HP Forum Archive 20

[Return to Index | Top of Index]

Matrix functions on the WP 34s Build 1685: in a word "incredible"

*Message #1 Posted by **gene wright** on 5 Oct 2011, 12:19 p.m.*

I have been encouraging the smart people on the 34S to give us great matrix functionality in the machine. It's here.

A couple of years ago, Valentin gave us some simple-looking matrices of two digit integers whose determinants equaled 1. The inverses of the matrices were also composed of integers and the determinant of the inverses were exactly equal to 1.

AM3 was the most difficult matrix that Valentin generated and you can read about it here:

Valentin's AM1, AM2 and AM3 thread

I have just run the matrix determinant on AM3.

It comes out to exactly 1 with no Tiny Element flag, no "Hey the matrix has integers so the determinant must be an integer" adjustments. It is simply 1 to the entire precision of the result displayed.

Then, I computed the inverse of AM3 and the determinant of the resulting inverse.

The result?

1 to the entire precision of the result.

Incredible. Valentin needs to come back and get himself a WP-34S!

P.S. By the way, the determinants, inverses, etc are computed as near to instantly as I can see.

Edited: 5 Oct 2011, 12:33 p.m.

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #2 Posted by ***Steve Simpkin*** *on 5 Oct 2011, 12:38 p.m.,*
in response to message #1 by gene wright

> Quote:
>
> ---
>
> I have been encouraging the SMART people on the 34S to give us great matrix functionality in the machine. It's here.
>
> ---

Oh, Way to go Gene! Excluding all of the not-so-smart people like me from testing that Matrix stuff. Next you will be telling us that the WP 34s is so easy to use, even a caveman could do it. Hmmmph.
:)

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #3 Posted by ***gene wright*** *on 5 Oct 2011, 12:44 p.m.,*
in response to message #2 by Steve Simpkin

Lol. Now Steve... anyone can download the current revision and test to your heart's content!

Beta software can be troubling, however, as yesterday afternoon, a version of these commands locked my machine where I had to remove the batteries. Caveat FREE emptor.

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #4 Posted by ***Richard J Nelson*** *on 5 Oct 2011, 2:10 p.m.,*
in response to message #1 by gene wright

This is great progress. i will have to check on other matrix challenges that I may have in my library.

X < > Y,

Richard

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #5 Posted by ***Palmer O. Hanson, Jr.*** *on 5 Oct 2011, 9:50 p.m.,*
in response to message #4 by Richard J Nelson

> Quote:
>
> ---

This is great progress. i will have to check on other matrix challenges that I may have in my library.

Page 24 of Kahan's paper "Mathematics Written in Sand" proposed a difficult problem. namely the inversion of a modified 8x8 Hilbert matrix where the elements are defined by A(i,j) = 360360/(i + j - 1). The inverse correct to 12 significant figures is

```
 1.77600177600E-4     -5.59440559441E-3      5.59440559441E-2     -0.256410256410      0.615384615385    .....

-5.59440559441E-3      0.234965034965       -2.64335664336         12.9230769231       .....

 5.59440559441E-2     -2.64335664336         31.7202797203        -161.538461538       .....

-0.256410256410        12.9230769231        -161.538461538         346.153846154       .....

 0.615384615385       -32.3076923077         415.384615385        .....

-0.8                   43.2                 -567                  .....

 0.533333333333       -29.4                  392                  .....

-0.142857142857        8                    -108                  .....
```

where only the elements of the first, second and third columns and parts of the fourth and fifth columns are shown. The exact values for the elements of the first column are 8/45045, -4/715, 8/143, 10/39, 8/13, 8/10, 8/15 and -1/7.

The following table presents the results for only the first column of the inverse (I get tired typing in all the numbers) as found by Stefan's program on the HP-35s, the HP-28s, the HP-28s with one iteration of refinement and the TI-85.

```
     True                 HP-35S               HP-28S               HP-28S+               TI-85

 1.77600177600E-4      1.77637306166E-4      1.77585836871E-4      1.77600204303E-4      1.77599778919E-4

-5.59440559441E-3     -5.5964307186E-3      -5.59363119661E-3     -5.59440698893E-3     -5.59438445849E-3

 5.59440559441E-2      5.59708386791E-2      5.59338926659E-2      5.59440738932E-2      5.59437817772E-2
```

-0.256410256410	-0.256556617662	-0.25635505057	-0.256410352726	-0.256408778659
0.615384615385	0.615781713919	0.615235562229	0.615384873348	0.615380646744
-0.8	-0.800565424016	-0.799788616834	-0.80000036382	-0.799994393047
0.533333333333	0.533737824252	0.533182624964	0.533333591708	0.533329346788
-0.142857142857	-0.142971771814	-0.142814556361	-0.14285721566	-0.142856018664.

where the most striking thing is the major improvement with the iterative refinement on the HP-28.

Edited: 5 Oct 2011, 10:16 p.m.

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

*Message #6 Posted by **Paul Dale** on 6 Oct 2011, 12:27 a.m.,*
in response to message #5 by Palmer O. Hanson, Jr.

The 34S seems to be getting 16 digit accuracy for this example assuming I've done things properly:

```
001  LBL A
002  .
003  0
004  8
005  STO K
006  M.ALL
007  STO I
008  LBL 00
009  3
010  6
011  0
012  3
013  6
014  0
015  RCL I
016  RCL K
017  M.IJ
018  +
019  1
020  -
```

```
021  /
022  STO[->]I
023  ISG I
024  GTO 00
025  RTN
```

Then .08 M.INV to get the inverse.

- Pauli

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #7 Posted by ***Rodger Rosenbaum*** *on 9 Oct 2011, 4:23 p.m.,*
in response to message #6 by Paul Dale

Valentin gave a matrix with an even higher condition number in this thread: http://www.hpmuseum.org/cgi-sys/cgiwrap/hpmuseum/archv015.cgi?read=84694#84694

He didn't call it AM7 in that thread, but in this document:

http://membres.multimania.fr/albillo/calc/pdf/DatafileVA014.pdf

he identifies it as AM7.

Valentin's AM7 determinant on the WP 34S

Message #8 Posted by ***gene wright*** *on 9 Oct 2011, 5:27 p.m.,*
in response to message #7 by Rodger Rosenbaum

is 1.0000... to the limits of precision of the machine.

The determinant of the inverse of this AM7 matrix is also 1.000... to the limits of the precision of the machine.

Amazing!

P.S. the timing is as near to instantaneous as I can imagine for the determinant and the display barely has time to flash "Wait..." or such before the inverse is computed. The determinant of the inverse is instantaneous.

Edited: 9 Oct 2011, 5:32 p.m.

Re: Valentin's AM7 determinant on the WP 34S

*Message #9 Posted by **Paul Dale** on 9 Oct 2011, 5:50 p.m.,*
in response to message #8 by gene wright

> Quote:
>
> ---
>
> P.S. the timing is as near to instantaneous as I can imagine for the determinant and the display barely has time to flash "Wait..." or such before the inverse is computed. The determinant of the inverse is instantaneous.
>
> ---

I'll likely take out that wait display. I thought these would be slower than they are.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

*Message #10 Posted by **gene wright** on 9 Oct 2011, 5:59 p.m.,*
in response to message #9 by Paul Dale

Maybe change it to say:

"Don't blink"

:-)

Good, take that thing out and it will look as fast as it is.

Re: Valentin's AM7 determinant on the WP 34S

*Message #11 Posted by **fhub** on 9 Oct 2011, 6:05 p.m.,*
in response to message #9 by Paul Dale

> Quote:
>
> ---
>
> I'll likely take out that wait display. I thought these would be slower than they are.
>
> ---

But does this still hold for a 10x10 matrix?

Re: Valentin's AM7 determinant on the WP 34S

*Message #12 Posted by **Paul Dale** on 9 Oct 2011, 6:51 p.m.,*
in response to message #11 by fhub

A second or less for a 10x10.

You do see the wait message long enough to read it.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

*Message #13 Posted by **Valentin Albillo** on 9 Oct 2011, 6:56 p.m.,*
in response to message #11 by fhub

> Quote:
>
> ---
>
> But does this still hold for a 10x10 matrix?
>
> ---

Who knows ... try this one and see how it fares:

```
Albillo Matrix #10 (AM#10):


  29   23   40   37   30   32   66   48   38   44
  24   22   45   49   16   39   65   72   38   56
  67   44   92   37   66   20   69   14   14   37
  37   28   63   70   36   35   52   43   26   72
  21   10   19   20   23   16   27   12   13   21
  49   65   93   71   65   39   83   57   42   77
  36   26   60   68   35   33   46   38   22   69
  51   42   63   39   71   16   57   13   15   40
  31   19   43   39   22   22   53   36   35   43
  57   28   76   38   37   23   89   45   38   44


Determinant :


   1
```

The HP-71B gives DET(AM#10) as **59.9605462429** instead of 1 so it's losing *all* significant digits and then some.

You can quickly check whether you've inputted it correctly by computing its Frobenius norm, which should give:

```
    FNORM(AM#10) ->  466.407547109
```

Best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

Message #14 Posted by ***Walter B*** *on 9 Oct 2011, 7:02 p.m.,*
in response to message #13 by Valentin Albillo

Buenas tardes, Valentin!

Long time no see - welcome back :-)

Walter

Re: Valentin's AM7 determinant on the WP 34S

Message #15 Posted by ***Valentin Albillo*** *on 9 Oct 2011, 7:10 p.m.,*
in response to message #14 by Walter B

> Quote:
>
> Buenas tardes, Valentin!
>
> Long time no see - welcome back :-)
>
> Walter

Thanks, Walter. Regrettably, I can't visit the forum as frequently as in times past, just too busy ... :D

Best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

Message #16 Posted by ***gene wright*** *on 9 Oct 2011, 9:46 p.m.,*

in response to message #15 by Valentin Albillo

Valentin! Good to see you again.

Email me through the forum if you can or use an old email if you have one from the past. Thanks!

Re: Valentin's AM7 determinant on the WP 34S

Message #17 Posted by ***Dan W*** *on 9 Oct 2011, 7:09 p.m.,*
in response to message #13 by Valentin Albillo

Since a lot of us use Excel these days, I took several of these ill conditioned matrices and tried them in Excel (version 2007, on a Windows 7 PC). The determinants are:

AM1 0.99999986183

AM2 0.99999977967

AM3 1.00000101670

AM10 1.00096388386

Edited: 9 Oct 2011, 7:10 p.m.

Re: Valentin's AM7 determinant on the WP 34S

Message #18 Posted by ***Bunuel66*** *on 10 Oct 2011, 6:42 p.m.,*
in response to message #17 by Dan W

Well, the mystery is still there, det(AM10) on HP39gs gives...1 exactly. Even doing the trick (det(AM10)-1)*10000 gives 0!

Regards

Re: Valentin's AM7 determinant on the WP 34S

Message #19 Posted by ***Valentin Albillo*** *on 14 Oct 2011, 7:34 p.m.,*
in response to message #18 by Bunuel66

Quote:

> Well, the mystery is still there, det(AM10) on HP39gs gives...1 exactly. Even doing the trick (det(AM10)-1)*10000 gives 0!

I suggest people with a WP 34S (and other calcs as well) should try (in non-exact mode) these two nifty 10x10 matrices I've concocted for the occasion

#AM 11:

```
 65   66  -58   74   -3  -46   28   29   11    6
-19   33   67    6   56   -6   25   20   57   49
 72   19   85  -20   46   14   39   -4   52   43
-52   -4  -37   95   39   32   79   90    4   -4
-16   29   71    2   60    3   30   17   61   52
-39   24  -23   88   23  -12   63   57    4   -1
 63   82   42   28   20  -71   57   49   -7  -12
-61  -26   47   30   77   63   77   27   26   16
 26  -43   45  -39   47   97   57  -29   43   32
 60   19   34   23   33   10   81    8  -12  -20


True determinant   = 1


HP-71B determinant = 754557.820054
Frobenius norm     = 461.055311215
Row norm           = 458
Column norm        = 536
Sum of elements    = 2540
```

#AM 12:

```
-19   33   56   -6  -23   44   25   49   57   20
 26  -43   47   97  -32   13   57   32   43  -29
-39   24   23  -12   77   54   63   -1    4   57
 65   66   -3  -46   97   39   28    6   11   29
-52   -4   39   32   84   47   79   -4    4   90
 72   19   46   14  -33   52   39   43   52   -4
-61  -26   77   63  -10   37   77   16   26   27
 63   82   20  -71   19   61   57  -12   -7   49
-16   29   60    3  -26   45   30   52   61   17
 60   19   33   10   19   53   81  -20  -12    8
```

```
True determinant   = 1


HP-71B determinant = 683755.24004
Frobenius norm     = 452.653288953
Row norm           = 441
Column norm        = 536
Sum of elements    = 2597
```

Despite the very small (2-digit or less) integer elements I expect non-exact calc algorithms to lose **20-23** significant digits while computing the determinant. The norms and sum of elements are included to check correct input.

Best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

Message #20 Posted by ***Paul Dale*** *on 14 Oct 2011, 8:39 p.m.,*
in response to message #19 by Valentin Albillo

Valentin, your guess as to the accuracy loss is spot on...

```
Matrix  Returned result       Internal result
 AM11   1.000000000000000     1.000000000000000480370825780757899759б
 AM12   1.000000000000000     1.000000000000000576478156381655514360 5
                                               ^ 17th digit
```

On more lost digit and the 34S will get the answer wrong. No that isn't a challenge, I'm sure it is possible.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #21 Posted by ***gene wright*** *on 14 Oct 2011, 8:48 p.m.,*
in response to message #20 by Paul Dale

Of course, at one point you were computing internally with more than 39 digits... but at a VERY large expense of memory, which would *not* be worth it at all.

The real point (to me) is that the 34S matrix commands seem to be more accurate by far than anything we've ever had that was calculator-portable and *ought* to handle most anything that gets thrown at them.

Great job, and someone get Valentin a 34S !!

Re: Valentin's AM7 determinant on the WP 34S

Message #22 Posted by ***Paul Dale*** *on 14 Oct 2011, 8:51 p.m.,*
in response to message #21 by gene wright

Currently, I'm doing the LU decomposition using temporaries of 34 digits and the multiply/accumulate calculations to 39. We're just fitting into the volatile RAM at the moment which is perfect.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #23 Posted by ***Paul Dale*** *on 14 Oct 2011, 9:20 p.m.,*
in response to message #22 by Paul Dale

Only two calculations are done to more than the internal 39 digits, they aren't used by the matrix code.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #24 Posted by ***Dan W*** *on 15 Oct 2011, 2:04 p.m.,*
in response to message #20 by Paul Dale

Here are the results in Excel 2007 for comparison. AM12 is really tough!

AM1 0.99999986183

AM2 0.99999977967

AM3 1.00000101670

AM10 1.00096388386

AM12 4.49647974387

Re: Valentin's AM7 determinant on the WP 34S

Message #25 Posted by **Marcus von Cube, Germany** *on 15 Oct 2011, 2:29 p.m.,*
in response to message #24 by Dan W

WP 34S does an LU decomposition to 34 digits with pivoting and then computes the determinant from the diagonal. This seems to be quite stable.

Pauli, correct me, if have misread your code.

Re: Valentin's AM7 determinant on the WP 34S

Message #26 Posted by **Paul Dale** *on 15 Oct 2011, 5:50 p.m.,*
in response to message #25 by Marcus von Cube, Germany

This is correct. I'm using Dolittle's algorithm like the 15C. This is stable but not the fastest.

The only extra thing to note is that the multiply/subtract steps used in the calculation of the lower triangular matrix are done using 39 digits which could help avoid a little extra cancellation if we're lucky.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #27 Posted by **Valentin Albillo** *on 16 Oct 2011, 11:27 a.m.,*
in response to message #20 by Paul Dale

> Quote:
>
> One more lost digit and the 34S will get the answer wrong. No that isn't a challenge, I'm sure it is possible.

Indeed it is. Enter **AM#13**, another one of my 10x10 matrices entirely consisting in very small (2-3 digit) integer elements:

AM #13:

```
  34   33  195  -18  213  238  -66   13   24  -56
  39  148  -51   95 -388  -11   28   31  -35   49
-125  124 -129  130   86  -99  156  -31  -53  181
 248  -65  -44  128  107   28   71   84 -119  -10
  87  201   40  291  -25   40  -68  268  176   31
  28 -148  147   75   89  139  187 -146 -156  -22
 175   69  140  178 -143  306  182  -51 -108 -129
-113  101 -126   97  165 -112  127   36   45  123
-127  202  148  356  277  224  258   24   63   98
  52 -116  173   93   38  -14 -188  132   76   19



True determinant   = 1


HP-71B determinant = 288,676,439,828
Frobenius norm     = 1387.86490697
Row norm           = 1777
Column norm        = 1531
Sum of elements    = 5167
```

Despite its simplicity I expect non-exact calc algorithms to lose **25-28** digits while computing its determinant.

This will surely make the final 16-digit 34S result *inexact in its last 4 or 5 digits*. Less capable calcs or computing software (say Excel) will probably lose *all* their digits, as seen in the result given above for the HP-71B Math ROM's **DET** function.

Best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

*Message #28 Posted by **Walter B** on 16 Oct 2011, 4:10 p.m.,*
in response to message #27 by Valentin Albillo

Buenas tardes Valentin,

please forgive me if my following question is mathematically trivial. But are there any smaller matrices of similar "nastyness" like AM#13? E.g. an 8x8 matrix? I'd estimate the probability for somebody keying in a 10x10 matrix into a pocket calculator being <0.01 even in an high math environment like this forum, and <1e-5 elsewhere.

TIA for your response,
Walter

Re: Valentin's AM7 determinant on the WP 34S

Message #29 Posted by **Werner** *on 17 Oct 2011, 3:08 a.m.,*
in response to message #28 by Walter B

There's no need to go even that far.

Take a 2x2 matrix

```
[[  a   a-1 ]
 [ a+1   a  ]]
```

its determinant is 1.
Now, for decimal machines, take a=2^39 = 549'755'813'888, and compute the determinant.
12-digit machines return 0. Free42 with its 25 decimal digits of precision returns

```
    0.956630091747
```

The condition number of this 2x2 matrix is 1e24, even greater than Valentins 10x10 - of course, not with the same small elements.
The 34S carries 16 digits normally, so try a=2^50.
The condition number is (2*a+1)^2 or about 5e30, you'll get maybe 4 digits correct in the determinant.

Cheers, Werner

Edited: 17 Oct 2011, 7:05 a.m. after one or more responses were posted

Re: Valentin's AM7 determinant on the WP 34S

Message #30 Posted by **Paul Dale** *on 17 Oct 2011, 3:20 a.m.,*
in response to message #29 by Werner

Try calculating this via:

```
    5
    0
    2^x
    FILL
```

```
    DEC T
    INC Y
    cmplx *
```

:-)

The matrix determinant code isn't as good.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

*Message #31 Posted by **Werner** on 17 Oct 2011, 3:41 a.m.,*
in response to message #30 by Paul Dale

On the 42S I had to create the equivalent 3x3 matrix

```
[[  a  a-1  0 ]
 [ a+1  a   0 ]
 [  0   0   1 ]]
```

to bypass what you just demonstrated, because it uses the straightforward formula a*b-c*d to calculate 2x2 determinants. Here (as in the 34S) the matrix code must resort to

```
    a
 ( --- * a - (a-1))*(a+1)
   a+1
```

to calculate the determinant, and the first division introduces the small roundoff error.
There's nothing to be done about that, that's my point.

Cheers, Werner

Edited: 17 Oct 2011, 3:43 a.m.

Re: Valentin's AM7 determinant on the WP 34S

*Message #32 Posted by **Valentin Albillo** on 17 Oct 2011, 3:33 p.m.,*
in response to message #28 by Walter B

Hi, Walter:

> Quote:
>
> [...] are there any smaller matrices of similar "nastyness" like AM#13? E.g. an 8x8 matrix?

Certainly. Have a look at this 8x8 matrix o'mine. Not as nasty as it sticks to small (2-3 digit) integer elements but still pretty nasty nevertheless:

AM#8:

```
  -65   153  -222   257   306   520  -121   461
  131    13   184   -69  -202    13   253  -121
   11    27   -81    88    92    44   -71    99
  347   320   267   171  -328   577   463    88
   87   237   -21   277    55   336   107   104
 -354  -223  -337  -563   548   333    63   323
  208   306    73   243  -115   563   196   215
  165   243    19   112    61   566   453   109


True determinant =  1
```

Go and try your favourite calculator against it and see how it fares and how many digits are lost.

> Quote:
>
> I'd estimate the probability for somebody keying in a 10x10 matrix into a pocket calculator being <0.01 even in an high math environment like this forum, and <1e-5 elsewhere.

Maybe but you know what they say: *"No pain, no gain"*.

The Spanish version of said proverb begins with *"El que quiera peces ..."* and common decency prevents me from posting the end ... XD.

> Quote:
>
> TIA for your response

You're welcome.

Best regards from V.

Hey V ... check your email!

Message #33 Posted by ***gene wright*** *on 17 Oct 2011, 3:40 p.m.,*
in response to message #32 by Valentin Albillo

ha!

Re: Valentin's AM7 determinant on the WP 34S

Message #34 Posted by ***Ángel Martin*** *on 17 Oct 2011, 4:03 p.m.,*
in response to message #32 by Valentin Albillo

> Quote:
>
> "El que quiera peces ..."

definitely much more a poetic version than the prosaic saxon one :-)

Glad to see you're in top shape, as usual.

Best, 'AM

Re: Valentin's AM7 determinant on the WP 34S

Message #35 Posted by ***Paul Dale*** *on 17 Oct 2011, 5:30 p.m.,*
in response to message #32 by Valentin Albillo

0.9999999999999997

Which displays as 1 of course :-)

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #36 Posted by ***Paul Dale*** *on 16 Oct 2011, 5:47 p.m.,*
in response to message #27 by Valentin Albillo

Yep, that does it.

The returned value is 0.9999999999908109 instead of 1.

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #37 Posted by ***Marcus von Cube, Germany*** *on 17 Oct 2011, 2:03 a.m.,*
in response to message #36 by Paul Dale

Still slightly better then the 71B. ;)

There are 11 nines in the result so rounded to 10 digits it will still return a 1.

Walter, Pauli has committed a .wp34s source file that inputs the matrix. No need to type it in. :-)

Re: Valentin's AM7 determinant on the WP 34S

Message #38 Posted by ***Werner*** *on 17 Oct 2011, 2:32 a.m.,*
in response to message #36 by Paul Dale

condition nr of AM13 is (about) 4e23, so you get 11 digits correct with 34-digit arithmetic.
Free42 Decimal uses 25 digits and returns 1.01769242024 as determinant, so indeed 2 correct digits, and I guess I'm that one out of a hundred that did key in the matrix ;-)
An 8x8 matrix with a similar condition number is possible, but then the individual elements will have to be larger.
And that's my point: 34-digit arithmetic for 10x10 matrices is a bit of overkill. There will always be a matrix that returns completely wrong results. It would be better to return an estimate of the condition number instead, so that you know how many digits of the result can be trusted. If too many digits are lost, then the matrix is the problem (and the way it was obtained), not the algorithm or the number of digits the calculator uses.

Cheers, Werner

How about a M-COND command?

Message #39 Posted by ***gene wright*** *on 17 Oct 2011, 7:52 a.m.,* *in response to message #38 by Werner*

that returns the condition number of a matrix?

Re: How about a M-COND command?

Message #40 Posted by ***Werner*** *on 17 Oct 2011, 8:49 a.m.,*
in response to message #39 by gene wright

a short and easy way would be what I use on a 42S: (it lacks CNRM, so I have to use row norm on the transpose)

```
TRAN
RNRM
LASTX
INVRT
RNRM
*
```

but of course, that implies inverting the matrix..
Another option would be to estimate it the way they do in LAPACK (SGECON), but that would be quite a lengthy routine, and as I've come to understand, life is short and flash is full - to paraphrase Bill Wickes.

Cheers, Werner

Re: How about a M-COND command?

Message #41 Posted by ***fhub*** *on 17 Oct 2011, 8:55 a.m.,*
in response to message #39 by gene wright

> Quote:
>
> that returns the condition number of a matrix?

That leads me to the following question: HOW do you compute this condition number of a matrix???

Since I didn't know its exact definition I searched a bit and found that cond(A)=norm(A)*norm(A^-1).

Well, now I have a problem with this definition: if A is ill-conditioned then computing A^-1 will give a quite inaccurate result, so also norm(A^-1) and thus cond(A) will be inaccurate.
This is what we call in German "the cat biting its own tail".

So again my question: what's the usual way to compute this cond(A)?

Franz

Re: How about a M-COND command?

Message #42 Posted by ***Werner*** *on 17 Oct 2011, 10:11 a.m.,*
in response to message #41 by fhub

The condition number does not have to be calculated to any great accuracy to be useful. If it is in the order of magnitude of 10^a, then we can expect to lose 'a' digits when calculating the determinant or solving a system of equations. If a is near the number of digits carried by your calculator, the matrix is said to be singular to working precision.

Cheers, Werner

Re: How about a M-COND command?

Message #43 Posted by ***Marcus von Cube, Germany*** *on 17 Oct 2011, 10:17 a.m.,*
in response to message #42 by Werner

Then it would be more appropriate to compute the $\log_{10}$ of the condition number as an integer.

Re: How about a M-COND command?

Message #44 Posted by ***fhub*** *on 17 Oct 2011, 10:27 a.m.,*
in response to message #42 by Werner

> Quote:
>
> The condition number does not have to be calculated to any great accuracy to be useful.

Well, that doesn't change the principle problem at all! :-(

If you have to calculate A^-1 to get cond(A), then this A^-1 could already be SO wrong for a VERY-ill-conditioned matrix (just see some DET results in this thread!), that the computed cond(A) with this wrong A^-1 would not only be of "no great accuracy" but even completely wrong.

It's the same as if you would try to measure the precision of any measuring instrument with this (unprecise) instrument itself.

Franz

Edited: 17 Oct 2011, 10:28 a.m.

Re: How about a M-COND command?

Message #45 Posted by ***Walter B*** *on 17 Oct 2011, 12:44 p.m.,*
in response to message #44 by fhub

> Quote:
>
> ---
>
> It's the same as if you would try to measure the precision of any measuring instrument with this (unprecise) instrument itself.
>
> ---

FWIW, this is one of the easiest jobs d:-) The solution won't help you in the matrix problem, however ...

Re: How about a M-COND command?

Message #46 Posted by ***fhub*** *on 17 Oct 2011, 1:21 p.m.,*
in response to message #45 by Walter B

> Quote:
>
> ---
>
> FWIW, this is one of the easiest jobs d:-)
>
> ---

Arrogant as usual ... ;-)

Re: How about a M-COND command?

*Message #47 Posted by **Walter B** on 17 Oct 2011, 5:49 p.m.,*
in response to message #46 by fhub

> Quote:
>
> ---
>
> Arrogant as usual ... ;-)
>
> ---

It's really easy - it doesn't have to be complex just because you don't know it ;-) But since some folks earn their $$$ teaching that method I won't disclose it here.

Re: How about a M-COND command?

*Message #48 Posted by **fhub** on 17 Oct 2011, 6:18 p.m.,*
in response to message #47 by Walter B

> Quote:
>
> ---
>
> It's really easy - it doesn't have to be complex just because you don't know it ;-) But since some folks earn their $$$ teaching that method I won't disclose it here.
>
> ---

Pfff, what a lame excuse! But I didn't expect anything else from you, because I already know you and your vacuous statements here since a few months.

Re: How about a M-COND command?

*Message #49 Posted by **Walter B** on 18 Oct 2011, 12:51 a.m.,*
in response to message #48 by fhub

> Quote:
>
> ---
>
> I didn't expect anything else from you, because I already know you and your vacuous statements here since a few months.
>
> ---

:-/ Shall I say "ditto"? No, I won't follow your track. Else you'll eventually quit for the 16th time, and we all know already what will happen thereafter ;-)

Re: How about a M-COND command?

Message #50 Posted by ***Valentin Albillo*** *on 17 Oct 2011, 12:51 p.m.,*
in response to message #44 by fhub

Hi, Franz:

> Quote:
>
> ---
>
> Well, that doesn't change the principle problem at all! :-([...] It's the same as if you would try to measure the precision of any measuring instrument with this (unprecise) instrument itself.
>
> ---

You're absolutely correct that this is kinda chicken-and-egg problem, you need the inverse to compute the condition number and if the matrix is very ill-conditioned your computed inverse will be practically useless.

The way out of this annoying conundrum is to simply estimate the necessary norm of the inverse as economically as possible without actually computing the inverse proper. You may want to have a look at this paper for a feasible approach:

http:/www.math.ufl.edu/~hager/papers/condition.pdf

Best regards from V.

Edited: 17 Oct 2011, 12:53 p.m.

Re: How about a M-COND command?

Message #51 Posted by ***fhub*** *on 17 Oct 2011, 1:26 p.m.,*
in response to message #50 by Valentin Albillo

> Quote:
>
> ---

> http:/www.math.ufl.edu/~hager/papers/condition.pdf
>
> ---

Thanks Valentin!

A short look at this paper tells me that it isn't worth the troubles. ;-)

I've also found some other estimation algorithms on the internet, quite a few of them use the LU-decomposition of the matrix A. This M-LU is already coded in WP34s, so bringing it back to the user might not be the worst idea. :-)

Franz

Re: How about a M-COND command?

*Message #52 Posted by **Paul Dale** on 17 Oct 2011, 5:45 p.m.,*
in response to message #39 by gene wright

The condition number is one I'd like to have included. We're out of flash again and scraping any back is getting much harder so it is unlikely to ever go native.

The reason I've not done this is as Franz has already pointed out: I don't know how without suffering the effects of any ill-conditioning.

Valentin's link looks interesting.

How many people want M-LU exposed again? Assuming we can squeeze it back in.

- Pauli

Re: How about a M-COND command?

*Message #53 Posted by **Valentin Albillo** on 18 Oct 2011, 7:36 a.m.,*
in response to message #52 by Paul Dale

> Quote:
>
> ---
>
> How many people want M-LU exposed again? Assuming we can squeeze it back in.
>
> ---

I'm about the last person to be qualified to have a say on this but I would hazard that the LU decomposition should *always* be exposed (i.e., available) to end users, as it's pretty useful for many advanced matrix processing.

On the M-COND command, computing it reliably is not practical and a decent estimate (plus or minus one order of magnitude) is about the best that can be done within reasonable time and memory limits. However, on the other hand we should remember that what's important is the *final* goal and the condition number *isn't* it.

The final goal is to get <u>a decent estimation on the number of digits lost</u> in the final result when computing the matrix determinant. Abour three years ago I wrote a full 16-page article discussing and implementing a novel idea I came up with in order to achieve this goal, with worked examples aplenty and intended for its immediate publication in **HPCC Datafile** magazine (together with four or five other very worthwhile articles, even though I say so myself) but most regrettably things went South then and there through no fault of my own and the articles never saw the light.

They wanted both my articles *and* my money, though they had next to none of the former and simply way too much of the latter. They got neither.

Best regards from V.

Re: How about a M-COND command?

Message #54 Posted by ***Paul Dale*** *on 18 Oct 2011, 9:41 a.m.,*
in response to message #53 by Valentin Albillo

> Quote:
>
> I'm about the last person to be qualified to have a say on this but I would hazard that the LU decomposition should *always* be exposed (i.e., available) to end users, as it's pretty useful for many advanced matrix processing.

I'd have said quite the opposite. When it comes to hard core mathematics, I value your opinion quite highly. I'll try hard to squeeze the exposed LU decomposition back in again.

Maybe not in the upcoming release, but the one after...

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

Message #55 Posted by ***Rodger Rosenbaum*** *on 19 Oct 2011, 5:55 a.m.,*
in response to message #38 by Werner

AM#8 has a condition number of about 5E19 according to Mathematica, but my HP50 says the condition number is about 5E15. If the goal is to determine the true condition number, then arithmetic with many more digits than the 15 digits internally in the HP50 would be necessary.

However, if one's goal is to solve some system using the given matrix, knowing that the condition number is at least 5E15 is enough to know that any solution derived from that matrix is likely to have no correct digits (on an HP50)--we don't need to know that the true condition number is 5E19.

Testing the COND function on the HP50, I have been unable to find a matrix with a true condition number greater than E12 which was inaccurately calculated to have a smaller condition number. The calculator doesn't seem to ever seriously underestimate the condition number.

Using a column matrix of: b=[1289 202 209 1905 1182 -210 1689 1728]T along with A=AM#8, we have a linear system Ax=b. The HP50 solution of this system, using the / key, is:

x=[-85.97057 -1328.61068 452.27432 61.401457 1570.67801 1770.08111 2566.08833 471.67052]T

but, the exact solution is:

[1 1 1 1 1 1 1 1]T

We can see that the high condition number makes for no correct digits in the solution.

Using the LSQ function to solve the system rather than the / key method gives much better results on the HP50--showing the first 4 digits of the results:

[.7876 1.181 .7576 1.015 .7726 .8039 1.211 1.212]T

This shows the advantage of orthogonal methods of solution rather than the Gaussian method.

Re: Valentin's AM7 determinant on the WP 34S

Message #56 Posted by ***Werner*** *on 22 Oct 2011, 1:43 p.m.,*
in response to message #55 by Rodger Rosenbaum

Hi, Rodger! Long time no hear.

> Quote:
>
> ---
>
> This shows the advantage of orthogonal methods of solution rather than the Gaussian method
>
> ---

But LSQ performs a rank determination, and will probably deem the matrix rank 7 (which, in 15-digit arithmetic, it is). Orthogonal transformations on the 8x8 matrix would not yield better results. I think ;-) (but I will be sure to verify)

Cheers, Werner

Re: Valentin's AM7 determinant on the WP 34S

Message #57 Posted by ***Valentin Albillo*** *on 17 Oct 2011, 4:09 p.m.,*
in response to message #36 by Paul Dale

Hi, Pauli:

> Quote:
>
> ---
>
> *(the underlining is mine)* Yep, that does it.
>
> The returned value is 0.9999999999<u>08109</u> instead of 1.
>
> ---

So the last 5 digits are lost as well, just as predicted ... :D

Now it's quite simple to concoct a 10x10 matrix with integer elements, still relatively small (6 digits or less), which should make the 34S result lose ***all its digits*** (and about 10 more if they were available !) and there's no need to painstakingly key in the 100 elements.

You simply use **AM#13** *squared*, i.e., form a new matrix **AM^2#13** by multiplying **AM#13** times itself. You'll get:

AM^2#13:

```
 -20180   47048    7076   98251   41242   14030   64203   13262   -3203   47956
  11359  -73359  -11370  -94757  -41339   -5116   15477  -82406  -79528  -19197
 100607   -2350   18324   54227 -118108   11778  -46780   68188    6781  -32421
  70760   -7344   40129   10754   58137   61653  -23236   33140   -2354  -34295
  13986   62051  -18247  135133   83179   15409   97207   39458  -19488   67225
  74931  -41113   32420   12012    -678   57658   21586  -20364  -55150  -41049
  76033  -51699   37581    2730    3139  107847  114841  -98712 -138280  -36810
  69960   46252   16119  100837  -72889   21405  -18475   86423   35487  -10128
 143252   65026   33559  263485  -53355   94084  133288   73613  -61885   21603
 -54883   23173  -39308   40609  149155  -56851   20293   30429   26420   71570
```

```
True determinant   = 1

HP-71B determinant = -5.1369256228 E32 (!)
Frobenius norm     = 661395.105928
Row norm           = 943150
Column norm        = 812795
Sum of elements    = 1928143
```

If I'm correct you should lose **45-50** digits at the very least so completely ruining the 34S result (let alone other models' !).

Of course it is possible to produce a 10x10 matrix with smaller integer elements and similarly high condition number but it would be necessary to key it in which would be a pain in the *derriere*.

Best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

Message #58 Posted by ***Marcus von Cube, Germany*** *on 17 Oct 2011, 4:33 p.m.,*
in response to message #57 by Valentin Albillo

I think you can do this ad nauseam. Squaring an ill conditioned matrix should roughly square its condition number thus doubling the number of lost digits. Still interesting that you can do this with such harmless looking integers.

Re: Valentin's AM7 determinant on the WP 34S

*Message #59 Posted by **Paul Dale** on 17 Oct 2011, 5:38 p.m.,*
in response to message #58 by Marcus von Cube, Germany

I think the numbers stopped looking harmless when they hit three digits positive and negative ;-)

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

*Message #60 Posted by **Valentin Albillo** on 17 Oct 2011, 6:43 p.m.,*
in response to message #59 by Paul Dale

> Quote:
>
> I think the numbers stopped looking harmless when they hit three digits positive and negative ;-)

Oh really ?

So I take it you're saying that the big bad 34-digit full-floating-point 34S is afraid of being harmed by those meanie *three-digit* positive and negative *integers* ? ... ;-)

Pathetic ! ...

Perhaps **the 34S is not for me** after all, I much prefer real-macho calculators which are afraid of *nothing* whatsoever I may throw at them ... XD

Best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

*Message #61 Posted by **Walter B** on 18 Oct 2011, 1:09 a.m.,*
in response to message #60 by Valentin Albillo

Buenas dias Valentin,

> Quote:
>
> ---
>
> So I take it you're saying that the big bad 34-digit full-floating-point 34S is afraid of being harmed by those meanie *three-digit* positive and negative *integers* ? ... ;-)
>
> ...
>
> I much prefer real-macho calculators which are afraid of *nothing* whatsoever I may throw at them ...
>
> ---

Can't prevent you from taking Pauli's post your way ;-) But at the bottom line you must admit the WP 34S is proven to be the most macho matrix matador met so far :-)

Re: Valentin's AM7 determinant on the WP 34S

*Message #62 Posted by **Paul Dale** on 18 Oct 2011, 2:09 a.m.,*
in response to message #61 by Walter B

Personally, I think you took Valentin's post the wrong way :-)

Anyway, I never said the 34S was scared of anything, I only said I was. My years of pure mathematics never involved numbers as large as these, we'll not written out explicitly at any rate...

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

*Message #63 Posted by **Valentin Albillo** on 18 Oct 2011, 5:50 a.m.,*
in response to message #62 by Paul Dale

> Quote:
>
> ---
>
> Personally, I think you took Valentin's post the wrong way :-)
>
> ---

Yes, definitely. Doesn't matter, though, it happens all the time ...

> Quote:
>
> ---
>
> Anyway, I never said the 34S was scared of anything, [...]
>
> ---

Well, it should. Just try AM#13 squared, as above, and post what the computed result is, I'd be curious to know ...

TIA and best regards from V.

Re: Valentin's AM7 determinant on the WP 34S

*Message #64 Posted by **Paul Dale** on 18 Oct 2011, 6:02 a.m.,*
in response to message #63 by Valentin Albillo

Too many program steps for the matrix itself :-(Make the matrix simpler for goodness sake....

Assuming I've squared the matrix properly & I'm not sure I have after more half a dozen semi-decent (export) German beers and a third of a bottle of bad wine, the determinant is -142456776964.0436.

Not really an unexpected result given the ill-conditioned nature of the matrix.

- Pauli

PS: Valentin, if you need a 34S, I'm willing to reflash one of mine and send it your way.

Re: Valentin's AM7 determinant on the WP 34S

*Message #65 Posted by **fhub** on 18 Oct 2011, 6:16 a.m.,*

in response to message #64 by Paul Dale

> Quote:
>
> ...after more half a dozen semi-decent (export) German beers and a third of a bottle of bad wine...

Ohhh, then I hope you don't plan to make any WP34s code 'improvements' in the next 24 hours ... ;-)

Franz

Re: Valentin's AM7 determinant on the WP 34S

*Message #66 Posted by **Paul Dale** on 18 Oct 2011, 6:30 a.m.,*
in response to message #65 by fhub

Come on, most hard-core programmers code best after a bit of ethanol. Sadly, I think I'm a bit past that peak :-(

To reassure our user base, I'm not coding anything on the 34S tonight....

- Pauli

Re: Valentin's AM7 determinant on the WP 34S

*Message #67 Posted by **Paul Dale** on 9 Oct 2011, 7:26 p.m.,*
in response to message #13 by Valentin Albillo

Welcome back.

The determinant returned is 1.000000000000000.

The internal working result is 0.99999999999999999998617199680615581171.

So fifteen or so digits are lost during the computation.

- Pauli

Again, simply incredible...

Message #68 Posted by ***gene wright*** *on 9 Oct 2011, 9:54 p.m.,*
in response to message #67 by Paul Dale

The jaw just drops...

Re: Valentin's AM7 determinant on the WP 34S

Message #69 Posted by ***Werner*** *on 10 Oct 2011, 2:41 a.m.,*
in response to message #67 by Paul Dale

The condition number of AM10 is about 3e14, so you can expect to lose at least 14 digits, that's about right then. Interestingly, even trying to estimate the condition number on a real 42S gives the wrong result since there, it loses all digits and the column norm of the inverse matrix is off by an order of magnitude... had to use Free42 ;-)

Werner

Re: Valentin's AM7 determinant on the WP 34S

Message #70 Posted by ***Peter Murphy (Livermore)*** *on 9 Oct 2011, 7:43 p.m.,*
in response to message #13 by Valentin Albillo

Valentín,

Unless you have been following the Forum closely, you may not know how much AM #1 has contributed to improvement in the WP-34S matrix-handling capability: a lot.

Testing that capability with AM #10 should be interesting to observe at least, and it may lead to further improvement.

Many thanks for those matrices, which continue to have beneficial effects even in your unfortunate absence from this Forum.

Peter Murphy Livermore, CA

Re: Valentin's AM7 determinant on the WP 34S

Message #71 Posted by ***Marcus von Cube, Germany*** *on 9 Oct 2011, 7:10 p.m.,*
in response to message #9 by Paul Dale

Quote:

> I'll likely take out that wait display. I thought these would be slower than they are.

Try on a 10x10 in SLOW mode and consider again, please.

Slow mode?

*Message #72 Posted by **gene wright** on 9 Oct 2011, 9:47 p.m.,*
in response to message #71 by Marcus von Cube, Germany

I didn't know it had AOS...

Re: Slow mode?

*Message #73 Posted by **Marcus von Cube, Germany** on 10 Oct 2011, 2:13 a.m.,*
in response to message #72 by gene wright

It does not have Another Operating System but Another Operating Speed. :-)

SLOW reduces the speed but also the power draw on the poor button cells.

Re: Valentin's AM7 determinant on the WP 34S

*Message #74 Posted by **Paul Dale** on 9 Oct 2011, 10:32 p.m.,*
in response to message #71 by Marcus von Cube, Germany

11 or 12 ticks for a 10x10 matrix inversion in SLOW mode. It should be okay to leave the waiting message out.

7 ticks in FAST mode.

- Pauli

Edited: 9 Oct 2011, 10:33 p.m.

Re: Valentin's AM7 determinant on the WP 34S

*Message #75 Posted by **Marcus von Cube, Germany** on 10 Oct 2011, 2:15 a.m.,*
in response to message #74 by Paul Dale

As log as the watchdog doesn't kick in you can leave out the message. If you see "Reset" but it back.

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #76 Posted by ***Crawl*** *on 5 Oct 2011, 11:05 p.m.,*
in response to message #1 by gene wright

Of course, if you did expansion in minors (iteratively), you should get the exact result as well ... right? Because then it's just multiplication and addition of integers, no division.

This would probably not be true for finding the determinant of the inverse, though, because in that case, the entries are so big that multiplying them out would lead to truncation error.

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #77 Posted by ***Paul Dale*** *on 5 Oct 2011, 11:13 p.m.,*
in response to message #76 by Crawl

Expansion by minors is O(n!) time. The algorithm I've used is $O(n^3)$ time.

For a 10x10 matrix this is likely to be significant. For small matrices there is no real difference. For the 7x7 examples here, I've no real feeling what the speed differential would be.

Expansion by minors is also going to be at a great risk of bad cancellation if the intermediate results get truncated.

- Pauli

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #78 Posted by ***Eddie W. Shore*** *on 6 Oct 2011, 12:22 a.m.,*
in response to message #1 by gene wright

Nice! Congratulations to the WP 34S Team!

Re: Matrix functions on the WP 34s Build 1685: in a word "incredible"

Message #79 Posted by ***Crawl*** *on 6 Oct 2011, 9:16 a.m.,*
in response to message #1 by gene wright

For what it's worth, if you replace the 1,1 entry in matrix 1 (58) with 58+x, the determinant is

1+96360245x

For matrix 2, it would be

1+193969587x

and for matrix 3,

1+294228951x

giving some hint as to why these three matrices are "ill conditioned".

[Return to Index | Top of Index]

GTO Go back to the main exhibit hall