Symbolic-Numeric Computation

Lihong Zhi
Chinese Academy of Sciences, China

30th Applications of Computer Algebra - ACA 2025

Symbolic-Numeric Computation (SNC) has become a central area within computer algebra, aiming to bridge the gap between exact symbolic methods and efficient numerical techniques. Its historical development reflects a broader trend in the evolution of computer algebra: the pursuit of algorithms that combine mathematical rigor with computational robustness.

The foundations of SNC were laid in the 1960s and 1970s, with the advent of early symbolic computation systems focused on exact algebraic manipulation. As computational demands increased in areas such as robotics, control theory, and computer-aided geometric design, the limitations of purely symbolic or purely numeric approaches became increasingly evident. The 1980s and 1990s marked a turning point, with foundational advances in approximate polynomial GCDs, symbolic-numeric factorization, and hybrid solvers for systems of equations.

In parallel, recent years have witnessed growing interest in the formalization of mathematics, driven by proof assistants such as Lean 4, which enable the precise encoding of mathematical definitions, theorems, and algorithms. While formal verification has traditionally been rooted in pure mathematics, its interaction with symbolic-numeric computation opens a promising new direction. By formalizing key concepts from computational algebraic geometry and numerical analysis, researchers can now verify both the correctness and the stability of hybrid algorithms, a critical step as such methods become increasingly sophisticated and indispensable in applications across optimization, scientific computing, and data science.

This talk will trace the historical development of symbolic-numeric computation, highlight our contributions to the field, and examine how formal systems like Lean 4 may help shape the future of computer algebra.