HP Forum Archive 20

[Return to Index | Top of Index]

Numerical Integration: HP, TI, etc.

Message #1 Posted by ***Chuck*** *on 23 May 2011, 3:09 p.m.*

I feel like this has been discussed before, but I can't find any info.

It's generally claimed that many, if not most, calculators that have numerical integration capabilities usually use an Adaptive Gauss-Kronrod Quadrature method.

Today in my Calc II class I introduced the students to this numerical method by first integrating Cos[1/(8-x)] from 0 to b, changing b from 5 to 6, 7, and 8 to show them the computing time increases substantially. From there we looked at a little G-K theory.

We discovered the TI calculators will eventually compute a result for b=8, but takes several minutes due to the improper upper limit. The value it gives is 6.491864009, while Mathematica gives 6.4916765549 (claiming to use Guass-Kronrod). So, the TI was accurate to 3 decimal places using an undetermined number of nodes.

However, testing with Mathematica, a Gauss-Kronrod 63 point quadrature only gives, 6.42284; way off. Even a 1000+ point quadrature doesn't give the same accuracy as the TI. And I know the TI isn't doing a 2000 point quadrature. So, what other strategies are being used?

Also, back to HP. My HP-42S, when setting the accuracy to 0.001, I get 6.4811 (almost within 0.01 not 0.001) after several minutes. So, what method is the 42S using?

I wish I took more numerical analysis 30 years ago.

Thanks, CHUCK

Re: Numerical Integration: HP, TI, etc.

Message #2 Posted by ***Gerson W. Barbosa*** *on 23 May 2011, 3:38 p.m.,*
in response to message #1 by Chuck

I hope these and (the still working) links therein may help:

http://www.hpmuseum.org/cgi-sys/cgiwrap/hpmuseum/archv016.cgi?read=94446

http://www.hpmuseum.org/cgi-sys/cgiwrap/hpmuseum/archv016.cgi?read=95394

http://www.hpmuseum.org/cgi-sys/cgiwrap/hpmuseum/archv016.cgi?read=104550

Re: Numerical Integration: HP, TI, etc.

Message #3 Posted by ***Dieter*** *on 23 May 2011, 5:16 p.m.,*
in response to message #1 by Chuck

The classic implementation of the Integrate function (HP-34C, 15C and others) uses a kind of adaptive Romberg method. With every iteration more and more (clevery placed) nodes are used. The integration limits are not evaluated so that a result can be determined even if the function is not defined there.

Please take a look at this page in the articles section. There you will also find a reference to a detailled discussion of the Integrate function and its properties on the 34C, written by W.H. Kahan in HP magazine issues more than thirty years ago.

FWIW: the 35s, set to FIX 3, returns 6,491 for the mentioned integral.

Dieter

Re: Numerical Integration: HP, TI, etc.

Message #4 Posted by ***Chuck*** *on 23 May 2011, 8:21 p.m.,*
in response to message #3 by Dieter

Much thanks Gerson and Dieter. I now have some good reading to do. And a lot of experimenting. Other good integrals to test are the Fresnel functions. I've found a few calculators have trouble with these as well.

CHUCK

Re: Numerical Integration: HP, TI, etc.

Message #5 Posted by ***Kiyoshi Akima*** *on 24 May 2011, 1:09 p.m.,*
in response to message #4 by Chuck

While you're at it, try integrating FRAC(x) (the fractional portion of x, sometimes called FRAC, sometimes called FP, depending on the calculator) from 0.0 to 6.4 . All HP calculators fail spectacularly on this simple function, but at least they do it quickly :-(

Re: Numerical Integration: HP, TI, etc.

Message #6 Posted by ***Thomas Okken*** *on 24 May 2011, 10:13 p.m.,*
in response to message #5 by Kiyoshi Akima

FP is a spectacularly non-analytical function... All the clever quadrature algorithms assume that the function to be integrated is well-behaved. Functions that go wild like cos(1/x) around x=0, or FP which is discontinuous at all integers, are always going to be a problem. This is why numerical integrators on calculators will never be a complete substitute for actual mathematical insight.

Re: Numerical Integration: HP, TI, etc.

Message #7 Posted by ***Gerson W. Barbosa*** *on 24 May 2011, 10:55 p.m.,*
in response to message #6 by Thomas Okken

WolframAlpha gets it right:

http://www.wolframalpha.com/input/?i=integrate+%5Bx+-+Floor%28x%29%5D+from+0+to+6.4

Re: Numerical Integration: HP, TI, etc.

Message #8 Posted by ***Thomas Klemm*** *on 26 May 2011, 7:59 p.m.,*
in response to message #6 by Thomas Okken

> Quote:
>
> All the clever quadrature algorithms assume that the function to be integrated is well-behaved.

Now consider the following polynomial of 6th degree:

```
f(x) = 44 (x-1) (5 x-27) (5 x-16) (8 x-49) (8 x-35) (40 x-81)/4671275427
     + 4 (x-1) (5 x-27) (5 x-16) (8 x-49) (8 x-35) (40 x-11)/701819181
     + 25 (49-8 x) (x-1) (5 x-27) (8 x-35) (40 x-81) (40 x-11)/3658919121
     + 4 (x-1) (5 x-27) (5 x-16) (8 x-49) (40 x-81) (40 x-11)/233939727
     + (49-8 x) (x-1) (5 x-16) (8 x-35) (40 x-81) (40 x-11)/159262983
     + 4 (x-1) (5 x-27) (5 x-16) (8 x-35) (40 x-81) (40 x-11)/4671275427
```

It was chosen to match exactly the FRAC function at the points mentioned by Kiyoshi Akima:
{0.275, 1, 2.025, 3.2, 4.375, 5.4, 6.125}.

This polynomial is well behaved but still the programs that fail with FRAC will also fail with this function.

Cheers
Thomas

Here's a C++ program for those who'd like to verify:

```
#include <stdio.h>
#include <math.h>


double f(double x) {
    return
        44*(x-1)*(5*x-27)*(5*x-16)*(8*x-49)*(8*x-35)*(40*x-81)/4671275427LL
        +4*(x-1)*(5*x-27)*(5*x-16)*(8*x-49)*(8*x-35)*(40*x-11)/701819181LL
        +25*(49-8*x)*(x-1)*(5*x-27)*(8*x-35)*(40*x-81)*(40*x-11)/3658919121LL
        +4*(x-1)*(5*x-27)*(5*x-16)*(8*x-49)*(40*x-81)*(40*x-11)/233939727LL
        +(49-8*x)*(x-1)*(5*x-16)*(8*x-35)*(40*x-81)*(40*x-11)/159262983LL
        +4*(x-1)*(5*x-27)*(5*x-16)*(8*x-35)*(40*x-81)*(40*x-11)/4671275427LL
    ;
}


double x[] = {
    0.275, 1.0, 2.025, 3.2, 4.375, 5.4, 6.125
};


int main() {
    for (int i = 0; i < 7; i++) {
        printf("%d: x = %5.3f    frac(x) = %8.6f   f(x) = %8.6f\n", i, x[i], x[i] - floor(x[i]), f(x[i]));
    }
}
```

It creates the following output:

```
0: x = 0.275    frac(x) = 0.275000   f(x) = 0.275000
1: x = 1.000    frac(x) = 0.000000   f(x) = 0.000000
2: x = 2.025    frac(x) = 0.025000   f(x) = 0.025000
3: x = 3.200    frac(x) = 0.200000   f(x) = 0.200000
4: x = 4.375    frac(x) = 0.375000   f(x) = 0.375000
5: x = 5.400    frac(x) = 0.400000   f(x) = 0.400000
6: x = 6.125    frac(x) = 0.125000   f(x) = 0.125000
```

Re: Numerical Integration: HP, TI, etc.

Message #9 Posted by ***Paul Dale*** *on 26 May 2011, 9:45 p.m.,*
in response to message #8 by Thomas Klemm

Why would all estimations fail on this polynomial?

Gauss/Kronrod quadratures provide some guarantees about correctness when integrating polynomials. I believe Gaussian quadratures of order n are exact for polynomials of degree 2n-1 or less (I might be out by +/- 1 on the degree, it has been a while). Kronrod quadratures extend the Gauss quadrature by using the Gauss points and additional points that provide two estimates of the integral, They do, however, lose something on the degree of polynomial they handle exactly when compared to a Gaussian quadrature with the same number of points (the degree change here has slipped my mind for the moment).

- Pauli

Re: Numerical Integration: HP, TI, etc.

Message #10 Posted by ***Thomas Klemm*** *on 27 May 2011, 4:00 a.m.,*
in response to message #9 by Paul Dale

> Quote:
>
> Why would all estimations fail on this polynomial?

I wrote: "programs that fail with FRAC will also fail with this function."

My assumption was that they fail with FRAC for the same reason as explained in Message #22.

It's obvious that numeric integration algorithms may have problems with "functions that go wild" but this is not obvious in the case of the FRAC function.
You can always construct a polynomial for which a specific integration algorithm fails. What surprised me was that the algorithm used in HP calculators fails for such a simple function as FRAC.

Those who think the polynomial given above is too complicated may try to integrate x^2(x^2-47^2)(x^2-88^2)(x^2-117^2) from -128 to 128.

My HP48 gives 0 as a result whereas WolframAlpha returns 2.72663x10^{16}.

Cheers
Thomas

Edited: 27 May 2011, 4:19 a.m.

Re: Numerical Integration: HP, TI, etc.

Message #11 Posted by ***Paul Dale*** *on 27 May 2011, 4:20 a.m.,*
in response to message #10 by Thomas Klemm

The 34s gets 2.72663118082 x 10^{16}.

This isn't surprising given that the algorithm used should be exact for this polynomial. However, it fails badly for FRAC.

Pauli

Edited: 27 May 2011, 4:21 a.m.

Re: Numerical Integration: HP, TI, etc.

Message #12 Posted by ***Kiyoshi Akima*** *on 27 May 2011, 12:45 p.m.,*
in response to message #10 by Thomas Klemm

> Quote:
>
> What surprised me was that the algorithm used in HP calculators fails for such a simple function as FRAC.

It's also a matter of the limits. The built-in integrator works perfectly well integrating FRAC from 0 to 1, for example. Or from 0 to 3. It has a little more difficulty with integrating from 0 to 2. Also note that it does come up with a more-or-less correct answer if the upper limit is changed from 6.4 to 5.4 .

Re: Numerical Integration: HP, TI, etc.

Message #13 Posted by ***Chuck*** *on 25 May 2011, 12:24 a.m.,*
in response to message #5 by Kiyoshi Akima

Cool. They also fail spectacularly slowly. I played around with the limits a bit. Lower=0 and upper=2 on the HP42 was integrating for about an hour (forgot it was on and let it sit unattended). I ended up hitting EXIT so as to not run out of batteries. (accuracy was set at 0.00001)

Re: Numerical Integration: HP, TI, etc.

Message #14 Posted by ***Valentin Albillo*** *on 25 May 2011, 7:06 a.m.,* *in response to message #5 by Kiyoshi Akima*

> Quote:
>
> While you're at it, try integrating FRAC(x) (the fractional portion of x, sometimes called FRAC, sometimes called FP, depending on the calculator) from 0.0 to 6.4 . All HP calculators fail spectacularly on this simple function, but at least they do it quickly :-(

The HP71 does indeed fail (and quickly at that):

```
  >INTEGRAL(0,6.4,0,FP(IVAR))


     1.28
```

but just halving the interval gives a decently correct result:

```
  >INTEGRAL(0,3.2,0,FP(IVAR))+INTEGRAL(3.2,6.4,0,FP(IVAR))


     3.08006142853
```

My own experimental integration program succeeds as well at minimum settings:

```
  >200 DEF FNF(X)=FP(X)
  >RUN
     X1,X2,Err,# Points: 0,6.4,1E-10,2


        32  3.08  .16
```

which is the exact result, not a 5-digit approximation like the one obtained with the halving trick above.

Best regards from V.

Re: Numerical Integration: HP, TI, etc.

Message #15 Posted by ***Crawl*** *on 26 May 2011, 1:30 p.m.,*
in response to message #14 by Valentin Albillo

Boy, even that halving trick doesn't work for the HP50g. Maybe it *eventually* gives the right answer, but after it ran for several minutes I just aborted the evaluation.

The TI89 still gives exact answers for the halved intervals, and fairly quickly.

Re: Numerical Integration: HP, TI, etc.

Message #16 Posted by ***Tim Wessman*** *on 25 May 2011, 2:24 p.m.,*
in response to message #5 by Kiyoshi Akima

>All HP calculators fail spectacularly

Hmm, the one I am using doesn't. Returns 3.08 as expected. . . ;-)

TW

Re: Numerical Integration: HP, TI, etc.

Message #17 Posted by ***Thomas Klemm*** *on 26 May 2011, 3:45 p.m.,*
in response to message #16 by Tim Wessman

Now I'm curious. Which calculator are you using?

Cheers
Thomas

Re: Numerical Integration: HP, TI, etc.

Message #18 Posted by ***Pal G.*** *on 26 May 2011, 4:20 p.m.,*
in response to message #17 by Thomas Klemm

I was hoping Tim was using an HP 15c+, but the "nonpareil-15c" app, on OSX, result is 1.2800.

Re: Numerical Integration: HP, TI, etc.

Message #19 Posted by ***Tim Wessman*** *on 26 May 2011, 5:15 p.m.,*
in response to message #17 by Thomas Klemm

'Tis a secret. :-)

TW

Re: Numerical Integration: HP, TI, etc.

Message #20 Posted by ***gene wright*** *on 26 May 2011, 5:40 p.m.,*
in response to message #19 by Tim Wessman

He's using the 30b with a 64 point Gaussian Quad program.

Re: Numerical Integration: HP, TI, etc.

Message #21 Posted by ***Pal G.*** *on 26 May 2011, 7:36 p.m.,*
in response to message #20 by gene wright

In the spirit of this thread, I hope Tim is bragging about an HP scientific that accurately integrates Kiyoshi Akima's challenge out of the box. That is, using HP Integrate; no programming required.

Otherwise he could be using a WP34s!

Re: Numerical Integration: HP, TI, etc.

Message #22 Posted by ***Paul Dale*** *on 26 May 2011, 9:06 p.m.,*
in response to message #21 by Pal G.

> Quote:
>
> Otherwise he could be using a WP34s!

The 34s isn't that accurate. It is using an 10/21 point Gauss Kronrod method with no adaptive component.

If someone wants to write a better integration routine, we'd be very interested.

- Pauli

Re: Numerical Integration: HP, TI, etc.

Message #23 Posted by ***Crawl*** *on 26 May 2011, 10:51 a.m.,*
in response to message #5 by Kiyoshi Akima

Yeah, the HP50g gives 1.28 in much less than a second.

The TI89 gives the correct answer of 3.08, taking about 15 seconds.

It looks like what's happening with the HPs is that it first does a midpoint rectangle rule (FP(6.4/2) = FP(3.2) = 0.2. Then 0.2*6.4 = 1.28).

I'm not sure why it terminates there, though.

I'd guess if it wanted to add more points, it would go to three point normal Gaussian integration (so it could reuse the middle point). But if you do that, you get 3.057777777..., not the same answer, and in fact one that is much closer to the correct answer.

Re: Numerical Integration: HP, TI, etc.

Message #24 Posted by ***Kiyoshi Akima*** *on 26 May 2011, 2:33 p.m.,*
in response to message #23 by Crawl

The scheme used, as far as I can tell on every HP calculator that has a built-in integrator, is a modified Romberg with argument scaling designed to reduce problems with periodic functions. The Romberg terminates when three passes evaluate to the same value under the current display setting.

The first pass is a one-point midpoint rule, evaluated at 3.2, giving 1.28.

The second pass adds evaluations at 1.0 and 5.4, which also gives 1.28.

The third pass adds evaluations at 2.025, 4.375, .275, and 6.125, which also happens to come out as 1.28. Since these three results agree regardless of the display setting, the routine decides it must be correct and terminates.

I stumbled across this particular example quite by accident. But when the result came out this bad, I had to investigate.

Re: Numerical Integration: HP, TI, etc.

Message #25 Posted by ***Eddie W. Shore*** *on 26 May 2011, 9:37 p.m.,*
in response to message #23 by Crawl

```
I get
1.28 on the HP 50g almost instantly[br]
3.08 on the Casio Prizm fx-CG 10 almost instantly[br]
3.08 on the TI 84+ after a half of second
1.28 on the HP 35S almost instantly
3.08 on the nSpire CX almost instantly
And 3.1121563505 on SpaceTime iPod/iPad app
```

Edited: 26 May 2011, 9:46 p.m.

Re: Numerical Integration: HP, TI, etc.

Message #26 Posted by ***Oliver Unter Ecker*** *on 27 May 2011, 6:41 p.m.,*
in response to message #25 by Eddie W. Shore

ND1 (iPhone):

FP(x): **3.08** IERR: 1e-2 (compute time: 3.5s)

cos(1/(8-x)): **6.4916** IERR: 3e-4 (compute time: 3.7s)

That's ND1 v1.3.9 (the next update).

Does anyone have a pointer to a good test suite of functions for quality-testing numerical integration? (Or a set, willing to share?)

EDIT: Reflected change in software to only returning digits believed to be accurate.

Edited: 28 May 2011, 5:56 a.m.

Re: Numerical Integration: HP, TI, etc.

Message #27 Posted by ***Ángel Martin*** *on 25 May 2011, 9:47 a.m.,*
in response to message #1 by Chuck

I thought about giving a positive example where INTEG (nested with SOLVE, no less) *works* as advertized - albeit painfully slowly, that's for sure.

Here's a program to calculate the zeroes of the Bessel functions of integer order, J(n,x). Just input the order and wait for the calculated values to appear. Hitting R/S again will continue the search in ascending order.

Using V41 in TURBO mode is highly recommended :-)

Cheers, ÁM.

```
1       LBL "ZJYX"
2       RAD
3       N=?
4       PROMPT
5       STO 00
6       1
7       LBL 00
8       3
9       +
10      "JYX"
```

```
11      SOLVE
12      STOP
13      GTO 00
14      RTN
15      LBL "JYX"
16      RAD
17      STO 01
18      "*JN"
19      0
20      PI
21      INTEG
22      PI
23       /
24      RTN
25      LBL "*JN"
26      RCL 00
27      *
28      X<>y
29      SIN
30      RCL 01
31      *
32      -
33      COS
34      END
```

PS. use this link to check the results:

http://mathworld.wolfram.com/BesselFunctionZeros.html

Edited: 25 May 2011, 10:19 a.m.

Re: Numerical Integration: HP, TI, etc.

Message #28 Posted by ***Oliver Unter Ecker*** *on 28 May 2011, 5:47 a.m.,*
in response to message #1 by Chuck

Re your original question about the TI and Mathematica results for cos(1/(8-x)).

You're saying "while Mathematica gives 6.4916765549 (claiming to use Guass-Kronrod)".

Can you tell us exactly how you tested? Are you sure about it using Gauss-Kronrod?

I spent some time in keisan which permits playing with a number of integration schemes, incl. Gauss-Kronrod. keisan num integration

I can confirm that Gauss-Kronrod (for n=15, 41, 71), does *not* return good results, just as you found in Mathematica. (Using some *other* kind of testing, I assume.)

Interestingly, the result for n=63 does not match the one you got in Mathematica. At 6.46.. it's a little better.

```
n=15
Gauss-Legendre  7       6.876098950292528094576
Gauss-Kronrod   15      6.574616683680514255335
accuracy        -       7


n=41
Gauss-Legendre  20      6.57836956143155286915
Gauss-Kronrod   41      6.566852251450387088799
accuracy        -       6.6


n=63 (slow)
Gauss-Legendre  31      6.46414119548917882461
Gauss-Kronrod   63      6.460665043695674082197
accuracy        -       6.46


n=71 (slow)
Gauss-Legendre  35      6.392977564788352852733
Gauss-Kronrod   71      6.479392744161763499238
accuracy        -       6.5
```

I'm unable to obtain a result good for two decimal places using *any* of the integration methods and ranges for n supported by keisan, even when the server is busy for 15s or so!

Instead, I'm getting frequent input errors with some methods (=function not accepted) and compute errors for "high" values of n.

Using a different implementation, I know that a 2K Gauss-Legendre integration returns 6.4902... which is still worse than your TI result.

WolframAlpha returns a result good to 5 decimal places. Assuming the keisan results are correct, and Gauss-Kronrod does not return a good result for this function using "typical" values of n, I suppose it either
- uses much higher values for n
- or, uses a different integration scheme for this function
- or, quite simply (for Mathematica), integrates symbolically and evaluates!

(ND1's result, which is accurate to 4 decimal places, is obtained via double exponential integration with a computational expense roughly equivalent to a 512-point Gauss integration.)

Edited: 28 May 2011, 6:01 a.m.

Re: Numerical Integration: HP, TI, etc.

Message #29 Posted by ***Gerson W. Barbosa*** *on 28 May 2011, 11:07 a.m.,*
in response to message #28 by Oliver Unter Ecker

> Quote:
>
> WolframAlpha returns a result good to 5 decimal places...
>
> ... I suppose it either
>
> - uses much higher values for n
>
> - or, uses a different integration scheme for this function
>
> - or, quite simply (for Mathematica), integrates symbolically and evaluates!

Most likely the latter, as we can verify by submitting "Integrate Cos[1/(8-x)]dx, from 0 to 8" to WolframAlpha:

http://www.wolframalpha.com/input/?i=Integrate++Cos%5B1%2F%288-x%29%5Ddx%2C+from+0+to+8

The closed form solution is:

```
8*cos(1/8) - pi/2 + Si(1/8)
```

Of course Si(1/8) requires a numerical method, but Si(x) is a well-behaved function. By using this result, we can get nine correct digits on the HP-15C:

```
01- f LBL A
02-   SIN
03-   x<>y
04-   /
05- g RTN
```

That's the program used to integrate Si(x) or sin(x)/x in the HP-15C manual.

```
keystrokes              display


f FIX 9 g RAD 0         0
```

```
ENTER 8 1/x            0.125000000
Integrate A            0.124891544         (after 57 seconds)
8 1/x COS 8 * +        8.062472882
pi 2 / -               6.491676555
```

The WolframAlpha result, when asking once for "More digits" is:

```
6.4916765549444080584305794664323142891737666776228076872897082214341676072377297600588699656218932943
```

Now, changing the upper limit a bit and multiplying the integral by 100 for a weird result :-)

```
Integrate  100 Cos[1/(8-x)]dx, from 0 to 100*(1/3+Exp[-pi])^2/W[100*(1/3+Exp[-pi])^2]
```

Edited: 28 May 2011, 11:17 a.m.

Re: Numerical Integration: HP, TI, etc.

Message #30 Posted by ***Valentin Albillo*** *on 28 May 2011, 10:23 p.m.,*
in response to message #29 by Gerson W. Barbosa

> Quote:
>
> Now, changing the upper limit a bit and multiplying the integral by 100 for a weird result :-)
>
> ```
> Integrate  100 Cos[1/(8-x)]dx, from 0 to 100*(1/3+Exp[-pi])^2/W[100*(1/3+Exp[-pi])^2]
> ```

Let's see ...

```
   1   DEF FNW(X)=FNROOT(0,10,FVAR*EXP(FVAR)-X)
   2   N=100*(1/3+EXP(-PI))^2 @ DISP INTEGRAL(0,N/FNW(N),0,100*COS(1/(8-IVAR)))


   >RUN
        665.999991757
```

... so I guess your *"weird result"* was intended to be **666**.

As for the original troublesome integral, this keyboard statement ...

```
    INTEGRAL(0,7.99,0,COS(1/(8-IVAR)))+INTEGRAL(7.99,8,0,COS(1/(8-IVAR)))
```

```
        6.4916773758
```

... gives the result correct to **7** digits.

Best regards from V.

Edited: 28 May 2011, 10:36 p.m.

Re: Numerical Integration: HP, TI, etc.

*Message #31 Posted by **Gerson W. Barbosa** on 28 May 2011, 10:51 p.m.,*
in response to message #30 by Valentin Albillo

> Quote:
>
> ```
> >RUN
>         665.9999991757
> ```

Emu71 does it instantly. I had to try it on the real thing: only 35 seconds! The HP-71B with the MATH ROM never stops amazing me!

Best regards,

Gerson.

[Return to Index | Top of Index]

GTO Go back to the main exhibit hall