The Art of Symbolic Computation

Erich Kaltofen
North Carolina State University, USA

Abstract

Symbolic Computation in the past 40 years has brought us remarkable theory: Berlekamp-Zassenhaus; Groebner; Risch; Gosper, Karr and WZ; cylindrical algebraic decomposition; sparse polynomial interpolation; LLL; Wiedemann and block Lanczos; matrix Pade; straight-line and black box polynomial calculus; baby-steps/giant-steps and black-box linear algebra and polynomial factorization; symbolic/numeric GCD, factorization and sparse interpolation; Tellegen's principle; sparse resultants; Giesbrecht/Mulders-Storjohann diophantine linear solvers; Sasaki/van Hoeij power sums and Bostan et al. logarithmic derivatives; fast bit complexity for linear algebra over the integers; over the integers; essentially optimal polynomial matrix inversion; Skew, Ore and differential polynomial factorization; Barvinok short rational functions and supersparse polynomial factorization; and many more.

The discipline has lead to remarkable software like Mathematica and Maple, which supply implementations of these algorithms to the masses. As it turned out, a killer application of computer algebra is high energy physics, where a special purpose computer algebra system, SCHOONSHIP, helped in work worthy of a Nobel Prize in physics in 1999.

In my talk I will attempt to describe what the descipline of Symbolic Computation is and what problems it tackles. In particular, I will discuss the use of heuristics (numerical, randomized, and algorithmic) that seem necessary to solve some of today's problems in geometric modeling and equation solving. Thus, we seem to have come full cycle (the discipline may have started in the 1960s at the MIT AI Lab), but with a twist that I shall explain.