Solving Polynomial Equations Using Linear Algebra

Michael Peretzian Williams

Quadric intersection is a common class of nonlinear systems of equations. Quadrics, which are the class of all degree-two polynomials in three or more variables, appear in many engineering problems, such as multilateration. Typically, numerical methods are used to solve such problems. Unfortunately, these methods require an initial guess and, although rates of convergence are well understood, convergence is not necessarily certain. The method discussed in this article transforms the problem of simultaneously solving a system of polynomials into a linear algebra problem that, unlike other root-finding methods, does not require an initial guess. Additionally, iterative methods only give one solution at a time. The method outlined here gives all solutions (including complex ones).

INTRODUCTION

Multivariate polynomials show up in many applications. Polynomials are attractive because they are well understood and they have significant simplicity and structure in that they are vector spaces and rings. Additionally, degree-two polynomials (conic sections that are also known as quadrics) show up in many engineering applications, including multilateration.

This article begins with a discussion on finding roots of univariate polynomials via eigenvalues/eigenvectors of companion matrices. Next, we briefly introduce the concept of a ring to enable us to discuss ring representation of polynomials that are a generalization of companion matrices. We then discuss how multivariate polynomial systems can be solved with ring representations. Following that, we outline an algorithm attributable to Emiris[1] that is used to find ring representations. Finally, we give an example of the methods applied to a trilateration quadric-intersection problem.

COMPANION MATRICES: FINDING ROOTS OF UNIVARIATE POLYNOMIALS AS AN EIGENVALUE/EIGENVECTOR PROBLEM

Recall that by definition an $n \times n$ matrix M has eigenvectors v_i and eigenvalues λ_i if $Mv_i = \lambda_i v_i$.

Such a matrix can have at most n eigenvalues/eigenvectors. Solving for λ_i is equivalent to solving $\det(M - xI) = 0$. This calculation yields a polynomial in x called the "characteristic polynomial," the roots of which are the eigenvalues λ_i. Once the λ_i values are obtained, the eigenvectors v_i can be calculated by solving $(M - \lambda_i I)v_i = 0$. Hence, it is not surprising that eigenvalues/eigenvectors provide a method for solving polynomials.

Next, we consider the univariate polynomial $p(x) = \sum_{i=0}^{n} C_i x^i$, where $C_n = 1$. The matrix

$$M = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -C_0 & -C_1 & \cdots & -C_{n-1} \end{bmatrix}$$

is known as the "companion matrix" for p. Recall that the Vandermonde matrix is defined as

$$V(x_1, x_2, \ldots, x_n) = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \ddots & x_n \\ \vdots & \vdots & \cdots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{bmatrix}.$$

As it turns out, the roots of the characteristic polynomial of M [i.e., $\det(M - xI) = 0$] are precisely the roots of the polynomial p, and the eigenvectors are Vandermonde vectors of the roots.

Indeed, if λ_j is a root of p [i.e., $p(\lambda_j) = 0$] and v is a Vandermonde vector such that $v = [1 \;\; \lambda_j \;\; \cdots \;\; \lambda_j^{n-1}]^T$, then

$$Mv = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -C_0 & -C_1 & \cdots & -C_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ \lambda_j \\ \vdots \\ \lambda_j^{n-1} \end{bmatrix} = \begin{bmatrix} \lambda_j \\ \lambda_j^2 \\ \vdots \\ -\sum_{i=0}^{n-1} C_i \lambda_j^i \end{bmatrix} = \begin{bmatrix} \lambda_j \\ \lambda_j^2 \\ \vdots \\ \lambda_j^n - p(\lambda_j) \end{bmatrix} = \begin{bmatrix} \lambda_j \\ \lambda_j^2 \\ \vdots \\ \lambda_j^n - 0 \end{bmatrix} = \lambda_j \begin{bmatrix} 1 \\ \lambda_j^2 \\ \vdots \\ \lambda_j^{n-1} \end{bmatrix} = \lambda_j v.$$

Hence, the eigenvalues of M are the roots of p. Likewise, the right eigenvectors of M are the columns of the Vandermonde matrix of the roots of p.

The most important property of companion matrices in this article can be stated as follows:

> Given a polynomial p, the companion matrix defines a matrix M such that the characteristic polynomial of M is p [i.e., $\det(M - xI) = \pm p(x)$].

To give a better understanding of the above statement, consider the following concrete example.

$$\begin{aligned} \text{Let } p(x) &= (x-2)(x-3)(x-5) \\ &= x^3 - (2+3+5)x^2 + (2 \cdot 3 + 2 \cdot 5 + 3 \cdot 5)x - 2 \cdot 3 \cdot 5 \\ &= x^3 - 10x^2 + 31x - 30 \end{aligned}$$

We can verify (for this case) that the eigenvalue/eigenvector statements made above hold [i.e., $\det(M - xI) = \pm p(x)$]. The companion matrix for $p(x)$ is

$$M = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 30 & -31 & 10 \end{bmatrix}.$$

Since the roots of p are (2, 3, 5) by construction, the eigenvectors of M should be given by the Vandermonde matrix

$$V(2,3,5) = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 3 & 5 \\ 4 & 9 & 25 \end{bmatrix}.$$

We observe that

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 30 & -31 & 10 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 30 - 2 \cdot 31 + 4 \cdot 10 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 8 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 30 & -31 & 10 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 9 \end{bmatrix} = \begin{bmatrix} 3 \\ 9 \\ 30 - 3 \cdot 31 + 9 \cdot 10 \end{bmatrix} = \begin{bmatrix} 3 \\ 9 \\ 27 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 3 \\ 9 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 30 & -31 & 10 \end{bmatrix} \begin{bmatrix} 1 \\ 5 \\ 25 \end{bmatrix} = \begin{bmatrix} 5 \\ 25 \\ 30 - 5 \cdot 31 + 25 \cdot 10 \end{bmatrix} = \begin{bmatrix} 5 \\ 25 \\ 125 \end{bmatrix} = 5 \begin{bmatrix} 1 \\ 5 \\ 25 \end{bmatrix}.$$

Hence, the eigenvalues of M are given by $\lambda = 2, 3, 5$, the roots of $p(x)$ as claimed. Also, we note that p is the characteristic polynomial for M [i.e., $\det(M - xI) = \pm p(x)$]. Indeed,

$$\begin{aligned} \det(M - \lambda I) &= \det\left(\begin{bmatrix} -\lambda & 1 & 0 \\ 0 & -\lambda & 1 \\ 30 & -31 & 10-\lambda \end{bmatrix}\right) \\ &= -\lambda \det\left(\begin{bmatrix} -\lambda & 1 \\ -31 & 10-\lambda \end{bmatrix}\right) - 1 \det\left(\begin{bmatrix} 0 & 1 \\ 30 & 10-\lambda \end{bmatrix}\right) + 0 \det\left(\begin{bmatrix} -\lambda & 0 \\ -31 & 30 \end{bmatrix}\right) \\ &= -\lambda(-\lambda(10-\lambda) + 31) + 30 \\ &= -\lambda^3 + 10\lambda^2 - 31\lambda + 30 = -p(\lambda). \end{aligned}$$

Hence, $\det(M - xI) = \pm p(x)$ as claimed.

In this section, we have outlined univariate examples of the companion matrix and Vandermonde matrix. Indeed, we have demonstrated that finding the roots of a univariate polynomial is an eigenvalue/eigenvector problem. In a later section, we discuss the u-resultant algorithm, which is used to construct a generalization of the companion matrix. These generalizations, known herein as "ring representations," can be used to solve polynomial systems, including quadric intersection.

RINGS AND RING REPRESENTATIONS

Previously, we discussed transforming univariate polynomial root finding into an eigenvalue/eigenvector problem. For the remainder of this article, we will generalize the method above to simultaneously solve systems of multivariate polynomial equations. More precisely, we want to find coordinates $(x_1, x_2, \ldots, x_n)$ such that for polynomials $f_1, f_2, \ldots, f_n$

$$\begin{aligned} f_1(x_1, x_2, \ldots, x_n) &= 0 \\ f_2(x_1, x_2, \ldots, x_n) &= 0 \\ &\vdots \\ f_n(x_1, x_2, \ldots, x_n) &= 0\,. \end{aligned}$$

As with many multivariate generalizations, the univariate case does not require understanding of the deeper structure inherent to the problem. The last section is no exception. A brief discussion regarding the algebraic concept of a ring is necessary to advance the generalization.

Rings are an algebraic structure having both a multiplication and addition but not necessarily a division. The most common rings are integers, univariate polynomials, and multivariate polynomials. For any a, b, c in a ring R, the following properties hold.

Addition:

A. Closure, $a + b$ are in R
B. Associativity, $(a + b) + c = a + (b + c)$
C. Identity, $a + 0 = 0 + a = a$
D. Inverse, $a + (-a) = (-a) + a = 0$
E. Commutativity, $a + \mathrm{b} = \mathrm{b} + a$

Multiplication:

A. Closure, ab is in R
B. Associativity, $(ab)c = a(bc)$
C. Identity, $a1 = 1a = a$ (optional)
D. Inverse, $a(a^{-1}) = (a^{-1})a = 1$ (optional)
E. Commutativity, $ab = ba$ (optional)

Algebra, the field of mathematics in which rings are studied, commonly looks at an operation called *adjoining*, or including new elements. If R is a ring, then adjoining is accomplished by including some new element j and creating the ring $R[j]$ in the following fashion: $R[j] = a_0 + a_1 j + a_2 j^2 + a_3 j^3 \ldots$ for all $a_0, a_1, a_2, a_3, \ldots$ in R. A common instance of this operation is adjoining the imaginary unit i to the real numbers to obtain the complex numbers.

Because matrices are well understood and easily implemented in code, ring representations (a special family of matrices particular to a ring) are a common way of handling algebraic applications. In the same way that logarithms map multiplication to addition ($\log(ab) = \log(a) + \log(b)$) and Fourier transforms map convolution to multiplication ($\Im[f * g] = \Im[f]\Im[g]$), ring representations map ring multiplication and ring addition to matrix multiplication and addition.

There are many ways to view a matrix. It can be, and typically is, viewed as a transformation whose eigenvalues could be adjoined to a ring if they were not present. Alternatively, because $p(\mathrm{M}) = 0$ (where p is the characteristic polynomial of M), the matrix can itself be viewed as an element that can be adjoined because matrices are roots of their characteristic polynomial. Let R be any ring. The mapping to adjoin a matrix M to the ring R obtaining the ring $R[\mathrm{M}]$ is as follows: $R[\mathrm{M}] = u_0 I + u_1 \mathrm{M} + u_2 \mathrm{M}^2 + \cdots + u_n \mathrm{M}^n$ for all $u_0, u_1, u_2, \ldots, u_n \in R$.

Each element $u_0 I + u_1 \mathrm{M} + u_2 \mathrm{M}^2 + \cdots + u_n \mathrm{M}^n$ in $R[\mathrm{M}]$ is a ring representation of the element $u_0 + u_1\lambda + u_2\lambda^2 + \cdots + u_n\lambda^n$ in $R[\lambda]$.

For a concrete example, consider the complex numbers, which form a ring as they satisfy all of the properties above. To cast the complex numbers as ring representations over the real numbers, consider the characteristic polynomial of the complex numbers, i.e., $p = x^2 + 1 = x^2 + 0x + 1 = 0$. In order to form a ring representation, we are interested in adjoining a matrix whose eigenvalues are the roots of p. The companion matrix of this polynomial, as discussed above, will provide just such a matrix. The companion matrix is

$$J = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

Indeed,

$$\det(J - \lambda I) = \det\left(\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} - \lambda I\right) = \det\left(\begin{bmatrix} -\lambda & 1 \\ -1 & -\lambda \end{bmatrix}\right) = \lambda^2 + 1, \lambda = \pm i.$$

We can now define a ring representation of the complex numbers as $rep(u_0 + u_1 i) = u_0 I + u_1 J$ for all u_0, $u_1 \in R$. More concretely, we have

$$rep(u_0 + u_1 i) = \begin{bmatrix} u_0 & u_1 \\ -u_1 & u_0 \end{bmatrix}.$$

This construction preserves addition. Indeed,

$$\begin{aligned} rep(u_0 + u_1 i) + rep(v_0 + v_1 i) &= \begin{bmatrix} u_0 & u_1 \\ -u_1 & u_0 \end{bmatrix} + \begin{bmatrix} v_0 & v_1 \\ -v_1 & v_0 \end{bmatrix} \\ &= \begin{bmatrix} u_0 + v_0 & u_1 + v_1 \\ -(u_1 + v_1) & u_0 + v_0 \end{bmatrix} \\ &= rep((u_0 + v_0) + (u_1 + v_1)i). \end{aligned}$$

Also, the construction preserves multiplication as

$$\begin{aligned} rep(u_0 + u_1 i) rep(v_0 + v_1 i) &= \begin{bmatrix} u_0 & u_1 \\ -u_1 & u_0 \end{bmatrix}\begin{bmatrix} v_0 & v_1 \\ -v_1 & v_0 \end{bmatrix} \\ &= \begin{bmatrix} u_0 v_0 - u_1 v_1 & u_0 v_1 + u_1 v_0 \\ -(u_0 v_1 + u_1 v_0) & u_0 v_0 - u_1 v_1 \end{bmatrix} \\ &= rep((u_0 v_0 - u_1 v_1) + (u_0 v_1 + u_1 v_0)i) \\ &= rep((u_0 + u_1 i)(v_0 + v_1 i)). \end{aligned}$$

Remarkably, this process can be generalized to include the roots of several multivariate polynomials.

MOTIVATION FOR USING RING REPRESENTATIONS

As stated in the *Companion Matrices* section, one interpretation of the companion matrix is as an answer to the inverse eigenvalue problem. Namely, given a polynomial p, find a matrix M such that p is the characteristic polynomial of M. We will show that ring representations are a multivariate generalization of companion matrices. In this section, we will motivate discussion of Emiris' algorithm for finding ring representations with an example.

Consider the following system of equations:

$$\begin{aligned} f_1 &= x^2 + y^2 - 10 = 0 \\ f_2 &= x^2 + xy + 2y^2 - 16 = 0. \end{aligned}$$

Let R now be the real numbers. These above equations are elements in the polynomial ring $R[x, y]$. If we could find ring representations X and Y for x and y, respectively, in a ring where $f_1 = f_2 = 0$, the following would be true by the properties of ring representations:

$$\begin{aligned} f_1 &= X^2 + Y^2 - 10I = 0 \\ f_2 &= X^2 + XY + 2Y^2 - 16I = 0. \end{aligned}$$

Furthermore, because the ring defined by the equations is commutative, X and Y would necessarily commute. As a result, if X and Y are nondefective (have distinct eigenvalues), then they are simultaneously diagonalizable (they can be diagonalized with the same basis or, equivalently, they share all of their eigenvectors). Consequently, the diagonal of the representations must be the coordinates of the roots. If V is the matrix of eigenvectors for X and Y, then the statements above can be expressed symbolically as follows:

$$V^{-1}(X^2 + Y^2 - 10I)V = 0$$

$$V^{-1}(X^2)V + V^{-1}(Y^2)V - V^{-1}(10I)V = 0$$

$$\begin{bmatrix} x_1 & 0 & 0 & 0 \\ 0 & x_2 & 0 & 0 \\ 0 & 0 & x_3 & 0 \\ 0 & 0 & 0 & x_4 \end{bmatrix}^2 + \begin{bmatrix} y_1 & 0 & 0 & 0 \\ 0 & y_2 & 0 & 0 \\ 0 & 0 & y_3 & 0 \\ 0 & 0 & 0 & y_4 \end{bmatrix}^2 - \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} x_1^2 + y_1^2 - 10 & 0 & 0 & 0 \\ 0 & x_2^2 + y_2^2 - 10 & 0 & 0 \\ 0 & 0 & x_3^2 + y_3^2 - 10 & 0 \\ 0 & 0 & 0 & x_4^2 + y_4^2 - 10 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Similarly,

$$V^{-1}(X^2 + XY + 2Y^2 - 16I)V =$$

$$\begin{bmatrix} x_1^2 + x_1y_1 + y_1^2 - 16 & 0 & 0 & 0 \\ 0 & x_2^2 + x_2y_2 + y_2^2 - 16 & 0 & 0 \\ 0 & 0 & x_3^2 + x_3y_3 + y_3^2 - 16 & 0 \\ 0 & 0 & 0 & x_4^2 + x_4y_4 + y_4^2 - 16 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Indeed, it turns out for this example that ring representations for x and y are as follows:

$$X = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 3 & 2 & 0 \end{bmatrix}, \quad Y = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 6 & 0 & 0 & -1 \\ 0 & 3 & -2 & 0 \end{bmatrix}.$$

It is easy to verify

$$f_1 = X^2 + Y^2 - 10I$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 3 & 2 & 0 \end{bmatrix}^2 + \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 6 & 0 & 0 & -1 \\ 0 & 3 & -2 & 0 \end{bmatrix}^2 + \begin{bmatrix} -10 & 0 & 0 & 0 \\ 0 & -10 & 0 & 0 \\ 0 & 0 & -10 & 0 \\ 0 & 0 & 0 & -10 \end{bmatrix}$$

$$= \begin{bmatrix} 4 & 0 & 0 & 1 \\ 0 & 7 & 2 & 0 \\ 0 & 3 & 2 & 0 \\ 12 & 0 & 0 & 5 \end{bmatrix} + \begin{bmatrix} 6 & 0 & 0 & -1 \\ 0 & 3 & -2 & 0 \\ 0 & -3 & 8 & 0 \\ -12 & 0 & 0 & 5 \end{bmatrix} + \begin{bmatrix} -10 & 0 & 0 & 0 \\ 0 & -10 & 0 & 0 \\ 0 & 0 & -10 & 0 \\ 0 & 0 & 0 & -10 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Similar calculations can be done for f_2. The eigenvalues for X and Y that should yield the coordinates of the roots of the polynomial system are $\lambda_X = (\pm\sqrt{8}, \pm 1)$ and $\lambda_Y = (\pm\sqrt{2}, \mp 3)$. Indeed,

$$\begin{aligned} (\pm\sqrt{8})^2 + (\pm\sqrt{2})^2 - 10 &= 0 \\ (\pm\sqrt{8})^2 + (\pm\sqrt{8})(\pm\sqrt{2}) + 2(\pm\sqrt{2})^2 - 16 &= 0 \\ (\pm 1)^2 + (\mp\sqrt{3})^2 - 10 &= 0 \\ (\pm 1)^2 + (\pm 1)(\mp 3) + 2(\mp 3)^2 - 16 &= 0\,. \end{aligned}$$

EMIRIS' ALGORITHM

We have established that, if we have a way of determining the ring representations for X and Y, we can solve the system of equations. Emiris' algorithm, a method to accomplish just that, is discussed notionally here. For a full proof and description see Ref. 1 or 2.

Given polynomials $f_1, f_2, \ldots, f_n$ each of degree $d_1, d_2, \ldots, d_n$, respectively, the algorithm is as follows:

1. Introduce an extra polynomial $f_0 = u_0 + u_1 x_1 + u_2 x_2 + \ldots + u_n x_n$, where $u_0, u_1, \ldots, u_n$ are arbitrarily chosen constants with at least one being non-zero
2. Consider $B = \{x_1^{i_1} x_2^{i_2} \cdots x_n^{i_n} \mid i_0 + i_1 \cdots + i_n = D,\ 0 \le i_0, i_1, \ldots, i_n \le D\}$, where $D = d_1 + d_2 + \ldots + d_n - n + 1$
3. Consider

$$\begin{aligned} S_n &= \{b \in B \mid x_n^{d_n} \text{ divides } b\} \\ S_{n-1} &= \{b \in B \mid x_{n-1}^{d_{n-1}} \text{ divides } b \text{ but } x_n^{d_n} \text{ does not}\} \\ S_{n-2} &= \{b \in B \mid \text{divides } b \text{ but } x_n^{d_n}, x_{n-1}^{d_{n-1}} \text{ do not}\} \\ &\vdots \\ S_1 &= \{b \in B \mid x_1^{d_1} \text{ divides } b \text{ but } x_n^{d_n}, x_{n-1}^{d_{n-1}}, \ldots, x_2^{d_2} \text{ do not}\} \\ S_0 &= \left\{ b \in B \mid b \notin \bigcup_{i=1}^{n} S_i \right\} \end{aligned}$$

4. Find M a linear transformation on the monomials in B such that

$$B(x_1, x_2, \ldots, x_n) = B(\bar{x}) = \begin{bmatrix} S_0(\bar{x}) \\ S_1(\bar{x}) \\ \vdots \\ S_n(\bar{x}) \end{bmatrix}$$

$$M \cdot B(\bar{x}) = M \begin{bmatrix} S_0(\bar{x}) \\ S_1(\bar{x}) \\ \vdots \\ S_n(\bar{x}) \end{bmatrix} = \begin{bmatrix} S_0(\bar{x}) f_0(\bar{x}) \\ (S_1(\bar{x}) / x_1^{d_1}) f_1(\bar{x}) \\ \vdots \\ (S_n(\bar{x}) / x_n^{d_n}) f_n(\bar{x}) \end{bmatrix}$$

5. Consider

$$Mv(p_1, p_2, \ldots, p_n) = M \cdot B(\bar{p}) = M \begin{bmatrix} S_0(\bar{p}) \\ S_1(\bar{p}) \\ \vdots \\ S_n(\bar{p}) \end{bmatrix} = \begin{bmatrix} S_0(\bar{p}) f_0(\bar{p}) \\ (S_1(\bar{p}) / p_1^{d_1}) f_1(\bar{p}) \\ \vdots \\ (S_n(\bar{p}) / p_n^{d_n}) f_n(\bar{p}) \end{bmatrix} = \begin{bmatrix} S_0(\bar{p}) f_0(\bar{p}) \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

6. Break M into blocks as follows:

$$M \cdot B(\bar{p}) = \left[\begin{array}{c|c} M_{00} & M_{01} \\ \hline M_{10} & M_{11} \end{array}\right] \left[\begin{array}{c} S_0(\bar{p}) \\ \hline S_1(\bar{p}) \\ \vdots \\ S_n(\bar{p}) \end{array}\right] = \left[\begin{array}{c} S_0(\bar{p}) f_0(\bar{p}) \\ \hline (S_1(\bar{p}) / p_1^{d_1}) f_1(\bar{p}) \\ \vdots \\ (S_n(\bar{p}) / p_n^{d_n}) f_n(\bar{p}) \end{array}\right] = \left[\begin{array}{c} S_0(\bar{p}) f_0(\bar{p}) \\ \hline 0 \\ \vdots \\ 0 \end{array}\right]$$

7. Block triangularlize with the following transformation:

$$\left[\begin{array}{c|c} I & -M_{01}M_{11}^{-1} \\ \hline 0 & I \end{array}\right] \left[\begin{array}{c|c} M_{00} & M_{01} \\ \hline M_{10} & M_{11} \end{array}\right] \left[\begin{array}{c} S_0(\bar{p}) \\ \hline S_1(\bar{p}) \\ \vdots \\ S_n(\bar{p}) \end{array}\right] = \left[\begin{array}{c|c} \tilde{M} & 0 \\ \hline M_{10} & M_{11} \end{array}\right] = \left[\begin{array}{c} S_0(\bar{p}) \\ \hline S_1(\bar{p}) \\ \vdots \\ S_n(\bar{p}) \end{array}\right] = \begin{bmatrix} I & -M_{01}M_{11}^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} f_0(\bar{p}) S_0(\bar{p}) \\ 0 \end{bmatrix} = \begin{bmatrix} f_0(\bar{p}) S_0(\bar{p}) \\ 0 \end{bmatrix}$$

8. As a result of the above, $\tilde{M} = \tilde{M}(u_1, u_2, \ldots, u_n) = M_{00} - M_{01}M_{11}^{-1}M_{10}$
9. Solve the following eigenvector problem:
   $\tilde{M}S_0(\bar{p}) = f_0(\bar{p})S_0(\bar{p}) = (u_0 + u_1p_1 + \cdots + u_np_n)\, S_0(p_1, p_2, \ldots, p_n)$
10. Since $S_0(p_1, p_2, \ldots, p_n) = [\alpha, \alpha p_1, \alpha p_2, \ldots, \alpha p_n, \alpha p_1 p_2, \alpha p_1 p_3, \ldots]^T$ (because it will be derived numerically), extracting the root coordinates can be done by dividing the eigenvector by α and collecting the 2nd through nth entries

The roots of the polynomial system are the coordinates $(p_1, p_2, \ldots, p_n)$.

APPLICATION

One application for Emiris' algorithm is trilateration. Trilateration is achieved by measuring the time difference of arrival between pairs of sensors. The so-called "TDOA" (time difference of arrival) equation is $\|x - S1\| - \|x - S2\| = \pm TDOA * C$, where S1 and S2 are sensors, x is an emitter position, *TDOA* is the time difference of arrival, and C is the emission propagation rate. It is well known that this can be represented as a polynomial (indeed, it is hyperbolic). Thus, the polynomial that describes the level surface above is given by

$$\left\|x - S1\right\| - \left\|x - S2\right\| = \pm TDOA * C$$
$$\left\|x + [(S1 + S2)/2 + (S1 - S2)/2]\right\| - \left\|x + [(S1 + S2)/2 - (S1 - S2)/2]\right\| = \pm TDOA * C$$

Let

$$\begin{aligned} u &= (S1 + S2)/2 \\ v &= (S1 - S2)/2 \\ y &= x + u \\ TDOA * C &= 2\delta \end{aligned}$$

The equation above becomes

$$\begin{aligned} \left\|y + v\right\| - \left\|y - v\right\| &= \pm 2\delta \\ \left\|y + v\right\| &= \pm 2\delta + \left\|y - v\right\| \\ \left\|y + v\right\|^2 &= 4\delta^2 \pm 4\delta\left\|y - v\right\| + \left\|y - v\right\|^2 \\ y^T y + 2y^T v + v^T v &= 4\delta^2 \pm 4\delta\left\|y - v\right\| + y^T y - 2y^T v + v^T v \\ 4y \cdot v &= 4\delta^2 \pm 4\delta\left\|y - v\right\| \\ y \cdot v - \delta^2 &= \pm\delta\left\|y - v\right\| \\ (y^T v)^2 - 2\delta^2 y^T v + \delta^4 &= \delta^2\left\|y - v\right\|^2 \\ (y^T v)^2 - 2\delta^2 y^T v + \delta^4 &= \delta^2 y^T y - 2\delta^2 y^T v + \delta^2 v^T v \\ (y^T v)^2 + \delta^4 &= \delta^2 y^T y + \delta^2 v^T v \\ y^T v v^T y + \delta^4 &= \delta^2 y^T y + \delta^2 v^T v \\ y^T (vv^T - \delta^2 I) y - \delta^2 v^T v + \delta^4 &= 0 \end{aligned}$$

As an example, let's consider the problem of locating a gunshot within a city. Suppose there are three acoustic sensors (S1, S2, and S3) at (9, 39), (65, 10), and (64, 71), respectively, in a local Cartesian coordinate system with meters as the distance unit. Suppose further that a gunshot is heard by S1 at 19:19:57.0875, by S2 at 19:19:57.1719, and by S3 at 19:19:57.1797, and the speed of sound (C) is 341 m/s. Also suppose the unknown emitter location is at (27, 42) and that it emitted at 19:19:57:0340 (see Fig. 1).

The time difference of arrival between S1 and S2 is 0.0844 and between S1 and S3 it is 0.0078. Hence, we wish to solve the system

$$\sqrt{(x-9)^2 + (y-39)^2} - \sqrt{(x-65)^2 + (y-10)^2} = \pm 0.0844 * 341$$
$$\sqrt{(x-9)^2 + (y-39)^2} - \sqrt{(x-64)^2 + (y-71)^2} = \pm 0.0078 * 341.$$

Equivalently, we solve

$$-207485.6417 - 19846.0765x + 31843.375y + 537.0281x^2 - 812xy - 36.7219y^2 = 0$$
$$2480777.5687 - 88508.5223x - 37529.9994y + 549.4318x^2 + 880xy + 49.1818y^2 = 0.$$

To apply Emiris' algorithm, we pick $f_0 = u_0 + u_1 x + u_2 y = 8 + 28x + 80y$ and calculate the ring representation $F_0 = u_0 + u_1 X + u_2 Y = 8 + 28X + 80Y$:

$$F_0 = \begin{bmatrix} 8 & 28 & 80 & 0 \\ -48618.3988 & 2548.0324 & -112.9542 & 84.5799 \\ -2483454.438 & 62895.7242 & 64660.2744 & -1549.6064 \\ -73249499.44 & 1969279.3923 & 1768850.5654 & -43682.6736 \end{bmatrix}.$$

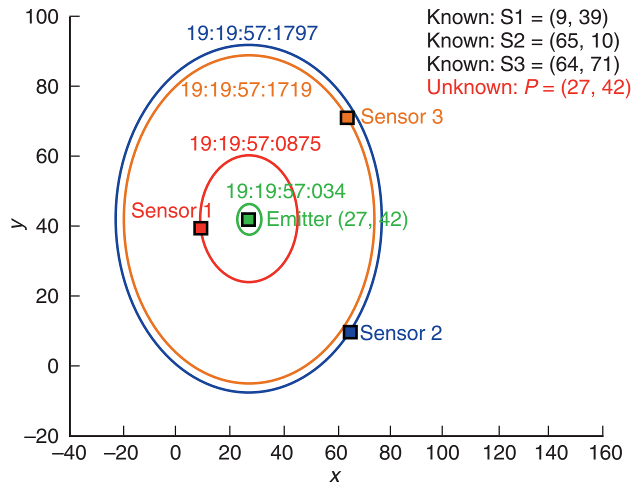


Figure 1. Emitter and sensor locations.

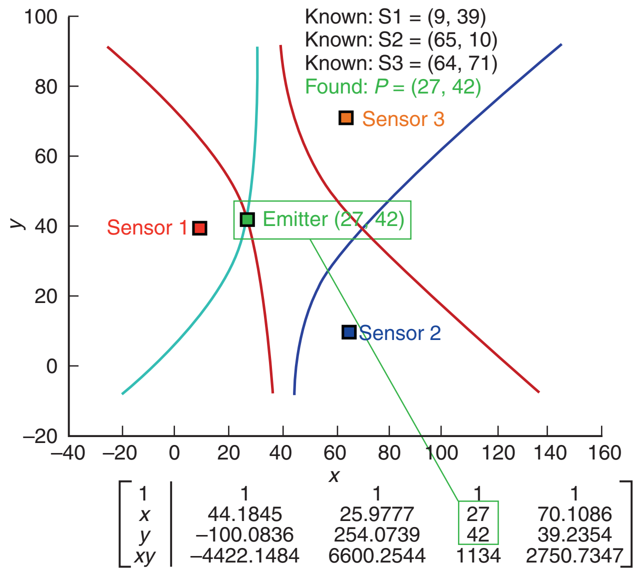


Figure 2. Emitter and sensor locations and eigenvector solutions.

The eigenvectors of F_0 are

$$V = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 25.9777 & 44.1845 & 27 & 70.1086 \\ 254.0739 & -100.0836 & 42 & 39.2354 \\ 6600.2544 & -4422.1484 & 1134 & 2750.7347 \end{bmatrix}.$$

The second and third rows are the solutions, and we have recovered the emitter location at (27, 42), as seen in Fig. 2.

CONCLUSION

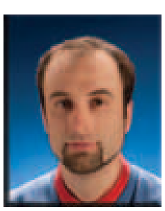
We have discussed some eigenvector methods for finding the roots of multivariate polynomials. Unlike iterative, numerical methods typically applied to this problem, the methods outlined in this article possess the numerical stability of numerical linear algebra, do not require a good initial guess of the solution, and give all solutions simultaneously. Furthermore, if the initial guess is poor enough, the methods outlined herein may converge more quickly than iterative methods.

REFERENCES

[1]Emiris, I. Z., "On the Complexity of Sparse Elimination," *J. Complexity* **12**, 134–166 (1996).

[2]Cox, D., Little, J., and O'Shea, D., *Using Algebraic Geometry*, Graduate Texts in Mathematics Series, Vol. 185, Springer, New York, 2nd Ed., pp. 122–128 (2005).

The Author



Michael P. Williams

Michael Peretzian Williams received his B.S. in Mathematics in 1998 from the University of South Carolina. He earned his M.S. in Mathematics in 2001, again from the University of South Carolina, focusing on Number Theory. He received his Ph.D. in Mathematics in 2004 from North Carolina State University, working in Lie Algebras. From 2005 to 2007, Dr. Williams worked as a researcher for Northrop Grumman, developing algorithms for Dempster–Shafer fusion. He has been with APL's Global Engagement Department since 2007, and his current work is in upstream data fusion. His e-mail address is michael.williams@jhuapl.edu.