Impact of Computer Algebra on Physics

D. Waxman

School of Mathematical and Physical Sciences
University of Sussex, Brighton BN1 9QH, Sussex, U.K.

1. Introduction

Physics is, by its nature, a mathematically based subject and its laws are expressed in a concise mathematical form. Many tools from mathematics are used to find these laws and once these laws are established, physicists exploit the fact that using mathematical reasoning allows us to infer consequences of these laws in new situations. For this reason, a significant fraction of a physicist's time is spent, both in training and beyond, learning and applying mathematics.

Until recently, the only mathematical assistance that physicists have had is in problems of a numerical nature ("number crunching"), where computers have been employed. There have, however, been developments along different lines in the field of computation in the last few years. With the availability of cheap yet quite powerful personal computers and with advances in programming, it is now possible to do far more mathematics on computers than that which comes into the category of numerical mathematics. In this article I discuss one of the relatively new computational tools that can simplify (or even eliminate) much of the mathematical manipulations that physicists do. This tool is called "Computer Algebra" or sometimes "Symbolic Computation." The most fundamental aspect of computer algebra is the ability of a computer to manipulate *symbols* such as x and y rather than numbers such as 0.1377 and 5.2342. Thus while conventional programs can efficiently perform numerical operations such as evaluating $(0.1377 + 5.2342)^3$, a computer algebra program can, amongst many other things, expand $(x+y)^3$ for *general* x and y and will produce the symbolic answer, namely $x^3+3x^2y+3xy^2+y^3$. For such a simple example it is, of course, possible to perform the elementary expansion with a pencil and paper, but it would be tedious and daunting to (correctly) evaluate the coefficient of e.g. x^{39} in $(1 + x)^{23}/(1 + 2x)^{17}$ by hand; with a computer algebra program it is straightforward and virtually effortless[1].

---

[1] The sequence of interactive operations necessary to find the coefficient of x^{39} in $(1+x)^{23}/(1+$

In short, much of the tedious algebra and other mathematics (see below) that has to be performed to get at answers "of direct physical interest" may be relegated to computers and I will explore some of the implications of computer algebra systems for Physics.

This article is aimed at a reader who has little or no experience of computer algebra and gives a brief glimpse of the great potential of programs of this type, indeed I hope that after reading this article, readers will be sufficiently interested to go out and try a computer algebra program.

I should say that what this article is not is a comparison of different computer algebra programs that are in existence; most of the computer algebra programs available are able to admirably cope with the small number of topics covered here. In order to illustrate the simplicity of the instructions needed to command one of these programs, I have given examples of these in the language of the computer algebra program Maple [1], which is the one I am most familiar with.

2. Visualisation

Before we discuss some of the direct mathematical capabilities of a computer algebra program, I wish to consider an important aspect of these programs that their name does not immediately suggest. This is their ability, via graphics, to give detailed insight into a mathematical object. With a simple one line command we can instruct the computer to plot (in 2 or 3 dimensions) a function we have been "working on." The graph appears as a coloured image on the computer screen a short time after the command has been issued and it is a matter of one or two key strokes to change parameters or the range plotted to see new aspects of the function. Repeated variations of parameters and replotting[2] leads us to acquire an insight into the shape of the function and its typical variation. We may indeed get to "know" the important aspects of the function and the insight gained in this

---

$2x)^{17}$ in a typical computer algebra program is to

   (i) enter the expression,

   (ii) type the instruction to expand the expression about $x = 0$ to $O(x^{40})$

   (iii) read off the required coefficient (which happened to be -398059188302289174528: It is perhaps needless to say that I have not checked this answer by hand!).

  The actual time the computer runs for this straightforward example is a fraction of a second; typing the expression takes longer! Of course more demanding calculations or more complex operations may take longer.

   [2]It is possible to produce moving plots, where e.g. a parameter in a function is varied with time and the successive plots are merged into a "movie."

way may well be one of the most profound benefits of such a program.

To give an example of the graphical abilities of one of these programs, let us consider a well known subject from mechanics[3].

2.1. Damped harmonic motion

For this case the particle, with coordinate $x(t)$ (t = time), is taken to obey Newton's equation of motion. With a mass of unity, this has the form

$$\frac{d^2x(t)}{dt^2} + 2\gamma\frac{dx(t)}{dt} + \omega^2 x(t) = 0 \tag{1}$$

and the solution of this that starts at the origin at time zero with unit velocity is, for the special case $\omega = 1$:

$$x(t) = \frac{e^{-\gamma t}\sin(t\sqrt{1-\gamma^2})}{\sqrt{1-\gamma^2}}, \qquad \gamma^2 \leq 1. \tag{2}$$

This is a function of two variables: γ and t whose character may be succinctly illustrated by a three dimensional plot of x against t and γ and in Fig. 1 we give

---

[3]High quality (publication quality) prints of graphs are generated by Maple if a postscript printer is used. Having a deskjet printer in my office made it more convenient to produce prints on this machine with a numerical package and in fact all the figures in this paper were produced in this way. At the present time Maple does not give full control on all aspects of figures (such as axis line-thickness) on a deskjet printer.
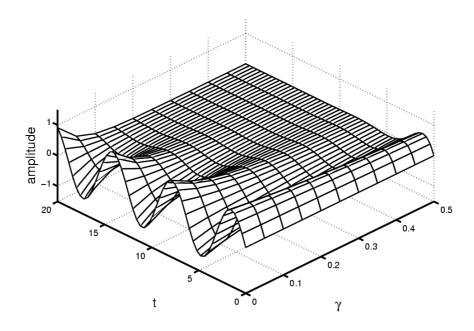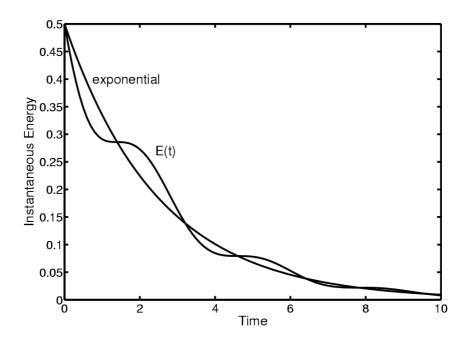
the plot of this quantity.

Figure 1

The plot illustrates, in a single view, the oscillatory character of the solutions for small γ and the damped character at larger γ.

Returning to the physics problem of damped motion, we are also in the position to see how the damping dissipates the energy of the particle as time progresses. For this we produce a two dimensional plot of the energy

$$E(t) = \frac{1}{2}\left(\frac{dx}{dt}\right)^2 + \frac{1}{2}x^2 \tag{3}$$

associated with the motion. The graphing facilities allow more than one curve to be plotted on the same page and in Fig. 2 we have plotted the energy (3) along

with the exponential curve $\frac{1}{2}e^{-2\gamma t}$ for damping parameter $\gamma = 0.2$.

Figure 2

Most textbooks assume or approximate the exact result for the instantaneous energy, $E(t)$, by that of the exponential curve. What we learn from this graph is that the decay of the energy is not a pure exponential curve but an oscillation about one. Thus with a relatively straightforward application of a computer algebra program, we can see the validity of a standard approximation and, if we wish, go beyond this. Possibly, the simplest idea that comes to mind is to look at the ratio of the two functions. We can thus plot $E(t)/\left(\frac{1}{2}e^{-2\gamma t}\right)$ as a function of t. In Fig. 3 we have plotted this quantity and a simple oscillatory function is
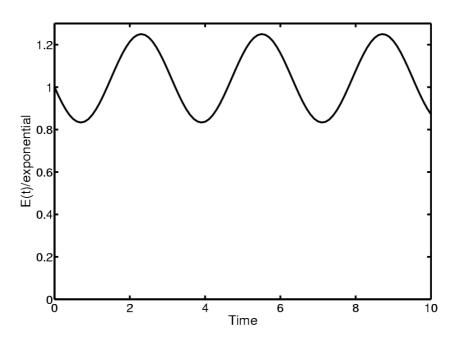
found.

Figure 3

This could spur us on to perform more analytical work (also possible on the computer) and in this way a rather full picture of what is going on may be obtained.
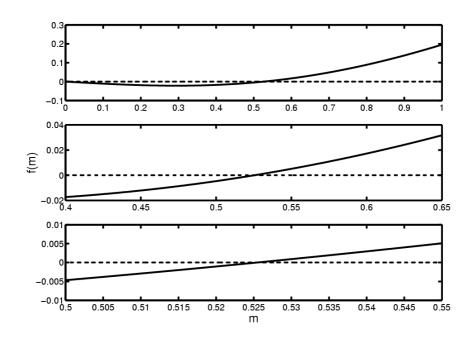
2.2. Zero of a non-linear function

Another case where the graphics facilities can come into play is when we want to find the zero of a non-linear function. In this case numerical packages exist within the program to deal with such problems however it is instructive (and sometimes even essential) to see explicitly what the function is doing in the vicinity of the zero. A "hands on" approach to this problem is to plot the function, see where it passes through zero, then replot the function in a smaller interval roughly centred on the place the function passes through zero. Repeating the process progressively focuses in on the crossing point until the zero is determined to sufficient accuracy.

Let us consider an explicit example where we require the zero of a function. In Statistical Mechanics, the determination of the magnetization of a mean field model of ferromagnetism in the absence of an external magnetic field [2] requires

the solution of a non-linear equation. Apart from being of interest in its own right, the magnetization determines a number of important thermodynamic properties of the ferromagnet. It is found that a mean field model leads to a relation between the magnetization m and temperature T (when expressed in appropriate units) given by the equation

$$m - \tanh(m/T) = 0. \tag{4}$$

This is to be solved for the non-zero value of m (if it exists) that satisfies the equation; here we shall do so for a single temperature.

With the choice $T = 0.9$, we have produced plots of $f(m) \equiv m - \tanh(m/T)$ versus m. In Fig. 4 plots with three succesively smaller intervals of m that include the zero of $f(m)$ are given.

Figure 4

By reading from the final graph the value of m at the crossing point we are able to estimate its value as 0.52...Thus with nothing that is really recognizable as programming, we have been able to solve for the zero of a non-linear equation!

2.3. Probability distribution for the quantum mechanical oscillator

The graphics facilities of computer algebra programs allow us to produce textbook-like graphs of quantities that we may currently be working on or studying, with the remarkable advantage that we are in full control of what is plotted. An example of this that we shall consider is the probability distribution of the quantum mechanical harmonic oscillator associate with the potential $V(x) = \frac{1}{2}m\omega^2 x^2$. For this case the probability distribution is the modulus squared of the wavefunction and when the oscillator lies in the energy eigenstate labelled by quantum number n ($n = 0, 1, 2, 3...$), its probability density is given by [3]

$$P_{qm}(n, x) = \frac{1}{\sqrt{\pi}2^n n!} [H_n(x)]^2 e^{-x^2}$$

when we have set the quantities $\hbar$, m and ω associated with the oscillator to unity and where $H_n(x)$ is a Hermite polynomial.of order n. In Fig. 5 the probability density is plotted for the relatively high quantum number, $n = 20$, along with result following from classical mechanics when the same energy has been selected:

$$P_{cl}(n, x) = \begin{cases} \dfrac{1}{\pi}\dfrac{1}{\sqrt{2n + 1 - x^2}}, & x^2 \leq 2n + 1 \\[4mm] 0, & x^2 > 2n + 1. \end{cases} \tag{5}$$
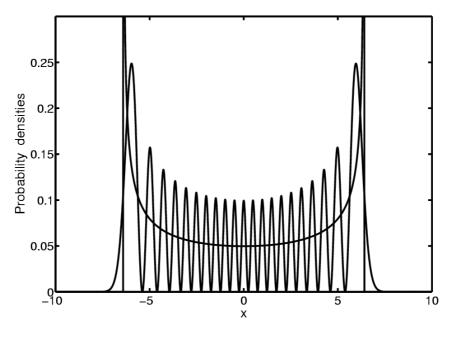
Figure 5

We can go further. Suppose we assume the actual distribution observed is not the straightforward quantum mechanical result but rather a "smeared" result obtained by adding contributions from a finite range, L, of x associated with imprecision of our measurement apparatus. This might be represented as a convolution

$$P_{smeared,qm}(n,x) = \sqrt{\frac{1}{\pi}} \frac{1}{L} \int_{-\infty}^{\infty} dx' \exp\left[-(x-x')^2/L^2\right] P_{qm}(n,x')$$

and we can use the computer algebra program to both analytically evaluate the integral (because it is a polynomial $\times$ a gaussian) and plot it. In Fig. 6 we give

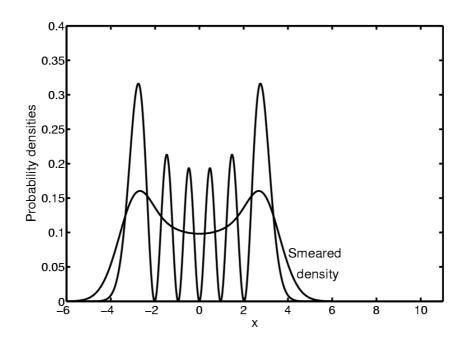a plot of $P_{smeared,qm}(5,x)$ for $L = 1$ along with $P_{qm}(5,x)$.

Figure 6

The smeared distribution plotted has greater similarity to the classical distribution than the quantum mechanical one as a result of the choice made for L, the range of smearing . Progressively reducing L causes $P_{smeared,qm}(n,x) \rightarrow P_{qm}(n,x)$, as the interested reader may verify.

The exploration of topics that by hand are more than a little tedious may, as the above example shows, be readily performed with a computer algebra program.

3. Interactive mathematics

In the introduction, section 1, two simple examples were given of calculations that all computer algebra programs can perform. These may seem impressive when met for the first time, but it has to be realised that straightforward algebra (as the examples were) is only the merest tip of the iceberg as far as these programs go. Currently available programs are remarkably comprehensive, having the capability to deal with many many more branches of mathematics than almost

any individual (student, teacher or professional) is likely to meet in their entire career and it would not be appropriate or interesting to list the topics covered by such program; basically, unless a mathematical subject is remarkably esoteric, it will be included in the package! Instead, I shall merely choose some examples that illustrate the ability of these programs to deal with a few different areas of commonly met mathematics. What is particularly interesting is the *interactive way* such a program can be used to carry out the mathematical manipulations.

3.1. Matrix problem

Let us consider a matrix problem as an example.

It is very common in Physics to have solve for the eigenvalues of a matrix. A basic case occurs when we have coupled oscillators in classical mechanics. The equations of motion for the oscillators are written as

$$\frac{d^2 x_1}{dt^2} + \omega_1^2 x_1 + \epsilon x_2 = 0$$

$$\frac{d^2 x_2}{dt^2} + \omega_2^2 x_2 + \epsilon x_1 = 0$$

(6)

where ω_1 and ω_2 are the angular frequencies of the uncoupled oscillators and the coupling between the oscillators is given by ϵ. To determine the new fundamental frequencies of this system (the frequencies of the normal modes), we are led by textbook methods [4] to solve for the eigenvalues of the matrix

$$A = \begin{pmatrix} \omega_1^2 & \epsilon \\ \epsilon & \omega_2^2 \end{pmatrix}.$$

(7)

Can a computer algebra program help us here ? The answer is yes. Once the matrix is entered into the computer, we type (in Maple) the command "eigenvals(A)" and what is returned to the user are analytical results for the two eigenvalues. One of these is given by the computer as[4]

$$\lambda_1 = \frac{1}{2}\omega_1^2 + \frac{1}{2}\omega_2^2 + \frac{1}{2}\sqrt{\omega_1^4 - 2\omega_1^2\omega_2^2 + \omega_1^4 + 4\epsilon^2}$$

(8)

---

[4]We should notice that the eigenvalue in (8) could have been written in the simpler (or nicer form) $\frac{1}{2}\left(\omega_1^2 + \omega_2^2 + \sqrt{(\omega_1^2 - \omega_2^2)^2 + 4\epsilon^2}\right)$; however the program does not have much aesthetic sense and the user has to have some interaction with the computer, giving various instructions, to get the result in a desired form.

and the eigenvector that corresponds to this can, with the same ease, also be obtained.

Suppose for the problem in hand, we know that $\omega_1 \gg \omega_2$. This suggests we try an asymptotic expansion[5] to see the leading correction to the uncoupled result. The instruction in Maple "asympt($\lambda_1, \omega_1, 4$)" immediately[6] yields the result

$$\lambda_1 = \omega_1^2 + \frac{\epsilon^2}{\omega_1^2} + O(\frac{1}{\omega_1^4}). \tag{9}$$

If we need to perform the asymptotic expansion, to a higher order, say 6, then we simply type in "asympt($\lambda_1, \omega_1, 6$)" and the answer, which would by hand involve far more work, is automatically produced[7]. Given the results of (8) or (9), we can now manipulate them as desired, e.g. substituting in numerical values or performing further manipulations.

Thus in the above way we can interactively carry out mathematics with a computer algebra program.

It is commonplace, when using computer algebra programs, to generate expressions on the computer that are tens (or more) of pages long. These never normally see the light of day, since usually only a small piece of the final expression is required. Nonetheless, to perform all the steps leading to a compact result could take an enormous effort. Because we avoid writing down any of the intermediate steps[8] (which, if desired, can be saved by the computer) and because of the ability of the computer to rapidly perform fault free calculations on expressions of great complexity and length, computer algebra programs allow us to perform calculations that are outside the possibilities of most individuals calculating by hand. Thus just as the use of computers for numerical calculations opened new realms that were inaccessible by hand calculations, computer algebra opens a different set of realms involving symbolic results.

---

[5]An asymptotic expansion of λ_1 is an expansion of this eigenvalue in negative powers of ω_1 assuming ω_1 is large.

[6]The Maple instruction means determine the asymptotic expansion of λ_1 in ω_1 up to $O(\omega_1^{-4})$.

[7]We find the result $\omega_1^2 + \dfrac{\epsilon^2}{\omega_1^2} + \dfrac{\frac{1}{8}\omega_2^2(\omega_2^4+4\epsilon^2)+\frac{1}{4}(\frac{1}{2}\omega_2^2+2\epsilon^2)\omega_2^2-\frac{1}{4}\omega_2^6}{\omega_1^4} + O\left(\dfrac{1}{\omega_1^6}\right)$. Clearly this would a significant amount of work if carried out manually.

[8]I know of an explicit proof showing the equivalence of two different calculational schemes in quantum field theory that are made with the aid of one these programs where the intermediate results are not presented, only the end equivalence being interesting [5].

4. Integer and infinite precision arithmetic

Let us look at another feature of computer algebra programs. They can do integer arithmetic and also arithmetic to any pre-assigned precision.

Integer arithmetic is the ability to manipulate and simplify quantities such as $\frac{1}{2} + \frac{1}{3} + \frac{1}{279}$ without expressing the answer as a decimal. Thus in this case we obtain from the computer the result

$$\frac{1}{2} + \frac{1}{3} + \frac{1}{279} = \frac{467}{558}. \tag{10}$$

Perhaps more interesting is the infinite precision capability. What this means in practice is that by a simple command, all calculations are performed to e.g. 39 digits after the decimal point (39 can be replaced by any positive integer). As an example, to 39 digits precision,

$$4/\pi = 1.273239544735162686151070106980114896276... \tag{11}$$

We can use this facility in cases where something numerically subtle is occurring and the extra precision is needed. A simple example comes from chaos [6] where the nth iterate of the so called logistic map obeys

$$x_{n+1} = cx_n(1 - x_n) \tag{12}$$

and one uses this equation by starting with one value of x, say x_0 and using the above equation to determine x_1 and then substituting this back into the equation to generate x_2 etc.

When $c = 4$ the iterates exhibit chaotic behaviour and the nth iterate can be shown to take the form

$$x_n = \sin^2(2^n \theta_0) \tag{13}$$

where θ_0 is a constant, determined by the starting point, x_0. Here take $\theta_0 = 1$. We find for varying numerical accuracy in the evaluation of e.g. x_{40} quite different results:

$$
\begin{array}{lll}
x_{40} = 0.05862 & \text{to 5 digit accuracy} & \\
x_{40} = 0.9581109402 & \text{to 10 digit accuracy} & \\
x_{40} = 0.915671112188 & \text{to 12 digit accuracy} & (14) \\
x_{40} = 0.164596556382853 & \text{to 15 digit accuracy} & \\
x_{40} = 0.16459655638285268... & \text{to higher accuracy.} &
\end{array}
$$

The controllable precision of a computer algebra program has allowed us to show that a chaotic system is seriously susceptible to rounding errors. Furthermore, by considering x_n for sufficiently large n it may be illustrated that these rounding errors will overcome the preassigned precision of any computer.

5. Numerical packages

Typical computer algebra programs contain a number of very user-friendly numerical packages. Here we shall only discuss the package that numerically solves differential equations. Let us consider the non-linear ordinary differential equation:

$$\frac{d^2\theta(t)}{dt^2} + \sin\theta(t) = 0. \tag{15}$$

This equation describes the variation of θ, the angle of a pendulum in radians, with an appropriately time variable t. If $\theta(t) \ll 1$, the nonlinear component in the equation, $\sin\theta(t)$, may be replaced by its argument and we recover the familiar harmonic oscillator equation whose solutions are a combination of $\sin t$ and $\cos t$. In general, however, we have to deal with the intrinsically non-linear equation, (15). A numerical solution of this may be found using a differential equation integrator. It only takes a one line command to numerically solve this equation and two lines to instruct the computer to solve and plot the solution. In Fig 7, we present a plot of the solution of (15) that satisfies the initial conditions $\theta(0) = 3.1$, $d\theta/dt|_{t=0} = 0$ along with the plot of the solution of the linearised equation that is

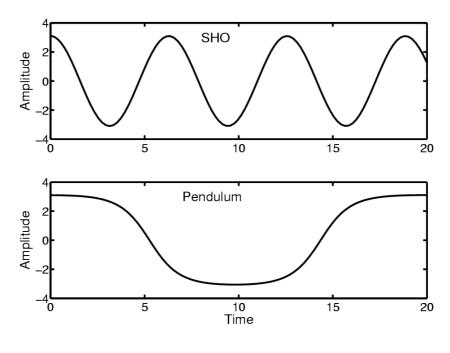subject to the same boundary conditions, namely $3.1 \cos t$.

Figure 7

In point of fact the solution of (15) is one of the "special functions" of mathematical physics - an elliptic function. Many of the special functions are defined by differential equations and as we have seen above, if we require only numerical information on these, it is as easy to use a computer algebra program to solve the differential equation as it is to look up or evaluate the special function in question.

6. Comparisons with conventional programs

As outlined above, computer algebra programs contain quite extensive numerical packages. Given the ease with which mathematical objects may be input into the computer in one of these programs, it makes it a very natural place to test out new ideas. The question that arises is given this user friendliness along with a large set of symbolic and numerical packages in one program, do we *only* require a computer algebra program for all our mathematical problems? The answer to this question depends on the user's requirements. Computer algebra programs are not

well suited to numerically intensive work; they are relatively slow compared with purely numerical programs. Thus e.g. the repeated solution of a set of differential equations over long time scales would probable be best delegated to a numerical program. It should be noted that quite often the work spent typing an equation into a computer algebra program for a quick trial of an idea may not be wasted since facilities exist in these programs for outputting the mathematics in another language (for example Fortran or C).

7. Drawbacks

7.1. General comments

It has often been said that pocket calculators were the downfall of arithmetic (or more precisely arithmetic skills). Inevitably, computer algebra programs will be blamed for the loss of the manipulative skills required of algebra. Taking myself as an example, I have to say this seems very likely. If I have to calculate the determinant of a 4×4 matrix, I prefer to type the matrix into the computer and let the "calculating engine" do the work; it is far quicker than doing it by hand and human algebra is certainly fallible. I will also allow the computer to have first attempt at integrals that are non-obvious[9]. This has inevitably resulted in a lowering of my manipulative skills, but given a computer at hand with an algebraic program, I believe my productivity has actually increased and I do not see computer algebra as a disadvantage provided one has a firm understanding of the mathematics.

7.2. Hidden assumptions

The hidden assumptions made by computer algebra programs are likely to be a source of error to the unwary.

Let us consider the following situation. A complicated algebraic analysis has been carried out and an analytic expression for a quantity "a" has been obtained. This then appears in the integral $\int_{-\infty}^{\infty} \exp(-ax^2)dx$. Naively, the integral has the value $\sqrt{\pi/a}$. Some computer algebra programs will yield this answer, while others

---

[9]The evaluation of integrals using algebraic programs is a slightly dangerous strategy since integrals are notoriously subtle. Complications such as the "choice of branch" of a root may result in the program yielding errors. Fortunately, these programs are also able to numerically evaluate many integrals and this may be used as an independent check of any analytic results.

will refuse to calculate an answer and yet others will pose questions concerning "a" before evaluating the integral. In fact the value of the integral depends sensitively on "a":

$$\int_{-\infty}^{\infty} \exp(-ax^2)dx = \begin{cases} \infty & \mathrm{Re}(a) < 0 \\ \infty & \mathrm{Re}(a) = 0, \ \mathrm{Im}(a) = 0 \\ \sqrt{\pi/a} & \mathrm{Re}(a) = 0, \ \mathrm{Im}(a) \neq 0 \\ \sqrt{\pi/a} & \mathrm{Re}(a) > 0. \end{cases} \tag{16}$$

Thus the blind application, in this case, of the (incomplete) answer $\sqrt{\pi/a}$ could lead to a nonsensical result.

Another example that different computer algebra programs treat differently are square roots. If x is real, does $\sqrt{x^2}$ have the value x or $|x|$. These differ when x is negative and some programs will automatically simplify $\sqrt{x^2}$ to x with possible errors.

These cases alert us to the possibility that hidden or non obvious assumptions may be made by computer algebra programs and these may be a source of possible error.

8. Discussion

I have pointed out above that computer algebra programs allow us to carry out more complex manipulations then we could do on our own. This allows all users (professionals and students) to tackle harder and more realistic problems, not simply restricted to linear or other exactly soluble problems. In the field of teaching this means that there will have to be significant changes in approach: it is pointless to set an exercise that can be done in seconds with the aid of the computer - and hours when done by hand[10]. The additional intuition about mathematics that one of these programs can provide needs also to be exploited in teaching, and hopefully this will compensate for the loss of skills in algebraic manipulation that will accompany these programs.

Given the remarkable wealth of mathematical material contained in one of these programs, they seem likely to turn the notions associated with computers and mathematics on their head. Instead of a computer being used to carry out *known* mathematics, computer algebra programs, with their extensive help facilities, will act as *source* of mathematical knowledge, for example, the user who

---

[10]The free availability of these programs means that students may already be using computer algebra to solve their homework problems.

knows a little of matrix theory (linear algebra) will, in the help sections find all sorts of special matrices and perhaps not so familiar operations on matrices. There is some room for improvement in this area; while computer algebra programs have an extensive help section, it is not really a "reference" section where the detailed properties of mathematical functions etc. are held. The incorporation of such a section would make a computer algebra program a largely self contained source of mathematics, allowing it to replace a handbook of mathematical functions such as that of Abramowitz and Stegun[7].

I have touched on only a few of the literally thousands of capabilities a computer algebra program has and have indeed not touched on the more conventional programming facilities that these programs can also support. It is likely that everyone who uses one of these programs will use only a small subset of the packages contained within it and it seems clear this comprehensiveness coupled with their ease of use means that they will have a lasting impact on the way physicists (and others) do and teach mathematics.

References

[1] Maple is a product of Waterloo Maple Software, 450 Phillip street, Waterloo, Ontario, Canada N2L 5J2.

   It may be purchased in the UK from Clecom, The Research Park, Vincent Drive, Edgbaston, Birmingham B15 2SQ.

[2] T Guenault, "Statistical Physics," Routledge, London, 1988. See pages 143–146.

[3] P Davies and D Betts, "Quantum Mechanics," 2nd Edition, Chapman and Hall, 1994. See pages 37–40.

[4] P Kahn, "Mathematical Methods for Scientists and Engineers,", Wiley, New York,1990. See page 24.

[5] B Sendhoff, Diplomarbeit, Theoretische-Physik, Ruhr Universität Bochum.

[6] G Rowlands, "Non-Linear Phenomena in Science and Engineering, Ellis Horwood, New York, 1990. See chapter 7.

[7] M Abramowitz and I Stegun,"Handbook of Mathematical Functions," Dover, New York, 1965.