

How to solve a Cubic Equation Part 3 – General Depression and a New Covariant

James F. Blinn
Microsoft Research
blinn@microsoft.com

Originally published in
IEEE Computer Graphics and Applications
Nov/Dec 2006, pages 92–102

TV is good for you. That's one of the conclusions of this month's book recommendation: *Everything Bad Is Good for You: How Today's Popular Culture Is Actually Making Us Smarter* by Steven Johnson. The basic premise is that television series (his primary example) have gotten so complex in their speed of exposition and involved interpersonal relationships that it really exercises your brain just to follow them. And each episode is not an individual story any more; a given program is one continuing mini-series that has to be seen in order without gaps for it to make sense. For example I've been totally sucked into the series *Lost*, and the whole fun of the series is watching the mystery unfold by seeing the episodes in sequence. But I missed the first few episodes of the second season, so I didn't dare look at any of the later episodes. I've been forced to wait for the second season to come out on DVD and not look at any fan sites until then.

Given that TV shows are now all miniseries I don't feel so bad about the fact that this column has evolved into various multi-part series. I had eight articles on 3D lines, two articles on solving quadratic equations, and this is the third one on cubic equations. (It would be nice to make it three articles on cubics, and four on quartics, but there is more than three articles worth say on cubics so it will probably be four). I do try to make each article have a theme so they are roughly self-contained, but I still must start out with the same phrase as all the serialized TV shows ...

Previously on Jim Blinn's Corner...

We are looking for the roots $[x, w]$ of the equation

$$f(x, w) = Ax^3 + 3Bx^2w + 3Cwx^2 + Dw^3 = 0$$

The Hessian matrix is a useful intermediate result and is calculated as

$$\begin{aligned}\delta_1 &= AC - B^2 \\ \delta_2 &= AD - BC \\ \delta_3 &= BD - C^2\end{aligned}\tag{0.1}$$
$$\mathbf{H} = \begin{bmatrix} 2\delta_1 & \delta_2 \\ \delta_2 & 2\delta_3 \end{bmatrix}$$

The discriminant of the cubic is the determinant of this matrix

$$\begin{aligned}\Delta &= \det \mathbf{H} \\ &= 4\delta_1\delta_3 - \delta_2^2 \\ &= -A^2D^2 + 6ABCD - 4AC^3 - 4B^3D + 3B^2C^2\end{aligned}\tag{0.2}$$

How to Solve a Cubic Equation – Part 3

If $\Delta > 0$ there are three real roots, if $\Delta = 0$ there is a double or triple root, and if $\Delta < 0$ there is a single real root.

The first step in solving the cubic is to “depress” it by transforming in parameter space to get a cubic with a second coefficient of zero. We transform by the substitution:

$$\begin{bmatrix} x & w \end{bmatrix} = \begin{bmatrix} \tilde{x} & \tilde{w} \end{bmatrix} \begin{bmatrix} t & u \\ s & v \end{bmatrix} \quad (0.3)$$

This gives a transformation in coefficient space of

$$\begin{bmatrix} \tilde{A} \\ \tilde{B} \\ \tilde{C} \\ \tilde{D} \end{bmatrix} = \begin{bmatrix} t^3 & 3t^2u & 3tu^2 & u^3 \\ t^2s & 2tus + t^2v & u^2s + 2tuv & u^2v \\ ts^2 & us^2 + 2tsv & 2usv + tv^2 & uv^2 \\ s^3 & 3s^2v & 3sv^2 & v^3 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} \quad (0.4)$$

We found two transformation matrices that result in a cubic with $\tilde{B} = 0$:

$$\begin{bmatrix} t & u \\ s & v \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -B & A \end{bmatrix} \text{ or } \begin{bmatrix} 0 & 1 \\ -D & C \end{bmatrix} \quad (0.5)$$

The two possibilities generate two variants of the algorithm (which I called A and D after the determinant of the transformation.) Each of these generate a depressed polynomial (though some authors use the more prosaic term “reduced polynomial”) with

$$\begin{bmatrix} \tilde{A} \\ \tilde{B} \\ \tilde{C} \\ \tilde{D} \end{bmatrix} = \begin{bmatrix} A \\ 0 \\ A(AC - B^2) \\ A(2B^3 - 3ABC + A^2D) \end{bmatrix} \text{ or } \begin{bmatrix} D \\ 0 \\ D(BD - C^2) \\ D(-D^2A + 3DCB - 2C^3) \end{bmatrix} \quad (0.6)$$

Each of these has a common homogeneous factor which we can toss out giving

$$\begin{bmatrix} \bar{A} \\ \bar{B} \\ \bar{C} \\ \bar{D} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ AC - B^2 \\ 2B^3 - 3ABC + A^2D \end{bmatrix} \text{ or } \begin{bmatrix} 1 \\ 0 \\ BD - C^2 \\ -D^2A + 3DCB - 2C^3 \end{bmatrix}$$

So both generate a canonical polynomial to solve:

$$\tilde{x}^3 + 3\bar{C}\tilde{x} + \bar{D} = 0$$

Solve this (see [1] for details) and transform by the original matrix to get the answer

$$\begin{bmatrix} x & w \end{bmatrix} = \begin{bmatrix} \tilde{x} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -B & A \end{bmatrix} \text{ or } \begin{bmatrix} \tilde{x} & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -D & C \end{bmatrix}$$

Both these variants work fine algebraically, but each is numerically good for regions in parameter space that is different from the other. So far, I have only finished the calculations for the case $\Delta < 0$ where there is just one real root, a , and a complex conjugate pair $(b \pm ic)$. We found that the best (so far) numerical solution to be:

if $B^3D \geq AC^3$ {equivalent to} $a^2 > b^2 + c^2$
 use algorithm A to calculate a ,
 use algorithm D to calculate b and c
 otherwise
 use algorithm D to calculate a ,
 use algorithm A to calculate b and c .

Generalizing the depressing transformation

Given the two choices of depressing transformations, Equation (0.5), it is natural to ask if there are any others. Of course there are. So let's find all possible transformations (values of s, t, u, v) that make $\tilde{B} = 0$. From Equation (0.4) we have

$$\tilde{B} = t^2sA + (2tus + t^2v)B + (u^2s + 2tuv)C + u^2vD \quad (0.7)$$

Proceeding as we did with quadratic polynomials we will parameterize the desired set of transformations in terms of the first row $[t, u]$. So given t, u we find the proper s, v that makes $\tilde{B} = 0$ by rewriting equation (0.7) as

$$\begin{aligned} \tilde{B} &= s(t^2A + 2tuB + u^2C) \\ &+ v(t^2B + 2tuC + u^2D) = 0 \end{aligned}$$

An s, v that satisfies this is :

$$\begin{aligned} s &= -(t^2B + 2tuC + u^2D) \\ v &= +(t^2A + 2tuB + u^2C) \end{aligned} \quad (0.8)$$

Actually any nonzero multiple of these values will also work, but for now I'll stick with just the values in Equation (0.8).

The transformation this generates is singular if the following is zero.

$$\begin{aligned} tv - su &= t(t^2A + 2tuB + u^2C) - (t^2B + 2tuC + u^2D)u \\ &= t^3A + 3t^2uB + 3tu^2C + u^3D \\ &= f(t, u) \end{aligned}$$

In other words, this transformation will be singular if the (t, u) we chose was a root of the original cubic. This situation will turn out to be not all that bad, but I'll defer a complete discussion of this until Part 5.

Now that we've transformed to get $\tilde{B} = 0$ let's see what the other coefficients are as functions of (t, u) .

Evaluating \tilde{A}, \tilde{C} and \tilde{D}

We first expand out the matrix formulation of equation (0.4) to get the formulas for $\tilde{A}, \tilde{C}, \tilde{D}$.

$$\begin{aligned} \tilde{A} &= t^3A + 3t^2uB + 3tu^2C + u^3D \\ \tilde{C} &= ts^2A + (us^2 + 2tsv)B + (2usv + tv^2)C + uv^2D \\ \tilde{D} &= s^3A + 3s^2vB + 3sv^2C + v^3D \end{aligned} \quad (0.9)$$

We recognize \tilde{A} as just $f(t,u)$ which is also the determinant of the transformation. Thinking of \tilde{A} as now a function of (t,u) we have simply $\tilde{A}(t,u) = f(t,u)$. The other two values \tilde{C} and \tilde{D} can be had in terms of just t and u by simply plugging the values from equation (0.8) into equation (0.9). This will result in an expression for \tilde{C} that is fifth order in (t,u) and for \tilde{D} that is sixth order in (t,u) . Needless to say, doing this explicitly will turn into an algebraic quagmire. And this quagmire obscures the very nice algebraic fact that the expression for $\tilde{C}(t,u)$ and $\tilde{D}(t,u)$ both contain the polynomial $\tilde{A}(t,u)$ as a factor. We saw a special case of this in equation (0.6) of the original Algorithm A (where $[t,u]=[1,0]$) and Algorithm D (where $[t,u]=[0,1]$). To see the general case more clearly I will drag in my favorite algebraic simplification tool, the tensor diagram.

Tensor Diagram Essentials

I've described this notation scheme in various previous articles [2]. But there are a few new constructions that we will need here so let's start out with a brief review.

Polynomials as Tensors

We are interested in linear, quadratic and cubic homogeneous polynomials

$$l(x, w) = Ax + Bw$$

$$q(x, w) = Ax^2 + 2Bxw + Cw^2$$

$$f(x, w) = Ax^3 + 3Bx^2w + 3Cwx^2 + Dw^3$$

We can write these as vector-matrix products (though the cubic polynomial is a bit clunky in this notation).

$$l(x, w) = [x \quad w] \begin{bmatrix} A \\ B \end{bmatrix}$$

$$q(x, w) = [x \quad w] \begin{bmatrix} A & B \\ B & C \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

$$f(x, w) = [x \quad w] \left\{ \left[\begin{bmatrix} A & B \\ B & C \end{bmatrix} \quad \begin{bmatrix} B & C \\ C & D \end{bmatrix} \right] \begin{bmatrix} x \\ w \end{bmatrix} \right\} \begin{bmatrix} x \\ w \end{bmatrix}$$

The coefficients form tensors of rank 1 (a vector for the linear polynomial), rank 2 (a matrix for the quadratic) and rank 3 (for the cubic). Next, we will give names to the tensors and give alternate, indexed, names to their elements.

$$\mathbf{L} = \begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} L_0 \\ L_1 \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} A & B \\ B & C \end{bmatrix} = \begin{bmatrix} Q_{00} & Q_{01} \\ Q_{10} & Q_{11} \end{bmatrix}$$

$$\begin{aligned} \mathbf{C} &= \left[\begin{bmatrix} A & B \\ B & C \end{bmatrix} \quad \begin{bmatrix} B & C \\ C & D \end{bmatrix} \right] \\ &= \left[\begin{bmatrix} C_{000} & C_{010} \\ C_{100} & C_{110} \end{bmatrix} \quad \begin{bmatrix} C_{001} & C_{011} \\ C_{101} & C_{111} \end{bmatrix} \right] \end{aligned}$$

The vector \mathbf{p} will be dot-producted with these. As a bookkeeping convention we will write its elements' indices as superscripts.

How to Solve a Cubic Equation – Part 3

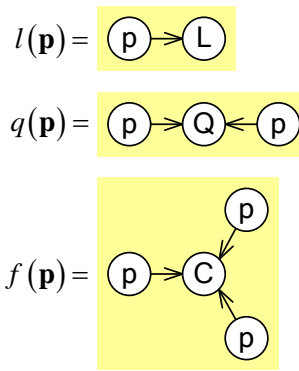
$$\mathbf{p} = [x \ w] = [p^0 \ p^1]$$

Then we can write the polynomial functions as nested sums.

$$\begin{aligned} l(\mathbf{p}) &= \sum_i p^i L_i \\ q(\mathbf{p}) &= \sum_{i,j} p^i p^j Q_{i,j} \\ f(\mathbf{p}) &= \sum_{i,j,k} p^i p^j p^k C_{ijk} \end{aligned} \tag{0.10}$$

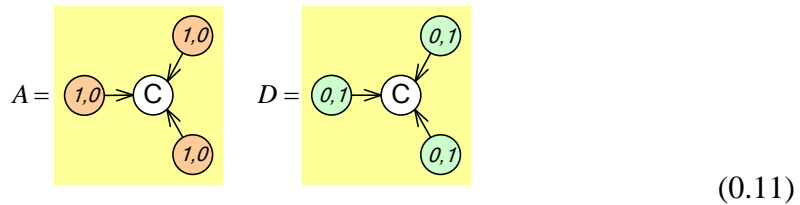
Note that I'm kinda running out of letters here so I've overloaded the notation a bit. I use boldface \mathbf{C} to represent the tensor and italic C with subscripts to represent its elements, and I use italic C without subscripts as one of the elements of \mathbf{C} . We will next dispense with explicit indices anyway so this is just temporary scaffolding.

To dispense with explicit indices we write equation (0.10) in diagram form as follows.

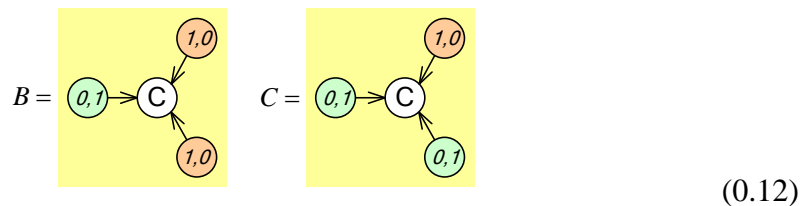


Each vector/matrix/tensor is a node. The number of connection points equals the number of indices. Each summed-over index is an arc connecting the two nodes containing that index. The arrow on the arc points from a connection point on a parameter vector \mathbf{p} (representing a superscript or “contravariant” index) to a connection point on a coefficient tensor (representing a subscripted or “covariant” index)

I want to point out something here that is pretty obvious, but that we will build on later. Given a tensor, we may extract out the coefficients by plugging in particular values for the parameter vector \mathbf{p} . For example we can plug in the parameter vectors $[1, 0]$ and $[0, 1]$ to get the A and D coefficients.



Slightly less obvious is the following. We can extract out the other coefficients by plugging different parameter vectors into the three inputs. I have colored the nodes here simply to emphasize the pattern.



Transformations

If we transform the parameter vector via a matrix

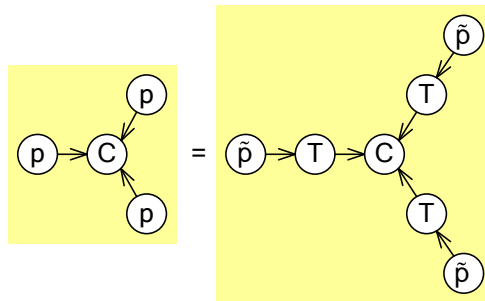
$$\begin{bmatrix} x & w \end{bmatrix} = \begin{bmatrix} \tilde{x} & \tilde{w} \end{bmatrix} \begin{bmatrix} t & u \\ s & v \end{bmatrix}$$

$$\mathbf{p} = \tilde{\mathbf{p}}\mathbf{T}$$

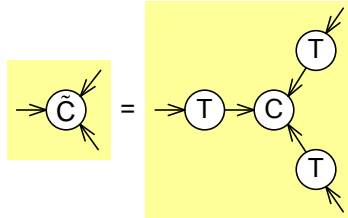
The diagram version is

$$\textcircled{\mathbf{p}} \rightarrow = \textcircled{\tilde{\mathbf{p}}} \rightarrow \textcircled{\mathbf{T}} \rightarrow$$

Note that the matrix \mathbf{T} has one arrow in and one out. This is called a mixed tensor in distinction to the pure covariant tensor \mathbf{Q} that represents a quadratic polynomial. To see the effect of \mathbf{T} on the tensor \mathbf{C} we form



So the transformed tensor $\tilde{\mathbf{C}}$ is just the inside portion



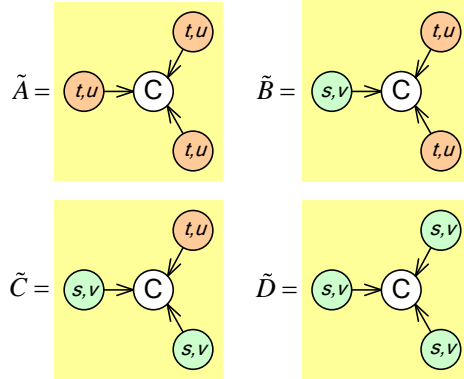
(0.13)

Given the tensor $\tilde{\mathbf{C}}$ we can extract the coefficients $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}$ by plugging in the parameter vectors $[1,0]$ and $[0,1]$ in the patterns from equation (0.11) and (0.12). Given the internal structure of $\tilde{\mathbf{C}}$ from equation (0.13) these will go straight into the transformation matrix. We observe that

$$\textcircled{1,0} \rightarrow \textcircled{\mathbf{T}} \rightarrow = \textcircled{t,u} \rightarrow$$

$$\textcircled{0,1} \rightarrow \textcircled{\mathbf{T}} \rightarrow = \textcircled{s,v} \rightarrow$$

This means that the coefficients $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}$ in terms of elements of the original tensor \mathbf{C} are:



(0.14)

This is the diagram version of equation (0.4) and is called blossoming.

The Epsilon Tensor

Two vectors $\mathbf{K} = [K_0 \ K_1]$ and $\mathbf{L} = [L_0 \ L_1]$ are homogeneously equivalent (differ by only a nonzero homogeneous scale factor) if the following is true

$$\frac{K_0}{L_0} = \frac{K_1}{L_1} \quad \text{or} \quad K_0 L_1 - K_1 L_0 = 0$$

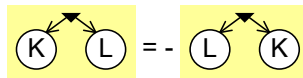
We can write this expression as a matrix product

$$K_0 L_1 - K_1 L_0 = [K_0 \ K_1] \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} L_0 \\ L_1 \end{bmatrix}$$

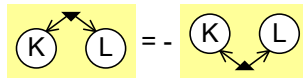
We give the name epsilon to the 2x2 anti-symmetric matrix. In diagram notation it would look like a node with two connecting arcs joined to the \mathbf{K} and \mathbf{L} nodes. But we have to be careful. Since epsilon is anti-symmetric reversing the positions of \mathbf{K} and \mathbf{L} will flip the sign:

$$[K_0 \ K_1] \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} L_0 \\ L_1 \end{bmatrix} = - [L_0 \ L_1] \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} K_0 \\ K_1 \end{bmatrix} \quad (0.15)$$

I want to make this visually apparent with the node diagram so I make the node shape itself asymmetric. The diagram version of equation (0.15) is



A rotation of a diagram or other connectivity-preserving rearrangement of its nodes doesn't affect its algebraic meaning. But a mirror reflection of an epsilon node will flip the sign.



It's important to note that, in this diagram, \mathbf{K} and \mathbf{L} don't have to be just single vectors. They can represent any other more complex collections of nodes and arcs; if you flip an epsilon anywhere inside a complex diagram you flip its algebraic sign.

The epsilon tensor is also good for representing determinants of matrices. Consider the following

$$\begin{matrix} \rightarrow & \text{Q} & \leftarrow \\ & \nearrow & \searrow \end{matrix} = \begin{bmatrix} A & B \\ B & C \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = \begin{bmatrix} -B & A \\ -C & B \end{bmatrix}$$

Now put two of these end to end

$$\begin{aligned}
 & \begin{array}{c} \text{---} \circlearrowleft \text{Q} \text{---} \circlearrowright \text{Q} \text{---} \\ \text{---} \circlearrowright \text{Q} \text{---} \circlearrowleft \text{Q} \text{---} \end{array} \\
 & = \begin{bmatrix} -B & A \\ -C & B \end{bmatrix} \begin{bmatrix} -B & A \\ -C & B \end{bmatrix} = \begin{bmatrix} B^2 - AC & 0 \\ 0 & B^2 - AC \end{bmatrix}
 \end{aligned} \tag{0.16}$$

This is (minus) the determinant of \mathbf{Q} , times an identity matrix. Diagrammatically:

$$\begin{array}{c} \text{---} \circlearrowleft \text{Q} \text{---} \circlearrowright \text{Q} \text{---} \\ \text{---} \circlearrowright \text{Q} \text{---} \circlearrowleft \text{Q} \text{---} \end{array} = -\det \mathbf{Q} \text{---} \tag{0.17}$$

You can write the determinant as a simple scalar by connecting the dangling arcs, which is the same as taking the trace of the matrix.

$$\begin{array}{c} \circlearrowleft \text{Q} \\ \circlearrowright \text{Q} \end{array} = -2 \det \mathbf{Q} \tag{0.18}$$

The factor of 2 on the right hand side comes from taking the trace of the right hand side of equation (0.16). Another way to look at this is to take the trace of the version in equation (0.17) and note that a single arc represents a 2x2 identity matrix whose trace is 2.

Why Tensor Diagrams are Cool

There are several things I like about tensor diagram notation. First of all, they are a good visualization tool since they represent certain algebraic properties visually: Symmetry of the tensors is reflected in the roundness of the nodes, so that exchanging any two input arcs leaves the algebraic values unchanged. And anti-symmetry of the epsilon is reflected in the shape of its node. But there are two main things that are particularly nice: tensor diagrams automatically represent transformationally invariant quantities, and they allow easy algebraic manipulations. Let's look at the first of these.

Invariants

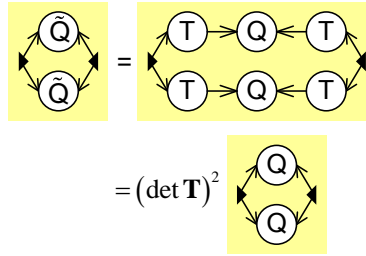
Consider the following construction

$$\begin{aligned}
 & \begin{array}{c} \text{---} \circlearrowleft \text{T} \text{---} \circlearrowright \text{T} \text{---} \\ \text{---} \circlearrowright \text{T} \text{---} \circlearrowleft \text{T} \text{---} \end{array} \\
 & = \begin{bmatrix} t & s \\ u & v \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} t & u \\ s & v \end{bmatrix} = \begin{bmatrix} 0 & tv - su \\ -tv + su & 0 \end{bmatrix}
 \end{aligned}$$

The right hand side is just the determinant of \mathbf{T} times the epsilon matrix. That is:

$$\begin{array}{c} \text{---} \circlearrowleft \text{T} \text{---} \circlearrowright \text{T} \text{---} \\ \text{---} \circlearrowright \text{T} \text{---} \circlearrowleft \text{T} \text{---} \end{array} = \det \mathbf{T} \begin{array}{c} \circlearrowleft \\ \circlearrowright \end{array} \tag{0.19}$$

Now look at what happens when we form the determinant of a transformed matrix \mathbf{Q}

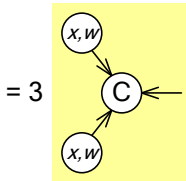


Each pair of **T**'s on either side of an epsilon “extracts out” into a factor of $\det \mathbf{T}$ for the resulting expression. Since there are two such factors here, the sign of the determinant of **Q** is unchanged by any parameter transformation. The mere writing of an expression as a tensor expression (in either diagram or index form) constitutes a proof that it represents a transformationally invariant quantity. Each epsilon in the diagram will turn into a factor of $\det \mathbf{T}$ when a transformation is applied. So if the diagram contains an even number of epsilons its sign is invariant under transformation. If it contains an odd number just the zeroness is invariant. The number of epsilon nodes in a diagram is called the *weight*.

An Invariant of the Cubic

As we stated in part 1 of this series [3], the Hessian of the cubic is a quadratic polynomial formed as the determinant of the matrix of second derivatives. What does this look like in diagram notation? The first derivative of $f(x,w)$ is a vector composed of the partial derivatives $[f_x, f_w]$ which can be written in diagram notation by simply lopping off one input node:

$$[f_x \quad f_w] = 3 \begin{bmatrix} Ax^2 + 2Bxw + Cw^2 & Bx^2 + 2Cxw + Dw^2 \end{bmatrix}$$



(0.20)

The second derivative matrix comes from lopping of two nodes

$$\begin{bmatrix} f_{xx} & f_{xw} \\ f_{xw} & f_{ww} \end{bmatrix} = 6 \begin{bmatrix} Ax + Bw & Bx + Cw \\ Bx + Cw & Cx + Dw \end{bmatrix} = 6 \begin{bmatrix} x,w & C \end{bmatrix}$$

The determinant of this, as prescribed by equation (0.18), will be

$$\det \begin{bmatrix} f_{xx} & f_{xw} \\ f_{xw} & f_{ww} \end{bmatrix} = -18 \begin{bmatrix} x,w & C & C & x,w \end{bmatrix}$$

I decided, for convenience, to define the Hessian matrix **H** (equation (0.1)) without the constant factor of 18 giving us

$$\begin{bmatrix} H \end{bmatrix} = - \begin{bmatrix} C & C \end{bmatrix}$$

(0.21)

Next we look at the cubic discriminant, $\Delta = \det \mathbf{H}$, which in diagram form is

$$\begin{array}{c} \text{H} \\ \leftarrow \quad \rightarrow \\ \text{H} \end{array} = -2 \det \mathbf{H} = -2\Delta$$

Plug in equation (0.21) twice, once for each \mathbf{H} , and the minus's from that equation go away leaving us with the cubic invariant

$$\begin{array}{ccc} & \text{C} & \\ \text{C} & \leftarrow \quad \rightarrow & \text{C} \\ & \text{C} & \\ & \text{C} & \end{array} = -2\Delta$$

The existence of six epsilons in this diagram immediately verifies the statement made in part 1 that the discriminant of a transformed cubic differs from the discriminant of the original by a factor of $(\det \mathbf{T})^6$

Covariants

Now let's look at what happens to the Hessian when we transform the \mathbf{C} tensor. The Hessian of the transformed tensor is

$$\begin{array}{c} \leftarrow \text{H} \rightarrow \end{array} = - \begin{array}{ccc} & \text{C} & \\ \leftarrow & \leftarrow \quad \rightarrow & \leftarrow \\ & \text{C} & \end{array}$$

Applying equation (0.13) and then equation (0.19) gives us

$$\begin{array}{c} \leftarrow \text{H} \rightarrow \end{array} = - \begin{array}{ccccccc} & & \text{T} & & \text{T} & & \\ & & \leftarrow & & \rightarrow & & \\ \leftarrow & \text{T} & \leftarrow & \text{C} & \rightarrow & \text{C} & \rightarrow \\ & & \leftarrow & & \rightarrow & & \\ & & \text{T} & & \text{T} & & \\ & & \leftarrow & & \rightarrow & & \end{array} \\ = - (\det \mathbf{T})^2 \begin{array}{ccc} & \text{C} & \\ \leftarrow & \leftarrow \quad \rightarrow & \leftarrow \\ & \text{C} & \end{array} \\ = (\det \mathbf{T})^2 \begin{array}{ccc} \leftarrow & \text{T} & \leftarrow \\ & \text{H} & \\ \leftarrow & \text{T} & \leftarrow \end{array}$$

So the Hessian transforms, as a quadratic, by the same transform that was applied to the cubic; the transform of the Hessian is the Hessian of the transform. This not-entirely-surprising fact is still important. It means that the Hessian is geometrically locked to the cubic and represents a transformationally invariant property. This type of object is called a *covariant* of the cubic. In fact, the main difference between an invariant and a covariant is that an invariant is a scalar and a covariant is a tensor—it still has some dangling arcs.

Much of the early work on invariants and covariants of homogeneous polynomials (also called *forms* or *quintics*) was due to David Hilbert. A good resource is [4] which is a recent reprint of a series of lectures originally given in 1897 and which gave me a lot of the inspiration for this and later derivations. Hilbert did not use diagrams, but rather developed what amounts to a set of lexicographic rules for testing whether polynomials in A, B, C, D like equation (0.2) were invariants. For example if you rename the coefficients

with indices $A = a_3, B = a_2, C = a_1, D = a_0$ and consider powers to be just replications of a quantity (so that $A^2 D^2 = a_3 a_3 a_0 a_0$) then a necessary (but not sufficient) condition for a polynomial in the a_i 's to be an invariant is that the sum of the subscripts in each term is the same. For equation (0.2) you can verify for yourself that that index sum for each term is 6 (which equals the *weight* of the expression, or diagramwise, the number of epsilons).

Depression in diagram form

To make $\tilde{B} = 0$ we had to choose the second row of the transformation according to Equation (0.8). We can write this in terms of epsilon with

$$[s \ v] = \begin{bmatrix} -t^2 B - 2tuC - u^2 D & t^2 A + 2tuB + u^2 C \\ t^2 A + 2tuB + u^2 C & t^2 B + 2tuC + u^2 D \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Comparing this to the first derivative diagram of equation (0.20) shows us the diagram form of $[s, v]$

$$[s, v] \Rightarrow = \begin{array}{c} \textcircled{t,u} \\ \swarrow \\ \textcircled{C} \\ \searrow \\ \textcircled{t,u} \end{array} \quad (0.22)$$

Plugging this into the diagrams of equation (0.14) gives us the following. For \tilde{B} we have

$$\tilde{B} = \begin{array}{c} \textcircled{t,u} \\ \swarrow \\ \textcircled{s,v} \rightarrow \textcircled{C} \\ \searrow \\ \textcircled{t,u} \end{array} = \begin{array}{c} \textcircled{t,u} \quad \textcircled{t,u} \\ \swarrow \quad \searrow \\ \textcircled{C} \quad \leftrightarrow \quad \textcircled{C} \\ \swarrow \quad \searrow \\ \textcircled{t,u} \quad \textcircled{t,u} \end{array}$$

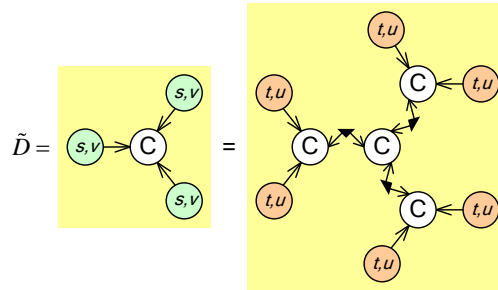
This is of form

$$\textcircled{K} \leftrightarrow \textcircled{K}$$

Which is identically zero, as we expect, since we designed (s, v) to make this happen. Next, for \tilde{C} we have

$$\tilde{C} = \begin{array}{c} \textcircled{s,v} \\ \swarrow \\ \textcircled{s,v} \rightarrow \textcircled{C} \\ \searrow \\ \textcircled{t,u} \end{array} = \begin{array}{c} \textcircled{t,u} \\ \swarrow \\ \textcircled{C} \quad \leftrightarrow \quad \textcircled{C} \\ \swarrow \quad \searrow \\ \textcircled{t,u} \quad \textcircled{t,u} \end{array}$$

and for \tilde{D} we have



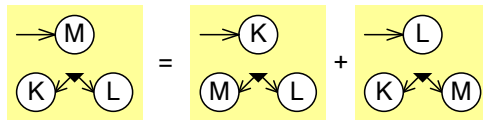
Now let's see how to manipulate these diagrams into something simpler.

Doing Things with Diagrams

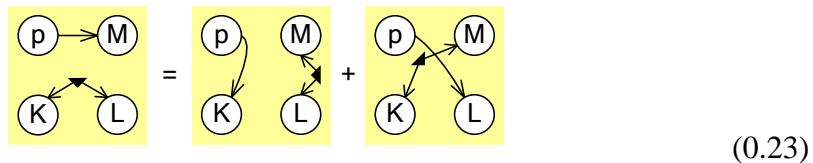
So far, I've used tensors only as a representation scheme for polynomial expressions. The real power is in using them to manipulate expressions. The technique is based on the observation that there can be only two linearly independent 2-vectors. Any third vector must be a linear combination of the first two. A little fiddling can show that the scale factors in the linear combination are given by the following identity:

$$\mathbf{M}(K_0L_1 - K_1L_0) = \mathbf{K}(M_0L_1 - M_1L_0) + \mathbf{L}(K_0M_1 - K_1M_0)$$

In diagram notation this looks like

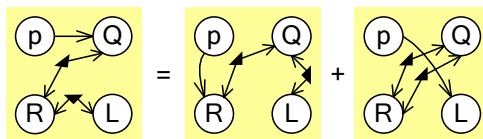


Rearranging this a bit to put **K**, **L** and **M** in a fixed position, and plugging in another tensor as a placeholder, we get



Now the cool thing is that this works even if **p**, **K**, **L**, and **M** are not simply vectors. They can be more general tensors with any number of other arrows pointing into and out of them that don't directly participate in the identity, but just go along for the ride. This basic identity then allows us to reconnect the arrows of any diagram to construct two new diagrams whose sum is equal to the original one.

As a useful example, suppose the nodes **K** and **M** were actually quadratics **R** and **Q** with an epsilon between them. Our diagram equation would look like



Clean this up with a couple of epsilon mirrors (and associated sign flips) for neatness and get the following identity:

$$\begin{aligned}
 & \text{Diagram: } \boxed{p \Rightarrow Q \rightleftarrows R \rightleftarrows L} = \\
 & - \text{Diagram: } \boxed{p \Rightarrow R \rightleftarrows Q \rightleftarrows L} + \begin{matrix} \text{Diagram: } \boxed{p \Rightarrow L} \\ \text{Diagram: } \boxed{R \rightleftarrows Q} \end{matrix}
 \end{aligned}
 \tag{0.24}$$

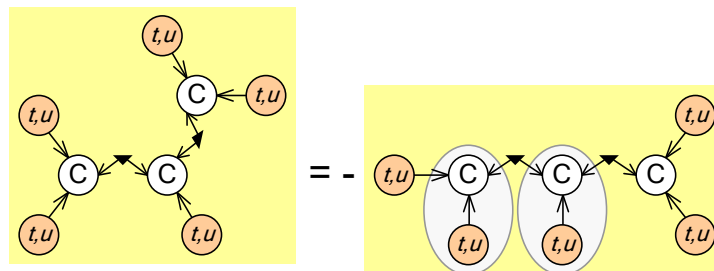
As a special case of this identity, consider what happens if you plug in $R=Q$. The identity turns into something we've seen before, a combination of equations (0.17) and (0.18).

$$\text{Diagram: } \boxed{p \Rightarrow Q \rightleftarrows Q \rightleftarrows L} = \frac{1}{2} \begin{matrix} \text{Diagram: } \boxed{p \Rightarrow L} \\ \text{Diagram: } \boxed{Q \rightleftarrows Q} \end{matrix}
 \tag{0.25}$$

So whenever we see a chain of two identical quadratics, we can apply this identity and split the diagram into two disjoint pieces. This trick applies immediately to our diagram for \tilde{C}

Simplifying \tilde{C}

Lets start with the \tilde{C} diagram and bend it around a bit (with a sign flip) to get



We can now apply identity (0.25), treating the two diagram pieces inside the gray ellipses as the quadratic Q . We get

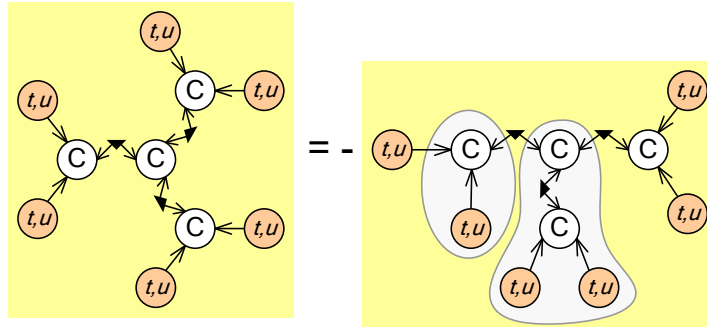
$$\tilde{C} = -\frac{1}{2} \text{Diagram: } \begin{matrix} \text{Diagram: } \boxed{t,u \Rightarrow C} \\ \text{Diagram: } \boxed{t,u \Rightarrow C \rightleftarrows C \rightleftarrows t,u} \end{matrix}
 \tag{0.26}$$

The diagram has fallen apart into two disjoint pieces. Each piece is an independent factor of the net expression. And one of the pieces is, as advertised, the value of \tilde{A} .

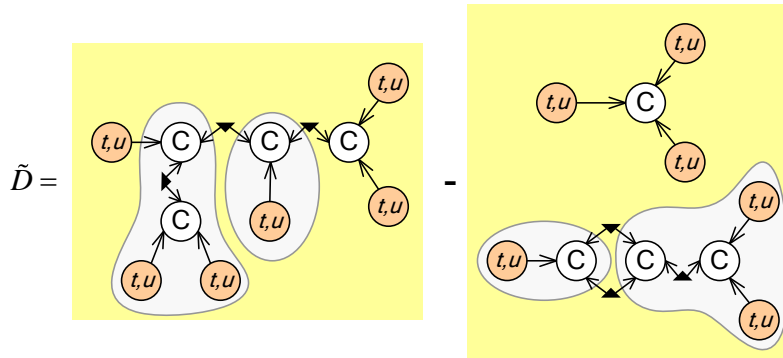
Simplifying \tilde{D}

To simplify the diagram for \tilde{D} we again start by bending the diagram around a bit and flipping an epsilon (and the sign).

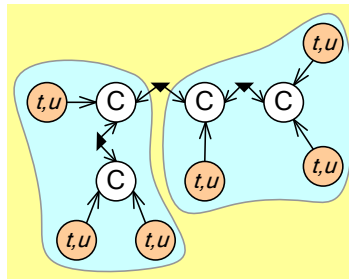
How to Solve a Cubic Equation – Part 3



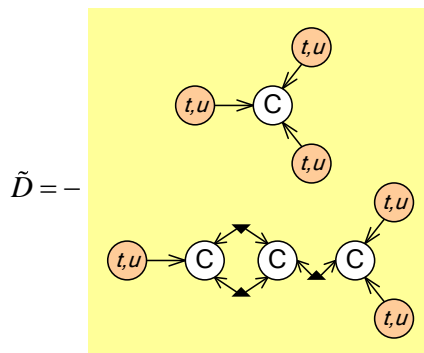
This time we apply the more general identity of Equation (0.24) to the two gray regions and get



Now look at the first term on the right hand side. What do you see? Let me give you a hint by the following grouping.



Remarkably, this is two identical things (indicated by the blue areas) on each side of an epsilon (the middle one of the three). So it's... ZERO. This is great. We like things that are zero. This makes \tilde{D} equal to just the second term.



(0.27)

Again notice the factor of \tilde{A} .

General Depression

Let's put all this together. Using the general transformation of equation (0.3) with the second row, $[s,v]$, defined by equation (0.22) we have depressed our polynomial into the form $\tilde{A}\tilde{x}^3 + 3\tilde{C}\tilde{x}\tilde{w}^2 + \tilde{D}\tilde{w}^3$. Then looking at equations (0.26) and (0.27) we see that we can remove the constant factor of $\tilde{A} = f(t,u)$ from this expression giving the net equation we need to solve as:

$$\tilde{x}^3 + 3\tilde{C}\tilde{x}\tilde{w}^2 + \tilde{D}\tilde{w}^3 = 0$$

Where

$$\bar{C} = -\frac{1}{2} \begin{array}{c} \text{---} \\ \circlearrowleft \text{---} \\ \text{---} \\ \circlearrowright \text{---} \\ \text{---} \end{array}$$

The diagram shows a central horizontal chain of two white circles labeled 'C'. To the left of the first 'C' is a red circle containing the text '(t,u)'. To the right of the second 'C' is another red circle containing '(t,u)'. Four curved arrows form a closed loop: two arrows on top pointing from right to left, and two arrows on bottom pointing from left to right.

$$\bar{D} = - \begin{array}{c} \text{---} \\ \circlearrowleft \text{---} \\ \text{---} \\ \circlearrowright \text{---} \\ \text{---} \end{array}$$

The diagram shows a central horizontal chain of three white circles labeled 'C'. To the left of the first 'C' is a red circle containing the text '(t,u)'. To the right of the third 'C' are two red circles, one above and one below, both containing '(t,u)'. Four curved arrows form a closed loop: two arrows on top pointing from right to left, and two arrows on bottom pointing from left to right.

(0.28)

Algorithms A and D are simply special cases of this with $[t,u]=[1,0]$ and $[t,u]=[0,1]$ respectively. But now we have a whole continuum of \bar{C} and \bar{D} values parameterized by $[t,u]$. We now think of \bar{C} and \bar{D} as polynomial functions of (t,u) . The expression for \bar{C} is something we've seen already; it's just the half of the Hessian evaluated at (t,u) . But what about \bar{D} ?

What is \bar{D} ?

\bar{D} is now a homogeneous cubic polynomial in (t,u) . And, since equation (0.28) shows it written as a tensor diagram, it is also a covariant of the original cubic tensor \mathbf{C} . (Hilbert calls this the *skew* covariant because the weight is odd.) So you can also think of \bar{D} as a mapping of one cubic polynomial to another one, $\bar{D}(\mathbf{C})$. For concreteness, let's explicitly find this mapping. Equation (0.28), written as a matrix product looks like

$$\bar{D}(t,u) = [t \quad u] \mathbf{H} \begin{bmatrix} s \\ v \end{bmatrix}$$

Plugging in equation (0.8) for $[s,v]$, expanding out and collecting like terms for t^3 etc gives

$$\bar{D}(t,u) = A_{\bar{D}}t^3 + 3B_{\bar{D}}t^2u + 3C_{\bar{D}}tu^2 + D_{\bar{D}}u^3$$

with

$$\begin{aligned} A_{\bar{D}} &= +A^2D - 3ABC + 2B^3 \\ B_{\bar{D}} &= -2AC^2 + ABD + B^2C \\ C_{\bar{D}} &= -ACD + 2B^2D - BC^2 \\ D_{\bar{D}} &= -AD^2 + 3BCD - 2C^3 \end{aligned} \tag{0.29}$$

Let's see what else we can find out about this mapping.

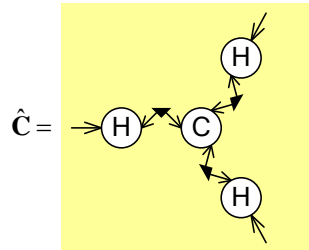
\bar{D} is a transformation of C

I will now show that the mapping $\bar{D}(C)$ is just a parameter-space transformation of C . Of course any cubic of a particular type is just a parameter transformation of other cubics of the same type. But this transformation is special; it's just the product of H with epsilon, $H\epsilon$. In diagram form this is

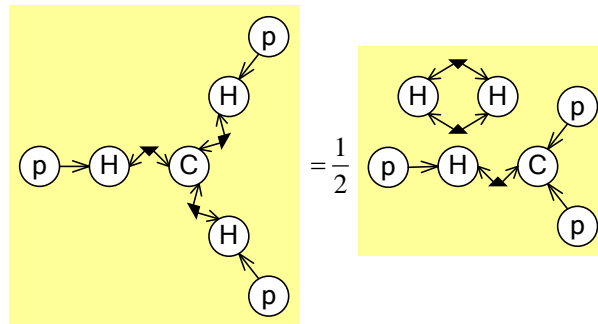
$$\Rightarrow \textcircled{T} \Rightarrow = \Rightarrow \textcircled{H} \begin{array}{l} \nearrow \\ \searrow \end{array}$$

Note that plugging in an epsilon has changed the two-arrows-in matrix H to a one-in-and-one-out matrix that typifies a transformation matrix.

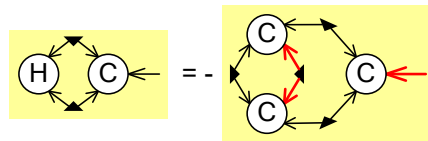
The transformed cubic is then



To see why \hat{C} is homogeneously equivalent to $\bar{D}(C)$ we again do a derivation using tensor diagrams. I'm not going to do this in detail here, but the process is very similar to what we've seen before. The net result is

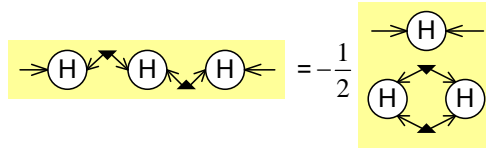


We can see that this is at least reasonable since the number of nodes, epsilons and arcs is the same; they are just connected together differently. The only slightly tricky part in the proof is that you will encounter the following diagram fragment



This fragment is identically zero, which you can prove by applying the basic identity of equation (0.23) to the arcs marked in red.

Now what about the Hessian of the cubic \bar{D} ? Since the Hessian operation is a covariant and since we transform C by $H\epsilon$ to get $\bar{D}(C)$ we can simply transform the Hessian of C by $H\epsilon$ to find the Hessian of \bar{D} . We get



So we get almost trivially that the Hessians of \mathbf{C} and $\bar{D}(\mathbf{C})$ are the same up to a homogeneous scalar. In part 1 we talked about the space of possible cubics as being foliated by a network of lines of constant Hessian. We have just shown that \mathbf{C} and $\bar{D}(\mathbf{C})$ lie on the same line in this space.

Finally let's think about what happens if we apply the \bar{D} operation twice; what is $\bar{D}(\bar{D}(\mathbf{C}))$? Well, Equation (0.17) basically tells us that the matrix $\mathbf{H}\mathcal{E}$ is its own inverse (up to a homogeneous scale factor). In other words applying the \bar{D} operation twice gets us back to the original cubic.

\bar{D} is a function of other covariants

At first it might seem that the discovery of \bar{D} gives us something new about the cubic polynomial. But in fact it turns out that \bar{D} is a function of the other three in/covariants that we already know about. Just looking at equation (0.28) it's not immediately obvious how this can be. Any attempt to process the diagram for \bar{D} into something else, using the basic identity in Equation (0.23), doesn't get us anywhere. But if we instead start playing with the *square* of \bar{D} (formed by just placing two copies side by side) we make some progress. The first step uses Equation (0.23) on this pair as follows

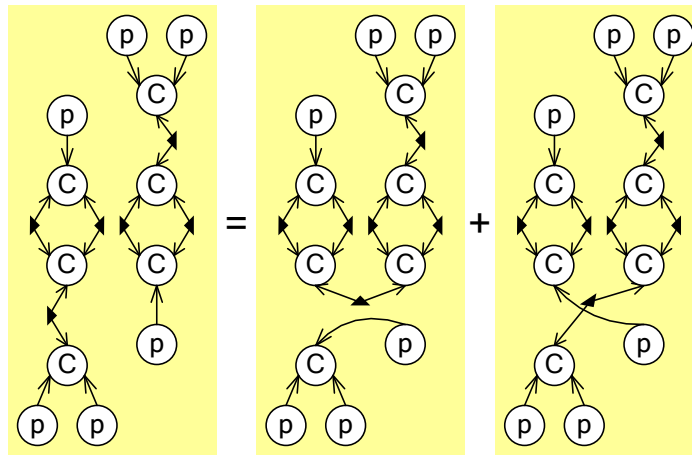


Figure 1 shows the ultimate result (which requires a few more applications of Equation (0.23)). Again, as a visual sanity check note that each term in figure 1 has the same number of \mathbf{C} 's, \mathbf{p} 's and epsilons; they are just connected together in different ways.

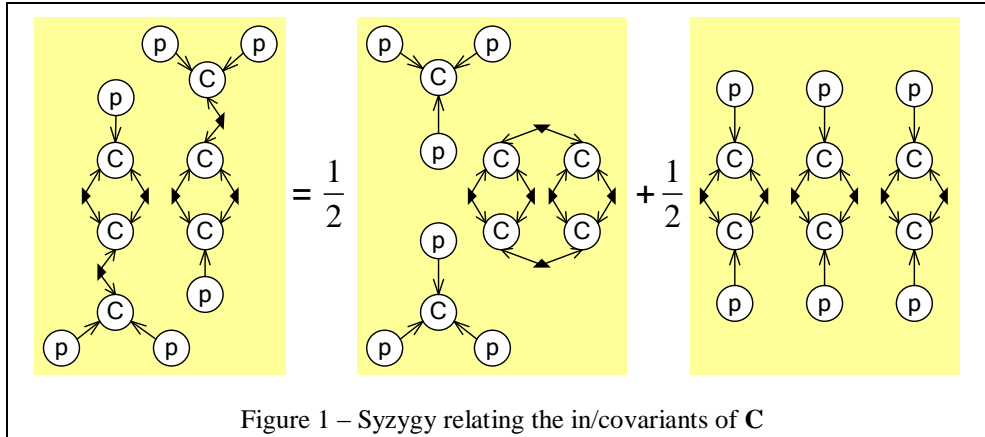


Figure 1 – Syzygy relating the in/covariants of C

We can write the diagram of figure 1 algebraically as

$$(-\bar{D}(x, w))^2 = \frac{1}{2} \tilde{A}(x, w)^2 (-2\Delta) + \frac{1}{2} (-2\bar{C}(x, w))^3$$

Which simplifies to

$$\bar{D}(x, w)^2 + 4\bar{C}(x, w)^3 = -\tilde{A}(x, w)^2 \Delta$$

This is the generalization of equation 13 of part 2 to a general $[t, u]$ instead of $[1, 0]$. This sort of relation is called a syzygy. As a general principle, any diagram that contains an odd number of epsilons can usually be related to simpler in/covariants by processing its square in this fashion.

Examples

Let's round out our intuition about the Hessian and the \bar{D} operation by looking at what they do to each of the four types of cubics. I'll start with a specific example of each type and apply equation (0.1) to get the Hessian, both in the form of the matrix \mathbf{H} and of its associated quadratic polynomial which I'll write as $H(f)$, and then apply equation (0.29) to calculate $\bar{D}(f)$. Then I'll discuss how this looks for a more general cubic of that type.

Type 3

The simplest exemplar of triple-root cubics is $f(x, w) = x^3$. Plugging this into equation (0.1) shows that the Hessian is identically zero (this is the defining characteristic of a triple-root cubic). Then equation (0.29) shows that \bar{D} is also identically zero. We'll catalog these as

$$\mathbf{H} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad H(x^3) = 0$$

$$\mathbf{H}\varepsilon = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \bar{D}(x^3) = 0$$

Type 21

This is typified by the function $f(x, w) = 3x^2w$ which leads to

$$\mathbf{H} = \begin{bmatrix} -2 & 0 \\ 0 & 0 \end{bmatrix} \quad H(3x^2w) = -2x^2$$

$$\mathbf{H}\mathcal{E} = \begin{bmatrix} 0 & -2 \\ 0 & 0 \end{bmatrix} \quad \bar{D}(3x^2w) = 2x^3$$

The transformation $\mathbf{H}\mathcal{E}$ that maps f to \bar{D} is singular, so it is able to change the type of the cubic from Type 21 to Type 3—that is, to the cubic that has a triple root where the original f has a double root. There's a nice geometric interpretation of this phenomenon. Recall that in part 1 we found that the discriminant surface $\Delta = 0$, where all type 21 cubics reside, is a ruled surface. A particular ruling line is the set of all type 21 cubics that share the same Hessian. The set of all type 3 cubics lies on a crease in the discriminant surface and each ruling line crosses this crease exactly once. The operation of forming the \bar{D} of a type 21 cubic then consists of following along its ruling line to the crease.

Generalizing the single root at $w=0$ to instead be at $Ax + 3Bw = 0$ gives us, after the appropriate computation:

$$\mathbf{H} = \begin{bmatrix} -2B^2 & 0 \\ 0 & 0 \end{bmatrix}, \quad H(Ax^3 + 3Bx^2w) = -2B^2x^2$$

$$\mathbf{H}\mathcal{E} = \begin{bmatrix} 0 & -2B^2 \\ 0 & 0 \end{bmatrix}, \quad \bar{D}(Ax^3 + 3Bx^2w) = 2B^3x^3$$

The coefficient A does not appear in the formulas for H and \bar{D} ; only the location of the double root matters. This means that a ruling line (a line of constant Hessian) is also a line of constant double root. Sliding up and down along this line just changes the single root until, at the crease, the single root momentarily coincides with the double root.

Type 111

A nice example of a type 111 cubic is $f(x, w) = x(x - \sqrt{3}w)(x + \sqrt{3}w) = x^3 - 3xw^2$. If you plot the three roots as lines through the origin in the x, w plane you find that they are equally spaced 60 degrees apart, Figure 2. Some computation gives:

$$\mathbf{H} = \begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix}, \quad H(x^3 - 3xw^2) = -2(x^2 + w^2)$$

$$\mathbf{H}\mathcal{E} = \begin{bmatrix} 0 & -2 \\ 2 & 0 \end{bmatrix}, \quad \bar{D}(x^3 - 3xw^2) = 2(-\sqrt{3}x + w)(\sqrt{3}x + w)w$$

The Hessian has no real roots. The matrix $\mathbf{H}\mathcal{E}$ that transforms f to \bar{D} is a 90 degree rotation so \bar{D} has three roots interleaved between the three roots of f , again see figure 2.

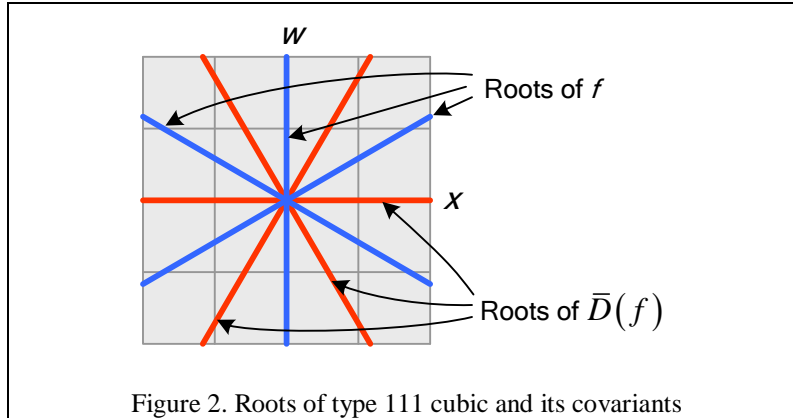


Figure 2. Roots of type 111 cubic and its covariants

For a general type 111 cubic, the Hessian will always have no real roots and be negative definite. Now recall that the roots of \mathbf{H} correspond to the eigenvectors of $\mathbf{H}\mathcal{E}$, so the transformation from f to \bar{D} has no real eigenvectors. This means that it is always a generalized rotation—that is, one with some possible additional shearing. So we have the situation that sliding a type 111 cubic along a line of constant Hessian basically rotates its roots until it gets to \bar{D} . But there's something else that's interesting. In figure 2 note that a rotation of 90 degrees is not the only rotation that can change f into \bar{D} . A rotation of 30 degrees and -30 degrees will also work. In the general case the matrix $\mathbf{H}\mathcal{E}$ is not the only transformation that does the job. Another matrix that works is one that is effectively the cube root of $\mathbf{H}\mathcal{E}$.

Type 1 $\bar{1}$

Here our concrete example will be $f(x, w) = x(x^2 + 3w^2) = x^3 + 3xw^2$ so it has one real root at $[x, w] = [0, 1]$. Computation reveals

$$\mathbf{H} = \begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix}, \quad H(x^3 + 3xw^2) = 2(x^2 - w^2)$$

$$\mathbf{H}\mathcal{E} = \begin{bmatrix} 0 & 2 \\ 2 & 0 \end{bmatrix}, \quad \bar{D}(x^3 + 3xw^2) = -2(3x^2w + w^3)$$

The Hessian has two real roots, $[x, w] = [1, 1]$ and $[x, w] = [-1, 1]$. The transformation matrix $\mathbf{H}\mathcal{E}$ simply exchanges x and w ; it is a mirror about the line $x=w$. So \bar{D} is a cubic with one real root at $[x, w] = [1, 0]$. See figure 3.

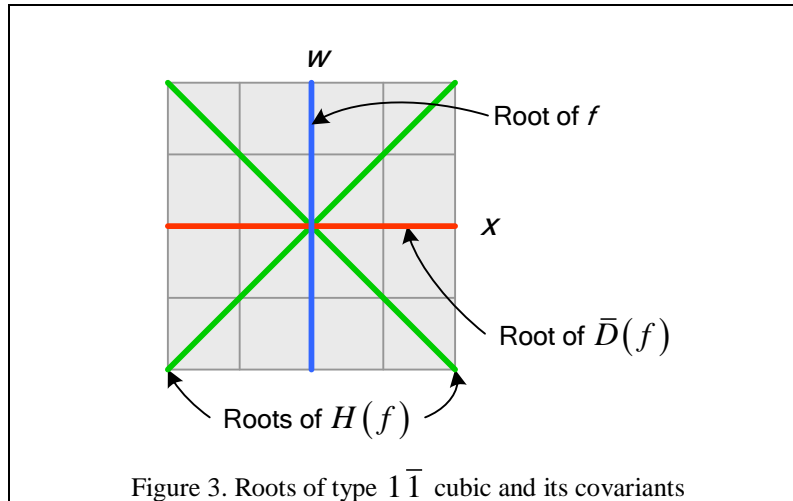


Figure 3. Roots of type $1 \bar{I}$ cubic and its covariants

In the general case $\mathbf{H}\varepsilon$ will always have two real eigenvectors and a negative determinant (since $\Delta < 0$). So it will mirror the single real root of f into the single real root of $\bar{D}(f)$. But here's the really interesting thing: Recall that in part 1 I showed that lines of constant Hessian for type $1 \bar{I}$ cubics intersect the triple-root curve at two points. These two triple-root cubics are the cubes of the roots of the Hessian. In our special case these are $(x+w)^3$ and $(x-w)^3$. Now we have a new cubic, $\bar{D}(f)$, that is also on this constant Hessian line. This means that these four cubic polynomials are linearly dependent; in particular in our special case we can easily see that:

$$f + \frac{1}{2}\bar{D}(f) = (x-w)^3$$

$$f - \frac{1}{2}\bar{D}(f) = (x+w)^3$$

So traveling along the constant-Hessian line we encounter, in order, $f, (x-w)^3, \bar{D}(f), (x+w)^3$ and then get back to f .

What can we do with this?

Our ultimate goal here is twofold. We want to get insight into how cubic polynomials work: what are the in/covariants and their geometric interpretations, how does a transformation in parameter space relate to the transformation in coefficient space, etc. But, more practically, we are also interested in finding numerically sound ways to calculate the roots of a cubic. To do this it is useful to have more than one way to calculate the roots. For example, when we were solving quadratics we found numerical problems in the expression $(-B + \sqrt{B^2 - AC})/A$ when B is positive and $AC \ll B^2$. But knowing about the problem was only useful because there was another way to calculate the same value: $C/(-B - \sqrt{B^2 - AC})$, that didn't have numerical problems. Similarly, last time we found two algorithms that complement each other in solving type $1 \bar{I}$ cubics so that at least one of them (pretty much) will be numerically nice. In part 4 we will show a preliminary solution to type 111 cubics, and in part 5, with the insights gained in this article, we will work on finding numerically nice solutions to all cubics.

Meanwhile, I'll get back to my *Lost DVD*'s. I wonder if Walt will get rescued. (I'll bet he does.)

References

- [1] J. F. Blinn, “How to Solve a Cubic Equation, Part 2 – The $1\bar{1}$ case”, IEEE CG&A, Vol 26, No 4, Jul/Aug 2006, pages 90 – 100.
- [2] J. F. Blinn, “Polynomial Discriminants Part 2 – Tensor Diagrams” IEEE CG&A, Jan/Feb 2001, and J. F. Blinn, “Quartic Discriminants and Tensor Invariants”, IEEE CG&A, Mar/Apr 2002, both reprinted in *Jim Blinn’s Corner: Notation, Notation, Notation*, Chapter 20, Morgan Kaufman, 2003.
- [3] J. F. Blinn, “How to Solve a Cubic Equation, Part 1 – The Shape of the Discriminant”, IEEE CG&A, Vol 26, No 3, 2006, pages 84 – 93.
- [4] D. Hilbert, *Theory of Algebraic Invariants*, Cambridge University Press, 1993.