

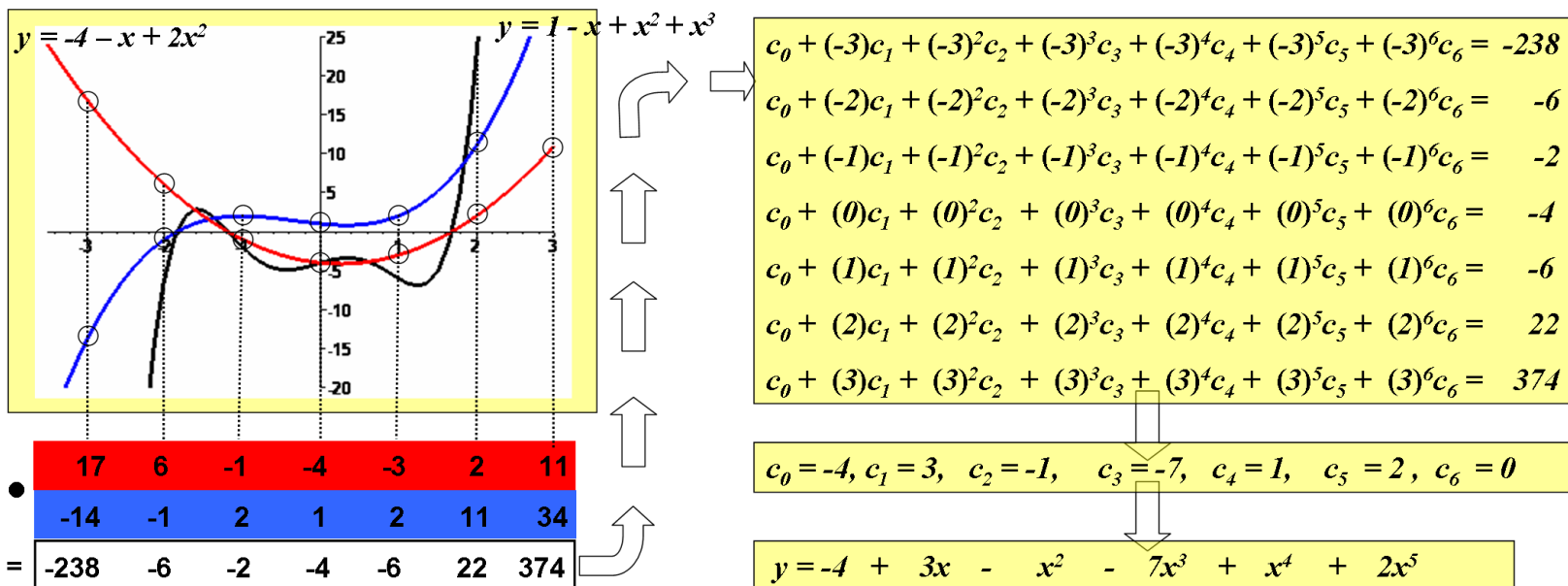


THEOREM OF THE DAY

The Convolution Theorem Let $a = (a_0, \dots, a_n)$ and $b = (b_0, \dots, b_n)$ be vectors in \mathbb{C}^{n+1} . The convolution of a and b , denoted $a \star b$, is the vector $c = (c_0, \dots, c_{2n})$, in \mathbb{C}^{2n+1} , defined by $c_i = \sum_{j=0}^n a_j b_{i-j}$, $i = 0, \dots, 2n$, with $b_k = 0$ whenever $k < 0$ or $k > n$. Then

$$a \star b = \mathcal{F}^{-1}(\mathcal{F}(a) \bullet \mathcal{F}(b)),$$

where \mathcal{F} is the $(2n+1)$ -dimensional Discrete Fourier Transform and \bullet is componentwise multiplication.



The picture shows a version of the Convolution Theorem for polynomials: $a \star b = \text{interpolate}(\text{evaluate}(a) \bullet \text{evaluate}(b))$. ‘Evaluate’ means evaluate a degree n polynomial at $2n+1$ points; the Discrete Fourier Transform (DFT) corresponds to a particular choice of points. Here we have merely chosen $2n+1 = 7$ integers at which the two polynomials $a : -4 - x + 2x^2$ and $b : 1 - x + x^2 + x^3$ are evaluated (polynomial a is taken to have degree 3 but with $a_3 = 0$). These values are multiplied and interpolation is applied: the 7 component products define a system of 7 simultaneous equations whose unique solution is precisely the list of coefficients of the product of the two polynomials (the coefficient of x^6 is zero because polynomial a had degree < 3). For polynomials, this is precisely what convolution achieves.

The DFT evaluates instead at complex numbers: the $2n+1$ powers of a primitive $(2n+1)$ -th root of unity. Via Euler’s Identity, these are expressed as powers of $e^{\tau i/(2n+1)}$ ($\tau = 2\pi$). In general, to apply the DFT to a vector v of length N , we multiply v by the $N \times N$ matrix \mathcal{F} whose ij -th element is $\omega^{(i-1)(j-1) \pmod{N}}$, $\omega = e^{\tau i/N}$. (To our polynomial coefficient vectors we first need to append n zeros to extend to length $2n+1$.) This special choice of points leads to a dramatic computational short-cut: the so-called *Fast Fourier Transform* (FFT) achieves convolution (and hence polynomial multiplication) in time $\mathcal{O}(N \log N)$ instead of $\mathcal{O}(N^2)$.

This discrete convolution theorem is intimately connected with the FFT known, in some form, to Gauss, as early as 1805; rediscovered by Cornelius Lanczos in 1940; and made widely known by James Cooley and John Tukey, 1965.

Web link: www-inst.eecs.berkeley.edu/~cs191/fa07/lectures/QFT_fa03.pdf

Further reading: *The Design and Analysis of Computer Algorithms* by A. Aho, J.E. Hopcroft and J.Ullman, Addison-Wesley, 1974, chapter 7.

