Computing separating linear forms for Bivariate systems

Yacine Bouzidi

April 13, 2014

Abstract

A fundamental problem in computational geometry is the computation of the topology of a real algebraic plane curve given by its implicit equation, that is, the computation of a set of polygonal lines that approximates the curve while preserving its topology.

A critical step in many algorithms computing the topology of a plane curve is the computation of the set of singular and extreme points (wrt $x$) of this curve, which is equivalent to the computation of the solutions of bivariate systems defined by the curve and some of its partial derivatives.

In this presentation, we address the problem of computing a linear separating form of a system of two bivariate polynomials with integer coefficients, that is a linear combination of the variables that takes different values when evaluated at the distinct solutions of the system. In other words, a separating linear form defines a shear of the coordinate system that sends the algebraic system in generic position, in the sense that no two distinct solutions are vertically aligned. The computation of such linear forms is at the core of most algorithms that solve algebraic systems by computing rational parameterizations of the solutions and this is the bottleneck of these algorithms in terms of worst-case bit complexity.

We first present for this problem, an algorithm of worst-case bit complexity $\tilde{O}_B(d^8 + d^7\tau)$ where $d$ and $\tau$ denote respectively the maximum degree and bitsize of the input polynomials (and where $\tilde{O}$ refers to the complexity where polylogarithmic factors are omitted and $O_B$ refers to the bit complexity). This complexity improves by a factor $d^2$ the best known complexity bound for this problem (in $\tilde{O}_B(d^{10} + d^8\tau)$). Then, we show how, using a similar approach, we can obtain an algorithm that simplifies the previous one while decreasing its complexity to $\tilde{O}_B(d^7 + d^6\tau)$ bit operations in the worst-case. This new algorithm also yields a probabilistic Las-Vegas algorithm of expected bit complexity $\tilde{O}_B(d^5 + d^4\tau)$. 