Interval iterative algorithm for computing the piecewise algebraic variety

Renhong Wang, Xiaolei Zhang

Abstract

As the set of the common zeros of the multivariate splines, the piecewise algebraic variety is a kind of generalization of the classical algebraic variety. In this paper, we discuss the computation problem of the piecewise algebraic variety. The approach presented here is the interval iterative algorithm by introducing the concept of $\varepsilon$-deviation solutions. Meanwhile, we present a simple method to evaluate the bound on the value of the derivative of the function on a given region.

Keywords: Piecewise algebraic variety; Interval iterative algorithm; Krawczyk algorithm; $\varepsilon$-deviation solutions; Derivative matrix

1. Introduction

A fundamental problem in CAGD is the intersection problem. However, most of the curves and surfaces in CAGD are piecewise algebraic curves and surfaces. Hence, how to compute intersection problem of the piecewise algebraic curves and surfaces becomes an important problem in CAGD. Indeed, this problem boils down to the computation problem of the piecewise algebraic variety.

The primary work on the computation of the piecewise algebraic variety is as follows. Wang et al. [1] presented the resultant algorithm for the intersection points of piecewise algebraic curves by the structure of bivariate spline functions.

In recent years, the interval iterative methods are successfully applied to solve nonlinear equations (cf. [2,3] and the references therein). However, how to apply the interval iterative method to the computation of piecewise algebraic varieties becomes an important problem. Grandine [4] first presented a method for finding all the real roots of a given univariate spline function by using interval Newton method. In this paper, we try to compute the real roots of a given piecewise algebraic variety by using interval iterative method with the introduction of a new concept of $\varepsilon$-deviation solutions. Meanwhile, we present an efficient and simple algorithm to evaluate the bound on the value of the derivative of the function on a given region.
The rest of this paper is organized as follows: The interval iterative method and piecewise algebraic variety are reviewed in Sections 2 and 3, respectively. In Section 4, the concept of ε-deviation solutions is introduced. In Section 5, an efficient algorithm is presented to evaluate the derivative matrix $F'(X)$. The interval iterative algorithm for computing the piecewise algebraic variety is given in Section 6. Lastly, a numerical example is provided to illustrate the flexibility of the proposed algorithm.

2. Interval iterative algorithm

An interval iterative algorithm for nonlinear systems has been introduced by Moore in [2], developed and modified in [5–7] etc.

We introduce some notations following [3]. For an interval $[a, b]$, define $w([a, b]) = b - a, m([a, b]) = (a + b)/2$. For an interval vector $X = (X_1, \ldots, X_n)$, define $m(X) = (m(X_1), \ldots, m(X_n))$. For an interval matrix $A$, define $\|A\| = \max_i \sum_{j=1}^n |a_{ij}|$, where $a_{ij}$ is the $i$th row and the $j$th column element (interval) of $A$.

Now, we recall the Moore form of the Krawczyk algorithm and use it in the later sections.

Let $f(x) = 0$ be a system of $n$ nonlinear equations in $n$ variables. Moore’s interval Newton algorithm is defined by:

$$N(X) = m(X) - V(X)f(m(X))$$

where $V(X)$ is an interval matrix containing the inversion of $[f'(x)]^{-1}$ for all $x$ in $X$.

Krawczyk has proposed another interval version of Newton’s method which does not require the inversion of interval matrix. The Krawczyk algorithm has the form:

$$K(X) = y - Yf(y) + (I - YF'(X))(X - y),$$

where $y$ is chosen from the region $X$, $F(X)$ is the inclusion monotone extension of $f(x)$, and $Y$ is an arbitrary nonsingular matrix.

In particular, if $y$ and $Y$ are chosen to be $y = m(X)$ and $Y = [m(F'(X))]^{-1}$ respectively, then the Moore form of the Krawczyk algorithm is:

$$K(X) = m(X) - [m(F'(X))]^{-1}f(m(X)) + (I - [m(F'(X))]^{-1}F'(X))(X - m(X)).$$

If $X$ contains a solution of $f(x) = 0$, then so does $K(X)$. Thus, it is the basis for the algorithm of Krawczyk:

$$X^{(k+1)} = X^{(k)} \cap K(X^{(k)}).$$

For the existence of solutions to nonlinear equations, Moore proved the following results:

**Theorem 2.1 ([6]).** If the two conditions

$$K(X^{(0)}) \subseteq X^{(0)}, \quad \text{and} \quad r_0 = \|I - YF'(X^{(0)})\| < 1$$

are satisfied simultaneously, then there exists a unique solution $x$ to $f(x) = 0$ in $X^{(0)}$ and the sequence $\{X^{(k)}\}_{k=0}^\infty$ converges at least linearly to $x$.

In fact, Moore further proved that the second condition was not essential with respect to this algorithm. Hence, this theorem still holds after deleting it [3].

3. Piecewise algebraic variety

As the set of the common zeros of the multivariate splines, the piecewise algebraic variety is a kind of generalization of the classical algebraic variety. Some fundamental definitions and properties of the piecewise algebraic variety have been discussed in papers [8–11].

Let $\mathbb{R}$ be a real number field, $\mathbb{R}^n = \{(a_1, a_2, \ldots, a_n) \mid a_i \in \mathbb{R}, i = 1, \ldots, n\}$ be an $n$-dimensional affine space over $\mathbb{R}$. Using finite number of hyperplanes in $\mathbb{R}^n$, we divide a region $D \subseteq \mathbb{R}^n$ into a finite number of simply connected regions, which are called the partition cells. Denote by $\Delta$ the partition of the region $D$, which is the union of all
Given a sufficiently small positive number \( \varepsilon \), the point \( K \) is called a \( \varepsilon \)-deviation solution if it satisfies the following three conditions:

1. \( K \subseteq X^\ast \)
2. \( w(X^\ast_i) < \varepsilon, \quad i = 1, \ldots, n \)
3. The point \( m(X^\ast) \in \delta \)

are satisfied simultaneously, then \( m(X^\ast) \) is an \( \varepsilon \)-deviation solution with respect to \( \delta \).

Obviously, the \( \varepsilon \)-deviation solution is a kind of generalization of the ordinary solution to nonlinear equations. In the following example, we will illustrate the advantage of \( \varepsilon \)-deviation solutions which greatly enables the interval iterative algorithm to solve a given algebraic variety.
Example 4.1. Consider the common real zeros of two algebraic curves \( f(x, y) = x^2 + y^2 - 1 = 0 \) and \( g(x, y) = x^2 - y = 0 \) on the triangle \( T = [V_1 V_2 V_3] \), where \( V_1 = (1.4041853, 0) \), \( V_2 = (0, 1.4041853) \) and \( V_3 = (1, 1) \).

Its exact solution on \( T \) is \( z(f, g) = \{(\sqrt{3}/2, \sqrt{3}/2)\} \). However, if we use float arithmetic to compute the solution, then we have \( z(f, g) = \{(0.786151, 0.618033)\} \notin T \). Therefore, two algebraic curves \( f \) and \( g \) have no common solution on the triangle \( T \), which is a contradiction.

Now, we apply the Krawczyk algorithm (1) to compute the solutions. Specify \( \varepsilon = 0.0025 \) and \( X^{(0)} = ([0.7853, 0.7870], [0.6168, 0.6193]) \), we have

\[
K(X^{(0)}) = ([0.78615, 0.786153], [0.618031, 0.618037]) \subset X^0,
\]

and

\[
m(X^{(0)}) = (0.78615, 0.61805) \in T.
\]

Therefore, there is a solution in \( X^{(0)} \). Moreover, \( m(X^{(0)}) \) is considered to be an \( \varepsilon \)-deviation solution of \( f \) and \( g \) with respect to \( T \).

Remark 4.1. The smaller the \( \varepsilon \) it takes, the more computational cost it requires. Fortunately, the numerical solutions are not sensitive to the choice of \( \varepsilon \) from the numerical experiments we tested. The general guidelines on how big \( \varepsilon \) should be taken in practice are not discussed.

5. Evaluation of \( F'(X) \)

One problem in the Krawczyk algorithm is of computing the value bounds of the derivative of the function. The crude bound will lead to slow convergence, whereas intervals with sharper bound on the value of the derivatives will lead to better performance. Grandine [4] applied the knot insertion algorithm to give the explicit bound of the derivative of a univariate spline function over an interval. For the general multivariate splines, we design a relatively convenient algorithm to evaluate \( F'(X) \).

Let \( X = z(f_1, \ldots, f_n) \), \( f_i(x_1, \ldots, x_n) \in S^\mu(\Delta) \) be a zero-dimensional piecewise algebraic variety. Suppose \( X^{(0)} \) is an arbitrary \( n \)-dimensional rectangle in the simply connected region \( D \) and there exist cells \( \delta_1, \ldots, \delta_s \) satisfying that \( \delta_i \cap X^{(0)} \) is non-empty. If we introduce the following notations

\[
m_{ijk} = \min_{\delta_i \cap X^{(0)}} \frac{\partial f_i}{\partial x_k}, \quad M_{ijk} = \max_{\delta_i \cap X^{(0)}} \frac{\partial f_i}{\partial x_k},
\]

\[
i = 1, \ldots, n; \quad j = 1, \ldots, s; \quad k = 1, \ldots, n,
\]

where \( f_{ij} = f_i|_{\delta_j} \) denotes the representation of \( f_i(x, y) \) on the cell \( \delta_j \), then the bounds of the derivative of \( f_i, i = 1, \ldots, n \) are given by

\[
\frac{\partial f_i}{\partial x_k}(X^{(0)}) = [m_{ik}, M_{ik}], \quad i = 1, \ldots, n; \quad k = 1, \ldots, n,
\]

where \( m_{ik} = \min_{1 \leq j \leq s} m_{ijk} \) and \( M_{ik} = \max_{1 \leq j \leq s} M_{ijk} \).

Therefore, \( F'(X^{(0)}) \) can be written in the form

\[
F'(X^{(0)}) = \begin{bmatrix}
[m_{11}, M_{11}] & \cdots & [m_{1n}, M_{1n}] \\
\vdots & \ddots & \vdots \\
[m_{n1}, M_{n1}] & \cdots & [m_{nn}, M_{nn}]
\end{bmatrix}.
\]

Next, we will analyze how to compute \( m_{ijk} (M_{ijk}) \) efficiently.

If the region \( \delta_j \cap X^{(0)} \) consists of \( r_j \) hyperplanes \( L_{jr}(x_1, \ldots, x_n) = 0, \ r = 1, \ldots, r_j \), then the minimum value of \( \frac{\partial f_i}{\partial x_k} \) on the region \( \delta_j \cap X^{(0)} \) can be determined by solving the following optimal problem:

\[
\min_{\delta_j \cap X^{(0)}} \frac{\partial f_i}{\partial x_k}(x_1, \ldots, x_n)
\]
subject to
\[ L_j r(x_1, \ldots, x_n) \geq 0, \quad r = 1, \ldots, r. \]
Here, \( L_j r(P) \) is assumed to be greater than zero when \( P \) locates in the interior of \( \delta_j \cap X^{(0)} \).

Of course, the maximum value of \( \frac{\partial f_j}{\partial x_k} \) on the region \( \delta_j \cap X^{(0)} \) can be determined similarly. Since it is complex to solve the nonlinear optimal problem in practice, we give another much simpler method to obtain the minimum (maximum) value.

Let \( R_j^{(0)} = ([mx_{j_1}^{(0)}, Mx_{j_1}^{(0)}], \ldots, [mx_{j_n}^{(0)}, Mx_{j_n}^{(0)}]) \) be the minimal \( n \)-dimensional rectangle containing \( \delta_j \cap X^{(0)} \).

We turn to compute the minimum and maximum values on a relatively larger region \( R_j^{(0)} \). Obviously, the polynomials \( \frac{\partial f_j}{\partial x_k} \) can be written in tensor Bézier form:

\[
\frac{\partial f_j}{\partial x_k}(x_1, \ldots, x_n) = \sum_{l_1=0}^{d_{i_1}^{j}} \cdots \sum_{l_n=0}^{d_{i_n}^{j}} a_{l_1, \ldots, l_n}^{i_1, \ldots, i_n} B_{l_1,1}^{i_1} \cdots B_{l_n,1}^{i_n} (t_1) \cdots B_{l_n,d_n}^{i_n} (t_n), \quad x \in R_j^{(0)}
\]

under the coordinate transformation

\[
t_1 = \frac{x_1 - mx_{j_1}^{(0)}}{Mx_{j_1}^{(0)} - mx_{j_1}^{(0)}}, \quad \ldots, \quad t_n = \frac{x_n - mx_{j_n}^{(0)}}{Mx_{j_n}^{(0)} - mx_{j_n}^{(0)}}, \quad t_1, \ldots, t_n \in [0, 1],
\]

where \( d_{i_k}^{j_1}, \ldots, d_{i_n}^{j_n} \) are the degrees of \( \frac{\partial f_j}{\partial x_k} \) with respect to \( x_1, \ldots, x_n \) and \( B_{l_1,1}^{i_1} \cdots B_{l_n,1}^{i_n} (t_1) \cdots B_{l_n,d_n}^{i_n} (t_n) \) are the Bernstein polynomials.

Therefore, the minimum and maximum values of \( \frac{\partial f_j}{\partial x_k} \) on the region \( \delta_j \cap X^{(0)} \) are taken to be

\[
m_{ij} = \min_{0 \leq l_1 \leq d_{i_1}^{j_1}, \ldots, 0 \leq l_n \leq d_{i_n}^{j_n}} a_{l_1,\ldots,l_n}^{i_1,\ldots,i_n}, \quad M_{ij} = \max_{0 \leq l_1 \leq d_{i_1}^{j_1}, \ldots, 0 \leq l_n \leq d_{i_n}^{j_n}} a_{l_1,\ldots,l_n}^{i_1,\ldots,i_n},
\]

respectively.

Obviously, \([m_{ij}, M_{ij}] \subseteq [\tilde{m}_{ij}, \tilde{M}_{ij}]\). That is to say, we obtain relatively larger bounds, which will slow down the convergence rate to some extent. Compared to the complexity of solving the optimal problems, it deserves to use our proposed approach.

6. Algorithm

With the above preparations, we give the interval iterative method for computing a given piecewise algebraic variety.

**Proposition 6.1.** With the above notations, if \( K(X^{(0)}) \cap X^{(0)} = \emptyset \), then \( z(f_1, \ldots, f_n) \) has no solution on \( X^{(0)} \).

It tells us that we can determine that there is no solution quickly on a relatively larger region without requiring to compute on every smaller region separately.

**Algorithm 6.1 (Interval Iterative Method for a Given Piecewise Algebraic Variety).** Input: A zero-dimensional piecewise algebraic variety \( z(f_1, \ldots, f_n) \), \( f_i(x_1, \ldots, x_n) \in S^i(\Delta), i = 1, \ldots, n \).

Output: The set \( P \) of \( \varepsilon \)-deviation solutions.

Step 1. Specify a tolerance \( \varepsilon \) and set \( P = \emptyset \).

Step 2. Set the current interval vector \( X^{(0)} \).

Step 3. Compute the interval matrix \( F'(X^{(0)}) \) and the matrix \( [m(F'(X^{(0)}))]^{-1} \).

Step 4. Compute \( K(X^{(0)}) \) according to the Krawczyk algorithm (1).

Step 5. If \( X^{(0)} \) is an \( \varepsilon \)-deviation solution, then set \( P := P \cup m(X^{(0)}) \) and stop; If \( K(X^{(0)}) \cap X^{(0)} = \emptyset \), then stop; Otherwise, bisect \( X^{(1)} = K(X^{(0)}) \cap X^{(0)} \), replace \( X^{(0)} := X_k^{(1)}, k = 1, \ldots, r \) where \( r \) is the number of subinterval vectors \( X_k^{(1)} \) of \( X^{(1)} \) satisfying \( X_k^{(1)} \cap D \neq \emptyset \), and go to Step 2.
Fig. 1. Two piecewise algebraic curves \( f = 0 \) and \( g = 0 \).

We can easily see that the algorithm must terminate after a finite number of steps.

Most work of this interval iterative algorithm occurs in Step 3, where the interval matrix \( F'(X^{(0)}) \) is required to be evaluated. Moreover, if the determinant of \( m(F'(X^{(0)})) \) is zero, then we bisect the initial interval vector \( X^{(0)} \) directly.

Remark 6.1. The bisection in the above algorithm might not be optimal. Take rectangular triangulation on \( \mathbb{R}^2 \) for example, subdivision along the partition lines ought to be optimal.

7. Numerical example

In this section, an example is given to illustrate that the proposed interval iterative algorithm for computing a given piecewise algebraic variety is flexible.

Example 7.1. Let \( \Delta = \{ \delta_1, \delta_2, \delta_3, \delta_4 \} \) be a triangulation of a quadrangle \( V_A V_B V_C V_D \) in \( \mathbb{R}^2 \), where \( \delta_1 = [V_D V_O V_A] \), \( \delta_2 = [V_D V_C V_O] \), \( \delta_3 = [V_C V_B V_O] \), \( \delta_4 = [V_O V_B V_A] \). \( V_A = (1.1577, 0) \), \( V_B = (0, -1) \), \( V_C = (-1, 0) \). \( V_D = (0, 1.1577) \) and \( V_O = (0, 0) \) (see Fig. 1).

Suppose that \( f, \ g \in S^1_{\Delta}(\Delta) \) and

- on cell \( \delta_1 \):
  \[
  f_1(x, y) = f|_{\delta_1} = x^3 + y^2 - \frac{5}{8} \\
  g_1(x, y) = g|_{\delta_1} = y^3 - x
  \]

- on cell \( \delta_2 \):
  \[
  f_2(x, y) = f|_{\delta_2} = f_1(x, y) + x^2(x + y) \\
  g_2(x, y) = g|_{\delta_2} = g_1(x, y) - x^2(2y)
  \]

- on cell \( \delta_3 \):
  \[
  f_3(x, y) = f|_{\delta_3} = f_2(x, y) + y^2(x - y + 2) \\
  g_3(x, y) = g|_{\delta_3} = g_2(x, y) - y^2(y - 3)
  \]

- on cell \( \delta_4 \):
  \[
  f_4(x, y) = f|_{\delta_4} = f_3(x, y) + y^2(x + y - 2) \\
  g_4(x, y) = g|_{\delta_4} = g_3(x, y) - y^2(y - 3)
  \]

Specify a tolerance \( \epsilon = 0.001 \) and take \( X^{(0)} \) to be \( X^{(0)} = (-1, 1.1577), (-1, 1.1577) \). Using Krawczyk algorithm (1), we can compute \( X^{(0)} \subseteq K(X^{(0)}) \). Hence, we subdivide the region \( X^{(0)} \) into four subregion \( \delta_1, \delta_2, \delta_3 \) and \( \delta_4 \) along \( x \)-axis and \( y \)-axis.

Take cell \( \delta_1 \) for example, we now take \( X^{(0)} = ([0, 1.1577], [0, 1.1577]) \) to be the minimal rectangle containing \( \delta_1 \). Using Krawczyk algorithm (1), we know that \( X^{(0)} \) belongs to \( K(X^{(0)}) = ([0.4379, 1.4984], [-0.0565, 1.5488]) \). Hence, we continue to subdivide the triangle \( \delta_1 \) into three subregions \( X_k^{(1)} \) satisfying \( X_k^{(1)} \cap \delta_1 \neq \emptyset \) and then repeat the similar procedure. After about one thousand iterations, we have the result that two piecewise algebraic curves \( f(x, y) = 0 \) and \( g(x, y) = 0 \) have an \( \epsilon \)-deviation solution (0.412980, 0.744692) with respect to the cell \( \delta_1 \).
Similarly, two piecewise algebraic curves \( f(x, y) = 0 \) and \( g(x, y) = 0 \) have another \( \varepsilon \)-deviation solution \((0.429801, -0.378506)\) with respect to the cell \( \delta_4 \).

From the numerical result, we can easily see that the interval iterative algorithm for computing the piecewise algebraic variety is feasible. However, computational procedure indicates that the proposed method for finding all the real solutions of the piecewise algebraic variety requires much computational cost. Therefore, how to construct a better interval iterative algorithm possessing faster convergence rate and the optimal choice of \( \varepsilon \) remain to be done in our future work.

Acknowledgement

We are very grateful to the referees for their careful reading of the original manuscript and many valuable comments, which greatly improved this paper.

References