Symbolic computation in the new century
The road ahead

Erich Kaltofen
North Carolina State University
www.kaltofen.net

KUNGL. VETENSKAPSAKADEMIEN
THE ROYAL SWEDISH ACADEMY OF SCIENCES

Information Department, PO Box 50005, SE-104 05 Stockholm, Sweden, webbsite: www.kva.se
Tel: +46-8-673 95 95, Fax +46-8-15 56 70, e-mail: info@kva.se

THE NOBEL PRIZE IN PHYSICS 1999

PRESS RELEASE 12 OCTOBER 1999
The Prize I Further reading I The laureates

The Royal Swedish Academy of Sciences has awarded
the 1999 Nobel Prize in Physics
jointly to

Professor **Gerardus 't Hooft**, University of Utrecht, Utrecht, the Netherlands, and
Professor Emeritus **Martinus J.G. Veltman**, University of Michigan, USA, resident in Bilthoven, the Netherlands.

The two researchers are being awarded the Nobel Prize for having placed particle physics theory on a firmer mathematical foundation. **...**

The Academy's citation:
"for elucidating the quantum structure of electroweak interactions in physics."

...
One person who had not given up hope of being able to renormalize non-abelian gauge theories was **Martinus J.G.Veltman**. At the end of the 1960s he was a newly appointed professor at the University of Utrecht. Veltman had developed the *Schoonschip* computer program which, using symbols, performed algebraic simplifications of the complicated expressions that all quantum field theories result in when quantitative calculations are performed. Twenty years earlier, Feynman had indeed systematised the problem of calculation and introduced *Feynman diagrams* that were rapidly accepted by researchers. But at that time there were no computers. Veltman believed firmly in the possibility of finding a way of renormalizing the theory and his computer program was the cornerstone of the comprehensive work of testing different ideas.

Overview

1. Faster algorithms:
   counting bit operations vs. counting arithmetic operations

2. Imprecise inputs
   With PhD student Markus Hitz

3. Lattice basis reduction

4. Component technology
   With PhD student Angel Díaz

5. Uncertain results
   With PhD student Wen-shin Lee

1. Linear Algebra

Strassen's [1969] $O(n^{2.81})$ matrix multiplication algorithm

$$
\begin{array}{l|l}
m_1 \leftarrow (a_{1,2} - a_{2,2})(b_{2,1} - b_{2,2}) & \\
m_2 \leftarrow (a_{1,1} + a_{2,2})(b_{1,1} + b_{2,2}) & \\
m_3 \leftarrow (a_{1,1} - a_{2,1})(b_{1,1} + b_{1,2}) & \\
m_4 \leftarrow (a_{1,1} + a_{1,2})b_{2,2}) & a_{1,1}b_{1,1} + a_{1,2}b_{2,1} = m_1 + m_2 - m_4 + m_6 \\
m_5 \leftarrow a_{1,1}(b_{1,2} - b_{2,2}) & a_{1,1}b_{1,2} + a_{1,2}b_{2,2} = m_4 + m_5 \\
m_6 \leftarrow a_{2,2}(b_{2,1} - b_{1,1}) & a_{2,1}b_{1,1} + a_{2,2}b_{2,1} = m_6 + m_7 \\
m_7 \leftarrow (a_{2,1} + a_{2,2})b_{1,1}) & a_{2,1}b_{1,2} + a_{2,2}b_{2,2} = m_2 - m_3 + m_5 - m_7
\end{array}
$$

Problems reducible to matrix multiplication:

linear system solving [Bunch and Hopcroft 1974],...

Coppersmith and Winograd [1990]: $O(n^{2.38})$

Life after Strassen: black box linear algebra

The black box model of a matrix

$A \in \mathbb{K}^{n \times n}$ singular

$\mathbb{K}$ an arbitrary, e.g., finite field

Perform linear algebra operations, e.g., $A^{-1}b$ [Wiedemann 86] with

$O(n)$	black box calls and
$n^2(\log n)^{O(1)}$	arithmetic operations in $\mathbb{K}$ and
$O(n)$	intermediate storage for field elements

Flurry of recent results

Lambert [96], Teitelbaum [98], Eberly & Kaltofen [97]	relationship of Wiedemann and Lanczos approach
Villard [97]	analysis of *block* Wiedemann algorithm
Giesbrecht [97] and Mulders & Storjohann [99]	computation of integral solutions
Giesbrecht, Lobo & Saunders [98]	certificates for inconsistency
Chen, Eberly, Kaltofen, Saunders, Villard & Turner [2K]	butterfly network, sparse and diagonal preconditioners
Villard [2K] & Storjohann [01]	characteristic polynomial

Diophantine solutions by Giesbrecht:
Find several rational solutions.

$$A(\tfrac{1}{2}x^{[1]}) = b, \quad x^{[1]} \in \mathbb{Z}^n$$
$$A(\tfrac{1}{3}x^{[2]}) = b, \quad x^{[2]} \in \mathbb{Z}^n$$
$$\gcd(2,3) = 1 = 2 \cdot 2 - 1 \cdot 3$$
$$A(2x^{[1]} - x^{[2]}) = 4b - 3b = b$$

Life after Strassen: bit complexity

Linear system solving $x = A^{-1}b$ where $A \in \mathbb{Z}^{n \times n}$ and $b \in \mathbb{Z}^n$:

With Strassen [McClellan 1973]:

Step 1: For prime numbers $p_1, \ldots p_k$ Do
Solve $Ax^{[j]} \equiv b \pmod{p_j}$ where $x^{[j]} \in \mathbb{Z}/(p_j)$

Step 2: Chinese remainder $x^{[1]}, \ldots, x^{[k]}$ to $A\bar{x} \equiv b \pmod{p_1 \cdots p_k}$

Step 3: Recover denominators of x_i by continued fractions of $\dfrac{\bar{x}_i}{p_1 \cdots p_k}$.

Length of integers: $k = (n \ \max\{\log \|A\|, \log \|b\|\})^{1+o(1)}$

Bit complexity: $n^{3.38} \ \max\{\log \|A\|, \log \|b\|\}^{1+o(1)}$

With Hensel  lifting [Moenck and Carter 1979, Dixon 1982]:

Step 1: For $j = 0, 1, \ldots, k$ and a prime p Do

Compute $\bar{x}^{[j]} = x^{[0]} + px^{[1]} + \cdots + p^j x^{[j]} \equiv x \pmod{p^{j+1}}$

1.a. $b^{[j]} = \dfrac{b - A\bar{x}^{[j-1]}}{p^j} = \dfrac{b - (A\bar{x}^{[j-2]} + Ap^{j-1}x^{[j-1]})}{p^j}$

1.b. $x^{[j]} \equiv A^{-1}b^{[j]} \pmod{p}$ reusing $A^{-1} \bmod p$

Step 3: Recover denominators of x_i by continued fractions of $\dfrac{\bar{x}_i^{[k]}}{p^k}$.

With classical matrix arithmetic:

Bit complexity of 1.a: $n(n \max\{\log \|A\|, \|b\|\})^{1+o(1)} + n^2(\log \|A\|)^{1+o(1)}$

Total bit complexity: $(n^3 \ \max\{\log \|A\|, \log \|b\|\ })^{1+o(1)}$

Note: the complexity of computing a floating-point solution to a linear system and the complexity of computing the exact solution of a linear system with classical matrix arithmetic are asymptotically the same.

New results:

- bit complexity of the determinant: $O(n^{2.697} \log \|A\|)$ [Kaltofen & Villard 2001]

- division-free complexity of determinant: $O(n^{2.697})$ additions, subtractions, and multiplications [Kaltofen & Villard 2001]

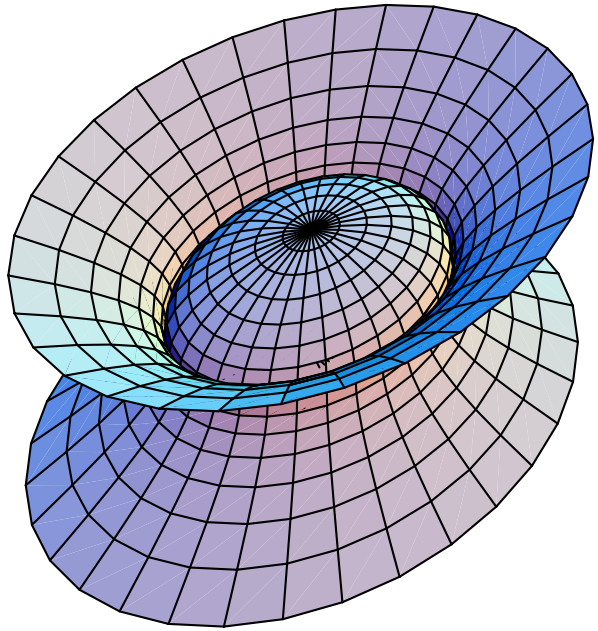
- bit complexity of linear systems: $O(n^{2.5} \log \max\{\|A\|, \|b\|\})$ [Villard 2001]

Problem 1

Improve the bit complexity of algorithms for the determinant, resultant, linear system solution, over the integers.

2. Factorization of nearby polynomials over the complex numbers

$$81x^4 + 16y^4 - 648z^4 + 72x^2y^2 - 648x^2 - 288y^2 + 1296 = 0$$



$$(9x^2 + 4y^2 + 18\sqrt{2}\,z^2 - 36)(9x^2 + 4y^2 - 18\sqrt{2}\,z^2 - 36) = 0$$

$$81x^4 + 16y^4 - 648.003z^4 + 72x^2y^2 + .002x^2z^2 + .001y^2z^2$$
$$- 648x^2 - 288y^2 - .007z^2 + 1296 = 0$$

Problem 2 *[Kaltofen LATIN'92]*
Given is a polynomial $f(x,y) \in \mathbb{Q}[x,y]$ *and* $\epsilon \in \mathbb{Q}$.

Decide in polynomial time in the degree and coefficient size if there is a factorizable $\tilde{f}(x,y) \in \mathbb{C}[x,y]$ *with*

$$\|f - \tilde{f}\| \leq \epsilon \textit{and } \deg(\tilde{f}) \leq \deg(f),$$

for a reasonable coefficient vector norm $\|\cdot\|$.

Note: fast algorithms are known for exact factorization
[Kaltofen 1985, Shuhong Gao 1999]

Theorem *[Hitz, Kaltofen, Lakshman ISSAC'99]*
We can compute in polynomial time in the degree and coefficient size if there is an $\tilde{f}(x, y) \in \mathbb{C}[x, y]$ *with a factor of a* ***constant degree*** *and* $\|f - \tilde{f}\|_2 \leq \epsilon$.

Numerical algorithms

Conclusion on my exact algorithm [JSC 1985]:

"*D. Izraelevitz at Massachusetts Institute of Technology has already implemented a version of algorithm 1 using complex floating point arithmetic. Early experiments indicate that the linear systems computed in step (L) tend to be* ***numerically ill-conditioned.*** *How to overcome this numerical problem is an important question which we will investigate.*"

Stetter, Huang, Wu and Zhi [ISSAC'2K]: Hensel lift factor combinations numerically and eliminate extraneous factors early

Corless, Kotsieras, van Hoeij, Watt [ISSAC'01]: use deformation theory to construct factors numerically

Univariate Problem: Given $f \in \mathbb{C}[z]$ and $\alpha \in \mathbb{C}$.
Find $\tilde{f} \in \mathbb{C}[z]$,such that

$$\tilde{f}(\alpha) = 0\,, \quad \text{and} \quad \|f - \tilde{f}\| = \min\,.$$

Let

$$\begin{aligned} f(z) &= a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0 \\ \tilde{f}(z) &= (z-\alpha)\,(u_{n-1} z^{n-1} + u_{n-2} z^{n-2} + \cdots + u_0) \\ &= u_{n-1} z^n + (u_{n-2} - \alpha) z^{n-1} + \cdots + (u_0 - \alpha u_1) z - \alpha u_0 \end{aligned}$$

In terms of linear algebra:

$$\|f - \tilde{f}\| = \min_{\mathbf{u} \in \mathbb{C}^n} \left\| \underbrace{\begin{bmatrix} -\alpha & & & 0 \\ 1 & -\alpha & & \\ & \ddots & \ddots & \\ 0 & & 1 & -\alpha \\ & & & 1 \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{bmatrix}}_{\mathbf{u}} - \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \\ a_n \end{bmatrix}}_{\mathbf{b}} \right\| \tag{1}$$

(1) is an over-determined linear system of equations:

Linear program, if $\|\cdot\|$ is the $\begin{cases} \infty\text{-norm, or} \\ 1\text{-norm} \end{cases}$

Least squares problem, if $\|\cdot\|$ is the 2-norm (Euclidean).

Solutions for the 2-norm in closed form:

$$\mathcal{N}_{min}(\alpha) = \|f - \tilde{f}\|^2 = \frac{\overline{f(\alpha)} f(\alpha)}{\sum_{k=0}^{n} (\overline{\alpha}\alpha)^k}, \quad f_j - \tilde{f}_j = \frac{(\overline{\alpha})^j f(\alpha)}{\sum_{k=0}^{n} (\overline{\alpha}\alpha)^k}$$

(also derived in Corless et al. [ISSAC'95] via SVD)

Constraining a Root Locus to a Curve

Let Γ be a piecewise smooth curve with finitely many segments, each having a parametrization $\gamma_k(t)$ in a single real parameter t.

For a given polynomial $f \in \mathbb{C}[z]$, we want to find a minimally perturbed polynomial $\tilde{f} \in \mathbb{C}[z]$ that has (at least) one root on Γ.

Parametric Minimization

We substitute the parametrization $\gamma_k(t)$ for the indeterminate α in $\mathcal{N}_{min}(\alpha)$. The resulting expression is a function in $t \in \mathbb{R}$.

It attains its minima at its *stationary* points. We have to compute the *real* roots of the derivative.

The derivative of the norm-expression is determined *symbolically*, but the roots can be computed numerically.

Bivariate factorization

Given $f = \sum f_{i,j} x^i y^j \in \mathbb{C}[x, y]$ **absolute irreducible**, find

$$\tilde{f} = (c_0 + c_1 x + c_2 y) u(x, y) \in \mathbb{C}[x, y], \quad \deg(\tilde{f}) \leq \deg(f),$$

such that $\|f - \tilde{f}\|_2$ is minimal.

("nearest polynomial with a linear factor").

Approach: minimize parametric least square solution in the real and imaginary parts of the $c_i = \alpha_i + \beta_i \, \mathbf{i}$.
$\longrightarrow$ must minimize least squares solution with 6 parameters.
$\longrightarrow$ yields polynomial system with a **fixed number of variables**, hence polynomial time.

3. Lattice basis reduction

$$\pi = \sum_{i=0}^{\infty} \frac{1}{16^i} \left(\frac{4}{8\,i+1} - \frac{2}{8\,i+4} - \frac{1}{8\,i+5} - \frac{1}{8\,i+6} \right)$$

Derivation by lattice reduction [Bailey&Borwein&Plouffe 1995]

$$\int_0^1 \frac{y^{k-1}}{1-\frac{y^8}{16}} dy = \int_0^1 \sum_{i=0}^{\infty} y^{k-1} \left(\frac{y^8}{16} \right)^i dy = \sum_{i=0}^{\infty} \frac{1}{16^i} \int_0^1 y^{8i+k-1} dy$$

$$= \sum_{i=0}^{\infty} \frac{1}{16^i (8i+k)}$$

Maple takes over

```
> latt := proc(digits)
> local k, j, v, saved_Digits, ltt;
> saved_Digits := Digits; Digits :=
> digits;
> for k from 1 to 8 do
>   v[k] := [];
>   for j from 1 to 10 do v[k] := [op(v[k]),
> 0]; od;
>   v[k][k] := 1;
>   v[k][10] := trunc(10^digits *
>                   evalf(Int(y^(k-1)/(1-y^8/16),
>                             y=0..1, digits), digits));
> od;
> v[9] := [0,0,0,0,0,0,0,0,1,
>             trunc(evalf(Pi*10^digits,digits+1))];
> ltt := [];
> for k from 1 to 9 do ltt:=[op(ltt),evalm(v[k])];od;
> Digits := saved_Digits;
> RETURN(ltt);
> end:
```

```
> L := latt(25);
```

$$L := [[1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 100718447641467622864476 00],$$
$$[0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 506476876667430480955939 4],$$
$$[0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 339230245245199072515585 3],$$
$$[0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 255412811882995341602757 0],$$
$$[0,\ 0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 205002557636423533944150 3],$$
$$[0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 171317070666497458966732 8],$$
$$[0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 147201934672635027195598 1],$$
$$[0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 0,\ 129077042275142343345847 8],$$
$$[0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 314159265358979323846264 34]]$$

```
> readlib(lattice):
> lattice(L);
```

$$[[-4,0,0,2,1,1,0,0,1,5],[0,-8,-4,-4,0,0,1,0,2,5],$$
$$[-61,582,697,-1253,453,-1003,-347,-396,10,559],$$
$$[-333,966,324,-1656,-56,784,1131,-351,-27,255],$$
$$[429,714,-1591,778,-517,-1215,598,362,-87,398],$$
$$[-1046,-259,-295,-260,1286,393,851,800,252,-1120],$$
$$[494,906,-380,-1389,1120,1845,-1454,-926,-218,400],$$
$$[1001,-1099,422,1766,1405,-376,905,-1277,-394,-30],$$
$$[-1144,491,-637,-736,-1261,-680,-1062,-1257,637,-360]]$$

```
> g := (8*y + 4*y^2 + 4*y^3 - y^6)/(1-y^8/16);
```

$$g := \frac{8\,y + 4\,y^2 + 4\,y^3 - y^6}{1 - \frac{1}{16}\,y^8}$$

```
> int(g, y=0..1);
```

$$2\,\pi$$

Goldreich&Goldwasser&Halevi [1997] public key crypto system

Public key: Lattice basis B (rows B_i are basis vectors).

Private key: *reduced* basis C for lattice spanned by B.

Clear text is represented as a vector x with *small* integer entries.

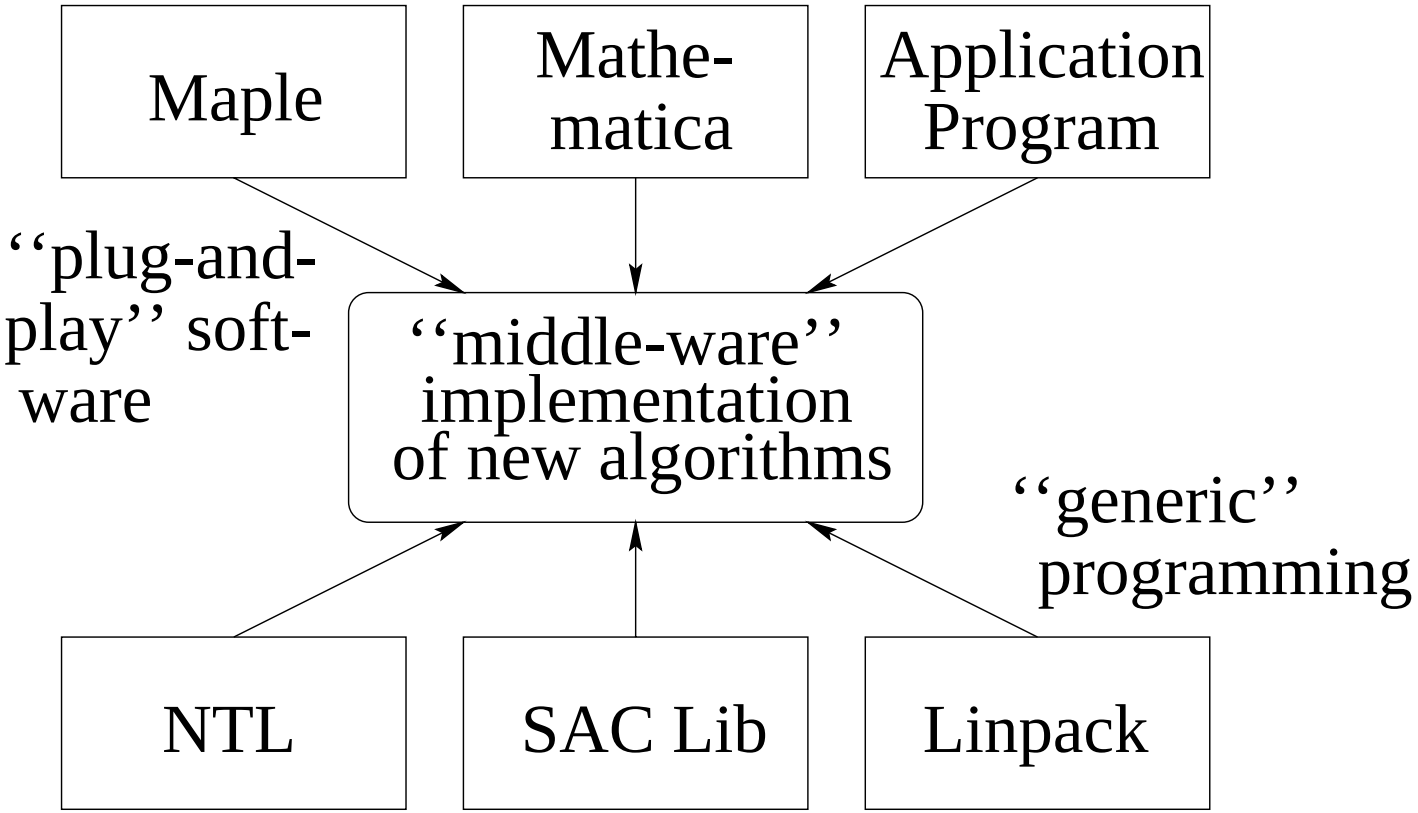
Encoded message: $y = x + \sum_i r_i B_i$ where $\sum_i r_i B_i$ is a random vector in the lattice.

Decryption based on Babai algorithm [1985] for nearest lattice point: Write $y = \sum_i s_i C_i$ with $s_i \in \mathbb{Q}$. Then $\sum_i \text{nearest-integer}(s_i) C_i$ is a near lattice point, probably $\sum_i r_i B_i$.

Problem 3

Devise a public key crypto-system that is based on diophantine linear algebra but that is safe from lattice reduction.

4. Plug-and-play systems



Problem solving environ's: end-user can easily custom-make symbolic software

Example: FoxBox [Díaz and K 1998]

```
# Call FoxBox server from Maple
> SymToeQ  := BlackBoxSymToe( BBNET_Q, 4, -1, 1.0 ):
> SymToeZP := BlackBoxSymToe( BBNET_ZP, 4, -1, 1.0 ):
> FactorsQ := BlackBoxFactors( BBNET_Q, SymToeQ, Mod, 1.0,
                               Seed ):
> FactorsZP := BlackBoxHomomorphicMap( BBNET_FACS, FactorsQ,
                                       SymToeZP ):
```

```
// construct factors of a symmetric Toeplitz determinant in C++
typedef BlackBoxSymToeDet< SaclibQ, SaclibQX > BBSymToeDetQ;
typedef BlackBoxFactors< SaclibQ, SaclibQX,
                                  BBSymToeDetQ > BBFactorsQ;

BBSymToeDetQ SymToeDetQ( N );
BBFactorsQ   FactorsQ( SymToeDetQ, Probab, Seed, &MPCard );
```

Plug-and-play

- Standard representation for transfer: MP, OpenMath, MathML
- Byte code for constructing objects vs. parse trees
- Visual programming environments for composition

Generic Programming

- Common object interface (wrapper classes),
  e.g., `K::random_generator(500)`
- Storage management vs. garbage collection
- Algorithmic shortcuts into the basic modules
- Parallel distribution of computation

MMLI2M: Presentation

- Describes appearance of math expression

$$(a+b)^2$$

```
<mrow>
  <msup>
    <mfenced>
      <mrow>
        <mi>a</mi>
        <mo>+</mo>
        <mi>b</mi>
      </mrow>
    </mfenced>
    <mn>2</mn>
  </msup>
</mrow>
```

Notice that the 2 is merely a superscript.

MMLI2M: Content

- Describes meaning of a math expression

$$(a+b)^2$$

```
<mrow>
  <apply>
    <power/>
    <apply>
      <plus/>
      <ci>a</ci>
      <ci>b</ci>
    </apply>
    <cn>2</cn>
  </apply>
</mrow>
```

MMLI2M: Mixing

```
<semantics>
  <mrow>
    <mrow>
      <mo>(</mo>
      <mi>a</mi>
      <mo>+</mo>
      <mi>b</mi>
      <mo>)</mo>
    </mrow>
    <mo>&InvisibleTimes;</mo>
    <mrow>
      <mo>(</mo>
      <mi>c</mi>
      <mo>+</mo>
      <mi>d</mi>
      <mo>)</mo>
    </mrow>
  </mrow>

  <annotation-xml encoding="MathML-Content">
    <apply>
      <and/>
      <apply>
        <xor/><ci>a</ci><ci>b</ci>
      </apply>
      <apply>
```

Problem 4

Devise a plug-and-play and generic programming methodology for symbolic mathematical computation that is widely adopted by the experts in algorithm design, the commercial symbolic software producers, and the outsider users.

MathML examples courtesy Stephen M. Watt.

5. Will our systems guarantee their answers?

Maple 6 allows calls to NAG numeric library routines

Basic polynomial algorithms with floating point coefficients are under development

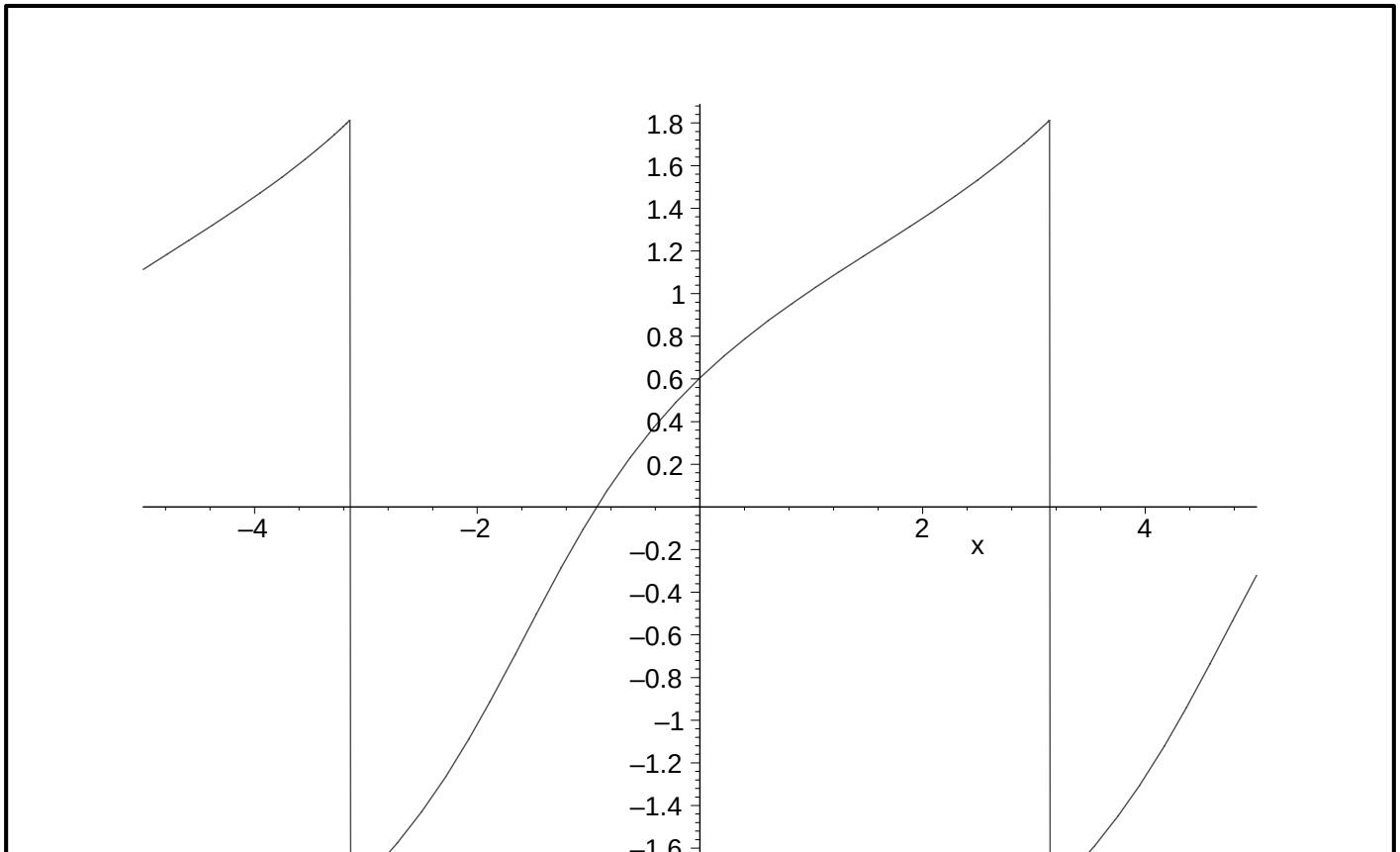
```
> # Example by Corless and Jeffrey
> f := 1/(sin(x) + 2);
```

$$f := \frac{1}{\sin(x) + 2}$$

```
> g := int(f, x);
```

$$g := \frac{2}{3}\sqrt{3}\arctan(\frac{1}{3}(2\tan(\frac{1}{2}x) + 1)\sqrt{3})$$

```
> plot(g, x=-5..5);
```



Early termination strategies

Early termination in Newton interpolation

For $i \leftarrow 1, 2, \ldots$ *Do*

Pick ***random*** p_i *and from* $f(p_i)$

compute

$$f^{[i]}(x) \leftarrow c_0 + c_1(x - p_1) + c_2(x - p_1)(x - p_2) + \cdots$$
$$\equiv f(x) \pmod{(x - p_1) \cdots (x - p_i)}$$

If $c_i = 0$ *stop.*

End For

Threshold η :
In order to obtain a better probability, we require $c_i = 0$ more than once before terminating.

The early termination of Ben-Or/Tiwari's interpolation algorithm.

If $p_1, \ldots, p_n$ are chosen randomly and uniformly from a subset S of the domain of values then for the linearly recurrent sequence

$$a_i = f(p_1^i, \ldots, p_n^i), i = 1, 2, \ldots$$

the Berlekamp/Massey algorithm encounters $\Delta = 0$ (when $2L < r$) the first time for $r = 2t + 1$ with probability no less than

$$1 - \frac{t(t+1)(2t+1)\deg(f)}{6 \cdot \text{cardinality}(S)},$$

where t is the number of terms of f.

Threshold ζ:
In order to obtain a better probability, we require $\Delta = 0$ (when $2L < r$) more than once before terminating.

Show Maple worksheet now.

Problem 5

Provide reasonable correctness specifications for our systems in the presence of floating point numbers, randomizations, and multivalued functions.