

HP Forum Archive 20

[[Return to Index](#) | [Top of Index](#)]

Impressive TI Nspire CX CAS Multiple Integration

Message #1 Posted by [Namir](#) on 18 Dec 2011, 1:59 p.m.

I have been playing with double and triple integration using the Monte Carlo method and with nested Simpson 1/3 rules using Excel VBA. The Monte Carlo method is very easy to program. Both methods work well, with the nested Simpson method performing better than the Monte Carlo for the same number of function calls.

Out of curiosity, I powered up my TI Nspire CX CAS (the newest version) and tried to perform double and triple integrations. To my surprised, the machine gave me the correct answer in both cases, instead of complaining that it could only do single integration! Very impressive TI! Sorry 15C, you no can do!!!

Namir

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #2 Posted by [Crawl](#) on 18 Dec 2011, 7:49 p.m.,
in response to message #1 by Namir

Pretty sure the TI89 and HP50g can do multiple integration out of the box, too.

No idea what the point of your last line was. Obviously the HP15C can't do any symbolic integrals, and while it might not be able to do triple integrals numerically out of the box, I'm sure such abilities can be programmed.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #3 Posted by [Namir](#) on 18 Dec 2011, 9:04 p.m.,
in response to message #2 by Crawl

I certainly would like to see an HP-15C program that uses the built-in Integrate feature to perform multiple integration.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #4 Posted by [Valentin Albillo](#) on 19 Dec 2011, 6:27 a.m.,
in response to message #3 by Namir

Quote:

I certainly would like to see an HP-15C program that uses the built-in Integrate feature to perform multiple integration.

The **HP-34C Solution Book "Advanced Math"** features a program of mine which does exactly that to implement double integrals on the HP-34C, using 3-point Gauss for the outer integral and the built-in integration capability for the inner one.

Though specifically written for the HP-34C it can be ported to the **HP-15C** with little or no change. Come to that, the same applies to all programs featured in the book, but if you're interested just in the particular one which computes double integrals and you can't find the Solutions Book, it's also featured here:

<http://membres.multimania.fr/albillo/calc/pdf/DatafileVA024.pdf>

Best regards from V.

Edited: 19 Dec 2011, 6:43 a.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #5 Posted by **Namir** on 19 Dec 2011, 8:16 a.m.,
in response to message #4 by Valentin Albillo

Thank you Valentin! Your article has definitely answered my question!

Namir

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #6 Posted by **Valentin Albillo** on 19 Dec 2011, 10:58 a.m.,
in response to message #5 by Namir

Quote:

Thank you Valentin! Your article has definitely answered my question!

You're welcome. If you're interested in testing out some particular methods against hard multiple integrals you may want to check **problems 6, 7, and 10** in this interesting paper:

<http://crd.lbl.gov/~dhbailey/dhbpapers/tenproblems.pdf>

Problem 7 in particular, I included in this ancient post where I gave both the **HP-71B MATH ROM** implementation as well as a simple **Monte-Carlo BASIC** implementation:

<http://www.hpmuseum.org/cgi-sys/cgiwrap/hpmuseum/archv014.cgi?read=53268>

Best regards from V.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #7 Posted by [Ángel Martin](#) on 19 Dec 2011, 10:27 a.m.,
in response to message #4 by Valentin Albillo

Very portable to the 41/Advantage as well, and running great on the CL ;-)

Cheers, ÁM.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #8 Posted by [Ángel Martin](#) on 19 Dec 2011, 2:55 p.m.,
in response to message #4 by Valentin Albillo

Here's the listing for the 41 version - in case you wonder:- It uses R10 instead of "I", so the usage is simple

1. store number of intervals in R10, set FIX for precision
2. type the function program name in Alpha
3. enter integration limits in T,Z,Y,X
4. XEQ "2DITG"

Enjoy, 'AM

```

01      LBL "2DITG"
02      STO 03
03      RDN
04      STO 04
05      RDN
06      X<>Y
07      STO 06
08      -
09      RCL 10
10      /
11      STO 07
12      RCL 08
13      ST- 08
14      LBL 02
15      RCL 06
16      RCL 06
17      RCL 07
18      ST+ 06
19      +
20      STO 00

```

```
21 X<>Y
22 ST- 00
23 +
24 2
25 ST/ 00
26 /
27 STO 01
28 XEQ 00
29 +
30 XEQ 01
31 STO 02
32 RCL 01
33 XEQ 00
34 -
35 XEQ 01
36 ST+ 02
37 5
38 ST* 02
39 RCL 01
40 XEQ 01
41 8
42 *
43 ST+ 02
44 RCL 00
45 ST* 02
46 9
47 ST/ 02
48 RCL 02
49 ST+ 08
50 DSE 10
51 GTO 02
52 RCL 08
53 RTN
54 LBL 00
55 RCL 00
56 ,6
57 SQRT
58 *
59 RTN
60 LBL 01
61 STO 05
62 RCL 04
63 RCL 03
64 INTEG
65 END
```

Edited: 19 Dec 2011, 2:57 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #9 Posted by **Valentin Albillo** on 19 Dec 2011, 4:24 p.m.,
in response to message #8 by Ángel Martin

Hi, Ángel:

Thanks for your interest and for the translation.

Matter of fact this simple program translates with minimal effort to run in many other RPN models (HP-34C, HP-15C, HP-41** + Advantage ROM, etc) and an HP-71B version would be equally straightforward but moot as the **INTEGRATE** keyword (a funny function) allows for up to 5-dimensional integrals *per se*.

A useful template to compute multiple integrals with the HP-71B plus examples can be found at point 10 in one of my articles, namely:

<http://membres.multimania.fr/albillo/calc/pdf/DatafileVA012.pdf>

By the way, I don't remember whether I've already ranted about this here but I think that the HP programmer (or team) who decided to name the keywords **INTEGRATE** instead of **INTEG**, **FNROOT** instead of **SOLVE**, **FVAR** instead of **FV**, and **IVAR** instead of **IV** had *absolutely no idea of what "ergonomic" and "economic" mean*, as the shorter keywords are faster to type and most importantly you can fit more complex expressions in the command line for immediate execution without the hassle of having to write a program just because your expression doesn't fit in a line (which is easily the case with multiple integrals).

Best regards from V.

Edited: 19 Dec 2011, 4:33 p.m.

Multiple Integration - WP 34S

Message #10 Posted by **Marcus von Cube, Germany** on 19 Dec 2011, 5:05 p.m.,
in response to message #9 by Valentin Albillo

With the recent addition of local registers and recursive programming, WP 34S has the ability to do multiple integration with the built-in integrator (a non adaptive Gauss-Kronrod quadrature). I've not done it, but it should be straight forward to write a subroutine for the function to integrate and another which calls the integrator for the defined function. This subroutine can then be used as the argument for the integrator. Not as comfortable as just typing in a formula but still easier than creating your own integrator in RPN.

Re: Multiple Integration - WP 34S

Message #11 Posted by **Marcus von Cube, Germany** on 20 Dec 2011, 12:31 p.m.,
in response to message #10 by Marcus von Cube, Germany

Here is a short example of a recursive call to the WP 34S integrator. It computes $S(0,1,S(0,1 X*Y,Y),X)$. The result is 0.25. No global registers are used, the outer subroutine at LBL 00 creates a local register .00 which is visible to the inner subroutine at LBL 01.

```

LBL A
0
ENTER[^]
1
[integral] 00
RTN

LBL 00
LocR 00
STO .00
0
ENTER[^]
1
[integral] 01
RTN

LBL 01
RCL[times] .00
RTN

END

```

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #12 Posted by [Oliver Unter Ecker](#) on 19 Dec 2011, 5:32 a.m.,
in response to message #1 by Namir

Hi Namir,

Can you show us the examples you tried? I'm interested to see how the syntax looks for multiple integration.

Does it work for both definite and indefinite integrals?

Have you been seeing it fail for some other examples? (Would like to see those, too.)

Thanks!

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #13 Posted by [Namir](#) on 19 Dec 2011, 8:13 a.m.,
in response to message #12 by Oliver Unter Ecker

I tried the simple integrals:

$$\int \sqrt{x} \quad \int \sqrt{x} \quad \int \sqrt{x} \quad \int \sqrt{x}$$

$$\int_0^1 \int_0^1 X*Y \, dX \, dY = 0.25$$

$$\int_0^1 \int_0^1 \int_0^1 X*Y*Z \, dX \, dY \, dZ = 0.125$$

Syntax on the 50g

Message #14 Posted by [Oliver Unter Ecker](#) on 19 Dec 2011, 11:55 a.m.,
in response to message #13 by Namir

Thanks. (I was secretly hoping for something a bit more exciting...)

How about the syntax on the TI?

On the 50g, I notice that 'S(11,14,X^2+4*Y,X)' EVAL works, whereas 'S(7,10,S(11,14,X^2+4*Y,X),Y)' does **not** enter. ("Invalid syntax" is reported.)

In an RPL program, one could feed the (symbolic) output of the inner integral into the outer integral. So, yes, as per Crawl, there is out-of-box capability for multiple integration on the 50g.

(On ND2, in development, S(7,10,S(11,14,X^2+4*Y,X),Y) works and returns 1719.)

["S" is my poor man's integral symbol.]

Re: Syntax on the 50g

Message #15 Posted by [Crawl](#) on 19 Dec 2011, 1:45 p.m.,
in response to message #14 by Oliver Unter Ecker

It should work.

My tip for figuring out the syntax on the HP50g is to enter an expression in the equation editor (which is usually intuitive), but if you don't like that and want the "code", turn off "textbook" in display.

That gave me

'S(7,10,S(11,14,(x^2)+(4*y),x),y)'

which evaluates to 1719

I'm not able to duplicate your error. It enters fine (on the stack) even if I enter what you have (which is identical to what I have, minus the extra parenthesis)

Maybe you just made a typo on the calculator?

Or maybe there's a mode difference, but I don't know which mode should make a difference, or why.

Re: Syntax on the 50g

Message #16 Posted by [Crawl](#) on 19 Dec 2011, 1:49 p.m.,
in response to message #15 by Crawl

Quote:

I'm not able to duplicate your error. It enters fine (on the stack) even if I enter what you have (which is identical to what I have, minus the extra parenthesis)

And, actually, that's only because I had the "show all parens" flag checked. Without it, it looks absolutely identical to what you had.

Re: Syntax on the 50g

Message #17 Posted by [Oliver Unter Ecker](#) on 19 Dec 2011, 2:24 p.m.,
in response to message #15 by Crawl

Thanks, Crawl.

You're right. It's all good.

I had entered

'S(7,10,S(11,14,X^2+4*Y,X),Y

and expected auto-completion to provide the closing chars for me, but, that, apparently, doesn't work in this instance and caused the syntax error.

(Yes, I'm that lazy/trusting of auto-completion...)

Edited: 19 Dec 2011, 2:32 p.m.

Re: Syntax on the 50g

Message #18 Posted by [Crawl](#) on 19 Dec 2011, 10:01 p.m.,
in response to message #14 by Oliver Unter Ecker

Quote:

How about the syntax on the TI?

It's pretty intuitive on the TI89. Just nest the integrals like on the HP50g (except that the limits come at the end rather than the beginning of the syntax)



$$\int_7^{10} \int_{11}^{14} (x^2 + 4 \cdot y) dx dy$$

1719

...x^2+4*y, x, 11, 14, y, 7, 10

MAIN RAD AUTO 30 1/30

Re: Syntax on the 50g

Message #19 Posted by [Oliver Unter Ecker](#) on 20 Dec 2011, 4:33 a.m.,
in response to message #18 by Crawl

Thanks for that, Crawl.

Makes sense. I had wondered if there might be, perhaps, a special "SS" syntax.

Re: Syntax on the 50g

Message #20 Posted by [Crawl](#) on 20 Dec 2011, 9:38 a.m.,
in response to message #19 by Oliver Unter Ecker

The problem with that approach is that you then wouldn't be able to do quadruple, quintuple, etc., integrals.

The great thing about the CAS in these modern calculators is that they're fully integrated, so you can (for example) combine complex numbers with symbolic expressions in matrices, or (as in this case) combine integrals with integrals to do multiple integration, etc.

There's probably no limit to the number of integrals you can nest, other than calculator memory and the increasing amount of time it would take to evaluate them.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #21 Posted by [Crawl](#) on 19 Dec 2011, 8:35 a.m.,
in response to message #1 by Namir

Quote:

Both methods work well, with the nested Simpson method performing better than the Monte Carlo for the same number of function calls.

Monte Carlo eventually wins, because the error goes as 1 over the square root of the number of function evaluations, [i]regardless of the dimension of the integral[i].

On the other hand, with nested Simpsons, error goes as 1 over the number of function evaluations raised to the (4/dimension) power.

So when you have 8 dimension integrals, Monte Carlo and Simpsons are equally good. For 9 or more dimension integrals, Monte Carlo is better.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #22 Posted by [Crawl](#) on 20 Dec 2011, 3:47 p.m.,
in response to message #21 by Crawl

I tried the 6-dimensional integral of $\tan(x*y*z*t*a*b)$ over x,y,z,t,a,b , all variables ranging between 0 and 1.

In maybe about a minute I set up an Excel spreadsheet that calculated it with Monte Carlo simulation (30K points), getting 0.015.

Anyway care to try to do better?

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #23 Posted by [Valentin Albillo](#) on 20 Dec 2011, 6:30 p.m.,
in response to message #22 by Crawl

Quote:

I tried the 6-dimensional integral of $\tan(x*y*z*t*a*b)$ over x,y,z,t,a,b , all variables ranging between 0 and 1.

In maybe about a minute I set up an Excel spreadsheet that calculated it with Monte Carlo simulation (30K points), getting 0.015.

Anyway care to try to do better?

Well, this extremely simple **HP-71B** Monte Carlo code (80+ bytes):

```
10 DESTROY ALL @ RANDOMIZE 1 @ RADIANS
20 FOR K=1 TO 6 @ M=10^K @ S=0 @ DISP K;M, @ FOR I=1 TO M
30 S=S+TAN(RND*RND*RND*RND*RND*RND) @ NEXT I @ DISP S/M @ NEXT K
```

repeatably runs to give:

>RUN

```
1 10      6.5740257498E-3
2 100     1.64187909582E-2
3 1000    1.67187076886E-2
4 10000   1.51787016672E-2
5 100000  1.56906185411E-2
6 1000000 1.57093573935E-2
```

which agrees with your result. The final value should have about $\text{Log}_{10}(\text{Sqrt}(1000000)) = 3$ correct digits so **0.0157** should be accurate to all the digits shown.

Curiously enough, the final value (**0.015709**) closely resembles $\text{Pi}/200 = \mathbf{0.015708}$... I'll let you figure out whether that's a coincidence or not ... :D

Best regards from V.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #24 Posted by **Thomas Klemm** on 21 Dec 2011, 12:49 a.m.,
in response to message #23 by Valentin Albillo

Quote:

I'll let you figure out whether that's a coincidence or not ... :D

I used the Taylor expansion of $\tan(x)$ (up to the term of x^{19}) and got:

1.57094692997E-2

Compared to $\text{Pi}/200$ I'd say: coincidence.

Kind regards
Thomas

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #25 Posted by **Crawl** on 22 Dec 2011, 4:53 p.m.,
in response to message #24 by Thomas Klemm

Thomas Klemm probably has the winning technique.

I'll post some solutions one at a time since it'll take a while to get it all together.

This is Simpsons's Rule on an HP50g.

First you need this program

```
%%HP: T(3)A(R)F(.);
\<< \-> N F T
  \<< N 2 / DUP IP == N 0 \=/ AND N T \=/ AND
    IF
      THEN F 2 *
```

```

ELSE N 2 / DUP IP \=/
  IF
  THEN 4 F *
  ELSE F
  END
END
\>>
\>>

```

which I named SIMPCO.

Then this is the program you run:

```

%%HP: T(3)A(R)F(.);
\<< 0 0 0 0 0 0 0 0 0 0 0 0 \-> T A B C D E F S P1 P2 P3 P4 P5 P6
\<< 0 T
  FOR A A 1 T SIMPCO 'P1' STO 0 T
  FOR B B P1 T SIMPCO 'P2' STO 0 T
  FOR C C P2 T SIMPCO 'P3' STO 0 T
  FOR D D P3 T SIMPCO 'P4' STO 0 T
  FOR E E P4 T SIMPCO 'P5' STO 0 T
  FOR F F P5 T SIMPCO 'P6' STO A B C D E F * * * * T 6 ^ / TAN T 3 * 6 ^ / P6 * S + 'S' STO
  NEXT
  NEXT
  NEXT
  NEXT
  NEXT
  NEXT S
\>>
\>>

```

I named it SIMP6, but it doesn't really matter what you name it.

You give as an argument an even number (t). It will do (t/2) applications of Simpsons rule in each dimension, which works out to (t+1)^6 function evaluations.

Results:

t	Calculation Result	Time on Calculator	Time on Emulator
2	1.57143781232e-2	2min22sec	4sec
4	1.57096718319e-2	x	1min18sec
6	1.57095084843e-2	x	9min48sec

You probably could do the t=6 case on an actual calculator, provided you were willing to let it run for 6 or 7 hours. Better use USB power.

Edited: 22 Dec 2011, 4:54 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #26 Posted by [Crawl](#) on 22 Dec 2011, 6:04 p.m.,

in response to message #25 by Crawl

Here's the Taylor series solution.

```
%%HP: T(3)A(R)F(.);
\<< X TAN X ROT TAYLR X A B C D E F * * * * * = SUBST A RISCH B
RISCH C RISCH D RISCH E RISCH F RISCH A 1 = SUBST B 1 = SUBST
C 1 = SUBST D 1 = SUBST E 1 = SUBST F 1 = SUBST \->NUM
\>>
```

The argument is the number of terms to take in the Taylor series.

And results:

t	Result	Calculator Time	Emulator Time
1	1.5625e-2	9sec	
3	1.57063802083e-2	20sec	
5	1.57092380044e-2	32sec	
7	1.5709443877e-2	44sec	
9	1.57094657465e-2	1min	
...			
15	1.57094692786e-2		5sec
17	1.57094692959e-2		6sec
19	1.57094692997e-2	3min	7sec
21	1.57094693005e-2		9sec
...			
29	1.57094693006e-2	3min34sec	9sec

Also, I'm getting slight differences between the physical and the emulator. The emulator gives a result 1ulp less for the t=19 and 29 cases I checked on both.

Edited: 22 Dec 2011, 6:15 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #27 Posted by [Thomas Klemm](#) on 23 Dec 2011, 12:24 a.m.,

in response to message #26 by Crawl

Is it possible to just calculate the coefficients of the Taylor series using the HP-50G? I was trying to do that using the HP-48G without success. So I let WolframAlpha do this and entered them into the HP-42S which was quite a pain, you may believe me. Once I have stored them into the registers 01, 02, ... I used the following program:

```
00 { 33-Byte Prgm }
01 LBL "ITAN"
02 STO "n"
03 10
04 STO 00
05 CLX
06 LBL 00
07 RCL IND 00
```

```

08 RCL 00
09 2
10 *
11 RCL "n"
12 Y^X
13 /
14 +
15 DSE 00
16 GTO 00
17 END

```

To run the program you just enter the dimension, in this case $n = 6$.

With minor changes this program could be used with the HP-15C and most other models as well. So yes, with only a little cheating (i.e. using W|A for the coefficients) we can do multiple integration. Purists would first write a program to calculate the Bernoulli-numbers to get the coefficients without that ugly hack. Now I'm curious how the TI NSpire CX handles this problem.

This Taylor expansion converges only slowly for the upper bound $x = 1$. Therefore I tried to find a substitution that might transform that to something like $1/2$ but to no avail. Maybe somebody else has an idea?

Since the quotient of succeeding coefficients a_i/a_{i+1} converges to $(\pi/2)^2$ some kind of convergence acceleration could be helpful but I didn't try that.

Here we're just lucky because $n = 6$: the harder you're trying to make it by adding more dimensions, the faster it converges.

Kind regards
Thomas

Edited: 23 Dec 2011, 12:47 a.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #28 Posted by [Thomas Klemm](#) on 23 Dec 2011, 2:45 a.m.,
in response to message #27 by Thomas Klemm

Quote:

Purists would first write a program to calculate the Bernoulli-numbers to get the coefficients without that ugly hack.

For those too lazy to enter those numbers manually but still willing to enter a program into the calculator:

```

00 { 49-Byte Prgm }
01 LBL "INIT"
02 ENTER
03 +
04 LASTX
05 1E3

```

```

06 /
07 1
08 +
09 STO 00
10 RDN
11 LASTX
12 RCL+ ST Y
13 N!
14 1/X
15 LBL 00
16 RCL IND 00
17 RCL ST Z
18 N!
19 /
20 X<>Y
21 -
22 X<>Y
23 2
24 -
25 X<>Y
26 ISG 00
27 GTO 00
28 STO IND 00
29 RCL 00
30 IP
31 END

```

Now you enter the following:

```

1
STO 01
XEQ "INIT"
R/S
R/S
R/S
R/S
R/S
(... )

```

Just keep on pressing [R/S] until you feel like it's enough, let's assume stop with 20. Now replace line 03 in the other program ("TTAN") with this number.

BTW: In the end I didn't use the Bernoulli-numbers but simply the fact that: $\sin(x) = \cos(x) \tan(x)$. This gives a system of equations for the unknown coefficients of $\tan(x)$.

So I hope the purists are happy now.

Cheers
Thomas

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #29 Posted by [Crawl](#) on 23 Dec 2011, 6:25 p.m.,
in response to message #27 by Thomas Klemm

Quote:

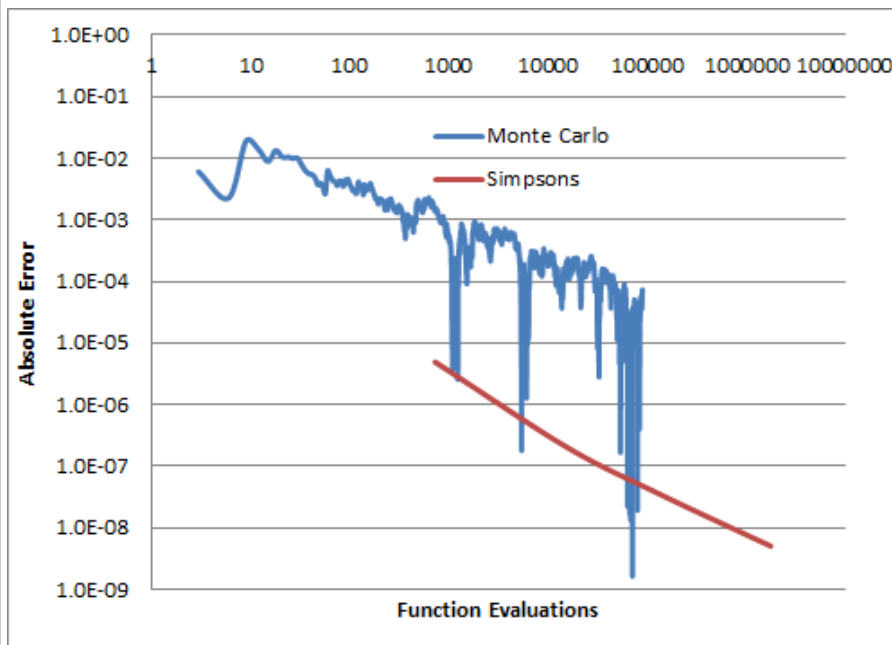
Is it possible to just calculate the coefficients of the Taylor series using the HP-50G? I was trying to do that using the HP-48G without success.

Yes, it can. I don't know about the 48.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #30 Posted by [Crawl](#) on 22 Dec 2011, 10:37 p.m.,
in response to message #25 by Crawl

And this compares the absolute error between using the Monte Carlo method versus Simpson's Rule.

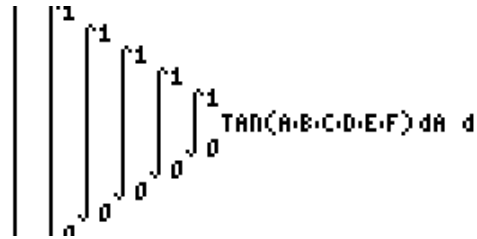


Even for this problem, Simpson's Rule is more efficient than Monte Carlo.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #31 Posted by [Crawl](#) on 23 Dec 2011, 6:21 p.m.,
in response to message #30 by Crawl

Another thing you could try would be to just enter the integral as-is and try to let the built-in integral routines try to deal with it.



I think we have evidence here that you should be able to get at least a few digits of accuracy this way, and by adjusting the FIX setting you can only ask for a few.

```
⌘ TIME '∫(0..1,∫(0..1,∫(0..1,  
∫(0..1,∫(0..1,∫(0..1,TAN(A*B*  
C*D*E*F),A),B),C),D),E),F)' →NUM  
TIME 600. 1. BEEP  
⌘
```

GRAPH **OK**

You don't really need a program for this, but this lets the calculator time itself, and I can also just let it run and do something else and have the calculator alert me when it's finished.

Fix	Returned	Full	Time (with Emulator)
0	2.e-2	1.57089974862e-2	2min26sec
1	1.6e-2	1.57089974862e-2	2min30sec
2	0.02	1.57089974862e-2	2min32sec
3	0.016	1.57090853677e-2	3min46sec
4	0.0157	1.57094436066e-2	43min45sec
5	?		Over 6 hours!

I didn't even bother trying this with a physical calculator.

Edited: 23 Dec 2011, 6:21 p.m.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #32 Posted by [Crawl](#) on 24 Dec 2011, 11:37 a.m.,
in response to message #31 by Crawl

The FIX5 case finally stopped running.

The clock says it took 25 hours, 13 minutes, but that's very approximate, since the computer was in sleep mode several times over that period.

The result is

1.57094677879e-2

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #33 Posted by **Thomas Klemm** on 25 Dec 2011, 3:18 p.m.,
in response to message #32 by Crawl

Quote:

25 hours, 13 minutes

That's amazing! Just for comparison: it takes ~60ms for Free42 on my iPhone to calculate this integral to 16 correct places (both "INIT" and "TAN") when I use 20 terms to get:

1.570946930078026084759965e-2

However with 40 terms it takes ~240ms to produce 24 correct places¹:

1.57094693007802638993157e-2

Just imagine how long it would take to do that with your method.

Cheers
Thomas

1. Of course I can't be sure whether all places are correct, but it's the case for $n = 1$ compared to $-\log(\cos(1))$ and $n = 2$ compared to Valentin's result when I use 100 terms. Therefore I assume the coefficients of the Taylor series are correct. And with $n = 6$ fewer terms are needed to get the same result.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #34 Posted by **Oliver Unter Ecker** on 23 Dec 2011, 6:04 p.m.,
in response to message #25 by Crawl

Nice job, Crawl.

I tried your code, unchanged, on my calcs.

Results:

t	Calculation Result	Time on ND1	Time on CalcPad/Mac
2	0.0157143781230784	12sec	0.7sec
4	0.0157096718412068	x	12.6sec
6	0.0157095084875193	x	88sec

Then I gave it a quick speed treatment.

I used what's called "natural math entry" on ND1 for SIMPCO and changed it to this:

$$f(N,F,T)=F * (N\%2 \text{ ? } 4 : (N \&\& N != T \text{ ? } 2 : 1))$$

Then I changed some RPL instructions into RPL+ for SIMP6:

```
\<< 0 \-> T S
\<< 0 T
  FOR A A 1 T simpco =P1 0 T
    FOR B B P1 T simpco =P2 0 T
      FOR C C P2 T simpco =P3 0 T
        FOR D D P3 T simpco =P4 0 T
          FOR E E P4 T simpco =P5 0 T
            FOR F F P5 T simpco =P6 A B C D E F * * * * T 6 ^ / TAN T 3 * 6 ^ / P6 * =+S
          NEXT
        NEXT
      NEXT
    NEXT
  NEXT
NEXT S
\>>
\>>
```

New timings:

t	Calculation Result	Time on ND1	Time on CalcPad/Mac
2	0.0157143781230784	0.8sec	0.052sec
4	0.0157096718412068	17s	0.9sec
6	0.0157095084875193	2min6sec	6.9sec
8	0.0157094817061892	x	32sec

Edited: 23 Dec 2011, 7:12 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #35 Posted by **Oliver Unter Ecker** on 23 Dec 2011, 7:37 p.m.,
in response to message #34 by Oliver Unter Ecker

A further 3x speed improvement is obtained by changing this RPL code

$$A B C D E F * * * * T 6 ^ / TAN T 3 * 6 ^ / P6 *$$

into this RPL+ code

$$P6 * \text{TAN}((A * B * C * D * E * F) / T^6) / (T * 3)^6$$

Another 10% is squeezed out by writing SIMPSO in JavaScript, like so:

```
function (N, F, T) { /*as is*/
  return F * (N%2 ? 4 : (N && N != T ? 2 : 1));
}
```

The improved timings:

t	Calculation Result	Time on ND1	Time on CalcPad/Mac
2	0.0157143781230784	0.3sec	0.022sec
4	0.0157096718412068	4.7sec	0.3sec
6	0.0157095084875193	33sec	2.2sec
8	0.0157094817061892	x	9.6sec
10	0.0157094743918417	x	31sec
12	0.0157094717597684	x	1min23sec

That's ~500x faster than a 50g for this task, on <10x faster HW (iPhone vs. 50g HW).

Edited: 23 Dec 2011, 7:45 p.m.

Re: Impressive TI Nspire CX CAS Multiple Integration

Message #36 Posted by [Oliver Unter Ecker](#) on 20 Dec 2011, 7:00 p.m.,
in response to message #22 by [Crawl](#)

I had W|A try better.

```
integrate (integrate (integrate (integrate (integrate (integrate (tan(x*y*z*t*a*b))
dx from 0 to 1) dy from 0 to 1) dz from 0 to 1) dt from 0 to 1) da from 0 to 1) db from 0 to 1)
```

First it wanted more time, then it volunteered another server, as the first decided to quit. (Again! It's pretty easy to get into that condition with W|A. I wonder if it really, physically, brings down a server. The icon of a server farm suggests as much to me.)

Maybe there's a better question to ask W|A that will return a result.

Anyway. See, that's where "SSSSSS" syntax may be useful: for higher dimensions, switch to an altogether different algorithm. Here: Monte Carlo. The nested syntax doesn't easily allow for that.

Here's home-baked Monte-Carlo in RPL+:

```
\<< 5 =1
  0 =sum
  1 1 START 1 1 START 1 1 START
  1 1 START 1 1 START 1 1 START
    random =x random =y random =z
    random =t random =a random =b
```

```

tan(x*y*z*a*b*t) =+sum
NEXT NEXT NEXT
NEXT NEXT NEXT
sum/1^6
\>>

```

Returns 0.015xxx, with latter digits varying by run, in ~7s on my iPhone.

That's for 5^6 (~16K) samples. Even with 7^6 (~118K) samples, it's still showing variability past the 5.

[Crawl: Off-topic, but since you had responded to my question about modular exponentation of a non-integer and the thread is archived: Here's how that played out: As suspected, this was not about any non-integer. The polynomial was special, and so were its roots. The secret sauce knowledge to have was about Pisot numbers, and Newton's Identities. Took me *weeks* to figure out, even with hints... but ultimately a great problem to solve. And running on an app calc in 1s/root.]

[EDIT: Added RPL+ code.]

Edited: 20 Dec 2011, 7:56 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #37 Posted by [Gerson W. Barbosa](#) on 20 Dec 2011, 8:15 p.m.,
in response to message #36 by Oliver Unter Ecker

Quote:

I had W|A try better.

```

integrate (integrate (integrate (integrate (integrate (integrate (tan(x*y*z*t*a*b))
dx from 0 to 1) dy from 0 to 1) dz from 0 to 1) dt from 0 to 1) da from 0 to 1) db from 0 to 1)

```

It will understand the following syntax:

```
integrate tan(x*y*z*t*a*b) dx dy dz dt da db, x=0..1, y=0..1, z=0..1, t=0..1, a=0..1, b=0..1
```

However, this is all it gives out:

Computation timed out. No more results available.

which might mean

Buy Mathematica, buy Mathematica... :-)

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #38 Posted by [Valentin Albillo](#) on 21 Dec 2011, 9:22 a.m.,
in response to message #37 by Gerson W. Barbosa

Quote:

However, this is all it gives out:

Computation timed out. No more results available.

which might mean

Buy Mathematica, buy Mathematica... :-)

Surely it does. W/A, while being an outstanding piece of software and a very useful one at that, also fails miserably in many surprisingly simple computational tasks with the added puzzlement that sometimes it will work alright for a very slight variation of the computation that's giving it the bends.

I initially submitted a lot of reports but never received a reply or noticed any action at all being taken on them so I don't bother anymore.

Best regards from V.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #39 Posted by [Thomas Klemm](#) on 23 Dec 2011, 3:03 a.m.,
in response to message #38 by Valentin Albillo

Quote:

I initially submitted a lot of reports but never received a reply or noticed any action at all being taken on them so I don't bother anymore.

My experiences with WolframAlpha were positive: I once reported a bug and was informed some time later, when it was fixed. So I encourage reporting them.

Kind regards

Thomas

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #40 Posted by [Valentin Albillo](#) on 21 Dec 2011, 4:37 p.m.,
in response to message #37 by Gerson W. Barbosa

Quote:

It will understand the following syntax:

integrate tan(x*y*z*t*a*b) dx dy dz dt da db, x=0..1, y=0..1, z=0..1, t=0..1, a=0..1, b=0..1

Trying that syntax out in W/A with the 2-dimensional version, namely:

```
integrate tan(x*y) dx dy, x=0..1, y=0..1
```

produces 0.275687 after a while and refuses to compute any more digits ("*Computation timed out*") which is pretty useless for identification purposes.

On the other hand, the **HP-71B** produces with the help of just a little code (and my constant recognition program):

```
> CAT

workfile BASIC 54 07/18/04 12:38

> LIST

10 DEF FNF(X)=INTEGRAL(0,1,0,TAN(X*IVAR))
20 DISP INTEGRAL(0,1,0,FNF(IVAR))

> CALL @ CALL IDENTIFY(RES,S$) @ S$

.2756872738 ("identified as") 5/179*Pi^2
```

which seems quite a reasonable agreement as the alleged symbolic identification evaluates to .275687273774, i.e., a *10-digit agreement* with the numerically computed value for the double integral.

Whether the symbolic identification is absolutely correct would require many more digits and that would only serve as empirical reassuring, never proof, that would demand symbolics.

Oh, and by the way, a Monte-Carlo run with 10^7 samples produces .27571⁶⁷⁵⁴⁷³⁵, which is good to nearly 5 digits, not bad for a dumb probabilistic method.

Best regards from V.

Edited: 21 Dec 2011, 4:50 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #41 Posted by **Gerson W. Barbosa** on 21 Dec 2011, 5:37 p.m.,
in response to message #40 by Valentin Albillo

Hello Valentin,

Quote:

Trying that syntax out in W/A with the 2-dimensional version, namely:

```
integrate tan(x*y) dx dy, x=0..1, y=0..1
```

produces **0.275687** after a while and refuses to compute any more digits ("*Computation timed out*") which is pretty useless for identification purposes.

I had tried that on W|A but I failed to notice any connection to pi. I had divided 0.275687 by pi on the HP-32SII and obtained 8.77538976e-2. Had I divided it again by pi I would have gotten 2.79329332e-2. Finally FDISP would have returned 0 5/179.

Best regards,

Gerson.

P.S.: By the way, one of these days I observed $e^{(e*\pi)} = 5113.98499978$, which multiplied by 200 equals 1022796.99996. A coincidence?

Edited: 21 Dec 2011, 5:43 p.m.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #42 Posted by **Gerson W. Barbosa** on 22 Dec 2011, 6:29 a.m.,
in response to message #40 by Valentin Albillo

Quote:

Trying that syntax out in W/A with the 2-dimensional version, namely:

```
integrate tan(x*y) dx dy, x=0..1, y=0..1
```

produces **0.275687** after a while and refuses to compute any more digits ("*Computation timed out*") which is pretty useless for identification purposes.

W|A returns "integral tan(x y) dx = -(log(cos(x y)))/y+constant" for "integrate tan(x*y) dx". We can use the HP-15C

```
f LBL A
  COS
g LN
  x<>y
  /
g RTN
and get
```


0 ENTER 1 Integrate A CHS -> 0.2756872439

We can also try [this online integrator](#) and obtain

Numerical Integration of $-(\log(\cos(x)))/x$ in the interval $[0,1]$

Method	Value	Estimated Error
Romberg	0.2756872738004371	2e-16
Gauss-Legendre	0.2756872738004297	9e-15
Double Exp	0.2756872738004369	9e-16

which is remarkably close to $5/179*\pi^2$:

0.2756872737734458

Best regards,

Gerson.

Re: Impressive TI NSpire CX CAS Multiple Integration

Message #43 Posted by [Valentin Albillo](#) on 23 Dec 2011, 4:13 a.m.,
in response to message #42 by Gerson W. Barbosa

Quote:

We can also try [this online integrator](#) and obtain

Numerical Integration of $-(\log(\cos(x)))/x$ in the interval $[0,1]$

Method	Value	Estimated Error
Romberg	0.2756872738004371	2e-16
Gauss-Legendre	0.2756872738004297	9e-15
Double Exp	0.2756872738004369	9e-16

Hehe, no need for a lousy online integrator, my very own integrator produces:

Integral $[-\log(\cos(x))/x]$ in $[0,1]$ = 0.275687273800437163889752061421593272955579883796079969049093394140279534475580649396385056763317646349196

which, as I stated, is remarkably close to:

$$5\pi^{2/179} = \underline{0.27568727377}3445771475823770946261204897030709699463425318808641793856000626234783324071882785434419883352$$

so both values aren't identical though in this case I don't think their similarity is coincidental. Thanks for your interest and

Best regards from V.

Re: Impressive TI NSpire CX CAS Multiple Integration

*Message #44 Posted by **Thomas Klemm** on 23 Dec 2011, 5:24 a.m.,
in response to message #42 by Gerson W. Barbosa*

Using the Taylor expansion of $\tan(x)$ I can confirm this result. With *Free42* we have something like 20 internal figures; I forgot the exact number. When I take the first 100 terms into account I get the following value for $n = 2$:

0.2756872738004371638897521

Cheers
Thomas

[[Return to Index](#) | [Top of Index](#)]



[Go back to the main exhibit hall](#)