Theorema:

Bruno Buchberger

Lecture for the students
of the DK “Computational Mathematics”
Johannes Kepler University

Theorema

Algorithms (on the “meta-level”) for

- invention (and “proof”) of definitions
- invention and proof of theorems
- invention of problems
- invention and proofs of algorithms
- organizing math knowledge

Mathematics:

Given a mathematical problem, whose solution, for each instance, at a given historical moment, needs human intelligence,

one strives at inventing, again by human intelligence, a general algorithm, that solves the problem in all infinitely many instances.

In other words, it is the essence of mathematics

to think once deeply (by “natural” intelligence) on a problem and its context

in order to replace thinking infinitely many times for solving each of the infinitely many problem instances individually by non-intelligent computation.

(In other words: Generate “artificial non-intelligence” by “natural intelligence”.)

Example:

At some stage, finding a shortest path in a graph needed an individual consideration for each graph.

At a next stage, somebody (Dijkstra) managed (and proved) one algorithm for finding a shortest path for all graphs.

Example:

At some stage, solving a Sudoku puzzle needed an individual consideration for each puzzle.

At a next stage, somebody (BB, …) managed (and proved) one algorithm for finding a solution for all Sudoku puzzles (e.g. by the Gröbner bases method).

Example:

At some stage, finding the integral (in “closed form”) of a function (expression) needed an individual consideration for each function.

At a next stage, somebody (Risch) managed (and proved) one algorithm for finding a solution for all functions (out of a particular class).

Mathematics **is**

automation of mathematical invention on some level

by a mathematical invention on a higher level.

In other words,

it is the goal of mathematics

to trivialize itself.

For making math easy (“non-intelligent”)
on one level
one must do more difficult (“intelligent”) math
on a higher level.

Discussion:
Is this always true?
Can general invention on higher level be easier
than the individual inventions on the lower level?

The principle of mathematics can be iterated.

Mathematics is a hierarchy of “intelligent” inventions:

One invention on a higher level
avoids infinitely many inventions on a lower level.

Historically,

this hierarchy went through
a couple of amazing rounds

full of emotion, effort, surprises, and excitement.

(With increasing speed.)

There is no upper bound
to the rounds of automating mathematical invention.

(Gödel’s Incompleteness Theorem
can be “felt” by all
who attempt to add a next round,)

In This Talk:
Automation of the Invention of Algebraic Algorithms

Is (considered to be) a typical AI problem.

(In my terminology, a problem of generating "artificial non-intelligence by natural intelligence")

"The Algorithm Synthesis Problem":

- Given: The specification of a problem.
- Find: An algorithm that solves the problem.

The rest of the talk will be an exposition

of my recent “lazy thinking” method
for automated algorithm synthesis.

The “lazy thinking” method mimics two strategies of human mathematicians for inventing (theorems and) algorithms:

- use general principles of algorithm design (which we cast in the form of “algorithm schemes”)
- analyze and derive inspiration from unsuccessful proofs.

We first explain the “lazy thinking” method in a simple example (sorting).

Then we show that the “lazy thinking” method has the potential of inventing non-trivial algorithms.

(Example: construct an algorithm for the “Gröbner bases” problem, which was open for 65 years.)

Conclusion

Artificial Intelligence =

artificial un-intelligence on the object level
by natural intelligence on the meta-level.

Don’t wait for artificial intelligence.

Rather, invest your natural intelligence
for generating artificial un-intelligence.

Future education:

Based on deep insight and intuition about nature

train the intellect

which is expressed in clear thinking and speaking.

Apply intuition and intellect

on two levels:

Think deeply on the meta-level
for automating the object level.

And amazing things will happen.

Discussion:

Must invention on the meta-level
necessarily be more difficult
than invention on the object level?

Another important principle
of math problem solving:

In order to solve a difficult problem
make it more difficult.