Towards a SymbolicComputational Philosophy (and Methodology!) for Mathematics

Doron ZEILBERGER [1]

Dedicated to Bruno Buchberger, on the occasion of his 5!/2!*-th Birthday*

One of the most *profound* mathematical concepts is the **diagonal** (cf. Pythagors, Cantor, Gödel, Turing). The format of a diagonal element is AA, but *my* favorite element is BB. My favorite movie-star used to be *Brigit Bardot*, my favorite playwright is *Bertolt Brecht*, my favorite Sesame Street character is *Big Bird*, and my favorite *mathematician, computer scientist, logician, philosopher, pedagogue, administrator, clarinet-player*, and *human-being* is: **Bruno Buchberger** (henceforth **BB**).

Math and science are *emergent* phenomena, and a new field is created *spontaneously* by the efforts of many people. So it is over-simplistic to talk about the 'founder' of a field. But, at least by one *criterion*, **BB** founded **SC** (Symbolic Computation), since he created **JSC** (Journal of Symbolic Computation).

By the way, **SC=CA=CF**, where **CA** stands for *Computer Algebra*, and **CF** is the French name, *Calcul Formel.*

Definition of CA (c. 1980): "CA is the part of CS which designs, analyses, implements, and applies, algebraic algorithms".

This definition was given by Rüdiger Loos[L] in the seminal volume edited by BB, Collins and Loos[BCL]. As you can see, the beginning was rather modest, and CA only claimed to be a tiny part of computer science.

A better definition was given recently by **BB** himself [B].

Definition of SC (2002): "The part of math that can be expressed by quantifier-free predicate-logic is the **SC** part of math".

But, according to me, **SC** is a *primitive*, *fundamental* entity, and it is *math* that needs a definition. So here is my own 2050 definition of math.

Definition of Math (2050): **Math:=SC**.

Let's pause to talk about the first word of the phrase **Symbolic Computation**, i.e. on *Symbolic*. Symbolism is as old as humanity, and is all around us, in art, science, religion, and of course,

[1] Department of Mathematics, Rutgers University (New Brunswick), Hill Center-Busch Campus, 110 Frelinghuysen Rd., Piscataway, NJ 08854-8019, USA. `zeilberg@math.rutgers.edu`
`http://www.math.rutgers.edu/~zeilberg/` . This version: Feb. 20, 2003. First Version: Dec. 12, 2002. Edited version of the transcript of an invited talk given at the Bruno Buchberger Symposium, LMCS2002, RISC-Linz, Hagenberg castle, Hagenberg, Austria, Oct. 22, 2002, 12:15-13:05. Supported in part by the NSF.

language. Here is what Alfred North Whitehead had to say about *language*, in his delightful little book [Wh].

"Language is such a symbolism... The word is a symbol, and its **meaning** *is constituted by the ideas, images, and emotions which it raises in the mind of the hearer...".*
"There is another sort of language, purely a written language, which is constituted by the mathematical symbols of the science of algebra".

Whitehead then goes on to say that in *algebra* the *meaning* is irrelevant, since the *symbols do the reasoning for you.*

It is this *magical* property of algebra, that is today amplified million-fold by *computer algebra*, that makes *humans*, those *semantical* creatures, so ambivalent about it.

Bordeaux 1991

1991 was a very good vintage for combinatorics, since at that year the historic 3rd 'Formal Power Series and Algebraic Combinatorics' conference took place. It was historic because amongst the five invited speakers, one was a Bourbaki (Pierre Cartier), while another one was a software developer (Gaston Gonnet). The three other invited speakers were Gilbert Labelle, of species fame, Asymptotics Guru Phillipe Flajolet, and myself. In my talk I made the famous statement: **EXTREME UGLINESS IS BEAUTIFUL.**

It was made in defense of the proofs generated by my beloved electronic servant, Shalosh B. Ekhad, that to the uninitiated human look very unmotivated and ugly. I claimed that it was a new art form, and it is exactly their 'inelegance' that made them so elegant.

The above sentence falls under the *paradigm*: **Extreme X is the Opposite of X**.

This *symbolic sentence* gives rise, by specializing **X**, to many 'profound statements'. E.g. try: **X**=Simplicity, Kindness, Love, Hate, Modesty, Fame, ...

The *slogan* of the present talk is: **Extreme Abstraction is Concrete.**

And this is made possible by *algebra*, and especially by *computer algebra.*

The most salient feature of mathematics is its *abstraction.* This was made explicit in Tim Gowers's fascinating recent booklet [G], where he described the 'abstract method' and epitomized it by the slogan 'A mathematical object is what it does'.

A Short History of Abstraction

Once upon a time there were *three* bears, *three* lions, *three* apples. All of which were designated by 3 strokes on the cave's wall, and from this was born the very abstract *concept* 'three', denoted by the *symbol* 3.

Thus the statement $2 + 3 = 3 + 2$ is really a deep theorem, containing infinitely many facts: 'two bears and three bears is the same as three bears and two bears', 'two lions and three lions is the same as three lions and two lions', 'two apples and three apples is the same as three apples and two apples' , etc. Then humans discovered other such deep theorems: $4+7 = 7+4$, $5+8 = 8+5$, ...

Much much later, came another leap. All these theorems were recognized as special cases of just *one*

Theorem: Let a and b be *arbitrary* integers, then $a + b = b + a$.

Here a and b are *symbols* that *symbolize* concrete numbers. As such this theorem requires a proof.

Proof: Since $a+0 = 0+a$, this is true for $b = 0$. Next let's prove this for $b = 1$. $a+1 = ((a-1)+1)+1 = (1+(a-1))+1 = 1+((a-1)+1) = 1+a$. Now, $a+b = a+((b-1)+1) = (a+(b-1))+1$. By induction, this equals $((b-1)+a)+1$, which, equals $(b-1)+(a+1) = (b-1)+(1+a) = b+(-1+1)+a = b+0+a = b+a$.

Traditionally, abstract symbols stand for more concrete objects, and they do have *meanings*.

In Symbolic Computation, a and b stand for *themselves*. In Maple, `type(a, symbol);` and `type(b,symbol);` are `true`. So $b + a := a + b$ by *fiat*, and it does not require proof. As our **Birthday Boy** said recently ([B]):

"Math can be viewed as a network of meta-theories: A theorem on a meta-level may 'trivialize' the invention/proof of many theorems in the object level".

Every time we abstract from one level to the next meta-level we trade a **pound of Semantics for an ounce of Syntax**.

Here is a parable. The infinitely many theorems $(1+2)^2 = 1^2+2*1*2+2^2$, $(2+3)^2 = 2^2+2*2*3+3^2$, ..., can be abstracted to the general 'theorem' $(a + b)^2 = a^2 + 2ab + b^2$, that if you think of a and b as symbols standing for numbers, requires a proof using the *axioms* of algebra. But if you encapsulate these 'axioms' into combinatorial rewriting rules (`expand` in Maple), and since there is a *canonical form* algorithm, this is now a purely routine fact, of the same epistemological stature as $2 + 2 = 4$.

In the same vein, $(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$ is purely routine, as well as the binomial theorem for $(a + b)^n$, for any specific n, although these theorems get *deeper and deeper* (i.e. requiring more time and memory) as n gets bigger, and if you use `expand` to prove it for $n = 10^{100}$, Maple will run out of time and memory.

Until 1988, the **Binomial Theorem**

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k} \quad ,$$

for arbitrary n, was considered a genuine *theorem*. But thanks to **Zeilberger's algorithm**, which is part of **WZ theory**, it is now completely routine, since both sides have the *canonical form* $N-(a+b), 1$. Here I used the shorthand for describing so-called holonomic sequences by giving the operator annihilating them (where N is the shift operator), followed by the "initial conditions".

Similarly, using the multisum procedure, the *trinomial* theorem, the *quartonomial theorem*, etc. are all routine, as is the *multinomial theorem* for each fixed number of variables. But we would have to wait for **WZ theory, Chapter II**, to make the full *multinomial theorem*

$$\left(\sum_{i=1}^{k} a_i\right)^n = \sum_{i_1+\ldots+i_k=n} \frac{n!}{i_1!\ldots i_k!} a_1^{i_1} \cdots a_k{}^{i_k}$$

with *symbolic* k, fully routine.

Not only **Math** progresses from the *concrete* to the *abstract*, also **Music** (e.g. Schoenberg, Stravinsky), **Art** (e.g. Picasso), **Literature** (e.g. Proust, Joyce) and even **Religion**.

The old testament God, Yehova (Jehovah), who was already much more abstract than His idolic predecessors, still enjoyed the good life, savoring the odor that came from the animal sacrifices offered by the *cohanim.* In comparison, the loving God of the new testament is much more civilized and abstract. Also Jewish Law progressed towards symbolism and abstraction. For example, the barbaric *eye for an eye* and *tooth for a tooth*, was changed, by Rabbi Hillel, to paying fines, simply by defining certain units of money to be called *eye* and *tooth*, that is replacing a real eye by a symbolic eye.

Yet greater feats of abstraction were achieved by the medieval Rabbi Abraham Ibn Ezra (alias Ben-Ezra), and, inspired by the latter, at the dawn of the Age of Enlightenment, by Baruch Spinoza. Sometimes being too abstract may risk your life, or at least make your life miserable. Spinoza was ex-communicated from the Jewish community of Amsterdam for his 'heresy'.

This 'abstraction trick' can be a very powerful argument in giving 'modern' humans *reasons to believe.* Here is a quotation from a great contemporary Christian theologian.

> *'The resurrection of Jesus was* **not** *a historical event, but it is* **symbolic**, *affirming that death does not have the* **last** *word about human life'.*

The above quotation is taken from Rev. Maurice Wiles's very interesting little book entitled *Reason to Believe*[Wi]. I don't think that it is a coincidence that the child of a great theologian turned out to be a great mathematician, since both math and religion are based on *abstract concepts.*

The Rise and Fall of Logocentrism.

Alfred North Whitehead claimed that all western philosophy is a footnote to Plato. The analogous thing may be said about Euclid and (western) mathematics. Indeed, the Euclidean axiomatic approach dominated mathematics for the last 2000 years and was considered a paradigm of rigor, that philosophy and other branches of knowledge tried to emulate. For example, Spinoza used the Euclidean mold of axioms and propositions to write his Ethics book. Also modern economics, in its struggle to become 'scientific', often uses axioms.

The precise logical thinking in Euclid's *elements* was also considered to be of great pedagogical, moral, and even *religious* significance, since the 'rational' thinking that studying it was supposed to engender supposedly made one a better person.

Ironically, the greatest *rationalist* of them all, René Descartes, demolished the Euclidean supremacy, at least in its immediate, geometrical, scope, by creating *Analytic Geometry.* I will soon argue that Descartes's breakthrough also shattered the Euclidean tradition in the *general* sense, in putting *Algebra* above *Logic*. However this, more general sense, took more than three hundred years to kick in.

Even though René Descartes already trivialized Geometry *in principle*, it took computer algebra, and the genius of our birthday boy, Bruno Buchberger, to make this *trivialization*, or more politely, *algorithmization*, feasible in *real time*.

What Buchberger did, via his revolutionary Gröbner bases, was to establish a *canonical form* for ideals in a polynomial ring. Since every entity in Plane Geometry can be described by an ideal, it gives us a decision procedure for proving any theorem. So one no longer must be clever, or have geometrical intuition. All that is needed is a knowledge of typing, the rest is done by the computer.

Alas, the strength of Gröbner bases is also their weakness. Because they can do so much, they are usually very slow, and one has the exponential-time-and-space curse. It turns out that for most theorems in Plane (and space) Geometry, one does not need ideals, and one can still prove things by making everything explicit, since via *parameterization*, things run much faster. Here is an example of Erdös's favorite theorem, called the **Butterfly Theorem**. Its statement, in Maple, is:

```
Butterfly:=proc() local P,t,i,R,Li,M,X,Y:for i from 1 to 4 do
P[i]:=ParamCircle([0,0],R,t[i]) od:M:=Pt(Le(P[1],P[3]),Le(P[2],P[4])):
Li:=PerpPQ([0,0],M):X:=Pt(Le(P[1],P[4]),Li):Y:=Pt(Le(P[2],P[3]),Li):
ItIsZero(DeSq(M,X)-DeSq(M,Y)):end:
```

This is taken from Shalosh B. Ekhad's **Geometry Textbook**[E]. It is a textbook written entirely in Maple, but that is fun to read for computers and humans alike. The above is a complete statement of the Butterfly theorem, and in order to prove it, all one has to do is type `Butterfly();` . In less than a second, the computer returns `true`.

For the sake of completeness, here are those marcos that are needed for `Butterfly`. So together with these, we have a completely self-contained *statement* and *proof* of the theorem.

```
#Def (Area of triangle ABC)
AREA:=proc(A,B,C):normal(expand((B[1]*C[2]-B[2]*C[1]-A[1]*C[2]+A[2]*C[1]
-B[1]*A[2]+B[2]*A[1])/2)):end:

#Def (Square of the distance of points A and B)
DeSq:=proc(A,B):(A[1]-B[1])**2+(A[2]-B[2])**2:  end:

#Def (Is it zero?)
ItIsZero:=proc(a):evalb(normal(a)=0):end:

#Def (The eq.  of the line joining A and B)
Le:=proc(A,B) AREA(A,B,[x,y]):end:

#Def (Generic point on a parametric circle center [c[1],c[2]] and radius R)
ParamCircle:=proc(c,R,t):[c[1]+R*(t+1/t)/2,c[2]+R*(t-1/t)/2/I]:end:

#Def (Line through Q perpendicular to PQ)
PerpPQ:=proc(P,Q):expand((y-Q[2])*(P[2]-Q[2])+(x-Q[1])*(P[1]-Q[1])):end:

#Def (The point of intersection of lines Le1 and Le2)
Pt:=proc(Le1,Le2) local q:q:=solve(
numer(normal(Le1)),numer(normal(Le2)),x,y):
[normal(simplify(subs(q,x))),normal(simplify(subs(q,y)))]:end:
```

The reason that it took less than one second of CPU time was that I used the *parametric equation* of the circle (implemented in `ParamCircle` above). In Humanese it is

$$x = R(t + t^{-1})/2 \quad , \quad y = R(t - t^{-1})/(2i).$$

Notice that everything is pure (high-school) algebra.

Now, had we insisted on using Gröbner bases, then , in the 'straightforward approach' we would have had to write the assumptions that the four points $P_1 = (x_1, y_1), P_2 = (x_2, y_2), P_3 = (x_3, y_3), P_4 = (x_4, y_4)$ lie on the circle, by introducing eight variables $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$, and introducing the ideal generated by $\{x_1^2 + y_1^2 - 1, x_2^2 + y_2^2 - 1, x_3^2 + y_3^2 - 1, x_4^2 + y_4^2 - 1\}$. (Dongming Wang pointed out that it is much more efficient to take three points, find their circumcircle, and demand that the fourth point lies on that circle). Next the computer would have had to compute its Gröbner basis. Then compute the *bottom-line* quantity (that has to be shown equal to 0) and finally use `normalf` to find it modulo the above ideal. This takes a little longer than before.

Of course, as you all know, Gröbner bases can do many other things besides proving 'high-school geometry' theorems, and it is impossible to imagine modern (and post-modern) algebra without it.

In all the many theorems in Ekhad's Geometry text, it was possible to avoid using Gröbner bases. Of course for *general* problems, one can't avoid it, but it so happens that for most theorems that come up in real life, it was possible to get away with parameterization (essentially equivalent to using Trigonometry). Here we have an example of a **targeted** subansatz. (Dongming Wang pointed out that the many cases where one can do without Gröbner bases are those where it is possible to 'linearize' quadratic relations.)

By successive abstraction, and going from one level to the next meta-level, one gets more and more symbolic. But once the symbols can be considered **qua symbols**, regardless of their *meaning*, things become **concrete** again. Ideally one should be able to write a computer program, and there is nothing more concrete than a computer program. So indeed **Ultimate Abstraction** is **Concrete**.

Analogously, **Ultimate Semantics is Syntactic**. But the converse is also true: **Ultimate Syntax is Semantic**. The symbols themselves are *marks on the paper* (so despised by Brouwer), and today *bytes*, obeying certain *combinatorial* conditions. So everything boils down to *combinatorics*, and what can be more concrete than combinatorial, finite, objects?

The **Axiomatic Method**, and **Formal Logic** (starting with Euclid and Aristotles) have had their 2000 years of glory. Thanks to Kurt Gödel we know that they can't do *everything* (even in principle). Another giant, Gregory Chaitin, quantified it with his beautiful uncomputable constant Ω. Gödel also proved that even for decidable statements, there exist short theorems with very long proofs. The Four-Color Theorem may be an example.

Hence, **humans** (and even **computers**) can formally and fully rigorously only prove facts of very low complexity (in the technical sense of computational (or program-length) complexity).

Realizing this, if we want to transcend our intrinsic triviality we need to **diversify**, and be more **inclusive**. We should welcome the whole gamut of mathematical truths. In addition to fully rigorous proofs, we also need semi-rigorous proofs (see [Z1]), ϵ-rigorous proofs, non-rigorous proofs, very plausible conjectures, plausible conjectures, all the way to wild guesses. Of course, for interesting statements it would be nice to upgrade their level of truth, but we should not spend too much time on these upgrades, since there are so many exciting new facts to discover.

We desperately need a new philosophy and methodology for doing mathematics, and I believe that the *practice* of symbolic computation is a very good *rough draft*, and *starting point* for this.

But before describing it, let's digress for one minute in order to critique the prevailing philosophies.

Formalism is too logocentric, while **Logicism** is even more logocentric. **Intuitionism** is too human-centric while **Humanism** (started by Phillip Davis and Reuben Hersh and fully developed by Hersh) is even more human-centric. **Platonism** is too platonic and metaphysical, while **Bourbakism** is too structured and semantical.

The **"new" philosophy** and **methodology** that I am proposing here, inspired by computer algebra, is really a revival of two very old traditions:

Pythagorianism, with its denial of 'real' numbers and the infinite (see [Z2]).

The Hindu-China-Babylonian-Persian-Arabic tradition of 'high-school', algorithmic, algebra.

In the new **methodology**, we should forget both about **syntax** and **semantics**, and focus instead on **PRAGMATICS**!: Learn to serve IRH (Its Royal Highness), the **Computer**.

We also need **Problem-Solving** methodologies, e.g. adapt Polya's heuristics (as given in his famous classic book 'How to solve it' and the series of books on plausible reasoning), to Computers and especially Computer Algebra Systems.

A very **important Principle**, originally intended as a **pedagogical principle**, but that I am sure has a much wider **scope** for doing **research** is, The **BBBBwBp**, which is short for: Bruno Buchberger's Black Box White Box Principle.

This great brainchild of our beloved Birthday Boy ([B1]) means that when we teach students a new concept or method or algorithm, in the first phase we should **not** let them use computers, but let them do it by hand, so that they can internalize what they are learning, with simple examples. But once they mastered it, it should be encapsulated into a **black box**, so that they can graduate to bigger, better, and deeper things, without being bogged down with details. This principle should be used recursively, of course, until an intricate web of knowledge will mature in the student's head.

But all of us are **students**! The computer is our **master**, and if we will learn to use **black boxes** efficiently, trusting their contents, without necessarily fully understanding them, we will be able to go much further. One of the reason mathematicians made so little progress so far is their obsession for knowing all the details, and not trusting previous results as black boxes. Furthermore, because math is so fragmented and specialized, there are lots of black boxes, developed by specialists in other areas, that are unaccessible to us, because of the language-barrier between sub-specialties. We really need an **Esperanto**, or at least a **lingua franca** that will bridge this tragic communication failures between mathematical subareas. A good start could be Maple (or Mathematica), that of course will have to keep expanding. The language of formal logic, with its quantifiers, turned out to do more harm than good because of its overwhelming generality and pedantry, and because it did not describe the day-to-day practice of doing mathematics. (I know that Bruno may disagree with me here, since he is a great fan of logic, and I agree that logic is a marvelous thing, but it should not be carried too far.)

The Bruno Buchberger Black Box-White Box principle is a great example of using wisdom gained in **teaching** to help do **research**. Most research mathematicians either dislike teaching, viewing it as a chore, or like it, but think of it as an activity unrelated to their research. This 'binary-opposites' pair, *teaching-research*, is yet another dichotomy that has to be abandoned. In the future research will be both **teaching** and **learning**. First teaching the computer to do mathematical research, and later learning from its output, and so on indefinitely.

One possible way to teach computers is to look for new **Ansatzes**, that will convert classes of theorems to 'high school algebra', amenable to computer search. I call this approach the **Ansatz Ansatz**, in analogy to Thomas Kuhn's approach to the philosophy of science, that may be called the **Paradigm Paradigm**.

Sometimes you have to transcend to a **superansatz**. Let's make-believe that the sequence $f(n)$ enumerating the number of legal bracketings with n pairs of brackets is 'intractable' (it is in fact the sequence of Catalan numbers, that satisfy a *non-linear* recurrence with constant coefficients, and a linear recurrence with *polynomial* coefficients, but let's forget this right now and pretend that we are linear, constant-coefficients, creatures). If you consider instead the more *general* function $F(m, n)$ of *prefixes* of legal bracketings, with m left parentheses and n right parentheses (and of course $m \geq n$), then $F(m, n)$ satisfies a *partial recurrence equation* $F(m, n) = F(m, n-1) + F(m, n-1)$ with the boundary conditions $F(m, 0) = 1$ and $F(m, m+1) = 0$, from which follows immediately that $F(m, n) = (m-n+1)(m+n)!/((m+1)!n!)$ (check!), and in particular, $f(n) = F(n, n) = (2n)!/((n+1)!n!)$.

What happened here was that we went up to a **superansatz** that made the problem more tractable.

Sometimes the ansatz is adequate, *in principle*, but is computationally inefficient. In that case one can look for **targeted subansatzes** that work faster for subclasses of problems or objects. We already saw that in the Plane Geometry example above, where Gröbner bases are relatively fast, but can be made yet faster with the 'right' parametrization, for an important subclass, by staying in the *rational function ansatz*.

Another example is the **Holonomic paradigm**, that plays a fundamental role with respect to algorithmic work in the field of special functions (e.g. NIST's Digital Mathematical Functions Library). My slow algorithm [Z3] (vastly improved by Chyzak[C], and in fact it is not so slow anymore at all), is very general. Then my 'fast' algorithm ([Z4]) is much faster, but can only do *proper hypergeometric* summands. Finally the WZ-pairs are yet faster, and also very elegant, but only work when the right hand side is closed form (i.e. the recurrence outputted by Zeilberger's algorithm is first-order).

Another important subansatz is Wegschaider's use of ideas of Verbaeten, that can be seen as a targeted subansatz for WZ/Sister Celine in the multiple sum case. Recently there was an exciting speed-up achieved by Axel Riese and Bruno Zimmermann obtained by supplementing this with another paradigm, namely with random modular checking.

Peter Paule's recent exciting work on *contiguous relations* can be viewed as a *superansatz* of my creative telescoping ansatz and is analogous to the above-mentioned generalization of the Catalan function f(n) to its bivariate version F(m,n).

Once you understand your **ansatz**, you can have your computer, discover **from scratch**, all the theorems in the field up to any given complexity.

A Confession

The masterpiece [E] that was supposedly 'downloaded from the future' was entirely written, by hand, by a human (myself). All that (the current) Shalosh B. EKhad (IV) did was to *run the program* and *prove correctness* for all the 'statement-proofs', and also draw the beautiful diagrams.

Not only did I lie about the author and date, but I did something much worse. In the 'Preface by the downloader', I pretended that the text was automatically generated by the computer Shalosh B. Ekhad, XIV, by using another *meta-program* that started with three generic points, and then by iterating a few macros, viz. **Pt, Le, Ce**, the computer kept getting new points, lines, circles, etc. Whenever a new object coincides with a previously defined one, the computer *discovered*, and at the very same time *proved*, a new theorem,

But, even though it was **not** done that way, it **could have been done that way**, I was just too lazy. I hope that someone will soon write this 'Geometry from scratch' computer program.

Automated DISCOVERY (and Proof) of ALL Binomial Coefficient Identities (Up to a Prescribed Complexity)

The so-called Zeilberger algorithm can *prove* any binomial coefficient identity (alias hypergeometric series) identity, once conjectured. WZ theory can do even better. By starting with one of the **golden oldies**, like Gauss, Saalschutz, Dixon, or Dougall, and *specializing and dualizing* (see [PWZ]), it can discover, and automatically prove, lots of *new* 'strange' hypergeometric identities.

BUT suppose that you do not know anything about the human heritage. Do you really need these humans to get started? Of course you don't! In principle all that humans have to do is *define* a concept. In this case the appropriate concept is that of *WZ pair*, invented by the humans Herb Wilf and Doron Zeilberger.

Let's first review *WZ theory*.

WZ Theory in a Nutshell

Suppose you want to prove an identity of the form

$$\sum_k NICE(n,k) = NICE'(n) \quad ,$$

where 'NICE' means hypergeometric in its arguments, i.e. ($NICE(n+1,k)/NICE(n,k)$ and $NICE(n,k+1)/NICE(n,k)$ are rational functions of (n,k) and $NICE'(n+1)/NICE'(n)$ is a rational function of n). Actually, we need $NICE(n,k)$ to be *proper-hypergeometric* (see [PWZ]), but let's forget about this technicality now.

The first step is to divide by $NICE'(n)$, and since $NICE(n,k)/NICE'(n)$ is also nice, let's rename the latter $F(n,k)$ and try to prove

$$\sum_k F(n,k) = 1 \quad . \qquad (NiceIdentity)$$

It so happens that in %99 of the cases[2] the *raison d'être* for such an identity is the existence of another *nice* discrete function of (n,k), let's call it $G(n,k)$, the so-called **WZ-mate**, such that

$$F(n+1,k) - F(n,k) = G(n,k+1) - G(n,k) \qquad (WZ)$$

Furthermore, $G(n,k)/F(n,k)$ is a *rational function* $R(n,k)$, called the *certificate*. Once $R(n,k)$ is given it is routine to prove (WZ), since defining the rational functions

$$R_1(n,k) = \frac{F(n+1,k)}{F(n,k)} \quad , \quad R_2(n,k) = \frac{F(n,k+1)}{F(n,k)}. \qquad (Ratios)$$

and dividing (WZ) by $F(n,k)$, reduces the proof of (WZ), at any given case, to the verification of the routine identity amongst rational functions (and by clearing denominators, amongst polynomials in n,k):

$$R_1(n,k) - 1 = R(n,k+1)R_2(n,k) - R(n,k) \quad . \qquad (WZ')$$

Now once (WZ') is established, $(NiceIdentity)$ follows immediately, by summing (WZ) with respect to k, yielding that $a(n) := \sum_k F(n,k)$ satisfies $a(n+1) - a(n) = 0$, and hence that $a(n)$ is a constant, and to prove that that constant is 1 all we have to do is verify that $a(0) = 1$.

Of course the "we" above is done completely automatically by the computer, as is the discovery of $R(n,k)$ (whenever it exists), which is done by applying Gosper's algorithm (w.r.t. k) to $F(n+1,k) - F(n,k) = (R_1(n,k) - 1)F(n,k)$.

But, how can we find *new* identities, *completely from scratch*? Every WZ-pair is really a *miracle*. First note that the rational functions $R_1(n,k)$ and $R_2(n,k)$, in order to arise from a hypergeometric $F(n,k)$ by $(Ratios)$, must satisfy the obvious compatibility condition

$$R_1(n,k+1)R_2(n,k) = R_1(n,k)R_2(n+1,k). \qquad (Compatibility)$$

So the (WZ) miracle hinges on the existence of a triple of rational functions (R_1, R_2, R) depending on (n,k) such that the non-linear equations $(Compatibility)$ and (WZ') are satisfied.

However, not all such triples are *interesting*, since for any nice $A(n,k)$, defining

$$F(n,k) = A(n,k+1) - A(n,k) \quad , \quad G(n,k) = A(n+1,k) - A(n,k),$$

automatically yields a (WZ)-pair, from which we can get lots of trivial solutions for the system $((Compatibility), (WZ'))$. In order to get the 'nice and interesting identities' we must 'mod out' by these 'exact forms'.

[2] There is always something more general that guarantees a proof.

Conclusion

The SymbolicComputational tail is starting to wag the mathematics dog, and will continue to do so more and more vigorously, and very soon, (traditional) math will become the tail of computer algebra. In order to accomplish this rôle-reversal, we desperately need a new outlook, and **working style**, and this article proposes a very rough draft. The details of the emerging philosophies and methodologies are yet to be worked out, but one thing is certain: the **pioneers** of the **SymbolicComputational Revolution**, most notably **Bruno Buchberger**, will become legendary and mythical heroes as long as human beings (and/or computers) will continue to do mathematics.

Acknowledgement

I wish to thank Bruno Buchberger and Dongming Wang for numerous suggestions and comments on an earlier draft.

REFERENCES

[B] B. Buchberger, *Theorema and mathematical knowledge management*, lecture delivered at the IMA (Minneapolis), Aug. 1, 2002, available from:
`http://www.risc.uni-linz.ac.at/people/buchberg/invited_talks.html`.

[B1] B. Buchberger, *Should Students Learn Integration Rules?*, SIGSAM Bulletin Vol.**24/1** (1990), 10-17.

[BCL] B. Buchberger, G. E. Collins, and R. Loos, *"Computer Algebra Symbolic and Algebraic Computation"*, 2nd Edition, Springer-Verlag, 1982.

[C] F. Chyzak, *An extension of Zeilberger's fast algorithm to general holonomic functions*, Discrete Mathematics **217** (2000), 115-134. Preliminary version available on-line at
`http://algo_inria_fr/chyzak/publications.ps` .

[E] S. B. Ekhad, XIV, "*Plane Geometry: An Elementary School Textbook (c. 2050)*"
`http://www.math.rutgers.edu/~zeilberg/GT.html`.

[G] T. Gowers, *"Mathematics - A Very Short Introduction"*, Oxford University Press, 2002.

[L] R. Loos, *Introduction to [BCL]*, (pp. 1-11).

[PWZ] M. Petkovsek, H. Wilf, and D. Zeilberger, *"A=B"*, A. K. Peters, 1996. [On-line version available from the authors' websites.]

[Wh] A. N. Whitehead, *"Symbolism"*, Macmillan, 1927 (reprinted by Fordham University Press, 1985) .

[Wi] M. Wiles, *"Reason to Believe"*, Trinity Press International, 1999.

[Z1] D. Zeilberger, *Theorems for a price: Tomorrow's semi-rigorous mathematical culture*, Notices of the Amer. Math. Soc. **40 # 8**, 978-981 (Oct. 1993). Reprinted: Math. Intell. **16**, no. 4, 11-14 (Fall 1994).

[Z2] D. Zeilberger, *"Real" analysis is a degenerate case of discrete analysis*, Proc. of ICDEA7 (2001), B. Aulbach, ed., Taylor and Francis, to appear. Available on-line from
`http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/real.html`.

[Z3] D. Zeilberger, *A Holonomic systems approach to special functions identities*, J. of Computational and Applied Math. **32**, 321-368 (1990). Available on-line from
`http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/holonomic.html`.

[Z4] D. Zeilberger, *A Fast Algorithm for proving terminating hypergeometric identities*, Discrete Math **80**, 207-211, (1990). Available on-line from
`http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/fast.html`.