# Notes on Computer Systems for Solving Symbolic Equations

Richard J. Fateman

March, 1991, revisited 2005

**Abstract**

Math students learn to solve single equations symbolically beginning in elementary or junior high school. Some computer programs solve equations in ways quite different from the methods we were taught, and other programs simulate the elementary "human" methods at some level of control. Which is right? Which is more capable? Do we have to make a choice?

## 1 Introduction

This note[1] was inspired by a problem which appears among those posed in the British A-level examinations, and one which was solved by the program PRESS (see [1], [3]).

Solve the equation $\cos(x) + \cos(3x) + \cos(5x) = 0$ for $x$.

The reader is invited to try to solve the problem before proceeding.

[1]Originally written in March, 1991, and recently revised

## 2 A solution based on a cute heuristic

PRESS solves this by a trick that apparently will work for 3 cosine terms whose arguments are in arithmetic progression. Perhaps this is a well-known trick for A-level exams.

Notice that $\cos(x)+\cos(5x) = 2\cos(3x)\cos(2x)$. Therefore we can transform $\cos(x)+\cos(3x)+\cos(5x) \Rightarrow \cos(3x)(1+2\cos(2x))$ Then $\cos(3x) = 0$ and $1+2\cos(2x) = 0$ can be solved separately. The set of solutions becomes $\{\frac{1}{3}\arccos(0), \frac{1}{2}\arccos(-\frac{1}{2})\}$.

## 3 What PRESS actually produces

By simplifying the arccos, the set of solutions produced by PRESS is actually $\{60n+30, 180m+60, 180m-60\}$ (in degrees), for integers $n$ and $m$[2].

In the more usual radian measure, and simplifying somewhat, this could be expressed as $\{\frac{(2n+1)\pi}{6}, \frac{(3m\pm1)\pi}{3}\}$

## 4 The garden path

From the initial set of $\{\frac{1}{3}\arccos(0), \frac{1}{2}\arccos(-\frac{1}{2})\}$ one might be fooled into thinking there were two roots to the equation.

Since $\cos(x) = \cos(-x)$, each of the roots blossoms into two. Our roots set becomes $\{\pm\frac{\pi}{6}, \pm\frac{\pi}{3}\}$.

The arccos function is periodic, so we are free to add an integer multiple of $\pi$ to any root without changing the situation. By adding $\pi$ to $\{-\frac{\pi}{6}, -\frac{\pi}{3}\}$ we get $\{\frac{5\pi}{6}, \frac{2\pi}{3}\}$.

At this point it should become clear that there are an infinite number of solutions, and they include the following multiples of $\pi$: $\{\frac{1}{6}, \frac{1}{3}, \frac{2}{3}, \frac{5}{6}\}$.

Now that you've seen two sets of solutions, can you answer the question: "How many roots are in the interval $[0, \pi]$?"

Unfortunately, adding multiples of $\pi$ is not enough to get all the distinct arccos values. In particular another arccos(0) is $\frac{3\pi}{2}$ and this, divided by 3 means that $\frac{\pi}{2}$ is also a root. A careful examination of the values generated by PRESS demonstrates that they generate the same values, 5 in number in the given interval: the roots in $[0, \pi]$ are the following multiples of $\pi$: $\{\frac{1}{6}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{5}{6}\}$.

Another way of writing the root set might be $\left(\frac{n\pi}{6} \mid n \not\equiv 0 \pmod 6\right)$.

Plotting this function is actually fairly helpful.

## 5 What does Macsyma do?

In 1991 Macsyma did not solve this problem, but now it produces a solution set

$$\left\{x = \frac{\pi\ (12\,\%\text{n}20 + 5)}{6}, x = \frac{\pi\ (12\,\%\text{n}19 - 5)}{6}, x = \frac{\pi\ (12\,\%\text{n}18 + 1)}{6}, x = \frac{\pi\ (12\,\%\text{n}17 - 1)}{6}, x = \frac{2\,\pi\ (3\,\%\text{n}16 + 1)}{3}, x = \frac{2\,\pi\ (3\,\%\right.$$

Most likely it converted the problem to exponentials equation for $y = \exp(ix)$:

$$\frac{y^{10}+y^8+y^6+y^4+y^2+1}{2y^5} = 0$$

The numerator factors into

$$(y^2-1)(y^2-y+1)(y^2+y+1)(y^4-y^2+1).$$

Finding the zeros of these factors is easy since they are all quadratics but the last one. But that one is a quadratic in $y^2$ and is almost as easy.

[2] The form of the solution produced by the program does not appear in the literature (e.g. [3]), but was communicated to me by Bernard Silver (July 10, 1986).

Computing the logarithms of these results, and then coercing the answer to a more useful "rectangular" form $(a+bi)$, eventually gets a set of ten specific solutions (The ones in the interval $[-\pi,\pi]$ ). In the process, a specific value for $\arctan(\sqrt{3})$ was used. If a "generic" arctan is used, (For example, $\arctan(\sqrt{3}) = \frac{(3k+1)\pi}{3}$), then all the solutions are produced.

## 6 What does Maple do?

In 1991, Maple version 5, and in 2001, Maple version 7 respond as follows:

```
> solve(cos(x)+cos(3*x)+cos(5*x)=0,x);
                       1/2 Pi, 2/3 Pi, 1/3 Pi, 1/6 Pi, 5/6 Pi
```

This is rather nice (and version 7 typesets it on the display) although it still misses the fact that there are an infinite set of solutions. Perhaps a perusal of the code for the solve command (available on-line in Maple) would explain how this was done and what might be altered to make the solution more complete.

## 7 What does Mathematica do?

In 1991, Mathematica version 2.0 could not find any solutions, although it warned that it might be missing some of them.

```
In[2]:= Solve[Cos[x]+Cos[3 x] + Cos[5 x]==0,x]

Solve::ifun: Warning: inverse functions are being used by Solve,
   so some solutions may not be found.

Out[2]= {}
```

In 2001, Mathematica version 4.1 does better, and after the same warning, it finds 10 solutions:

$$\{\{x \to \frac{-5\,\pi}{6}\}, \{x \to \frac{-2\,\pi}{3}\}, \{x \to \frac{-\pi}{2}\}, \{x \to \frac{-\pi}{3}\}, \{x \to \frac{-\pi}{6}\}, \{x \to \frac{\pi}{6}\}, \{x \to \frac{\pi}{3}\}, \{x \to \frac{\pi}{2}\}, \{x \to \frac{2\,\pi}{3}\}, \{x \to \frac{5\,\pi}{6}\}\}$$

But not the rest of them.

## 8 What if we change the problem slightly?

Solving the equation $\sin(6x)/(2\sin(x)) = 0$ is quite similar to the initial problem. Except for a problem at zeros of $\sin(x)$, where a limit calculation is required, it is identical. Macsyma finds 11 expressions covering the same roots, unfortunately including $2\pi n$ which is wrong when $n = 0$ (taking a limit, the expression is equal to 3, not 0). Mathematica 2,0 finds none, although it says it found $x = 0$ and then discarded it, since it is not a solution. Various possible transformations can coerce Mathematica into producing some answers (`TrigExpand`, for example). Maple finds

$$\{-1/2\,\pi, 5/6\,\pi, 2/3\,\pi, -2/3\,\pi, -1/3\,\pi, -5/6\,\pi, -1/6\,\pi, 1/6\,\pi, 1/2\,\pi, 1/3\,\pi\}$$

but by setting `_EnvAllSolutions` to true, Maple inserts a collection of integer-valued "arbitrary constants" which will generate an infinite set based on 10 generators. This is an improvement over the 1991 version which found no solutions. PRESS (Thanks to Alan Bundy, March, 1991 for running it), returns the answer $x = 30n$ (recall, that's in degrees). So PRESS mistakenly believes there is a root at $x = 0$ (etc). The (human) argument offered is that naturally all solutions offered by PRESS should be checked, just as we do when we solve problems by hand. This is not so trivial in the case that an infinite set of solutions is offered and an infinite number of elements in that set are not solutions. Nevertheless, in the explanation given by the PRESS trace facility, it is clear what has happened:

```
| ?- solve(sin(6*x)/sin(x)=0).

Solving sin(6 * x) / sin(x) = 0 for x
Tidying to sin(x) ^ -1 * sin(6 * x) = 0

Solving factor sin(6 * x) = 0

        Letting n1 denote an arbitrary integer
    x = 30 * n1
     (by Isolation)

Solving factor sin(x) ^ -1 = 0
sin(x)^-1 = 0 has no real roots, sin(x)^-1 can not be 0

Answer is : X1
  where :  X1 =  x = 30 * n1

yes
```

I think that we should reject the contention that it is satisfactory for a program to provide incorrect solutions from `solve` on the assumption that a human would naturally check the solutions. One can write a program that tries out the solutions and separates the ones that are confirmed from the ones that are either unconfirmed or provably wrong [3].

# 9 How should we solve equations?

Other than showing a cute example or two, what is the point of this?

The following questions (and our partial answers) remain.

1. What should the role of heuristics be in solving equations?

   Heuristics, if they sometimes fail and cannot be corrected, are not acceptable. In the example here, and in a large class of similar problems, a canonical conversion to exponentials would work to reduce the problem to an algebraic one. Also, just as clearly, the form of the answer that is obtained from this "radical" change of form is less convenient than the trigonometric form.

   To what extent should we be permitted to search for the easy solution and neglect the hard? The 10 answers returned by Macsyma are right: should we now write a program to convert them to a more compact form? How can we continue to compute with such forms? The arguments for simplification of the treatment of multiple-valued inverses of ordinary functions like cos and exp continue to proliferate, but it should be clear that choosing any single arccos will simply not solve this problem. What is necessary is to face up to the fact that there are, in some cases, an infinite number of solutions that must be represented and that this situation deserves further treatment. (We continue this discussion in the next section.)

2. Can we find a useful technique for proving the equivalence of different ways of denoting the same set (simplifying multiple values)? (Again, we continue this discussion in the next section.)

[3] Having skilled humans check over the results of a computer-based medical diagnosis x-ray examination makes sense: humans are very good at image understanding, and medical treatment, which can be expensive and painful, does not need to be done at microsecond time scales.

Having humans check over the results of a computer-based mathematics program is not as plausible. Humans are not that good at algebra, and proceeding to the next step in a lengthy calculation should be done without delay. We would not consider using a compiler whose specification said that "You should check that the output is correct binary code" so why should we consider a computer algebra system whose specification (i.e. Correct Behavior) states that the output may be (without warning) wrong.

3. How should strong methods (e.g. polynomial factoring) be controlled? They can be very expensive if used casually on large examples. In spite of increases in computer speed and capacity, a strategy of “try all simplification programs one after another to try to make the expression smaller” works only on smallish examples.

4. To what extent can we use additional techniques (hash-coding, parallel search), to make heuristics and/or strong methods better?

5. If our solution methods are stronger than our checking methods, we may not be able to confirm the correctness of some solutions. Is this inevitable? Should we use heuristics for checking, too? In the case of $\sin(6x)/\sin(x)$ it is necessary to compute a limit at $x = n\pi$ specifically at $n = 0$ to show the result is false.

6. Perhaps we should consider programs to prove the non-existence of solutions? For example, having failed to find a value of $x$ for which $f(x) = 0$, should one try to prove $f(x) \neq 0$?

7. Finally, to what extent should symbolic solvers appeal to solutions from numerical methods? (In particular, solving some univariate transcendental or polynomial equations can be done far more expeditiously by numerical or even graphical root-finding. Further computations (e.g. integration, finding eigenvectors, etc. are not necessarily hampered by this approach.)

# 10 Computing with sets

We seem to be faced with the need to solve such problems as: Compute that $\{n\pi/3 \mid n \in \mathbf{Z}\}$ is a subset of $\{k\pi/6 \mid k \in \mathbf{Z}\}$

One subproblem is the simplification of expressions with arbitrary constants. The simplest example may be the observation that $\{n + k \mid n \in bfZ, k \in \mathbf{Z}\}$ is simply $\mathbf{Z}$ We thought about addressing this over 20 years ago; some results emerged in the 1981 master’s project of Neil Soiffer[2] which essentially shows how to collapse constants in a multivariate rational form to a minimum number of arbitrary constants (or at most one extra). As an example, consider

$$\frac{3(r_1 - r_2)r_3 x + 2(r_1 + r_2)y}{x^2 + (r_2 - r_1)r_3 y + (r_1^2 - r_2^2)r_3^2}$$

with constants $\{r_i\}$. This can be collapsed to

$$\frac{s_1 x + s_2 y}{x^2 - (1/3)s_1 + (1/6)s_1 s_2}.$$

# 11 Conclusion: suggestions for further work

Our two points of the previous section deserves some elaboration. Just because one’s favorite symbolic manipulation system cannot find a symbolic solution does not mean there is none or that a solution cannot be found by other methods. For symbolic systems to be helpful, the most capable numerical solvers should be waiting in the wings, as well as graphics programs that are particularly sensitive to the requirements of “difficult” plots. Proving that solutions do *not* exist somewhere are of interest in some applications.

As a secondary consideration, we should realize that not all solutions are comparably interesting, even symbolically. We may wish to know how many roots are in a given interval or region. We may want to find (or merely verify the existence of) only *one* symbolic or numerical solution. We may want to find the smallest root, or the one closest to a given point. Each of these tasks can be accomplished in a separate way, and each is of importance in different contexts in which the general problem is quite unapproachable. Programs to solve each of these tasks can be far more useful than a general program, especially one which falls unnecessarily into the several possible “solving” time-sinks of computing resultants, Grobner bases, multivariate polynomial factorizations, explosive growth via trigonometric expansions or similar algorithms.

## References


[1] Richard Fateman, Alan Bundy, Richard O'Keefe, Leon Sterling. "Commentary on: Solving Symbolic Equations with PRESS" Res. Paper No. 357, Dep't of Artificial Intelligence, Univ. of Edinburgh, 1987. also published in SIGSAM Bulletin (22), no. 2, April, 1988, 27-40.

[2] Neil Soiffer, "Collapsing Constants," MS report, EECS Dept, Univ. Calif, Berkeley. 1981.

[3] Leon Sterling, Alan Bundy, Lawrence Byrd, Richard O'Keefe, Bernard Silver. "Solving Symbolic Equations with PRESS" *J. Symbolic Computation* (1989) **7,** 71-84.