Logic, Mathematics, Computer Science:

The Accumulated Thinking Technology of Mankind

Bruno Buchberger
RISC
Talk at LMCS 2002
"Logic, Mathematics, Computer Science: Interactions"
Castle of Hagenberg, October 22, 2002

Copyright Note: This file may be copied under the following conditions:
- The file must be kept unchanged and must include this copyright note.
- A mail should be sent to buchberger@risc.uni-linz.ac.at
- If material in this file is used, the talk should be cited appropriately.

The Current Situation

- **Some people believe that ...**
  - mathematics has to do with computing,
  - mathematics is logical,
  - computers have to do with computing,
- **and, hence, that ...**

mathematics, logic, and computer science are intimately related.

- **However,**
  - many mathematicians are proud that they never touch a computer and are happy that they are not touched by the computer,
  - many mathematicians are also proud and happy that they do not need "formal" logic,
  - many computer scientists are proud and happy that they do not need any significant mathematics.
- **In fact, ...**

the gap between logic, mathematics, and computer science widens instead of becoming smaller.

- **I deeply deplore this.**
  - The gap between the three fields is theoretically and practically bad for all three fields.
  - There is no theoretical reason for this gap. (Except that computer science also has a "hardware" aspect.)

- **Psychologic and Sociologic reasons for the separation:**

  Each of the three fields is difficult. It seems next to impossible to "master" all three of them.

  Many mathematicians, notably "pure" mathematicians, look down on computer mathematics and computer science because they believe that computer math / science just is "very fast repetition of trivial mathematics on dull machines".

  Many mathematicians don't see the necessity of formal / logical quality because the addressees of their work, typically, are other (hopefully very intelligent) mathematicians.

  Logic talks *about* the language of mathematics. Logicians don't bother to make their work useful for the working mathematicians, who talk *in* the language of mathematics.

  Computer scientists don't "see" mathematics in their field because they normally don't "see" integrals anywhere.

Mathematics: The Art of Trivialization

Observing and Reasoning

What is Mathematics?

First Answer.

The theory of numbers, of geometrical figures, of functions, of graphs, ..., i.e. of certain classes of abstract objects.

Second Answer.

Mathematics: Gaining insight and solving problems by reasoning.

Mathematics can not be characterized by the objects of study but only by the method.

In contrast: natural sciences gain insight by observing.

At the Beginning, Mathematics was / is Observing

Language and Reality: Names and Objects

names	objects
zero	
one	●
two	● ●
three	● ● ●
four	● ● ● ●
five	● ● ● ● ●

Language and Reality: Names and Operations

one more ● ● ⟶ ● ● ●

one more ● ● ● ⟶ ● ● ● ●

add ● ● ● ● ● ⟶ ● ● ● ● ●

equal ● ● ● ● ⟶ true

equal ● ● ● ● ● ⟶ false

equal ● ● ● ● ● ● ● ● ● ● ⟶ true

equal ● ● ● ● ● ● ● ⟶ false

▫ Gaining Insight by Observing: First Example

"Two and three added is equal to five"

This statement need not be proved. Rather it can be observed.

two and three added ● ● ● ● ● ⟶ ● ● ● ● ●

five ● ● ● ● ●

In this moment, this statement is a statement of "physics", not of mathematics.

▫ Gaining Insight by Observing: Second Example

"Two and three added is equal to three and two added"

This statement need not be proved. Rather it can be observed.

two and three added ● ● ● ● ● ⟶ ● ● ● ● ●

three and two added ● ● ● ● ● ⟶ ● ● ● ● ●

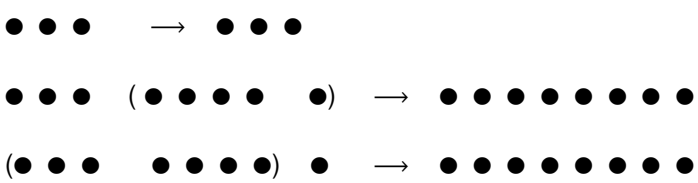
■ "True" Mathematics Starts When Reasoning Comes into Play

▫ The Motivation for Reasoning Stems from (Potential) Infinity

When we want to gain insight / solve problems
for (potentially) infinitely many situations,
we need reasoning
(in a language in which certain finitary language constructs - "variables" - can cover infinitely many situations):

General Definition of Addition

```
Definition["addition", any[m, n],
    add[m, zero] = m                                "base"
    add[m, one–more[n]] = one–more[add[m, n]]   "general"]
```



Derivation of Known Statements

In the finitely many cases we already considered
we can check that
the statements observed
can be derived, by pure reasoning, from the above two statements:

```
Compute[
  add[one–more[one–more[one–more[zero]]],
    one–more[one–more[zero]]],
  using → Definition["addition"]]
```

```
one–more[one–more[one–more[one–more[one–more[zero]]]]]
```

Now,
we may stipulate that "adding" described by the two statements above
is the operation "adding" which we "have in mind".
(Note that we can never prove this! This is a kind of "ubiquitous" Church thesis!!!)

Derivation of New Statements

```
Compute[
  add[one–more[one–more[one–more[one–more[one–more[one–more[zero]]]]]],
    one–more[one–more[one–more[one–more[one–more[one–more[one–more[one–more[zero]]]]]]]]],
  using → Definition["addition"]]
```

```
one–more[one–more[one–more[one–more[one–more[
        one–more[one–more[one–more[one–more[one–more[one–more[one–more[one–more[zero]]]]]]]]]]]]]
```

Objection

Non-formal mathematics is much easier than "formal" mathematics! ? ?

The Miracle of Mathematics

Gaining Interesting Insight on Infinitely Many Situations by Reasoning

Now, by pure reasoning, without any further observation, we can gain the insight

```
Proposition["commutativity of addition", any[m, n],
        add[m, n] = add[n, m]   "com" ]
```

Proof: By nested induction on m and n. Exercise! (The proof can also be generated, completely automatically, by the *Theorema* system, see the papers for downloading in the home page

http://www.risc.uni-linz.ac.at/people/buchberg/)

(When we allow induction, we implicitly stipulate much more knowledge on "zero" and "add" than the one shown explicitly above.)

Reasoning (Proving) and the Ideas for Propositions and Proofs

The activity of proving is (must be) purely formal.

The idea for propositions and proofs may well come from observations in concrete examples.

The Miracle of Mathematics

By finitely many reasoning steps,
(starting from statements involving variables that range over potentially infinitely many objects)
we may derive insight (statements) that holds for infinitely many situations!

In Other Words

The practical success of mathematics stems from the fact that,
by working (hard but finitely and abstractly), i.e. reasoning, once
one can subsequently be lazy infinitely many times in concrete situations.

In Other Words

The incredible power of mathematics stems from the fact that,
by working once on the language level (which is the meta-level of the object level)
one can gain insight / solve problems for infinitely many situations on the object level.

▫ This Could Conclude My Talk

The unfolding of mathematics is just a continuous repetition of this principle.

Hence, I could stop my talk here.

However, a principle needs sufficiently many examples and implications!

Also, this principle generates excitement that may last a life long.

■ Trivialization of Mathematics by Non-Trivial Mathematics

■ Example: Polynomials

▫ First Round

language	objects
x ((2 x - 5) x + 3 x)	real numbers

▫ A Problem

Find an x such that

x ((2 x - 5) x + 3 x) = 0.

Solution: By Reasoning (in a sub-language of predicate logic):

x = 0 or x = 0 or (2 x - 5) + 3 = 0,

x = 0 or x = 1.

▫ Second Round

language	objects
f, g, ... +, -, ...,	$x^5 + 3\,x^4 + 0\,x^3 + -2\,x^2 + \frac{2}{5}\,x^1 + 3\,x^0$
{ | },	(or any "isomorphic" representation)
' (differentiation)	

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Note

- a jump from object level to meta-level

- by non-trivial mathematics (analysis, ...) this problem is "trivialized".

- in some instances, the "old" method may still be more useful ("intelligent") than the new one.

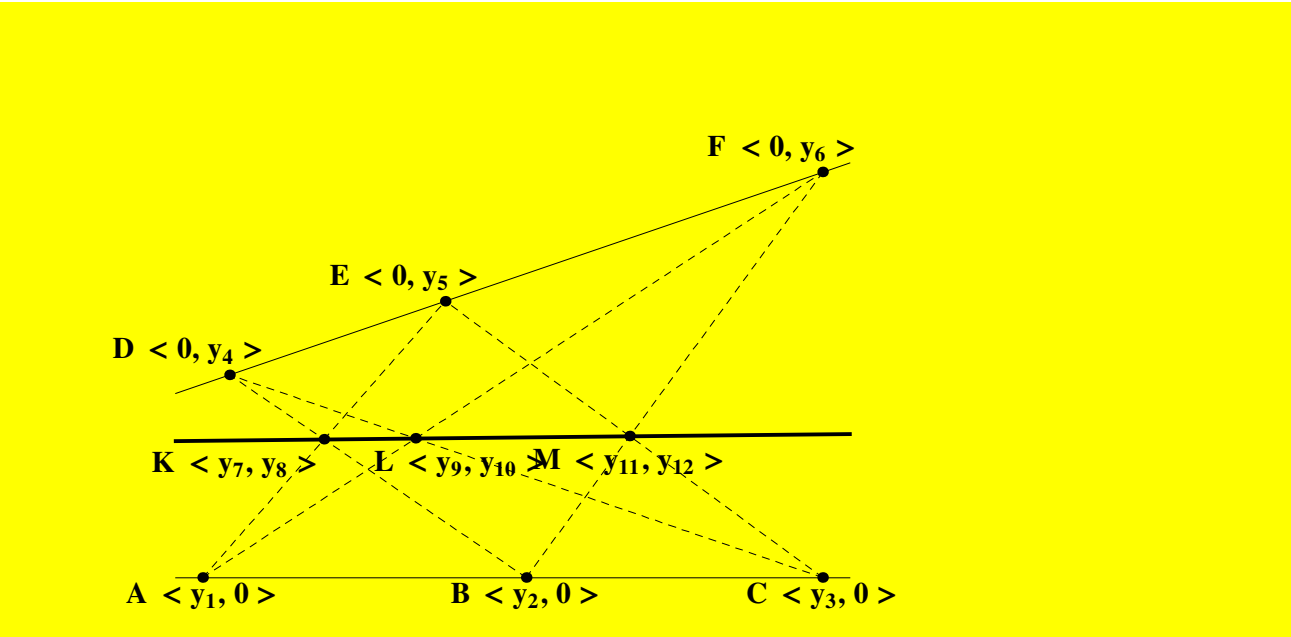
Example: Geometry

First Round

language objects

is–on–line[P,L] points and lines
....

if is–on–line[...] ∧ ...
then is–on–line[....]



A Question

Question is this always true?

Answer: by reasoning in the language of points and lines (predicate logic).

Second Round

Language Objects

about polynomials polynomials

With a translator (Rabinowich trick, proposed by Deepak Kapur) every question of the above kind can be translated into a question of membership in polynomial ideals.

For answering such questions we have algorithms (e.g. Gröbner bases).

▫ An Example Call of the Method (in the *Theorema* System)

```
Formula["Test", any[x, y],
    ( (x^2 y - 3 x ) ≠ 0) ⋁ ((x y + x + y) ≠ 0) ⋁                                    "B1"
        (((x^2 y + 3 x = 0) ⋁ ((-2 x^2 + -7 x y + x^2 y + x^3 y + -2 y^2 + -2 x y^2 + 2 x^2 y^2) = 0))
    ⋀ ((x^2 + -x y + x^2 y + -2 y^2 + -2 x y^2) = 0))                                ]
```

```
Prove[Formula["Test"], using → {}, by → GroebnerBasesProver]
```

```
- ProofObject -
```

By the above call to the *Theorema* system, the following proof is generated, completely automatically:

Prove:

(Formula (Test): B1)

$$\forall_{x,y} \; (((x^2 * y - 3 * x) \neq 0) \vee (x * y + x + y \neq 0) \vee ((x^2 * y + 3 * x = 0) \vee ((-2) * x^2 + (-7) * x * y + x^2 * y + x^3 * y + (-2) * y^2 + (-2) * x * y^2 + 2 * x^2 * y^2 = 0)) \wedge (x^2 + (-x) * y + x^2 * y + (-2) * y^2 + (-2) * x * y^2 = 0))$$

with no assumptions.

Proved.

The Theorem is proved by the Groebner Bases method.

The formula in the scope of the universal quantifier is transformed into an equivalent formula that is a conjunction of disjunctions of equalities and negated equalities. The universal quantifier can then be distributed over the individual parts of the conjunction. By this, we obtain:

Independent proof problems:

(Formula (Test): B1.1)

$$\forall_{x,y} \; ((x^2 + (-x * y) + x^2 * y + (-2) * y^2 + (-2) * x * y^2 = 0) \vee ((-3) * x + x^2 * y \neq 0) \vee (x + y + x * y \neq 0))$$

(Formula (Test): B1.2)

$$\forall_{x,y}\ ((3*x+x^2*y=0)\vee ((-2)*x^2+(-7)*x*y+x^2*y+x^3*y+(-2)*y^2+(-2)*x*y^2+2*x^2*y^2=0)\vee ((-3)*x+x^2*y\neq 0)\vee (x+y+x*y\neq 0))$$

We now prove the above individual problems separately:

Proof of (Formula (Test): B1.1):

This proof problem has the following structure:

(Formula (Test): B1.1.structure) $\forall_{x,y}\ ((\text{Poly}[1]\neq 0)\vee(\text{Poly}[2]\neq 0)\vee(\text{Poly}[3]=0))$,

where

$$\begin{aligned}\text{Poly}[1] &= (-3)*x+x^2*y\\ \text{Poly}[2] &= x+y+x*y\\ \text{Poly}[3] &= x^2+(-x*y)+x^2*y+(-2)*y^2+(-2)*x*y^2\end{aligned}$$

(Formula (Test): B1.1.structure) is equivalent to

(Formula (Test): B1.1.implication) $\forall_{x,y}\ ((\text{Poly}[1]=0)\wedge(\text{Poly}[2]=0)\Rightarrow(\text{Poly}[3]=0))$.

(Formula (Test): B1.1.implication) is equivalent to

(Formula (Test): B1.1.not-exists) $\nexists_{x,y}\ (((\text{Poly}[1]=0)\wedge(\text{Poly}[2]=0))\wedge(\text{Poly}[3]\neq 0))$.

By introducing the slack variable(s)

$\{\xi\}$

(Formula (Test): B1.1.not-exists) is transformed into the equivalent formula

(Formula (Test): B1.1.not-exists-slack)

$$\nexists_{x,y,\xi}\ (((\text{Poly}[1]=0)\wedge(\text{Poly}[2]=0))\wedge((-1)+\xi\,\text{Poly}[3]=0))$$.

Hence, we see that the proof problem is transformed into the question on whether or not a system of polynomial equations has a solution or not. This question can be answered by checking whether or not the (reduced) Groebner basis of

$\{\text{Poly}[1], \text{Poly}[2], (-1)+\xi\,\text{Poly}[3]\}$

is exactly $\{1\}$.

Hence, we compute the Groebner basis for the following polynomial list:

$\{(-1) + x^2\,\xi + -x\,y\,\xi + x^2\,y\,\xi + -2\,y^2\,\xi + -2\,x\,y^2\,\xi,\ -3\,x + x^2\,y,\ x + y + x\,y\}$

The Groebner basis:

$\{1\}$

Hence, (Formula (Test): B1.1) is proved.

Proof of (Formula (Test): B1.2):

This proof problem has the following structure:

(Formula (Test): B1.2.structure) $\forall_{x,y}\ ((\text{Poly}[1] \neq 0) \vee (\text{Poly}[2] \neq 0) \vee (\text{Poly}[3] = 0) \vee (\text{Poly}[4] = 0))$,

where

$\text{Poly}[1] = (-3) * x + x^2 * y$
$\text{Poly}[2] = x + y + x * y$
$\text{Poly}[3] = 3 * x + x^2 * y$
$\text{Poly}[4] = (-2) * x^2 + (-7) * x * y + x^2 * y + x^3 * y + (-2) * y^2 + (-2) * x * y^2 + 2 * x^2 * y^2$

(Formula (Test): B1.2.structure) is equivalent to

(Formula (Test): B1.2.implication)

$\forall_{x,y}\ ((\text{Poly}[1] = 0) \wedge (\text{Poly}[2] = 0) \Rightarrow (\text{Poly}[3] = 0) \vee (\text{Poly}[4] = 0))$.

(Formula (Test): B1.2.implication) is equivalent to

(Formula (Test): B1.2.not-exists)

$\nexists_{x,y}\ (((\text{Poly}[1] = 0) \wedge (\text{Poly}[2] = 0)) \wedge ((\text{Poly}[3] \neq 0) \wedge (\text{Poly}[4] \neq 0)))$.

By introducing the slack variable(s)

$\{\xi 1, \xi 2\}$

(Formula (Test): B1.2.not-exists) is transformed into the equivalent formula

(Formula (Test): B1.2.not-exists-slack)

$\nexists_{x,y,\xi 1,\xi 2}\ (((\text{Poly}[1] = 0) \wedge (\text{Poly}[2] = 0)) \wedge \{(-1) + \xi 1\ \text{Poly}[3] = 0,\ (-1) + \xi 2\ \text{Poly}[4] = 0\})$.

Hence, we see that the proof problem is transformed into the question on whether or not a system of polynomial equations has a solution or not. This question can be answered by checking whether or not the (reduced) Groebner basis of

$\{\text{Poly}[1], \text{Poly}[2], (-1) + \xi 1\ \text{Poly}[3], (-1) + \xi 2\ \text{Poly}[4]\}$

is exactly {1}.

Hence, we compute the Groebner basis for the following polynomial list:

$$\{(-1) + 3\,x\,\xi 1 + x^2\,y\,\xi 1,\ (-1) + -2\,x^2\,\xi 2 + -7\,x\,y\,\xi 2 + x^2\,y\,\xi 2 + x^3\,y\,\xi 2 + -2\,y^2\,\xi 2 + -2\,x\,y^2\,\xi 2 + 2\,x^2\,y^2\,\xi 2,\ -3\,x + x^2\,y,\ x + y + x\,y\}$$

The Groebner basis:

{1}

Hence, (Formula (Test): B1.2) is proved.

Since all of the individual subtheorems are proved, the original formula is proved.

PhD thesis by J. Robu (2002): Implementation of this (and other) method(s) for geo theorem proving with translation to the geo language and graphical illustration of theorems.

▫ Note

- a jump from object level to meta-level

- by non-trivial mathematics (Gröbner bases theory, ...) this problem is "trivialized".

- in some instances, the "old" method may still be more useful ("intelligent") than the new one.

■ Example: Analysis

▫ First Round

language	objects
limit[f,a] :⇔ ...	real numbers, sequences, functions

....

limit[f,a]∧limit[g,b]⇒limit[f+g,a+b]

Question is this always true?

Answer: by reasoning in the language of real numbers, sequences, functions, ...
(predicate logic)

▫ Second Round

Language	Objects
about formulae	formulae with alternating quantifiers (and definitions)

With a (heuristic) proving algorithm ("PCS", BB 2000) many "prove" problems of the above kind can be proved by reduction to a "solve" problem.

An Example Call of the Method (in the *Theorema* System)

```
Definition["limit:", any[f, a],
    limit[f, a] ⟺ ∀_{ϵ>0} ∃_N ∀_{n≥N} |f[n] − a| < ϵ]
```

```
Proposition["limit of sum", any[f, a, g, b],
    (limit[f, a] ∧ limit[g, b])  ⇒  limit[f + g, a + b]]
```

```
Definition["+:", any[f, g, x],
    (f + g)[x] = f[x] + g[x]]
```

```
Lemma["|+|", any[x, y, a, b, δ, ϵ],
    (|(x + y) − (a + b)| < (δ + ϵ))  ⟸  (|x − a| < δ ∧ |y − b| < ϵ)]
```

```
Lemma["max", any[m, M1, M2],
    m ≥ max[M1, M2]  ⇒  (m ≥ M1 ∧ m ≥ M2)]
```

```
Theory["limit",
    Definition["limit:"]
    Definition["+:"]
    Lemma["|+|"]
    Lemma["max"]
]
```

```
Prove[Proposition["limit of sum"], using → Theory["limit"], by → PCS]
```

```
- ProofObject -
```

By the above call to the *Theorema* system, the following proof is generated completely automatically:

Prove:

(Proposition (limit of sum)) $\forall_{f,a,g,b} (\text{limit}[f, a] \wedge \text{limit}[g, b] \Rightarrow \text{limit}[f + g, a + b])$,

under the assumptions:

(Definition (limit:)) $\forall_{f,a} \left(\text{limit}[f, a] \Leftrightarrow \forall_{\epsilon>0} \exists_{N} \forall_{n \geq N} (|f[n] - a| < \epsilon) \right)$,

(Definition (+:)) $\underset{f,g,x}{\forall}((f+g)[x]=f[x]+g[x])$,

(Lemma (|+|)) $\underset{x,y,a,b,\delta,\epsilon}{\forall}(|(x+y)-(a+b)|<\delta+\epsilon \Leftarrow (|x-a|<\delta \wedge |y-b|<\epsilon))$,

(Lemma (max)) $\underset{m,M1,M2}{\forall}(m \geq \max[M1, M2] \Rightarrow m \geq M1 \wedge m \geq M2)$.

We assume

(1) $\text{limit}[f_0, a_0] \wedge \text{limit}[g_0, b_0]$,

and show

(2) $\text{limit}[f_0 + g_0, a_0 + b_0]$.

Formula (1.1), by (Definition (limit:)), implies:

(3) $\underset{\epsilon>0}{\forall}\underset{N}{\exists}\underset{n\geq N}{\forall}(|f_0[n]-a_0|<\epsilon)$.

By (3), we can take an appropriate Skolem function such that

(4) $\underset{\epsilon>0}{\forall}\underset{n\geq N_0[\epsilon]}{\forall}(|f_0[n]-a_0|<\epsilon)$,

Formula (1.2), by (Definition (limit:)), implies:

(5) $\underset{\epsilon>0}{\forall}\underset{N}{\exists}\underset{n\geq N}{\forall}(|g_0[n]-b_0|<\epsilon)$.

By (5), we can take an appropriate Skolem function such that

(6) $\underset{\epsilon>0}{\forall}\underset{n\geq N_1[\epsilon]}{\forall}(|g_0[n]-b_0|<\epsilon)$,

Formula (2), using (Definition (limit:)), is implied by:

(7) $\underset{\epsilon>0}{\forall}\underset{N}{\exists}\underset{n\geq N}{\forall}(|(f_0+g_0)[n]-(a_0+b_0)|<\epsilon)$.

We assume

(8) $\epsilon_0 > 0$,

and show

(9) $\underset{N}{\exists}\underset{n\geq N}{\forall}(|(f_0+g_0)[n]-(a_0+b_0)|<\epsilon_0)$.

We have to find N_2^* such that

(10) $\underset{n}{\forall}(n \geq N_2^* \Rightarrow |(f_0+g_0)[n]-(a_0+b_0)|<\epsilon_0)$.

Formula (10), using (Definition (+:)), is implied by:

(11) $\forall_n \, (n \geq N_2^* \Rightarrow |(f_0[n] + g_0[n]) - (a_0 + b_0)| < \epsilon_0)$.

Formula (11), using (Lemma (|+|)), is implied by:

(12) $\exists_{\substack{\delta,\epsilon \\ \delta+\epsilon=\epsilon_0}} \; \forall_n \, (n \geq N_2^* \Rightarrow |f_0[n] - a_0| < \delta \wedge |g_0[n] - b_0| < \epsilon)$.

We have to find δ_0^*, ϵ_1^*, and N_2^* such that

(13) $(\delta_0^* + \epsilon_1^* = \epsilon_0) \bigwedge \forall_n \, (n \geq N_2^* \Rightarrow |f_0[n] - a_0| < \delta_0^* \wedge |g_0[n] - b_0| < \epsilon_1^*)$.

Formula (13), using (6), is implied by:

$(\delta_0^* + \epsilon_1^* = \epsilon_0) \bigwedge \forall_n \, (n \geq N_2^* \Rightarrow \epsilon_1^* > 0 \wedge n \geq N_1[\epsilon_1^*] \wedge |f_0[n] - a_0| < \delta_0^*)$,

which, using (4), is implied by:

$(\delta_0^* + \epsilon_1^* = \epsilon_0) \bigwedge \forall_n \, (n \geq N_2^* \Rightarrow \delta_0^* > 0 \wedge \epsilon_1^* > 0 \wedge n \geq N_0[\delta_0^*] \wedge n \geq N_1[\epsilon_1^*])$,

which, using (Lemma (max)), is implied by:

(14) $(\delta_0^* + \epsilon_1^* = \epsilon_0) \bigwedge \forall_n \, (n \geq N_2^* \Rightarrow \delta_0^* > 0 \wedge \epsilon_1^* > 0 \wedge n \geq \max[N_0[\delta_0^*], N_1[\epsilon_1^*]])$.

Formula (14) is implied by

(15) $(\delta_0^* + \epsilon_1^* = \epsilon_0) \bigwedge \delta_0^* > 0 \bigwedge \epsilon_1^* > 0 \bigwedge \forall_n \, (n \geq N_2^* \Rightarrow n \geq \max[N_0[\delta_0^*], N_1[\epsilon_1^*]])$.

Partially solving it, formula (15) is implied by

(16) $(\delta_0^* + \epsilon_1^* = \epsilon_0) \wedge \delta_0^* > 0 \wedge \epsilon_1^* > 0 \wedge (N_2^* = \max[N_0[\delta_0^*], N_1[\epsilon_1^*]])$.

Now,

$(\delta_0^* + \epsilon_1^* = \epsilon_0) \wedge \delta_0^* > 0 \wedge \epsilon_1^* > 0$

can be solved for δ_0^* and ϵ_1^* by a call to Collins cad–method yielding a sample solution

$\delta_0^* \leftarrow \frac{\epsilon_0}{2}$,

$\epsilon_1^* \leftarrow \frac{\epsilon_0}{2}$.

Furthermore, we can immediately solve

$N_2^* = \max[N_0[\delta_0^*], N_1[\epsilon_1^*]]$

for N_2^* by taking

$N_2^* \leftarrow \max[N_0[\frac{\epsilon_0}{2}], N_1[\frac{\epsilon_0}{2}]]$.

Hence formula (16) is solved, and we are done.

Note

- a jump from object level to meta-level

- by (not completely) trivial mathematics (a little bit of systematic thinking plus a powerful solver for inequalities over the reals) this problem is "trivialized".

- in some instances, the "old" method may still be more useful ("intelligent") than the new one.

The Race Between Humans and Computers

Is Computer Mathematics Better Than Human Mathematics?

This question is based on a profound misunderstanding!

What Can the Computer Do Mathematicians Cannot Do?

This question is based on a profound misunderstanding!

The Simple Truth

As we have seen in the example, there is no race between "the computer" and "humans"

but, rather, for each problem P (with infinitely many "instances" P_n)

- a human H_n can solve P_n with an ad hoc method M_n
- or, alternatively, a human H may try to invent one method M that can solve P_n, for all n.

Computer Mathematics

In some (lucky) cases, an algorithmic method M may be found for P.

In such a case, a computer can be used for executing the method.

In such a case, "area P" of mathematics is "trivialized".

Non-Trivial Mathematics for Trivializing Mathematics

Non-Algorithmic Mathematics

In many cases, only non-algorithmic methods M can be found for a problem P.

In such a case, "area *P*" of mathematics is not completely "trivialized".

However, still, there is a distinct difference between a uniform method *M* for *P* or a spectrum of ad-hoc methods M_n for some of the instances P_n.

All of mathematics, as a tendency, is striving for uniform methods for problems!

Non-Trivial Mathematics

Typically, non-trivial mathematics is necessary for trivializing an area of mathematics.

Typically, the mathematics necessary for treating an area *P* is the more non-trivial the more trivialization (automation, computerization) is attempted.

Unfortunately, some / many pure mathematicians do not understand this simple but important point.

Unfortunately, some / many applied mathematicians / computer scientists do not understand this simple but important point.

The Limitations of Trivialization

There are Limits

It is known (by non-trivial mathematics) that certain problems *P* cannot be trivialized in the strict sense.

There are No Limits

However, there are no limits to trivialization (automation) because more and more problems (versions of problems) can be trivialized with better and better methods needing stronger and stronger mathematics.

This fact is a matter of experience but it also can be proved (by non-trivial mathematics): A kind of concrete version of Gödel's incompleteness theorem.

Also, "methods" is much more than only "algorithmic methods". (Compare: Computability w.r.t. to "oracles").

There are No Limits

What will be achieved will always be an infinitely small part in comparison to what is still untouched.

(Thus, fortunately, the jobs of mathematicians can never be killed by the trivialization of mathematics on which we work diligently.)

Some Views

A Uniform View of "Thinking Technology"

- Mathematics
- with logic as its (permanent) "jump to the meta-level" and
- computers as its ultimate trivialization measure

is the coherent thinking technology of mankind that

- evolved from the pre-stage in which observing, thinking, (and acting) was not separated
- through a conscious separation between observing, thinking (reasoning, proving), and acting
- and will reach higher and higher degrees of (relative) automation by jumps to higher and higher meta-levels
- with more and more excitement.

Implications for the Curricula

Mathematics, logic, computer science expand more and more into the breadth.

Of course, it is impossible to study, teach, cover "all of math / logic / computer science" in a life-time (and neither in 1000 life-times).

However, in some way, it becomes easier and easier to see and train the uniform thinking technology behind mathematics, logic, and computer science.

Thus, this uniform thinking technology should form the uniform and stable nucleus of our math and computer science curricula.

Standing on this firm basis, we may pursue the dramatic expansion of knowledge in math and computer science in a conscious but relaxed way.

The Role of Mathematics in Society

Mathematics is not (only) an accumulation of abstract knowledge on abstract notions, which a normal human will never need and never encounter in her life.

Rather, the thinking technology of mathematics is a treasure of mankind that has bearing, relevance, and benefits for everyone:

- It is the central source of all innovation in our technology-based society.

- It is the thinking culture of a truly democratic society.

Thus, we must devote some part of our time to make this clear to the public.

"Don't lean back and be angry that the public does not appreciate our work."

■ The Increasing Speed of the Technology Spiral

The speed of the technology spiral is increasing.

This is a natural consequence of the technique of "self-application" (jump from the object level to the meta-level) of math-based science and technology.

■ Beyond Technology

Although there is no limit to the advancements of math-based science and technology, we must ask ourselves "what comes next?", "is there something in life beyond technologies?", etc.

There are only personal, individual answers possible.

My own personal view on the past and future of society:

From an unconscious nature-based society
through a technology-based society
to a conscious nature-based society.