

Re: Maxima – precision

Source: <http://sci.tech–archive.net/Archive/sci.math.symbolic/2007–06/msg00005.html>

- *From:* adam majewski <adammaj1@xxxxx>
 - *Date:* Sat, 02 Jun 2007 18:18:22 +0200
-

rjf napisab(a):

On May 20, 12:59 am, adam majewski <adamm...@xxxxx> wrote:

Hi

I'm computing complex roots of polynomial:

```
P(n):=if n=0 then 0 else P(n-1)^2+c;  
bigfloat(allroots(%i*P(3)));
```

How can I set precision (number of significant digits) ?
Changing fpprec seems to have no effect .

You are computing the roots of a polynomial in machine precision. Allroots is just a transcription of the Jenkins–Traub floating point polynomial zero–finder, and does not use bigfloats at all.

given the machine–float roots, you then convert them to bigfloat precision. This does not make them any more accurate. At least you would do this if you were to spell bfloat(...) correctly.

What you want is a program different from allroots that computes the roots to a precision that depends on fpprec. This is not, so far as I know, part of the Maxima system, though a version could easily be produced.

It would probably have some glitches because zero–finding programs have to deal with nasty problems.

But maybe your polynomials are not nasty.

What you can do is choose a zero computed from allroots, and then refine it.

Usually you can use newton iteration, and run that until you are happy with the answer.

Thx for answer. If I properly understand :

Re: Maxima – precision

```
P(n):=if n=0 then 0 else P(n-1)^2+c;  
load(newton1);  
clist:allroots(%i*P(6));  
a:rhs(clist[4]);  
(%i9) newton(%i*P(3),c,a,0.00000000000000000001);Maxima encountered a Lisp error: Error in PROGN  
[or a callee]: The storage for CONS is exhausted.Currently, 82780 pages are allocated.Use ALLOCATE to  
expand the space.Automatically continuing.To reenable the Lisp debugger set *debugger-hook* to nil.
```

Adam

.