Mathematical Invention: How Much Can be Automated?

Bruno Buchberger
Research Institute for Symbolic Computation
Johannes Kepler University, Linz / Hagenberg, Austria

Inventing mathematical knowledge (theorems and methods), in interaction with proving and disproving, is the essence of mathematical creative activity. How much of this can be automated?

We argue that, paradoxically, automating the creative activity of mathematics is, ultimately, the goal of mathematics: Higher layers of creativity automate previous layers of creativity. The spiral of going through higher and higher levels of automation has no upper bound.

This view is intimately connected with teaching mathematics: Teaching is essentially explanation, i.e. "making seemingly difficult things plain". This is possible by deeper insight, "seeing and finding" the keys in seemingly complicated situations. A complete insight may typically lead to a method for inventing certain mathematical facts.

Automation of creative mathematical invention can be observed already in the early days of mathematics. It became more an more pronounced through the history of mathematics. Through the advent of computers, this automation is accelerating in present time. Teaching should follow this path from "creative" consideration of individual instances to an automated procedure.

In the talk, we will give a couple of examples of automating mathematical invention by (meta-)mathematical means. We start from simple algorithms that - in an appropriate perspective - may be viewed as automated invention methods. We proceed to considering developments in computer algebra of the past four decades, for example Wu's method, that may also be interpreted as - powerful - invention methods. Finally, we will explain two recent methods proposed by the speaker that allow to invent mathematical theorems and algorithms in the context of computer-supporting mathematical theory exploration. One of the methods is "scheme-based" invention that has a lot to do with the notion of "functors", the second method is a top-down method that systematizes the approach of "learning from failure".