Computer Based Maths

David Vaccaro

April 10, 2017

Abstract

This paper is written partly in response to Conrad Wolfram's 2010 TED Talk *Teaching Kids Real Math with Computers*. The aim is to distinguish between problems which are mere calculations and those which promote genuine mathematical understanding. I will argue that mathematical understanding can be promoted both by "traditional" abstract mathematics as well applications. In both cases I believe that computer algebra systems can be useful to ensure that the focus is on mathematical problem solving rather than algorithmic procedures.

1 Mathematics and Calculating

1.1 Two problems.

The following question is taken from a C2 examination paper (MEI, June 2013).

1) The gradient of a curve is given by

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \frac{18}{x^2} + 2$$

The curve passes through $(3, 6)$. Find the equation of the curve.

This is a relatively routine question on integration. However, while most candidates will know what to do, their level of success will vary, largely due to the technical difficulties presented by dealing with the negative power of x. Indeed it is a negative power which distinguishes this question from simpler integration questions on C1, while a more advanced question from C3 or C4 on the same topic would involve trigonometric, exponential or logarithmic functions which would result in even greater algebraic challenges. A significant part of A-Level assessment seems to be a test of pupils' computational proficiency. However does a failure to cope with the technical details of calculations really point to a lack of understanding?

It is probably familiar to everyone here that a simple natural language query in *WolframAlpha*, say, would generate not only the correct answer to any routine problem of this type but also a full "step-by-step" solution. The natural question is why pupils are seemingly being tested on simply performing calculations which can done more reliably using a computer? Indeed some, like Conrad Wolfram, may even question why "traditional" maths like this is still taught in schools at all. In his 2010 TED talk *Teaching Kids Real Math with Computers* he advocates that schools replace all calculations done by hand in favour of hard "real life" problems done with the aid of a computer.

I will return later to Conrad's Wolfram's suggestion of using Computer Algebra Systems in an applied context, but before doing so let us look at a second type of maths problem. This question is taken from an introductory set of problems for undergraduates starting at Oxford University. It most definitely does not satisfy Conrad Wolfram's call for practical problems as it is extremely abstract in nature. However, I suspect that many here will agree that its solution requires the skills of creative problem solving, logical reasoning and clear expression which makes mathematical studies so prized within academia and industry.

2) Show that if a, b and c are odd numbers and x is a real root of the quadratic equation

$$ax^2 + bx + c = 0$$

then x is irrational.

First of all it is clear that the question is beyond a direct assault from *WolframAlpha*, and its solution is by no means routine. Indeed, I would argue that while it is on a more elementary topic than the previous problem, it is not accessible to the overwhelming majority of school pupils, even those scoring close to full marks on A-Level Modules. Let us look in depth at the solution to this problem in order to highlight both why such a problem is currently out of reach to most pupils, and ultimately I seek to show why Computer Algebra Systems could be helpful.

Solving the problem

Firstly it is apparent that the roots of the quadratic equation $ax^2+bx+c=0$ are rational precisely if the discriminant b^2-4ac is a perfect square. It is not obvious why this cannot be the case if a, b and c are all odd, but a natural first step would be to play around with some values for a, b and c, calculating the discriminant and try to see why the resulting numbers could never be a square. When mentoring high achieving pupils preparing for school Olympiad Competitions one of the things I find myself needing to do most often is to encourage candidates to experiment rather than to expect to be able to see a solution straight away. Here are some values for the discriminant:

a	b	c	b^2-4ac
1	1	1	-3
1	3	1	5
1	5	3	13
1	5	1	21
1	7	5	29

An interesting pattern has indeed emerged, in that all discriminants are all three less than a multiple of 8. Experimenting with more values this seems always to be the case, but can we prove this conjecture?

If we write the odd numbers a, b and c in the form $a = 2p+1$ and $b = 2q+1$ and $c = 2r+1$ then a "simple" algebraic calculation gives:

$$\begin{aligned} b^2-4ac &= (2q+1)^2 - 4(2r+1)(2s+1) \\ &= 4q^2+4q+1-4(4rs+2r+2s+1) \\ &= 4q^2+4q-16rs-8r-8s-3 \\ &= 4q(q+1)-16rs-8r-8s-3 \end{aligned}$$

We wanted to show that this is three less than a multiple of 8. However as one of q and $q+1$ is even, $q(q+1)$ is even, and $4q(q+1)$ is a multiple of 8. The terms $16rs$, $8r$ and $8s$ are clearly multiples of 8, so the discriminant is indeed three less than a multiple of 8. This proves our conjecture, but it has not really solved the initial problem, as it is not clear why such a number could not be a perfect square.

We do know, however, that as the discriminant is three less than a multiple of eight, the discriminant is itself odd, so if it were a perfect square, then it would be the square of an odd number. A natural next step is to look at the odd square numbers, and see how they compare to multiples of 8.

x	x^2	
1	1	1
3	9	8+1
5	25	24+1
7	49	48+1
9	81	80+1
11	121	120+1

This seems to suggest that each odd square is always one more than a multiple of 8. Can we prove this conjecture? Well if we take the odd number $2n+1$ and square it then we get:

$$\begin{aligned}(2n+1)^2 &= 4n^2 + 4n + 1\\ &= 4n(n+1) + 1.\end{aligned}$$

Arguing as before we can say that $n(n+1)$ is even and $4n(n+1)$ is a multiple of 8. So we indeed have that every odd square number is one more than a multiple of 8. However as we know that the discriminant is three less than a multiple of 8, we know that the discriminant cannot be a perfect square number. So the roots are indeed irrational. Once a solution has been found it is likely that one would want to look back over the argument and seek to write things out as succinctly as possible. In particular the numerical experimentation, which was central to the finding the solution, is normally not included in the write-up. If I spoke earlier about children's reluctance to approach a problem with an initial period of experimentation then I as a school teacher need to bear some of the responsibility, as techniques like "guess and check" and "trial and error" are often presented as infantile compared to systematic algebraic approaches to problems.

It is easy to see why this question was a suitable introduction to University Level mathematics. It was a genuinely challenging problem whose method of solution was by no means immediately apparent. Like all such problems it would be hard to imagine *WolframAlpha* ever being able to give a computer generated "step by step" proof, not least because it is amenable to many different approaches, I suspect that lots of people would offer different solutions which are more elegant than the method described above.

While this question is at first glance about quadratic equations, a particular bete noir of Conrad Wolfram and his Computer Based Maths organisation, I feel that if schools produced pupils capable of solving problems like this, then it would be very hard to argue that school level mathematics was not doing its job well. However, the general consensus is that most school leavers are not robust problem solvers, and in analyzing the reason for this I agree that the problem is too much time spent in drilling procedures, like pen-and-paper calculus, but unlike Conrad Wolfram I am not convinced that one can completely write-off the need for abstract mathematics. What I feel is needed is a curriculum which equips pupils with the thinking skills necessary to solve mathematical problems, and I am convinced that computer algebra systems and computer programming are invaluable tools in making such problems accessible to a greater number of pupils.

2 Problem Solving Strategies

Looking back over our solution to the previous problem the overall strategy will no doubt be very familiar to all who have studied mathematics or who use mathematics to solve problems; the solution breaks neatly into three main stages:

1. Experiment (e.g. collecting numerical data).
2. Conjecture.
3. Proof.

Sadly, I think that this problem solving paradigm will be unknown to virtually all school pupils. It is not hard to see why such challenging problems are unfeasible within the mainstream school curriculum. Very few pupils would have the patience, stamina and computational alacrity required to perform sufficiently many calculations required to spot a pattern, nor the algebraic manipulation required within the proofs. I labeled the expansion of the discriminant above as "simple" but in reality for for the vast majority of children it would provide a significant technical challenge. If pupils are capable of doing calculations like this then it is largely due to the vast amount of time they have spent practising such calculations.

However I think all will agree that these two steps are conceptually the least important part of the solution. From a Mathematician's point of view the gathering of numerical data is a chore, and the algebraic manipulation is a routine procedure. All the real meat is in the logical reasoning, and rather optimistically I think that all the really important steps, including the underlying logical structure of the argument and the forming of conjectures are something which most pupils are capable of.

This situation is reminiscent of the historical situation when Mathematical discoveries were restricted to people like Euler and the Bernoullis who were blessed with almost superhuman calculating skills. The development of Mathematical theories has always gone hand in hand with computation. In contrast to the pristine and elegant proofs often presented in University lecture courses, historically mathematicians have always solved problems using a mixture of experimentation alongside what Bernoulli called *Ars Conjectandi* or the Art of Conjecture.

Today, however, because of the development of the computer, mathematics is no longer only the preserve of the super-numerate. Computational skills are no longer essential for quantitative professions. In the life sciences for example an understanding of how data is gathered and a nuanced understanding of which statistical test is appropriate in different situations is far more important than the ability to implement a statistical test by hand. Once the correct calculation has been decided upon it will always be done by a machine.

Like most people I can think of a large number of Classicist or Historian friends from University, who describe themselves as having been "bad at Maths" at school, who have gone on to work in the financial sector and who have proved themselves to be more than capable of using and interpreting computer based models on a daily basis.

However, this computational revolution which has democratized mathematics in the "real world" has not reached schools. School assessment still tests the ability to calculate accurately, and those pupils who lack computational ability are often disenfranchised from school examinations which, as noted earlier test the ability to reproduce routine calculations rather than the ability to reason mathematically.

3 Computer Algebra Systems

So here is the main the question: would Computer Algebra Systems help pupils tackle richer and more challenging problems? Is it possible for computer algebra systems and computer programming to take on the heavy-lifting of generating numerical data and performing algebraic manipulation, so that learners can focus on the part of the problem that is genuinely Mathematical?

My experience is that armed with a Computer Algebra System, like *Mathematica*, problems do indeed become approachable in a way that they would not have been beforehand. That is not to say that a hard problem open-ended problem becomes easy just because one has a computer. Indeed I suspect that initially a problem like the one presented earlier would need to be broken down into more manageable steps like.

1. Show that if n is an odd number, then $n^2 - 1$ is divisible by eight.
2. Find the discriminant, D, of the the quadratic expression $ax^2 + bx + c$ for a large number of odd values of a, b and c. Form a conjecture about the values that D can take?
3. Show that if a, b and c are odd numbers then $D + 3$ is divisible by 8.
4. Explain why if a, b and c are odd then the roots of $ax^2 + bx + c = 0$ are irrational.

Pedagogically, the goal of the teacher is to introduce the minimum level of scaffolding required, and to gradually work towards enabling pupils to solve the problem *for themselves*. In my experience, initially the tasks that are given to pupils are essentially an introduction to programming, for example calculating the discriminant for a range of odd values requires pupils to think both about how to represent odd numbers and how to set up suitable looping structures. These are transferable skills and once a certain degree of proficiency has been attained pupils can concentrate on problem solving.

4 Finding "Goldilocks" Problems

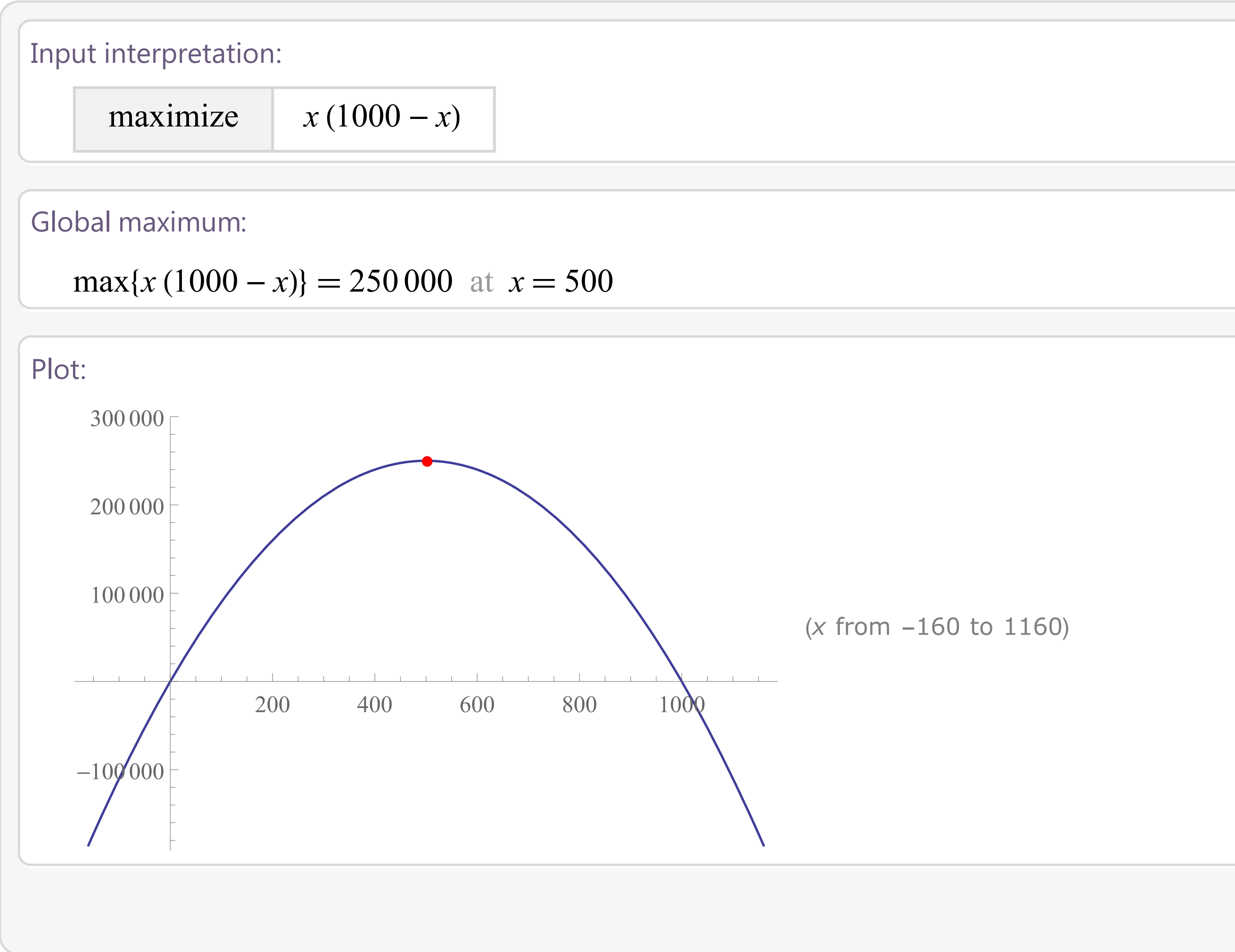
I have been using computer algebra systems within the classroom, especially Wolfram *Mathematica*, *Autograph* and *Geogebra*, for nearly ten years. Like most teachers, I initially used them as a way to demonstrate mathematical topics via graphical or dynamic interfaces, and I spent a lot of time doing programming as part of my lesson preparation. However, over the years my interest has changed, and now whenever I can I seek to use Computer Algebra Systems as a way to introduce problem solving. This has not proved to be altogether straightforward and the challenge is to find problems which are suitably rich so that they are rendered completely trivial by access to a computer (what I call "The *WolframAlpha* Test"), yet also accessible enough so as to be within reach of the majority of pupils.

4.1 An Example

Here is an examples of a problem that work well, serving as a great introduction to optimisation. While a full analytic solution would require a large amount of calculus, it has proved to be easily tractable to pupils in Year 10.

> A set of positive numbers add up to make 1000. What is the largest possible value of their product?

At first glance this problem is far too open ended for pupils to make any headway whatsoever, and I would encourage them to look at the simplest possible case where there are just two numbers. After some exploration, they will find that the maximum is attained if the two numbers are chosen to be equal to one another.



I would encourage pupils to try and explain why the maximum occurs when the two numbers are equal, and often they will pick out clues from the output from *WolframAlpha*, for example the complete square form. Natural curiosity also often attracts them to the unfamiliar symbols of Calculus.

It is not too hard a leap for pupils to realize that the maximum product would always be attained by setting all of the numbers to be equal, fo if the numbers were not all equal the product could be increased by setting two different numbers to their mean. What is not clear is how many numbers there should be within the set, this can be found using a computer. If the common values of all of the numbers were x then there would be $\frac{1000}{x}$ numbers included in the product, giving a total product of $x^{\frac{100}{x}}$.

This product ends up being maximized when x is approximately equal to some strange number $x \approx 2.7183\ldots$

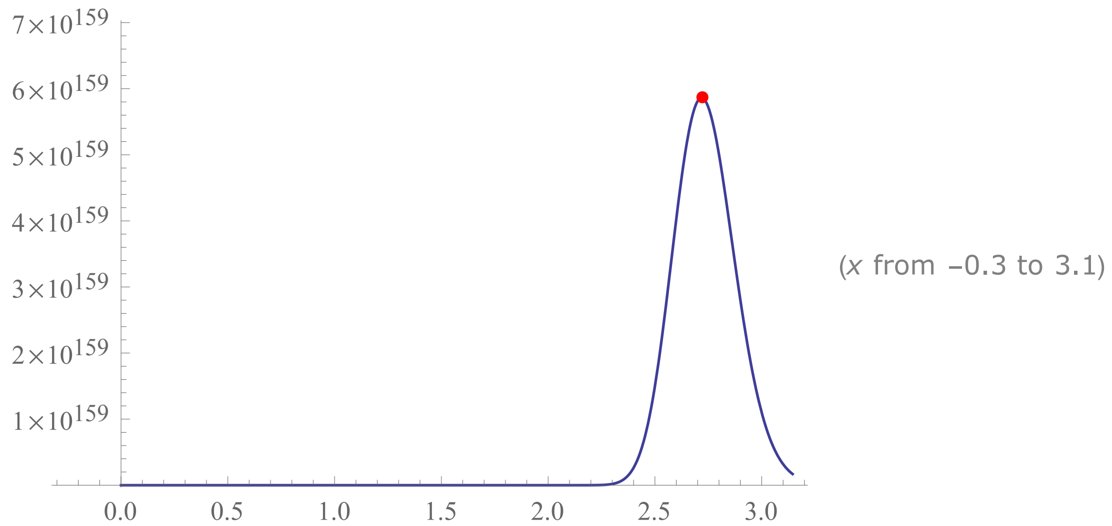
Input interpretation:

maximize	function	$x^{1000/x}$
	domain	$x > 0$

Global maximum:

$$\max\{x^{1000/x} \mid x > 0\} =$$
$$5\,861\,534\,242\,158\,375\,681\,775\,646\,093\,841\,747\,076\,798\,209\,657\,785\,578\,997\,893\,646$$
$$154\,384\,183\,459\,239\,863\,681\,993\,820\,827\,137\,876\,416\,266\,240 \text{ at } x \approx 2.7183$$

Plot:



Of course pupils investigating this problem will take various blind-alleys, and some wrong turnings which lead to interesting but unintended destinations, but hopefully this examples shows how with the aid of the computer “real ” mathematics (AM-GM Inequality and Euler’s Constant) is in very easy reach. Though I’d argue that the process of specializing and then generalizing is the most important problem solving approach contained within this.

One of the things I hope for most from this conference and the ongoing discussions about the role of Computer Algebra Systems is a compilation of a large number of similar challenging yet tractable problems.

5 Real World Problems

I mentioned earlier that I would return to Conrad Wolfram’s suggestion that there is no place for “traditional” Mathematics, which looks at artificially simplified problems, and pupils should cut their teeth on big, hairy real world problems. Within my own experience the most impressive pieces of work that my pupils have produced have indeed been on large open ended real world problems.

For the last four years I have run a week long summer school at the Royal Institution. The structure of the course is that after a crash course in *Mathematica* they are unleashed to work on a problem of their own choice. Some analyze data, some model a situation using a differential equation or stochastic simulation, and the best examples exceed anything that I would have imagined possible.

Once a certain degree of competence has been reached, pupils are genuinely able to research open ended questions. The problem solving cycle is slightly different to the experiment- conjecture- proof within Pure Mathematics, but is much more commonly involves modeling or data analysis. Data which is analyzed comes either from external data sources or from data generated by a random simulation. Successful projects have included:

1. **Modeling the evolution of a Ponzi Scheme**- The student set up a system of differential equations to model a Ponzi scheme.
2. **Modeling social newtorks using random graphs**- The student modeled a social network as a random graph on a lattice, and analyzed the size of connected subgroups.
3. **Modeling disease spread**- This is susceptible both to differential equations and random simulations.
4. **Optimal Strategies in Roulette**- The student was able to compare the results from different strategies both using random simulations and developing theoretical probability distributions.
5. **Hedging Strategies**- A student tested the effect of employing Black-Scholes Hedging Strategies against real stock market data.
6. **Experimental Analysis of Conway's Game of Life**- The student used data gathered from Conway's Game of Life to look at expected population size and population density.

Away from the Royal Institution I have built some resources which allows pupils to interact with real world data and build and refine Mathematical models. For example if planetary positions viewed from the position of the Earth have been given then pupils can really try to see if they, like Kepler, could spot any patterns to explain the perceived motion. As well as using external information, stochastic simulations can generate data to be analyzed, and Mathematics can end up feeling much more like an experimental science.

In practice I find that students need a certain amount of Mathematical maturity before they can tackle such research questions. While there is merit in Conrad Wolfram's view that school children can work on big, hairy, real world problems, it is most likely the case that some preparatory work needs to be done to give them the abstract reasoning skills necessary to formulate problems and translate them into a form suitable for a computer.

6 Conclusion

In this talk I have tried to distinguish between problems which require mathematical thinking from routine calculations. I have broadly agreed with the conclusion that too much time is spent in schools teaching and assessing the implementation of algebraic algorithms (by hand), and I hope to have made that case that computer algebra could enable pupils to spend their time on richer tasks, whether these are abstract or applied questions. Above all I feel that this can be done by promoting a more empirical, playful approach towards problem solving.