficients of pf with respect to the basis of \mathbf{P}_{m+n} .

A variation on this is the *division matrix* where, in the example above, we augment with two additional columns at the front with just a 1 in the diagonal spot. This makes a square upper triangular matrix which will be non-singular assuming $p_2 \neq 0$.

$$\text{Div}_3(p) = \begin{bmatrix} 1 & 0 & p_0 & 0 & 0 & 0 \\ 0 & 1 & p_1 & p_0 & 0 & 0 \\ 0 & 0 & p_2 & p_1 & p_0 & 0 \\ 0 & 0 & 0 & p_2 & p_1 & p_0 \\ 0 & 0 & 0 & 0 & p_2 & p_1 \\ 0 & 0 & 0 & 0 & 0 & p_2 \end{bmatrix}$$

We see that

$$\text{Div}_3(p) \begin{bmatrix} r_0 \\ r_1 \\ a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \text{Coeff}(r_0 + r_1x, m + n) + \text{Coeff}(pf, m + n)$$

If, instead, we replace the right hand side by $\text{Coeff}(g, m + n)$ where $g \in \mathbf{P}_{m+n}$ and solve for the vector on the left we see that we have simply performed the division algorithm to write the 5th degree polynomial g as $g = pf + r$ for some polynomial f of degree 3 and remainder r of degree 1 or less.

Generalizing this actually gives a proof of the division algorithm, since $\text{Div}_n(p)$ is non-singular if p has exactly degree m and hence there is a unique solution. Moreover, this triangular system is easy to solve by substitution, the work being equivalent to the usual division algorithm but more user friendly to a computer. It should be remarked that if one is working over \mathbb{R} or \mathbb{C} and one knows that the remainder should be zero then one can use the convolution matrix $C_n(p)$ and solve as a least square problem for a very accurate approximate solution. This was one of Zeng's motivations for replacing the division algorithm with this matrix formulation.

The Sylvester Matrix and the Resultant In a certain sense the use of the convolution matrix idea to study the GCD goes back at least a century, but because of difficulty in actually performing linear algebra computations without a computer, mathematicians only went half way, to decide if $\text{gcd}(f, g) = 1$ without actually calculating the GCD.

The idea is now simple for us: given a polynomial f of degree exactly m and a polynomial g of degree exactly n one forms a square $m + n \times m + n$ matrix $\text{Syl}(f, g) = [C_{n-1}(f) \mid C_{m-1}(g)]$ formed by adjoining the two convolution matrices $C_{n-1}(f)$ and $C_{m-1}(g)$. If we form a vector by putting a coefficient vector $\text{Coeff}(v, n - 1)$ of a, possibly unknown, polynomial v of degree $n - 1$ on top of a vector $\text{Coeff}(w, m - 1)$ of coefficients of a, also possibly unknown, polynomial w of degree $n - 1$ then multiplying by the Sylvester matrix gives the coefficient vector of the sum $fv + gw$.

If $\text{Syl}(f, g)$ is non-singular then this sum $fv + gw$ can be any $m + n - 1$ degree vector we want, choosing v, w correctly, in particular we can get the constant polynomial 1 so $\gcd(f, g) = 1$. Conversely if $\text{Syl}(f, g)$ is singular we can find polynomials v, w of degree less than or equal to $n - 1, m - 1$ respectively, not both zero, with $fv + gw = 0$. This gives $fv = -gw$ (so in fact neither v or w is zero) and hence, g divides fv . If it were true that $\gcd(f, g) = 1$ then it would follow that g divides v , but g has higher degree than v so this is impossible. Hence $\gcd(f, g) \neq 1$.

In the pre-computer days mathematicians used the determinant to decide if a square matrix was non-singular so they called the determinant of the Sylvester matrix (or actually the transpose of the Sylvester matrix, but this is the same number) the *resultant*, $R(f, g)$, of f and g . The theorem was

Resultant Theorem $\gcd(f, g) = 1$ if and only if $R(f, g) \neq 0$.

Example: Let $f(x) = x^3 + 2x^2 - 4x + 1, g(x) = x^2 - 3x + 2$. Then our Sylvester determinant (the transpose of the classical one) is

$$R(f, g) = \begin{vmatrix} 1 & 0 & 2 & 0 & 0 \\ -4 & 1 & -3 & 2 & 0 \\ 2 & -4 & 1 & -3 & 2 \\ 1 & 2 & 0 & 1 & -3 \\ 0 & 1 & 0 & 0 & 1 \end{vmatrix}$$

It is not hard to see that this determinant is zero so $\gcd(f, g) \neq 1$. The fact that we are calculating a determinant obscures the fact that the work we did is almost enough to actually find the GCD.

Zeng's Calculation Several modern methods for deciding if a matrix is singular or not actually give us, or at least make it easy to find, vectors in the nullspace if the matrix is singular. By organizing the work slightly differently Z. Zeng gives an efficient method for finding the GCD, whether or not it is 1.

First, for calculating convolution matrices, we can assume f, g lie in the same \mathbf{P}_m , that is pick $m = \max(\text{degree}(f), \text{degree}(g))$. Then define *Sylvester subresultant matrices* by $S_j(f, g) = [C_j(f)|C_j(g)]$ for $j = 0, ..m - 1$. That is adjoin two convolution matrices side by side. Note that $S_j(f, g)$ is an $m + j + 1 \times 2(j + 1)$ matrix.

The game is to check these matrices in order, $S_0(f, g), S_1(f, g), \dots$ to see if they are of full rank $2j + 2$. The first matrix to fail can be shown to be rank deficient by just 1, that is, of rank $2j + 1$. Thus the nullspace is of dimension 1, pick any convenient non-zero vector. Split this in half, the top $j + 1$ coordinates being the coefficient vector of a polynomial w of degree no more than j and the remaining coordinates being the coefficient vector of a polynomial v . It follows that

- $fv = -gw$
- $\gcd(v, w) = 1$

The first follows simply by definition of nullspace, the second because if $\gcd(v, w) \neq 1$ then dividing this GCD out would give v, w of smaller degree making the first equation true which would have made an earlier $S_j(f, g)$ rank deficient.

But the two properties above imply that v divides f and w divides g , moreover, dividing both sides by $-vw$ shows the quotients are equal, call this equal quotient u . u must be the GCD, it is certainly a common divisor, and a larger common divisor would again contradict the minimality of degrees of v, w . Using matrices, u can be found using linear algebra by the division matrices above.

If no such rank deficient $S_j(f, g)$ is found by the time one gets to $S_{m-1}(f, g)$ then, since $S_{m-1}(f, g)$ is square it is essentially the classical Sylvester matrix and the arguments above show $\gcd(f, g) = 1$.

Example We consider the same example as above: $f(x) = x^3 + 2x^2 - 4x + 1$ and $g(x) = x^2 - 3x + 2$. A trick is to shuffle the columns, that is to alternate columns from f and g rather than put all of those from f first. We then have, writing, for example, $S_0(f, g) = S_0$,

$$S_0 = \begin{bmatrix} 1 & 2 \\ -4 & -3 \\ 2 & 1 \\ 1 & 0 \end{bmatrix}, S_1 = \begin{bmatrix} 1 & 2 & 0 & 0 \\ -4 & -3 & 1 & 2 \\ 2 & 1 & -4 & -3 \\ 1 & 0 & 2 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, S_2 = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 & 0 \\ -4 & -3 & 1 & 2 & 0 & 0 \\ 2 & 1 & -4 & -3 & 1 & 2 \\ 1 & 0 & 2 & 1 & -4 & -3 \\ 0 & 0 & 1 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Using a computer we can just check the ranks of the matrices in order. By hand we could use Gaussian elimination row operations directly on S_2 relying on the band matrix structure to avoid doing actual work on the later columns unless we need to. We are looking for the first column which is a linear combination of the earlier ones. If it is the second we note that we already had this information in S_0 , the third or 4th was already in S_1 . In this particular example we were somewhat unlucky as it is the last column. Then completing Gauss-Jordan we can find a vector $[2, -1, -1, 3, 0, 1]^T$ in the nullspace. Since we shuffled the columns the coefficient vector for w is $[2, -1, 0]^T$ while that for v is $[-1, 3, 1]$ so $-w = 2 - x$ and $v = -1 + 3x + x^2$. Dividing by long division or division matrices gives $\gcd(f, g) = u = x - 1$.

The Extended GCD. One advantage in this linear algebra approach is in finding the extended GCD, that is find polynomials $a(x), b(x)$ so that $\gcd(f, g) = a(x)f(x) + b(x)g(x)$. This is not pleasant using the Euclidean Algorithm. Try it with the example above.

But using the linear algebra method it is fairly easy. The polynomials v, w obtained by the method were seen to have $\gcd(v, w) = 1$. Thus one can find the coefficients of v, w simply by using the

classical Sylvester matrix and solving

$$\text{Syl}(f, g) \begin{bmatrix} a_0 \\ \vdots \\ a_n \\ b_0 \\ \vdots \\ b_m \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

But multiplying $av + bw = 1$ by u gives $af + bg = u$ as desired.

Example In the example above we saw $v := -1 + 3x + x^2$ and $w = x - 2$. So we need only solve the 3×3 system

$$\begin{bmatrix} -1 & -2 & 0 \\ 3 & 1 & -2 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

getting $a = 1/9, b = -5/9 - 1/9x$ and hence

$$\frac{1}{9}f - \left(\frac{5}{9} + \frac{1}{9}x\right)g = x - 1$$

A final Word of Warning! When using linear algebra it is tempting to use decimal approximations of numbers rather than exact fractions. Or you may wish to use a more sophisticated rank finding method such as singular values. This latter method is the one used by Z. Zeng in his work. But the result is then the *approximate GCD* which may not be the same as the exact GCD. One must check to see if the computed GCD has the right kind of coefficients. For example, the fact the GCD can be found using the Euclidean Algorithm or the method above using exact arithmetic says that the GCD of polynomials with rational (fraction) coefficients also has rational coefficients. It can be shown using number theory that the GCD of integer polynomials will always be an integer polynomial, even if one is working in a larger coefficient field. The following example shows what can happen.

Example Let

$$\begin{aligned} f &= 225075x^5 - 20295 \\ g &= 88555x^4 - 12920 \end{aligned}$$

Applying Maple's GCD algorithm `gcd(f, g)` one gets the constant polynomial 5, however using the command `gcd(evalf(f), evalf(g))`, which does the same problem using floating point numbers, the answer is $x - 0.618034$. But the number 0.618034 is a known approximation to a very irrational number, the reciprocal of the golden mean. So no multiple of this polynomial could be an integer polynomial. Both answers are correct, however. The first answer is the exact GCD as used in algebra and number theory, the second is the approximate GCD which might be used to show that f, g have a common approximate real root of 0.618034.