[VA] SRC#001 - Spiky Integral

07-10-2018, 11:10 PM **Post: #1**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

[VA] SRC#001 - Spiky Integral

.

Hi all, welcome to my ***SRC#001 - Spiky Integral***:

Here I'll deal with a real-world math problem not of my own making but which did appear at a certain math competition. The problem introduces this *"spiky"* integral:

$$I(N) = \int_0^{2\pi} cos(x)\, cos(2x)\, cos(3x) \ldots cos(Nx)\, dx$$

and asks ***for which values of N in the interval [1,10] does I(N) have a non-zero value***.

It then specifies that the result must include proof of correctness but in my extensive experience with problems of this sort I've found that it's best to *first obtain the correct result* using whatever means (praying included, bribing not excluded), as this will usually provide a most helpful "hint" to afterwards try and get proof of its validity.

With this strategy in mind (and if praying and bribing have already proved ineffective), we can use our beloved **HP calcs** (yes, **HP calcs**, not *Excel*, not *Mathematica*, not *Wolfram Alpha*, not *Maple*, not *Matlab*, not *laptops*, you get the drift) to get some numerical evidence first, then use it to make an educated conjecture on what the correct result might probably be.

As for numerical evidence, this is how I'd obtain it using an **HP-15C** and a very simple, straightforward little program I wrote which goes as follows:

The program segment which computes *Cos(x)*Cos(2x)*Cos(3x)*...*Cos(Nx)*, where **N** is assumed to be in **R0** and **x** in stack register **X** (the display), is placed under ***LBL B*** and the main program which computes the integral **I(N)** for **N** from 1 to 10 is placed after it under ***LBL A***:

```
001  LBL B       015  LBL A
002  STO 1       016  1.01
003  RCL 0       020  STO 0
004  INT         021  LBL 0
005  STO I       022  0
006  1           023  PI
007  LBL 1       024  2
008  RCL I       025  x
009  RCLx 1      026  INTEG B
010  COS         027  RCL 0
011  x           028  R/S
012  DSE I       029  X<>Y
013  GTO 1       030  R/S
014  RTN         031  ISG 0
                 032  GTO 0
```

To run it, simply press:

```
[USER] [RAD] [FIX 3]

[A]   ->  1.010    [R/S] -1.083 -04
```

```
[R/S] ->  2.010    [R/S] -5.974 -05
[R/S] ->  3.010    [R/S]  1.571
[R/S] ->  4.010    [R/S]  0.785
[R/S] ->  5.010    [R/S]  3.778 -08
[R/S] ->  6.010    [R/S]  2.733 -09
[R/S] ->  7.010    [R/S]  0.393
[R/S] ->  8.010    [R/S]  0.344
[R/S] ->  9.010    [R/S] -3.600 -04
[R/S] -> 10.010    [R/S]  1.664 -09
```

Notes:

*- It's best to run this on much faster, modern **HP-15C**-based hardware or a fast software emulator as otherwise it will take several hours for a physical vintage **HP-15C** to get the results above, mostly because the function being integrated gets more an more "spiky" as **N** grows and so it takes more and more time to integrate it accurately.*

*- FIX 3 is recommended for speed as FIX 4 or higher takes significantly longer and the extra accuracy isn't needed here. On the other hand, FIX 2 can't be used in the **HP-15C** or the **HP-41C/Advantage** versions because for **N**=4 we get:*

I(4) = 0.86 +- 0.03

*which is **wrong**, the uncertainty is lying. This is similar to what Mr. Kahan said about one of his sample integrals in some issue of the **HP Journal**: the error is too small to notice but too big to ignore. The solution is of course to use FIX 3 as above or SCI 2, which also gives the correct result, namely:*

I(4) = 7.85E-01 +- 1.48E-03

With this evidence in mind and regarding anything smaller than 1E-3 as **0** (as we are computing the integrals in FIX 3) we can see that **I(N) is non-zero for N=3, 4, 7 and 8**.

Using this numerical evidence I have my own conjecture on what the result will be for **general N**, which I'll post in a few days together with my verbatim conversion of this **HP-15C** program for the **HP-41C/Advantage ROM** (which runs very fast in modern hardware/emulators) as well as a *completely different 6-line enhanced version* for the **HP-71B/Math ROM** which profitably uses a technique discussed in my latest ***S&SMC#23*** and further includes a couple of *very nice extras* not present in the other versions.

First though, **I'd like to see what your own conjectures are** (computing for N>10 might help to check them out) and of course **your very own programs** for your preferred hardware (*RPL* comes to mind, *Prime's PPL*, even *SHARP* or *Casio* models ...) or, if you're not up to the task (you know who you are), at least your very own personal comments or ideas ! (Googling for the solution is totally *laaaaaame,* boo !).

Regards.
V.
.

Find All My HP-related Materials here: Valentin Albillo's HP Collection

PM WWW FIND EDIT QUOTE REPORT

07-11-2018, 12:10 PM **Post: #2**

pier4r
Senior Member

Posts: 2,067
Joined: Nov 2014

RE: [VA] SRC#001 - Spiky Integral

Thanks for the post. What does SRC means in this case? Source problem?

Wikis are great, Contribute :)

EMAIL PM FIND QUOTE REPORT

07-12-2018, 05:23 AM **Post: #3**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-10-2018 11:10 PM)
> Using this numerical evidence I have my own conjecture on <u>what the result will be for **general N**</u>, which I'll post in a few days...

Hello, Valentin,

Here is my conjecture for the exact result when N = 39:

$$\frac{756388295}{68719476736}\pi$$

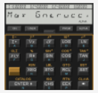
No programs. I’ve evaluated the integrals for N up to 12 on my CASIO fx-991 LA X, which does it fast enough and gives exact results for N = 1, 2, 3, 4, 7 and 8. No googling, except for a OEIS sequence.

Best regards,

Gerson.

EMAIL PM FIND QUOTE REPORT

07-12-2018, 07:10 AM **Post: #4**



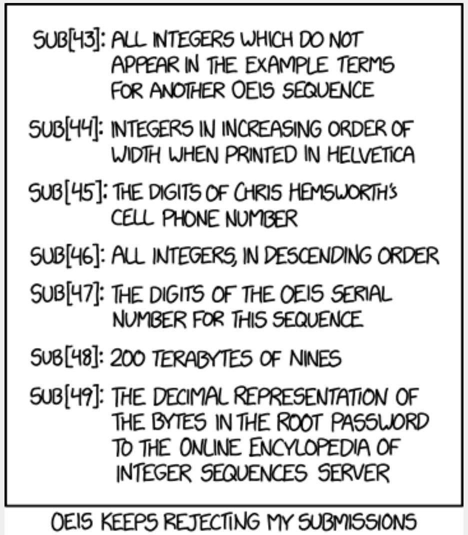
Massimo Gnerucci
Senior Member

Posts: 2,152
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Gerson W. Barbosa Wrote:** (07-12-2018 05:23 AM)
> No googling, except for a OEIS sequence.

I bet is one of these!



Greetings,
    Massimo

-+×÷ ↔ *left is right and right is wrong*

PM WWW FIND QUOTE REPORT

07-12-2018, 11:32 AM **Post: #5**

Pjwum
Member

Posts: 51
Joined: Jan 2018

RE: [VA] SRC#001 - Spiky Integral

With HP Prime

Code:

```
EXPORT SPIKES()
BEGIN
LOCAL N;
FOR N FROM 1 TO 20 DO
PRINT(int(product(COS(M*X),M,1,N,1),X,0,2*π));
END;
END;
```

we can go beyond 10

0
0
1/2 pi
1/4 pi
0
0
1/8 pi
7/64 pi
0
0
35/512 pi
31/512 pi
0
0
361/8192 pi
657/16384 pi
0
0
2055/65536 pi
1909/65536 pi

suggesting that I(N) will be non-zero if and only if
N = 4k+3 and N = 4k+4, for k = 0, 1, 2..

PM FIND QUOTE REPORT

07-12-2018, 09:10 PM **Post: #6**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Pjwum Wrote:** (07-12-2018 11:32 AM)
> With HP Prime
>
> we can go beyond 10
>
> 0
> 0
> 1/2 pi

1/4 pi
0
0
1/8 pi
7/64 pi
0
0
35/512 pi
31/512 pi
0
0
361/8192 pi
657/16384 pi
0
0
2055/65536 pi
1909/65536 pi

Sure we can:

```
%%HP: T(3)A(R)F(.);
\<< DUPDUP 'X' 1 ROT OVER SWAP
  FOR i OVER i ^ DUP INV + *
  NEXT NIP EXPAND FXND DROP \->STR "*X^" ROT DUPDUP * + 2 / DUP 2 + \->STR "+" + UNROT \->STR + "+" + "X^" ROT + PICK3 SWAP POS PICK3 ROT POS 1 - SUB DUP SIZE 1 -
  IF NOT
  THEN DROP2 0
  ELSE DUP SIZE OVER "+" POS 1 + SWAP SUB OBJ\-> 2 ROT 1 - ^ / \pi *
  END
\>>

'VASRC1' STO

40

%%HP: T(3)A(R)F(.);
\<< { } 1 ROT
  FOR i i VASRC1 +
  NEXT
\>>

EVAL

-->
```

{ 0 0 '1/2*π' '1/4*π' 0 0 '1/8*π' '7/64*π' 0 0 '35/512*π' '31/512*π' 0 0 '361/8192*π' '657/16384*π' 0 0 '2055/65536*π' '1909/65536*π' 0 0 '24955/1048576*π' '46923/2097152*π' 0 0 '316301/16777216*π' '299973/16777216*π' 0 0 '4136805/268435456*π' '15796439/1073741824*π' 0 0 '13853361/1073741824*π' '26585247/2147483648*π' 0 0 '756388295/68719476736*π' '182188585/17179869184*π' }

Not the best method, I fear. Expand Product{ k=1..n, x^N + 1/x^N } and take the coefficient of the power of x corresponding to the Nth triangular number in the numerator (if there is no correspondence, then the result will be zero). That's your numerator. Your denominator is 2^(N - 1). Multiply the resulting fraction by π. The cases where the results are zero should be handled more cleverly, as you have suggested, but this is only a test. The RPL program might not be fast enough on the real HP 50g as N get larger. For N = 12, it takes 16.75 seconds; for N = 20 it takes 118.41 seconds. The evaluation of the integrals would take much, much longer, I guess.

Gerson.

EMAIL PM FIND QUOTE REPORT

07-12-2018, 09:29 PM (This post was last modified: 07-12-2018 11:19 PM by Gerson W. Barbosa.) **Post: #7**

Gerson W. Barbosa
Senior Member

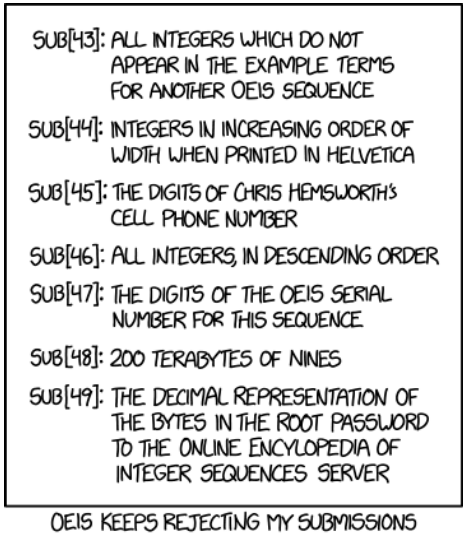
Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

Massimo Gnerucci Wrote: ⇒ (07-12-2018 07:10 AM)

> **Gerson W. Barbosa Wrote:** ⇒ (07-12-2018 05:23 AM)
>
> No googling, except for a OEIS sequence.

I bet is one of these!



I thought this one was not at OEIS yet, but it is:

2, 10, 12, 17, 18, 19, 200, 201, 202...

BTW, at first I searched for 140, 248, but these didn't match anything in the table. 70, 124 was successful.

Edited to fix a typo

Gerson.

EMAIL PM FIND QUOTE REPORT

07-12-2018, 11:43 PM **Post: #8**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, all:

Thanks for your interest in my ***SRC#001***, some excellent contributions so far, much appreciated. Next Sunday night (GMT+1) I'll post my **HP-41C** and **HP-71B** solutions plus extras, but meanwhile I'll comment on some of your recent posts, read on:

> **pier4r Wrote:**
>
> Thanks for the post. What does SRC means in this case? Source problem?

Nope, **SRC** = **S**emi-**R**egular **C**olumn.

Gerson W. Barbosa Wrote:

Here is my conjecture for the exact result when **N = 39**: 756388295 / 68719476736*Pi

Fully correct. You might want to check the exact result for **N=71**, namely:

```
I(71) = 335205518724079925 / 73786976294838206464 * Pi
```

which I got by actually *computing* the integral, then *identifying* the resulting constant, using dedicated programs I wrote myself (no *OEIS* involved). It's the hard way but hey, it's fun !

Pjwum Wrote:

we can go beyond 10 [...] **suggesting that I(N) will be non-zero if and only if N = 4k+3 and N = 4k+4**, for k = 0, 1, 2..

Fully correct as well, we have a winner ! Well done, congratulations. 8-)

Gerson (again) Wrote:

Sure we can: [...]

*{ 0 0 '1/2*π' '1/4*π' 0 0 '1/8*π' '7/64*π' 0 0 '35/512*π' '31/512*π' 0 0 '361/8192*π' '657/16384*π' 0 0 '2055/65536*π' '1909/65536*π' 0 0 '24955/1048576*π' '46923/2097152*π' 0 0 '316301/16777216*π' '299973/16777216*π' 0 0 '4136805/268435456*π' '15796439/1073741824*π' 0 0 '13853361/1073741824*π' '26585247/2147483648*π' 0 0 '756388295/68719476736*π' '182188585/17179869184*π' }*

Not the best method, I fear. [...] **The evaluation of the integrals would take much, much longer**, I guess.

It would take much, much longer or not, depending on how you go about computing them. Your results are correct, congratulations, but without explaining *why* would you compute the numerators and denominators that way it all seems a "magic trick" unrelated to the problem at hand as the relation to the integral seems shrouded in mystery ... 8-D

Regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

07-13-2018, 04:54 AM **Post: #9**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

Valentin Albillo Wrote: (07-12-2018 11:43 PM)

Pjwum Wrote:

we can go beyond 10 [...] **suggesting that I(N) will be non-zero if and only if N = 4k+3 and N = 4k+4**, for k = 0, 1, 2..

Fully correct as well, we have a winner ! Well done, congratulations. 8-)

Or, alternatively, if N(N + 1)/2 mod 2 = 0.

EMAIL PM FIND QUOTE REPORT

07-13-2018, 07:52 PM (This post was last modified: 07-14-2018 02:47 AM by ijabbott.) **Post: #10**

ijabbott
Senior Member

Posts: 921
Joined: Jul 2015

RE: [VA] SRC#001 - Spiky Integral

You don't really need to do any integration to solve this problem. You just need to use the trig identities to convert the product of cosines into a sum of cosines. When there is a non-zero constant term in the sum, the integral from 0 to 2π will be non-zero. The integrals of all the $\cos(kx)$ terms from 0 to 2π will be zero.

The general form for converting the product of cosines to a sum of cosines is:

$$\cos(s)\cos(t) = \frac{1}{2}\cos(s-t) + \frac{1}{2}\cos(s+t)$$

So:

$$\cos(1x) = \cos(1x)$$

$$\cos(1x)\cos(2x) = \frac{1}{2}\cos(1x) + \frac{1}{2}\cos(3x)$$

$$\begin{aligned}
\cos(1x)\cos(2x)\cos(3x) &= \frac{1}{2}\cos(1x)\cos(3x) + \frac{1}{2}\cos(3x)\cos(3x) \\
&= \frac{1}{2}\left(\frac{1}{2}\cos(2x) + \frac{1}{2}\cos(4x)\right) + \frac{1}{2}\left(\frac{1}{2}\cos(0x) + \frac{1}{2}\cos(6x)\right) \\
&= \frac{1}{4}\cos(2x) + \frac{1}{4}\cos(4x) + \frac{1}{4}\cos(0x) + \frac{1}{4}\cos(6x) \\
&= \frac{1}{4}\cos(0x) + \frac{1}{4}\cos(2x) + \frac{1}{4}\cos(4x) + \frac{1}{4}\cos(6x)
\end{aligned}$$

$$\begin{aligned}
\cos(1x)\cos(2x)\cos(3x)\cos(4x) &= \frac{1}{4}\cos(0x)\cos(4x) + \frac{1}{4}\cos(2x)\cos(4x) + \frac{1}{4}\cos(4x)\cos(4x) + \frac{1}{4}\cos(6x)\cos(4x) \\
&= \frac{1}{4}\left(\frac{1}{2}\cos(4x) + \frac{1}{2}\cos(4x)\right) + \frac{1}{4}\left(\frac{1}{2}\cos(2x) + \frac{1}{2}\cos(6x)\right) + \frac{1}{4}\left(\frac{1}{2}\cos(0x) + \frac{1}{2}\cos(8x)\right) + \frac{1}{4}\left(\frac{1}{2}\cos(2x) + \frac{1}{2}\cos(10x)\right) \\
&= \frac{1}{8}\cos(4x) + \frac{1}{8}\cos(4x) + \frac{1}{8}\cos(2x) + \frac{1}{8}\cos(6x) + \frac{1}{8}\cos(0x) + \frac{1}{8}\cos(8x) + \frac{1}{8}\cos(2x) + \frac{1}{8}\cos(10x) \\
&= \frac{1}{8}\cos(0x) + \frac{1}{4}\cos(2x) + \frac{1}{4}\cos(4x) + \frac{1}{8}\cos(6x) + \frac{1}{8}\cos(8x) + \frac{1}{8}\cos(10x)
\end{aligned}$$

$$\begin{aligned}
\cos(1x)\cos(2x)\cos(3x)\cos(4x)\cos(5x) &= \frac{1}{8}\cos(0x)\cos(5x) + \frac{1}{4}\cos(2x)\cos(5x) + \frac{1}{4}\cos(4x)\cos(5x) + \frac{1}{8}\cos(6x)\cos(5x) + \frac{1}{8}\cos(8x)\cos(5x) + \frac{1}{8}\cos(10x)\cos(5x) \\
&= \frac{1}{8}\left(\frac{1}{2}\cos(5x) + \frac{1}{2}\cos(5x)\right) + \frac{1}{4}\left(\frac{1}{2}\cos(3x) + \frac{1}{2}\cos(7x)\right) + \frac{1}{4}\left(\frac{1}{2}\cos(1x) + \frac{1}{2}\cos(9x)\right) + \frac{1}{8}\left(\frac{1}{2}\cos(1x) + \frac{1}{2}\cos(11x)\right) + \frac{1}{8}\left(\frac{1}{2}\cos(3x) + \frac{1}{2}\cos(13x)\right) + \frac{1}{8}\left(\frac{1}{2}\cos(5x) + \frac{1}{2}\cos(15x)\right) \\
&= \frac{1}{16}\cos(5x) + \frac{1}{16}\cos(5x) + \frac{1}{8}\cos(3x) + \frac{1}{8}\cos(7x) + \frac{1}{8}\cos(1x) + \frac{1}{8}\cos(9x) + \frac{1}{16}\cos(1x) + \frac{1}{16}\cos(11x) + \frac{1}{16}\cos(3x) + \frac{1}{16}\cos(13x) + \frac{1}{16}\cos(5x) + \frac{1}{16}\cos(15x) \\
&= \frac{3}{16}\cos(1x) + \frac{3}{16}\cos(3x) + \frac{3}{16}\cos(5x) + \frac{1}{8}\cos(7x) + \frac{1}{8}\cos(9x) + \frac{1}{16}\cos(11x) + \frac{1}{16}\cos(13x) + \frac{1}{16}\cos(15x)
\end{aligned}$$

It's a bit tedious to continue, but you can see a constant term $\frac{1}{4}\cos(0x) = \frac{1}{4}$ in the expansion of $\prod_{i=1}^{3}\cos(ix)$, a constant term of $\frac{1}{8}\cos(0x) = \frac{1}{8}$ in the expansion of $\prod_{i=1}^{4}\cos(ix)$, and the constant term goes away in the expansion of $\prod_{i=1}^{5}\cos(ix)$. The constant terms are the coefficients of $\cos(0x)$ in those expansions whose terms consist of cosines of *even* multiples of x.

Your challenge, should you choose to accept it, is to come up with a general formula for each term (or a summation formula for all the terms).

Of course, once you've found the coefficient for the constant $\cos(0x)$ term, calculating the integral from 0 to 2π is a piece of cake! (Just multiply the coefficient by 2π.)

— Ian Abbott

EMAIL PM FIND QUOTE REPORT

07-14-2018, 02:30 AM **Post: #11**

rprosperi
Senior Member

Posts: 4,439
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

I'm truly impressed at your diligence writing all those equations before essentially saying "... and so on". I'd probably be at least as impressed at the math, if I followed it, but I just wanted to give you kudos for all the work to explain your analysis.

--Bob Prosperi

EMAIL PM FIND QUOTE REPORT

07-14-2018, 02:39 AM **Post: #12**

ijabbott
Senior Member

Posts: 921
Joined: Jul 2015

RE: [VA] SRC#001 - Spiky Integral

> **rprosperi Wrote:** (07-14-2018 02:30 AM)
> I'm truly impressed at your diligence writing all those equations before essentially saying "... and so on". I'd probably be at least as impressed at the math, if I followed it, but I just wanted to give you kudos for all the work to explain your analysis.

Thanks! I'm slowly getting the hang of this MathJax lark!

— Ian Abbott

EMAIL PM FIND QUOTE REPORT

07-14-2018, 03:08 PM (This post was last modified: 07-14-2018 03:09 PM by ijabbott.) **Post: #13**

ijabbott
Senior Member

Posts: 921
Joined: Jul 2015

RE: [VA] SRC#001 - Spiky Integral

Converting the product of cosines $\prod_{i=1}^{n}\cos(ix)$ into a sum of cosines results in the angle multipliers being in the sum of cosines being all odd (when $(n \bmod 4) \in \{1,2\}$) or all even (when $(n \bmod 4) \in \{0,3\}$).

Identity:

$$\cos(s)\cos(t) = \frac{1}{2}\cos(s-t) + \frac{1}{2}\cos(s+t)$$

Let $s = \mathrm{a}x, t = \mathrm{b}x$. Then:

$$\cos(\mathrm{a}x)\cos(\mathrm{b}x) = \frac{1}{2}\cos((\mathrm{a}-\mathrm{b})x) + \frac{1}{2}\cos((\mathrm{a}+\mathrm{b})x)$$

When a and b are both odd or both even, then $(\mathrm{a}-\mathrm{b})$ and $(\mathrm{a}+\mathrm{b})$ are both even, otherwise $(\mathrm{a}-\mathrm{b})$ and $(\mathrm{a}+\mathrm{b})$ are both odd. This results in the factors k of x in the $\cos(\mathrm{k}x)$ terms of the summation switching between all odd and all even after every two $\cos(ix)$ factors are appended to the product of cosines.

As discussed in my earlier post, the summations with all odd k, $\cos(\mathrm{k}x)$ terms all integrate to 0 over the interval $[0, 2\pi]$, but the summations with all even k, $\cos(\mathrm{k}x)$ terms all include a constant term $\mathrm{q}\cos(0x)$ for some positive rational factor q which integrates to $2\mathrm{q}\pi$ over the interval $[0, 2\pi]$.

— Ian Abbott

EMAIL PM FIND QUOTE REPORT

07-14-2018, 09:38 PM **Post: #14**

Thomas Klemm
Senior Member

Posts: 1,447
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **ijabbott Wrote:** (07-13-2018 07:52 PM)
> It's a bit tedious to continue

We can use:

$$cos(nx) = \frac{e^{inx} + e^{-inx}}{2}$$

Thus:

$$\begin{aligned} cos(mx)cos(nx) &= (\frac{e^{imx} + e^{-imx}}{2})(\frac{e^{inx} + e^{-inx}}{2}) \\ &= \frac{e^{i(m+n)x} + e^{-i(m+n)x}}{4} + \frac{e^{i(m-n)x} + e^{-i(m-n)x}}{4} \\ &= \frac{cos(m+n)}{2} + \frac{cos(m-n)}{2} \end{aligned}$$

Let's forget the factor $\frac{1}{2}$ for a moment and define:

$$a_k = a^k + a^{-k}$$

For the same reason as above we have:

$$\begin{aligned} a_m a_n &= (a^m + a^{-m})(a^n + a^{-n}) \\ &= a^{m+n} + a^{-(m+n)} + a^{m-n} + a^{-(m-n)} \\ &= a_{m+n} + a_{m-n} \end{aligned}$$

But of course $a_k = a_{-k}$.

This allows us to rewrite it as:

$$a_m a_n = a_{m+n} + a_{|m-n|}$$

We want to calculate the product:

$$p_N = \Pi_{k=1}^N a_k$$

Let's assume we already have p_{N-1} represented as a sum of a_k with coefficients b_k:

$$p_{N-1} = \Sigma_{k=1}^{M} b_k a_k$$

Then $p_N = a_N p_{N-1}$ and thus:

$$\begin{aligned} p_N &= a_N \Sigma_{k=1}^{M} b_k a_k \\ &= \Sigma_{k=1}^{M} b_k a_N a_k \\ &= \Sigma_{k=1}^{M} b_k (a_{N+k} + a_{N-k}) \end{aligned}$$

This *Python* program allows us to calculate the coefficients b_k:

Code:

```
def spiky(N):
    b = [0] * N
    b[0] = [0]
    b[1] = [0, 1]
    for k in range(2, N):
        M = len(b[k-1])
        b[k] = [0] * (M + k)
        for i in range(M):
            b[k][k + i] += b[k - 1][i]
            b[k][abs(k - i)] += b[k - 1][i]
    return b
```

Here's the result for the first couple of values:

Code:

```
>>> for b in spiky(11):
...   print b
...
[0]
[0, 1]
[0, 1, 0, 1]
[1, 0, 1, 0, 1, 0, 1]
[1, 0, 2, 0, 2, 0, 1, 0, 1, 0, 1]
[0, 3, 0, 3, 0, 3, 0, 2, 0, 2, 0, 1, 0, 1, 0, 1]
[0, 5, 0, 5, 0, 4, 0, 4, 0, 4, 0, 3, 0, 2, 0, 2, 0, 1, 0, 1, 0, 1]
[4, 0, 8, 0, 8, 0, 7, 0, 7, 0, 6, 0, 5, 0, 5, 0, 4, 0, 3, 0, 2, 0, 2, 0, 1, 0, 1, 0, 1]
```

But we're only interested in the coefficient with the index **0**:

Code:

```
>>> for b in spiky(40):
...   print b[0]
...
0
0
0
1
1
0
0
4
```

Now it's time to deal with the factor $\frac{1}{2}$ that we neglected.
But that's trivial. We just have to add it to each factor a_k.
This leads to: $\frac{1}{2^{N-1}}$

And since we integrate the constant over 2π we loose another 2.

These are the examples for $N = 39$ and $N = 71$:

Code:
```
>>> b = spiky(100)
>>> print b[39][0], '/', 2**37
1512776590 / 137438953472
>>> print b[71][0], '/', 2**69
2681644149792639400 / 590295810358705651712
```
However the ratios haven't been reduced.

> **Gerson W. Barbosa Wrote:** (07-12-2018 09:10 PM)
>
> Expand Product{ k=1..n, x^N + 1/x^N } and take the coefficient of the power of x corresponding to the Nth triangular number in the numerator (if there is no correspondence, then the result will be zero). That's your numerator. Your denominator is 2^(N - 1). Multiply the resulting fraction by π.

Not sure if I understood that correctly but it might explain the expression:

$$x^N + \frac{1}{x^N}$$

Thanks both for the challenge and the contributions.

Cheers
Thomas

PM FIND QUOTE REPORT

07-14-2018, 11:54 PM **Post: #15**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi all,

First of all, thanks for your interest and outstanding contributions, I've really enjoyed them all, much appreciated. As promised, these are my versions for other HP models (**HP-71B** and **HP-41C**) plus relevant comments:

As already mentioned, an educated conjecture might be that ***I(N)*** *is non-zero if the reminder of dividing N by 4 (that is,* ***N mod 4****) is either* ***0*** *or* ***3*** *(i.e.: N=3, 4, 7, 8, 11, 12, 15, 16,...)* which actually is the correct answer.

1) My version for the HP-71B is the following 220-byte 6-liner:

```
1 DESTROY ALL @ DIM S$[200] @ T=1/10^8 @ S$="COS(IVAR)" @ FOR N=1 TO 16
2 IF N>1 THEN S$=S$&"*COS("&STR$(N)&"*IVAR)"
3 S=VAL("INTEGRAL(0,2*PI,"&STR$(T)&","&S$&")") @ IF ABS(S)<T THEN 6
4 CALL IDENTIFY(S/PI,T$) @ T$=T$&"*Pi"
5 DISP USING "3D,':',2DZ.10D,' = ',20A";N,S,T$
6 NEXT N
```

which makes good use of a new technique I discovered long ago and then explained in my recent ***S&SMC#23***, which in this case consist of incrementally creating an arbitrarily large string representing the function to be integrated, and then prepend to it the *INTEGRAL* keyword, the limits of integration and the tolerance before passing it to *VAL* to perform the actual computation. For instance, for N=16 this is the generated string passed to *VAL*:

```
INTEGRAL(0,2*PI,.00000001,COS(IVAR)*COS(2*IVAR)*COS(3*IVAR)*COS(4*IVAR)*COS(5*IVAR)*COS(6*IVAR)*COS(7*IVAR)*COS(8*IVAR)*COS(9*IVAR)
                          *COS(10*IVAR)*COS(11*IVAR)*COS(12*IVAR)*COS(13*IVAR)*COS(14*IVAR)*COS(15*IVAR)*COS(16*IVAR))
```

This means that, for each N, the string is parsed just the one time when passed to *VAL* but the function itself doesn't need any additional parsing no matter how many times the *INTEGRAL* keyword evaluates it, nor are there any loops whatsoever while evaluating it. This all results in extra simplicity (no loops) and much faster execution (neither loops nor calls to a multi-line user-defined function or extra parsing).

As a nice extra, *line 4* (which is optional, can be omitted) does *CALL* my *IDENTIFY* subprogram (see my ***"Boldly Going - Identifying Constants"*** article) to *identify* every numeric result after integration and display the *symbolic* value. Finally, taking advantage of ***J-F Garnier****'s* ***Emu71***'s much greater speed, the results are provided not just up to N=10 but up to **N=16**, and results below the tolerance are considered to be **0** and aren't output. Let's see:

```
>RUN
        3: 1.5707963268 = 1/2*Pi
        4: 0.7853981634 = 1/4*Pi
        7: 0.3926990817 = 1/8*Pi
        8: 0.3436116965 = 7/64*Pi
       11: 0.2147573103 = 35/512*Pi
       12: 0.1902136177 = 31/512*Pi
       15: 0.1384417661 = 361/8192*Pi
       16: 0.1259781722 = 657/16384*Pi
```

Without the identification calls this runs in 19 sec. in my *POPS* system.

2) My version for the HP-41C/Advantage ROM is a *verbatim* port of the one I gave for the *HP-15C*, namely:

```
01   LBL "FX"    01   LBL "CINT"
02   STO 01      02   1.01
03   RCL 00      03   STO 00
04   INT         04   "FX"
05   STO 02      05   LBL 00
06    1          06    0
07   LBL 01      07   PI
08   RCL 02      08   ST+ X
09   RCL 01      09   INTEG
10    x          10   RCL 00
11   COS         11   STOP
12    x          12   X<>Y
13   DSE 02      13   STOP
14   GTO 01      14   ISG 00
15   END         15   GTO 00
                 16   END
```

To run it (the results are exactly the same as the *HP-15C*'s but obtained *much* faster):

RAD, **FIX 3**
XEQ "CINT" -> 1.010 [R/S] -> -1.083 -04
[R/S] -> 2.010 [R/S] -> -5.974 -05
[R/S] -> 3.010 [R/S] -> **1.571**
[R/S] -> 4.010 [R/S] -> **0.785**
[R/S] -> 5.010 [R/S] -> 3.778 -08
[R/S] -> 6.010 [R/S] -> 2.733 -09
[R/S] -> 7.010 [R/S] -> **0.393**
[R/S] -> 8.010 [R/S] -> **0.344**
[R/S] -> 9.010 [R/S] -> -3.600 -04
[R/S] -> 10.010 [R/S] -> 1.664 -09

With *FIX 2*, for **N=4** we'd get *I(4) = 0.86 +- 0.03*, which is *10% wrong*. Details:

N=4, f(x) = cos(x)*cos(2*x)*cos(3*x)*cos(4*x)

FIX 3: **0.78539818**71 +- 0.003141592705, correct
FIX 2: **0.8**645215735 +- 0.03141592659 , 10% wrong

SCI 2: **0.78539818**71 +- 0.001484303379 . correct

The exact value is *Pi/4 = **0.785398163**3397* so we get almost 8 correct digits with *FIX 3* or *SCI 2* but hardly *one* with *FIX 2*.

3) Extra comments: A little theory now.

After some algebraic massaging, any ***product*** of *cosines* of *real* arguments can be converted into a ***sum*** of *exponentials* of *complex* arguments using the following identity:

Cos(x) = (e^(i*x) + e^(-i*x))/2

This can be checked with this little **HP-71B** routine:

```
10 DESTROY ALL @ COMPLEX A,B @ INPUT X
15 !
20 DEF FNC1(X)=COS(X)
25 DEF FNC2(X)=COS(X)*COS(2*X)
30 DEF FNC3(X)=COS(X)*COS(2*X)*COS(3*X)
35 DEF FNC4(X)=COS(X)*COS(2*X)*COS(3*X)*COS(4*X)
40 !
45 DEF FNE(X)=REPT(EXP((0,X))+EXP((0,-X)))
50 !
55 DEF FNE1(X)=1/2*FNE(X)
60 DEF FNE2(X)=1/4*(FNE(X)+FNE(3*X))
65 DEF FNE3(X)=1/8*(FNE(0)+FNE(2*X)+FNE(4*X)+FNE(6*X))
70 DEF FNE4(X)=1/16*(FNE(0)+2*FNE(2*X)+2*FNE(4*X)+FNE(6*X)+FNE(8*X)+FNE(10*X))
75 !
80 DISP 1;FNC1(X),FNE1(X) @ DISP 2;FNC2(X),FNE2(X) @ DISP 3;FNC3(X),FNE3(X) @ DISP 4;FNC4(X),FNE4(X)
```

where the *FNC1*, *FNC2*, *FNC3*, *FNC4* are user-defined functions (*UDFs*) implementing the **products** of 1, 2, 3 and 4 *cosines*, while the *FNE1*, *FNE2*, *FNE3*, *FNE4* are the equivalent **sums** of 1, 2, 4 and 8 complex *exponentials* (not all necessarily distinct). Let's run it:

```
>RUN

? .2018

      1 .979707385531 .97970738553
      2 .900990956269 .90099095627
      3 .740861912405 .740861912405
      4 .512323594236 .512323594236
```

As you may see, every *FNE1..4* gives the same result as the corresponding *FNC1..4* for an arbitrary argument (*.2018*).

What is to be gained by converting a *product* of *cosines* into a *sum* of *exponentials* ? Two main things:

- the *integral of a sum* of functions is the *sum of the integrals* of each function separately
- the integral of a simple *exponential* function is *another* simple *exponential* function so no need for numerical integration

In this way, the integral of the *product* of cosines becomes a *sum* of exponential functions and no numerical approximations are required. There's of course the drudgery of determining the components of the sum but that's another story ! XD

Again, thanks for your interest and regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

07-15-2018, 03:53 PM (This post was last modified: 07-15-2018 03:55 PM by Gerson W. Barbosa.) **Post: #16**

Posts: 1,361

Gerson W. Barbosa
Senior Member

Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** (07-14-2018 09:38 PM)
>
> These are the examples for $N = 39$ and $N = 71$:
>
> **Code:**
> ```
> >>> b = spiky(100)
> >>> print b[39][0], '/', 2**37
> 1512776590 / 137438953472
> >>> print b[71][0], '/', 2**69
> 2681644149792639400 / 590295810358705651712
> ```
> However the ratios haven't been reduced.

In exact mode, the HP 50g does it automatically.

> **Thomas Klemm Wrote:** (07-14-2018 09:38 PM)
>
> > **Gerson W. Barbosa Wrote:** (07-12-2018 09:10 PM)
> >
> > Expand Product{ k=1..n, x^N + 1/x^N } and take the coefficient of the power of x corresponding to the Nth triangular number in the numerator (if there is no correspondence, then the result will be zero). That's your numerator. Your denominator is 2^(N - 1). Multiply the resulting fraction by π.
>
> Not sure if I understood that correctly but it might explain the expression:
>
> $$x^N + \frac{1}{x^N}$$

I haven't trodden any of the hard (and beautiful) paths you all have done. Instead, I took an unallowed (according to Valentin's rules) and dull shortcut. It was not difficult to recognize pi/2 and pi/4 in his his three-digit results for n = 3 and n = 4. Then I evaluated the integrals for n up to 12 on my CASIO fx-991 LA X, which doesn't take too long to return exact results in terms of pi for small values of n (1/8*pi after 3m 30s, for n = 7; 0.3436116965 after 4m 31s, for n = 8). The pattern soon became apparent: a fraction of pi, the denominator being a power of 2. For denominators = 2^(n - 1), the first numerators, starting with n = 1, are 0, 0, 2, 2, 0, 0, 8, 14, 0, 0, 70, 124, 0..., that is, OEIS sequence A063865.

Quoting from the formula section:

"a(n) = constant term in expansion of Product_{ k = 1..n } (x^k + 1/x^k). - N. J. A. Sloane, Jul 07 2008"

Thus, since

(X + 1/X)(X^2 + 1/X^2)(X^3 + 1/X^3)(X^4 + 1/X^4)(X^5 + 1/X^5)(X^6 + 1/X^6)(X^7 + 1/X^7)(X^8 + 1/X^8)

=

X^36 + 1/X^36 + X^34 + 1/X^34 + X^32 + 1/X^32 + 2 X^30 + 2/X^30 + 2 X^28 + 2/X^28 + 3 X^26 + 3/X^26 + 4 X^24 + 4/X^24 + 5 X^22 + 5/X^22 + 6 X^20 + 6/X^20 + 7 X^18 + 7/X^18 + 8 X^16 + 8/X^16 + 9 X^14 + 9/X^14 + 10 X^12 + 10/X^12 + 11 X^10 + 11/X^10 + 12 X^8 + 12/X^8 + 13 X^6 + 13/X^6 + 13 X^4 + 13/X^4 + 13 X^2 + 13/X^2 + **14**

a(8) = 14

So, the numerator can be found by means of a polynomial expansion. However, when expanding this polynomial on the HP 50g, using the EXPAND command, I got

'(X^72+X^70+X^68+2*X^66+2*X^64+3*X^62+4*X^60+5*X^58+6*X^56+7*X^54+8*X^52+9*X^50+
10*X^48+11*X^46+12*X^44+13*X^42+13*X^40+13*X^38+**14***X^36+13*X^34+13*X^32+13*X^30+12*X^28+11*X^26+10*X^24+9*X^22+8*X^20+7*X^18+6*X^1
6+5*X^14+4*X^12+3*X^10+2*X^8+2*X^6+X^4+X^2+1)/X^36',

which is equivalent to the previous polynomial, except that the constant term in the numerator is 1. But notice 14 is the coefficient in the numerator that corresponds to the power of the denominator (...+14*X^36+.../X^36). Also, 36 = 8*(8 + 1)/2, that is, the 8th triangular number. This has worked for some othe values of n I tried, so I assumed it is a valid property.

The degree of the numerator polynomial is n*(n + 1), which means its size grows quadratically as n increases, which is both memory and time-consuming. Hopefully your method of generating the coefficients is faster (since I don't know python, I can't properly decode your algorithm).

I am disappointed there isn't a formula to directly compute the terms of the sequence. There is an asymptotic formula at OEIS, but it is not good enough for practical purposes. So I made some adjustments, which are by no means exact, but might give 5 or 6 correct significant digits for relatively low n, when computing the integral:

a(n) ~ sqrt(6/pi)*2^n*(1-6/(5*n)+21/(20*n^2)-1/(8*n^3)+3/n^4)/(n*sqrt(n))

or

a(n) ~ (sqrt(3/p) 2^(n - 5/2) (n (2 n (4 n (5 n - 6) + 21) - 5) + 120))/(5 n^(11/2))

The following HP-42S program computes the integral for n >= 1 and n up to about 20000 (on Free42) and returns exact results for n < 15.

Code:

```
00 { 90-Byte Prgm }

01▸LBL "INCS"
02 ENTER
03 ENTER
04 ENTER
05 5
06 ×
07 6
08 -
```

Gerson.

EMAIL PM FIND QUOTE REPORT

07-15-2018, 07:26 PM **Post: #17**

Thomas Klemm
Senior Member

Posts: 1,447
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

This *RPL* program calculates the coefficients:

Code:

```
«
  « → m
    « { 1 m } 0 CON OBJ→ DROP m →LIST
    »
  » → n ZEROS
  « { 0 1 } 2 n
    FOR k DUP SIZE → a s
      « k ZEROS EVAL a +
        a 1 k SUB 0 + REVLIST s 1 - ZEROS EVAL +
        ADD
        a k 1 + s SUB k 2 * ZEROS EVAL +
```

Example:

The value for **4** is:
{ 1 0 2 0 2 0 1 0 1 0 1 }

To get the value for **5** we create the following lists:
{ 0 0 0 0 0 1 0 2 0 2 0 1 0 1 0 1 }
{ 0 2 0 2 0 1 0 0 0 0 0 0 0 0 0 0 }
{ 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 }

And then we just ADD them up:
{ 0 3 0 3 0 3 0 2 0 2 0 1 0 1 0 1 }

The 2nd and the 3rd list is just the 1st list reversed and then again mirrored at the left border.
That's a consequence of $a_k = a^k + a^{-k}$ being symmetric, that is $a_k = a_{-k}$. We don't want negative indices.

Cheers
Thomas

PS: Is there a better way to create a list of m zeros?

PM FIND QUOTE REPORT

07-15-2018, 08:10 PM (This post was last modified: 07-15-2018 08:16 PM by Gerson W. Barbosa.) **Post: #18**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

The RPL program has been rewritten so as not to needlessly waste time expanding a polynomial when results are supposed to be zero.

```
%%HP: T(3)A(R)F(.);
\<< DUPDUP DUPDUP * + 2 / 2 MOD
  IF NOT
  THEN DUP 'X' 1 ROT OVER SWAP
    FOR i OVER i ^ DUP INV + *
    NEXT NIP EXPAND FXND DROP \->STR "*X^" ROT DUPDUP * + 2 / DUP 2 + \->STR "+" + UNROT \-
>STR + "+" + "X^" ROT + PICK3 SWAP POS PICK3 ROT POS 1 - SUB DUP SIZE OVER "+" POS 1 + SWAP SUB OBJ\-> 2 ROT 1 - ^ / \pi *
  ELSE DROP2 0
  END
\>>


71 -> '3352055187240799 25/73786976294838206464*π'

-> NUM -> 1.42718843886E-2
```

267.5 bytes, but it takes too long for this example (28m 34s... on the emulator!)

As a comparison, the RPN program on Free42 returns `1.4271(9054304)E-2` instantly (maybe a couple of seconds on a real 42S).

EMAIL PM FIND QUOTE REPORT

07-15-2018, 08:35 PM (This post was last modified: 07-15-2018 08:36 PM by Gerson W. Barbosa.) **Post: #19**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** (07-15-2018 07:26 PM)
> PS: Is there a better way to create a list of m zeros?

Better, but probably not the best way, at least on the 49/50g:

« { m } 0 CON AXL »

EMAIL PM FIND QUOTE REPORT

07-15-2018, 08:53 PM **Post: #20**

DavidM
Senior Member

Posts: 785
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Gerson W. Barbosa Wrote:** (07-15-2018 08:35 PM)
>
> > **Thomas Klemm Wrote:** (07-15-2018 07:26 PM)
> >
> > PS: Is there a better way to create a list of *m* zeros?
>
> Better, but probably not the best way, at least on the 49/50g:
>
> « { m } 0 CON AXL »

Gerson's method above is also how I would do it if the ListExt library isn't available.

Using the ListExt library:

« 0 m LMRPT »

Creating a list of 1000 0s with the CON method: 0.77s
Creating the same list with the ListExt command: 0.07s

Either should give reasonable performance in this scenario.

PM FIND QUOTE REPORT

07-15-2018, 10:03 PM **Post: #21**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:**
>
> I haven't trodden any of the hard (and beautiful) paths you all have done. Instead, **I took an unallowed (according to Valentin's rules) and dull shortcut**.

I never said any of that, Gerson, and I didn't stablish any **rules** whatsoever so you hardly could've taken *"an unallowed* { VA: by whom ? not me ... } *and dull shortcut."*

The only thing I said was this:

- *"Your results are correct, congratulations, but without explaining why would you compute the numerators and denominators that way it all seems a "magic trick" unrelated to the problem at hand as the relation to the integral seems shrouded in mystery"*

because until you posted your very detailed (thank you !) rationale for your direct computation that was exactly the case and none was the wiser.

Thanks again for your long and very interesting recent post detailing the process you followed to get your results, that's exactly the spirit and the way for interested people to learn interesting things but in the future please do not credit me with statements, whether direct or implied, which I didn't actually make.

Regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

07-15-2018, 10:22 PM **Post: #22**

John Keith
Senior Member

Posts: 615
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **DavidM Wrote:** (07-15-2018 08:53 PM)
>
> Gerson's method above is also how I would do it if the ListExt library isn't available.
>
> Using the ListExt library:
>
> « 0 m LMRPT »
>
> Creating a list of 1000 0s with the CON method: 0.77s
> Creating the same list with the ListExt command: 0.07s
>
> Either should give reasonable performance in this scenario.

Or equivalently, using only built-in commands:

<< 0 m NDUPN ->LIST >>

EMAIL PM FIND QUOTE REPORT

07-15-2018, 10:55 PM **Post: #23**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-15-2018 10:03 PM)
>
> .
> Hi, **Gerson**:
>
> > **Gerson W. Barbosa Wrote:**
> > I haven't trodden any of the hard (and beautiful) paths you all have done. Instead, **I took an <u>unallowed</u> (<u>according to Valentin's rules</u>) and <u>dull</u> shortcut**.
>
> *I never said any of that*, Gerson, and I didn't stablish any **rules** whatsoever so you hardly could've taken *"an unallowed* { VA: by whom ? not me ... } *and dull shortcut."*
>
> The only thing I said was this:
>
> - *"Your results are correct, congratulations, <u>but without explaining why would you compute the numerators and denominators that way</u> it all seems a "magic trick" unrelated to the problem at hand as the relation to the integral seems shrouded in mystery"*
>
> because until you posted your very detailed (thank you !) rationale for your direct computation <u>that</u> was exactly the case and none was the wiser.

Hello, Valentin,

Sorry for the misunderstanding! No, that was not what has elicited my comment about my possible infringement to your rules, which basically are “not googling for solutions”. Since I did google for an OEIS sequence, which eventually led me to a solution, it later appeared to me that I had done something that might be considered “illegal” or “unallowed”.
Thanks for your always interesting problems and solutions!

Gerson.

EMAIL PM FIND QUOTE REPORT

07-15-2018, 11:16 PM **Post: #24**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **John Keith Wrote:** (07-15-2018 10:22 PM)
>
> Or equivalently, using only built-in commands:
>
> **<< 0 m NDUPN ->LIST >>**

Very nice! Shorter and faster when compared to my suggestion (about 8 and 5 times on the 49G and on the 50g, respectively.

EMAIL PM FIND QUOTE REPORT

07-16-2018, 01:31 AM (This post was last modified: 07-16-2018 01:53 AM by Gerson W. Barbosa.) **Post: #25**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** (07-15-2018 07:26 PM)
>
> This *RPL* program calculates the coefficients:
>
> **Code:**
>
> ```
> «
>   « → m
>     « { 1 m } 0 CON OBJ→ DROP m →LIST
>     »
>   » → n ZEROS
>   « { 0 1 } 2 n
>     FOR k DUP SIZE → a s
>       « k ZEROS EVAL a +
>         a 1 k SUB 0 + REVLIST s 1 - ZEROS EVAL +
>         ADD
>         a k 1 + s SUB k 2 * ZEROS EVAL +
> ```

Thanks! We now can use it to compute the integrals much faster than I was able to before:

```
%%HP: T(3)A(R)F(.);
\<< DUPDUP DUPDUP * + 2 / 2 MOD
  IF NOT
  THEN
    \<<
      \<< \-> m
        \<< 0 m NDUPN \->LIST
        \>>
      \>> \-> n ZEROS
      \<< { 0 1 } 2 n
        FOR k DUP SIZE \-> a s
          \<< k ZEROS EVAL a + a 1 k SUB 0 + REVLIST s 1 - ZEROS EVAL + ADD a k 1 + s SUB k 2 * ZEROS EVAL + ADD
          \>>
        NEXT
```

```
      \>>
    \>> EVAL 1 GET 2 ROT 2 - ^ / \pi *
  ELSE DROP2 0
  END
\>>
```

(I've only replaced `{ 1 m } 0 CON OBJ→ DROP m →LIST` with John Keith's contribution elsewhere in this thread).

`20 -> '1909/65536*π'`, after 15.1 seconds (previously 118.4 seconds)

And here is a list containing the results of the integrals, starting from n = 1 up to n = 71:

```
{ 0 0 '1/2*π' '1/4*π' 0 0 '1/8*π' '7/64*π' 0 0 '35/512*π' '31/512*π' 0 0 '361/8192*π' '657/16384*π' 0 0 '2055/65536*π' '1909/65536*π' 0 0 '24955/1048576*π' '46923/2097152*π' 0 0
'316301/16777216*π' '299973/16777216*π' 0 0 '4136805/268435456*π' '15796439/1073741824*π' 0 0 '13853361/1073741824*π' '26585247/2147483648*π' 0 0 '756388295/68719476736*π'
'182188585/17179869184*π' 0 0 '20965992017/2199023255552*π' '20268008015/2199023255552*π' 0 0 '294245741167/35184372088832*π' '570497115729/70368744177664*π' 0 0
'4173319332859/562949953421312*π' '4055330794367/562949953421312*π' 0 0 '59723919552183/9007199254740992*π' '58153763705741/9007199254740992*π' 0 0 '430665931945033/72057594037927936*π'
'840170667413757/144115188075855872*π' 0 0 '12505857230438737/2305843009213693952*π' '12217503312833669/2305843009213693952*π' 0 0 '18265087511152103 3/36893488147419103232*π'
'44670833701814021/9223372036854775808*π' 0 0 '3352055187240799 25/73786976294838206464*π' }
```

Just a few minutes on the emulator.

Thanks again for providing both the program and an explanation why this works!

Gerson.

PS: Here is the result for n = 100, in case someone wants to check it :-)

`'432756001487181254158446581/158456325028528675187087900672*π'` (399 seconds, on the emulator)

EMAIL PM FIND QUOTE REPORT

07-16-2018, 04:27 AM **Post: #26**

Thomas Klemm
Senior Member

Posts: 1,447
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **John Keith Wrote:** (07-15-2018 10:22 PM)
> Or equivalently, using only built-in commands:
>
> **<< 0 m NDUPN ->LIST >>**

That's what I was originally looking for but it appears to be missing on the HP-48.
Thanks for all your suggestions.

> **Gerson W. Barbosa Wrote:** (07-16-2018 01:31 AM)
> We now can use it to compute the integrals much faster than I was able to before.

If you want to create a list of <u>all</u> values you better do that within the **FOR**-loop:

Code:

```
«
  « 0 SWAP NDUPN →LIST
  » → n ZEROS
  « { } { 0 1 } 2 n
    FOR k DUP SIZE → a s
      « π a HEAD * 2 k 3 - ^ / +
        k ZEROS EVAL a +
        a 1 k SUB 0 + REVLIST s 1 - ZEROS EVAL +
        ADD
```

```
a k 1 + s SUB k 2 * ZEROS EVAL +
ADD
```

Kind regards
Thomas

PM FIND QUOTE REPORT

07-16-2018, 02:18 PM **Post: #27**

John Keith
Senior Member

Posts: 615
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** (07-16-2018 04:27 AM)
>
> > **John Keith Wrote:** (07-15-2018 10:22 PM)
> >
> > Or equivalently, using only built-in commands:
> >
> > **<< 0 m NDUPN ->LIST >>**
>
> That's what I was originally looking for but it appears to be missing on the HP-48.
> Thanks for all your suggestions.

My mistake, I was assuming you were using the HP 50g.

EMAIL PM FIND QUOTE REPORT

07-18-2018, 01:36 AM **Post: #28**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:**
>
> PS: Here is the result for **n = 100, in case someone wants to check it** :-)
>
> `'432756001487181254158446581/158456325028528675187087900672*π'` (399 seconds, on the emulator)

Checked Ok. I first get this:

```
I(100) = 0.0085799230470872552899621320334918486875752516022037177062536244981437941118351804321866767529083052476306866829563033462749729 6
```

which then gets recognized as:

```
432756001487181254158446581 / 158456325028528675187087900672 * Pi
```

in perfect agreement with your result. In return, will you please confirm my result for **N=200**, namely:

```
I(200) = 195115902556687929766460554749767560813889646699192346811 / 200867255532373784442745261542645325315275374222849104412672 * Pi
```

Regards.
V.
.

Find All My HP-related Materials here: Valentin Albillo's HP Collection

PM WWW FIND EDIT QUOTE REPORT

07-18-2018, 04:07 AM **Post: #29**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-18-2018 01:36 AM)
> In return, will you please confirm my result for **N=200**, namely:
>
> I(200) = 195115902556687929766460554749767560813889646699192346811 / 200867255532373784442745261542645325315275374222849104412672 * Pi

I am sorry, but I fear the HP 50g is not up to this task (at least with the current algorithm). I(100) has required a clean emulated 50g and almost seven minutes, so I won't even try. Anyway, the first seven significant digits of my approximate result on Free42, **3.051640**78210624e-3, agree with your numerical result for I(200).
Not meaning to abuse your good will, would you please check how many significant digits I get right for **N=20000**?

3.06979593309e-6

Thank you very much!

Gerson.

EMAIL PM FIND QUOTE REPORT

07-18-2018, 07:17 AM **Post: #30**

Werner
Senior Member

Posts: 552
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

For those who want NDUPN on their 48:

straightforward

Code:
```
\<< \-> N \<< 0. N START DUP NEXT DROP2 N \>> \>>
```

without local variables

Code:
```
\<< 0. OVER START OVER SWAP NEXT ROT ROT DROP2 \>>
```

fast for large N

Code:
```
\<<
  \-> n
  \<<
    1 2 n
    FOR i
      DUPN i EVAL DUP
    STEP
    n -
    IF DUP 0 <
    THEN NEG DUPN
    ELSE DROPN END
```

Take your pick ;-)
Cheers, Werner

EMAIL PM FIND QUOTE REPORT

07-18-2018, 06:32 PM (This post was last modified: 07-18-2018 06:35 PM by ijabbott.) **Post: #31**

ijabbott
Senior Member

Posts: 921
Joined: Jul 2015

RE: [VA] SRC#001 - Spiky Integral

Is there a neat formula for just the constant term when converting the product to a sum? I tried Thomas Klemm's `spiky` function for N=20001 and it didn't like it very much! (Probably because it stores the coefficients for all the sequences up to N, eating lots of memory in the process.)

— Ian Abbott

EMAIL PM FIND QUOTE REPORT

07-19-2018, 12:39 AM **Post: #32**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:** (07-18-2018 04:07 AM)
> Not meaning to abuse your good will, **would you please check how many significant digits I get right for N=20000**?

No, sorry, I can't verify your alleged numeric result for **N=20,000**, because:

- ***f(x) = Cos(x)*Cos(2*x)*...*Cos(20000*x)*** is not just *'spiky'*, it's *"solid-area"* spiky, with tens of thousands of extremely thin spikes crowding the *[0,2*Pi]* interval so much that very few samples, if any, fall on each individual spike even using a million samples. Thus, I'd have to use *tens of millions of samples* to compute the integral in *[0,2*Pi]* to any useful accuracy.

- ***f(x) is the product of 20,000 cosines***, each of which is a number with absolute value ***<= 1*** (it's exactly ***1*** only at *x=0* and various fractions of *Pi*, none of which are ever sampled) and with *average absolute value* ***2/Pi***, so ***f(x)*** is typically *~ (2/Pi)^20000* ***~ 10^(-3922)*** for almost every *x* in *[0,2*Pi]*.

  This means I'd need to compute each sample using <u>at least 4,000 decimal digits of precision</u>, i.e.: evaluate the product of 20,000 cosines, each of them computed to 4,000 decimal digits, for **each* and every one of the many million samples. Trying to use less decimal digits, say *1,000* or *2,000,* would result in *f(x)* evaluating to ***0*** for *every* x sampled, as would the integral itself.

Needless to say, computing I(20000) this way would require a *truly humongous* amount of time, certainly it would for my *POPS (Pretty Old Pretty <u>Slow</u>)* system, and regrettably I can't allocate that much running time to this task.

Anyway, on a more feasible scale and in case it still might be useful to you, this is what a sufficiently accurate algorithm should return for **N=1,000**:

```
I(1000) = 0.0002742581536        (all digits shown are correct)
```

Regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

07-19-2018, 01:27 AM **Post: #33**

Thomas Klemm
Senior Member

Posts: 1,447
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **ijabbott Wrote:** (07-18-2018 06:32 PM)
> Is there a neat formula for just the constant term when converting the product to a sum?

From A058377:

> **Quote:**
> FORMULA
> a(n) is half the coefficient of q^0 in product('(q^(-k)+q^k)', 'k'=1..n) for n >= 1.

Thus I doubt there is a "neat formula".
However here's a table of n, a(n) for n = 1..3342

From this we can calculate the value of *I(1000)* in accordance with Valentin's result:

Code:
```
>>> pi * 23385429349552689006523846424823575124524010013195676079918742537593627724690522222877561004003130584906467004224646067900127143577719083836763175003199331
52618246357256959031783393818502766544769519367557819335776078670054986664044830878858579421236476481386740895972986819419273322159044663387 08 / 2**998
0.00027425815360837894
```

Since I had trouble with the *Python* program with bigger numbers I implemented it in *Clojure*:

Code:
```
  (replicate k 0))

(defn spiky [n]
  (loop [k 2 a '(0 1)]
    (if (> k n) (first a)
        (let [s (count a)
              [b c] (split-at k a)
              u (concat (zeros k) a)
              v (concat (conj (reverse b) 0) (zeros (dec s)))
              w (concat c (zeros (* 2 k)))]
          (recur (inc k) (map +' u v w))))))
```
You may notice similarities to the implementation in *RPL*.

It's fast for *n=100*, takes a couple of seconds for *n=200* and multiple minutes for *n=1000*.
I've tried it for *n=2000* and after a while that felt like eternity the correct result was given.
From these measurements I would assume that it would take days or even weeks to calculate the value for *n=20,000*. Thus I refrained from trying.

Kind regards
Thomas

PM FIND QUOTE REPORT

07-19-2018, 02:22 AM **Post: #34**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** (07-19-2018 01:27 AM)
>> **ijabbott Wrote:** (07-18-2018 06:32 PM)
>> Is there a neat formula for just the constant term when converting the product to a sum?

From A058377:

> **Quote:**
>
> FORMULA
> a(n) is half the coefficient of q^0 in product('(q^(-k)+q^k)', 'k'=1..n) for n >= 1.
>
> Thus I doubt there is a "neat formula".

There is an asymptotic formula, but it doesn't help much:

https://cs.uwaterloo.ca/journals/JIS/VOL...ivan8.html

(((n^2+n)/2+1) mod 2)*sqrt(6/pi)*2^(n-1)/(n*sqrt(n))

n = 3 -> 1.06385 (1)
n = 4 -> 1.38198 (1)
n = 8 -> 7.81764 (7)
n = 11 -> 38.7893 (35)
n = 27 -> 661051 (632602)
n = 1000 -> 2.34135e296 (2.3385429e296)

A small correction might help a bit:

(((n^2+n)/2+1) mod 2)* sqrt(6/pi)*2^(n-1)*(1-6/(5*n)+21/(20*n^2)-1/(8*n^3)+3/n^4)/(n*sqrt(n))

n = 3 ->0.7969 (1)
n = 4 -> 1.07157 (1)
n = 8 -> 6.77707 (7)
n = 11 -> 34.8986 (35)
n = 27 -> 632623 (632602)
n = 1000 -> n = 1000 -> 2.33854293231e296 (2.33854293496e296)

Regards,

Gerson.

EMAIL PM FIND QUOTE REPORT

07-19-2018, 02:31 AM **Post: #35**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-19-2018 12:39 AM)
>
> Needless to say, computing I(20000) this way would require a *truly humongous* amount of time, certainly it would for my *POPS (Pretty Old Pretty Slow)* system, and regrettably I can't allocate that much running time to this task.
>
> Anyway, on a more feasible scale and in case it still might be useful to you, this is what a sufficiently accurate algorithm should return for **N=1,000**:
>
> **I(1000) = 0.0002742581536** *(all digits shown are correct)*

It has been useful!

On Free42 I get 0.000274258153298, which is good to 9 significant digits. Since that's based on an asymptotic formula, the result for N=20,000 should be even more accurate.

Thanks again,

Gerson.

EMAIL PM FIND QUOTE REPORT

07-19-2018, 10:07 PM **Post: #36**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:** (07-19-2018 02:31 AM)
>
> > **Valentin Albillo Wrote:** (07-19-2018 12:39 AM)
> >
> > Anyway, on a more feasible scale and in case it still might be useful to you, this is what a sufficiently accurate algorithm should return for **N=1,000**:
> >
> > ```
> > I(1000) = 0.0002742581536        (all digits shown are correct)
> > ```
>
> It has been useful!
> On Free42 I get 0.000274258153298, which is good to 9 significant digits. **Since that's based on an asymptotic formula, the result for N=20,000 should be even more accurate**.

Seems likely. As I said in some previous post, I don't use any theoretical way to compute the numerators ***S(n)*** which, when divided by the respective powers of 2 (and times Pi), directly give the value of the integral. I simply compute *the integral itself* numerically using a quadrature algorithm, which for the case **N=1,000** goes as follows (9 iterations):

```
0     0.00031973267525170880671090486042929081382754O315
1     0.00029366466207370769882435773475819140874249698 9
2     0.00027232783588774944449352798692706228629184347 7
3     0.00027403463882402095373785663748392970850425967 0
4     0.00027425942664444567537509586766350495168580923 5
5     0.00027425815441455235746909640075385877816761332 2
6     0.00027425815360837555763283919688872868360492973 0
7     0.00027425815360837892680774111902830129968127660 0
8     0.00027425815360837892680773443266980800775290852 8
9     0.000274258153608378926807734432669808007979394750
```

So we get **I(1000) = 0.000274258153608378926807734432669808007979394750** *(all 48 decimal digits shown are correct)*

Applying a multilevel extrapolation scheme to those iterations quickly gives in excess of 100 correct decimal digits. These quadrature-provided results are useful to check that the *S(n)*-provided ones are correct.

Regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

07-20-2018, 03:30 PM (This post was last modified: 07-20-2018 10:41 PM by Albert Chan.) **Post: #37**

Albert Chan
Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-10-2018 11:10 PM)
>
> **Hi all**, welcome to my ***SRC#001 - Spiky Integral***:
>
> Here I'll deal with a real-world math problem not of my own making but which did appear at a certain math competition. The problem introduces this *"spiky"* integral:

$I(N) = \int_0^{2\pi} cos(x)\, cos(2x)\, cos(3x) \ldots cos(Nx)\, dx$

and asks ***for which values of N in the interval [1,10] does I(N) have a <u>non-zero</u> value***.

It then specifies that the result must include proof of correctness

It seems the problem is easier to solve with symmetry.

let F = cos(x) cos(2x) cos(3x) ... cos(Nx)

$I(N) = \int_0^{2\pi} F dx = 2 \int_0^{\pi} F dx = 2 \int_0^{\pi/2} F dx + 2 \int_{\pi/2}^{\pi} F dx$

If F is symmetric around x = Pi/2, the two terms are same sized and same sign.

A simple test is when x=Pi, F=1, which imply even number of odd values between 1 to N:
(since cos(odd Pi) = -1, even numbers of such terms restored the symmetry)

--> If N mod 4 = 0 or 3, $I(N) = 4 \int_0^{\pi/2} F dx$

Since F is maximized (= 1.0) when x=0, above should always be positive.
As N increases, F spike is "thinner", thus smaller $I(N)$, but still above 0

If odd number of odd values between 1 and N, symmetry is flipped.
The two terms still have same size, but opposite sign, cancelling each other.

--> If N mod 4 = 1 or 2, $I(N) = 0$

So, for interval [1, 10] and non-zero $I(N)$, N = 3,4,7,8

EMAIL PM FIND QUOTE REPORT

07-20-2018, 09:06 PM (This post was last modified: 07-21-2018 04:41 AM by Albert Chan.) **Post: #38**

Albert Chan
Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

Gerson W. Barbosa Wrote: (07-18-2018 04:07 AM)

would you please check how many significant digits I get right for **N=20000**?

```
3.06979593309e-6
```

Let F= cos(x) cos(2x) cos(3x) ... cos(N x)

Since N mod 4 = 20000 mod 4 = 0, $I(N) = \int_0^{2\pi} F dx = 4 \int_0^{\pi/2} F dx$

For big N, integral is dominated mostly by the area of spike:

$I(N) \sim 4 \int_0^{\pi/(2N)} F dx$

I did the integral in Python (plain float):

$I(20000) \sim 4 * 7.67448983276e\text{-}07 = 3.0697959331e\text{-}06$

Both values agreed each other, to 11 digits.

Comment:
it is not necessary to sum the full spike area.

For x = Pi / (20N) (one tenth of spike base), F = 1.547e-36, which contribution little to the sum.
With this tighter base, *I(20000)* still converge to the same value, but only take 16 sec (instead of 145 sec)

BTW, my computer is 20+ years old Dell P3, modern computer may only take few seconds.

EMAIL PM FIND QUOTE REPORT

07-21-2018, 05:06 AM **Post: #39**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Albert Chan Wrote:** ⇒ (07-20-2018 09:06 PM)
>
> For big N, integral is dominated mostly by the area of spike:
>
> *I(N)* ~ 4 $\int_0^{\pi/(2N)} F dx$
>
> I did the integral in Python (plain float):
>
> *I(20000)* ~ 4 * 7.67448983276e-07 = 3.0697959331e-06
>
> Both values agreed each other, to 11 digits.

Thanks both for the I(20000) result and for another interesting approach. These results might be useful to improve the correction terms of the approximation formula.

EMAIL PM FIND QUOTE REPORT

07-21-2018, 05:00 PM **Post: #40**

Albert Chan
Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

> **Gerson W. Barbosa Wrote:** ⇒ (07-21-2018 05:06 AM)
>
> Thanks both for the I(20000) result and for another interesting approach. These results might be useful to improve the correction terms of the approximation formula.

Hi Gerson,

I am a bit late to the game, and just realized your number is also an approximation.
So, both numbers matching are nice, but probably not good enough.

Instead, I tried *I(1000)* ~ 4 $\int_0^{\pi/2000} F dx$ = 0.00027425815360837926

From Valentin Albillo last post, my estimate match correct values, to 15 digits
Comparing values in binary form, accuracy is even better, mine is just 6 ULP over !

Thanks Valentin.

I am amazed at how your integration function work.
You mentioned the problems of it computing *I(20000)*.
Shrinking the integral range 800,000 times may help: (0 to Pi/400000) instead of (0, 2 Pi)

EMAIL PM FIND QUOTE REPORT

07-22-2018, 01:33 AM **Post: #41**

Valentin Albillo

Posts: 636
Joined: Feb 2015

Senior Member

Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.

Hi, Albert:

> **Albert Chan Wrote:** ⇒ (07-21-2018 05:00 PM)
>
> From Valentin Albillo last post, my estimate match correct values, to 15 digits. Comparing values in binary form, accuracy is even better, mine is just 6 ULP over !
>
> **Thanks Valentin**. I am amazed at how your integration function work.

You're welcome. Thanks to you for your fresh and effective insight, your contribution to this subject has been truly outstanding and I for one appreciate it. Also, I'm glad if my numeric quadrature results were useful for you to check your remarkably accurate approximations.

> **Quote:**
>
> You mentioned the problems of it computing *I(20000)*.
> **Shrinking** the integral range 800,000 times may help: (0 to Pi/400000) instead of (0, 2 Pi)

Sure it does. These are the results my quadrature program outputs for the various ranges you mention:

I(20000, 0..Pi/40000)

0 **0.0000007674489**75625012059728677812789736739179600994

1 **0.0000007674489832760**14084555342766291500235218569303

2 **0.0000007674489832760347202085**1084672424898967032756 3

3 **0.000000767448983276034720208518559793576182713599**9610

4 **0.000000767448983276034720208518559793576182713598529** *(all 51 decimals digits shown are correct)*

I(20000, 0..Pi/400000)

0 **0.000000767448983276034720208518559793576182640549996 31**
350641171121504026981165080771398176783847145495615 11

1 **0.000000767448983276034720208518559793576182640549996 31**
3503851137908194154883506920060816 43072573304299154 01

2 **0.000000767448983276034720208518559793576182640549996 31**
350385113790819415488350692006081642898064914750479 63 *(all 106 digits shown are correct)*

However, I concur with you that the fact that these results are accurate to *51* and *106* decimal digits, respectively, for both shrunk ranges does not necessarily mean that after multiplying them by the correct factor the resulting *approximate* values for the *full-range (0..2*Pi)* integral will be accurate to that many digits, in fact it's quite conceivable that they are *not* because both results above match only up to ***42*** digits, namely:

0.000000767448983276034720208518559793576182

Last but not least, using your remarkable *shrinking* technique, the value on the full integral for **N=20,000** indeed **can** be computed using an HP calc using just its native 12-digit precision (!!), as these *3 lines* of code I wrote for the **HP-71B** convincingly demonstrates:

```
DESTROY ALL @ SFLAG -1 @ DIM P,I @ DISP 4*INTEGRAL(0,PI/40000,1E-8,FNF(IVAR))
DEF FNF(X) @ P=1 @ FOR I=1 TO 20000 @ P=P*COS(I*X) @ IF NOT P THEN END
NEXT I @ FNF=P
```

>RUN

3.069795933556E-6 *(75 min. in **Emu71**)* wich is accurate to **10 digits**.

Note: That simplest of tricks in the function's definition *("IF NOT P THEN END")* <u>halves</u> the computing time.

Regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

07-22-2018, 05:50 AM (This post was last modified: 07-22-2018 11:37 AM by Albert Chan.) **Post: #42**

Albert Chan
Senior Member
Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-22-2018 01:33 AM)
>
> both results above match only up to ***42*** digits, namely:
>
> `0.000000767448983276034720208518559793576182`

Hi Valentin, Thanks for the check.

The fact that two integral only match 42 digits does not mean anything.
The spike formula asked for **full spike area**.

I chopped the spike only for speedier computation, trading accuracy for time.

Plotting N vs spike formula digits accuracy, and it followed a straight line,
already reaching 15 digits accuracy for N = 60

You might like to try the spike formula again, for N = 1000.
For *I(1000)*, you have a correct value to match against ...

07-22-2018, 03:01 PM **Post: #43**

Albert Chan
Senior Member
Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

Using Python + numpy, I managed to calculate *I(1000)*

Code:

```
import numpy

def spike(n):
    terms = 1 + n*(n+1) // 2    # terms count of Product[z^i + 1/z^i, {i, n}]
    if terms % 2 == 0: return 0 # no constant term
    b = numpy.array([0] * (terms//2 + 1), dtype=object)
    b[-2:] = 1                  # 1 + z
    for i in range(2, n+1):     # multiply by 1 + z^i
        b[:-i] += b[i:]         # build half terms
    return b[0]                 # constant term
```

>>> spike.spike(1000)
46770858699105378013047692849647150249048020026391352159837485075187255449381044
44575512200800626116981293400844929213580025428715543816767352635000639866305236
49271451391806356678763700553308953903873511563867155215734010997332808966175771
715884247295296277348179194597363883854664431808932677416L

Above calculation take **49 seconds** on my laptop

I(1000) = b[0] / 2^1000 * (2 Pi) =

0.000274258153608378926807734432669808007979394749673091726358234027755841714506
724234556964545380120825381783157659756758893238403974033229771909645027440110 04
817396115520260429033568814177090161109686357644415948319736196030024371755584 85
429100067602123263082583991209351012812036081763570096068925645759247750677 19 ...

Rounded to 45 digits, it match Valentin value **exactly**

Is it possible to explain how the integration code work ?
How does it handle so many spikes ?

EMAIL PM FIND QUOTE REPORT

07-22-2018, 10:06 PM (This post was last modified: 07-22-2018 10:07 PM by Valentin Albillo.) **Post: #44**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

,
Hi, **Albert**:

> **Albert Chan Wrote:**
> Hi Valentin, Thanks for the check.

You're welcome.

> **Quote:**
> The fact that two integral only match 42 digits does not mean anything. The spike formula asked for full spike area. [...]You might like to try the spike formula again, for N = 1000. For I(1000), you have a correct value to match against ...

I'll do it soon, when I get home in a few hours.

> **Quote:**
> Rounded to 45 digits, it match Valentin value **exactly**

Of course. That's why I stated *"(all 48 decimal digits shown are correct)"* ... :-) ... (the 48 decimal digits mentioned include the three initial zeros as well).

> **Quote:**
> Is it possible to explain how the integration code work ? How does it handle so many spikes ?

I developed the code myself as a program for the **HP-71B**, good for 12 digits and easily surpassing the speed of the assembly-language *Math ROM*'s *INTEGRAL* keyword, then ported it nearly *verbatim* to a multiprecision environment, which is the version I've used to produce the results I've posted in this thread.

The code was intended to be included in one of my PDF *Datafile* articles some 10 years ago, as explaining it and providing relevant examples would take at least some 10-12 pages. Alas, for reasons which I won't repeat here the article was never submitted for publication and remains unpublished to this day.

If (and when) I can find some adequate place where to publish it, I'll post an announcement here for interested people (like yourself) to download it for free.

Thanks for your interest and regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

07-24-2018, 01:13 AM (This post was last modified: 03-27-2019 07:02 PM by Albert Chan.) **Post: #45**

Albert Chan
Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

Using Gaussian Quadrature for **spike formula = 4*I(1000, 0 to Pi/2000)**, against exact **I(1000)**:

- 200 digits accuracy
- 204 matched digits after decimal point

EMAIL PM FIND QUOTE REPORT

« Next Oldest | Next Newest »

Enter Keywords Search Thread

NEW REPLY

View a Printable Version
Send this Thread to a Friend
Subscribe to this thread

User(s) browsing this thread: Valentin Albillo*