MoHPC The Museum of HP Calculators

Search Member List Calendar Help

Welcome back, Valentin Albillo. You last visited: Yesterday, 11:43 PM (**User CP** — Log Out)
View New Posts | View Today's Posts | Private Messages (Unread 0, Total 145)

Current time: 04-30-2019, 02:26 AM
Open Buddy List

HP Forums / HP Calculators (and very old HP Computers) / General Forum / [VA] SRC#001 - Spiky Integral

Pages (3): « Previous 1 **2** 3 Next »

NEW REPLY

[VA] SRC#001 - Spiky Integral

Threaded Mode | Linear Mode

07-15-2018, 11:03 PM **Post: #21**

Valentin Albillo
Senior Member

Posts: 347
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.
.

PM FIND EDIT QUOTE REPORT

07-15-2018, 11:22 PM **Post: #22**

John Keith
Senior Member

Posts: 389
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **DavidM Wrote:** ⇒ (07-15-2018 09:53 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, 11:55 PM **Post: #23**

Gerson W. Barbosa
Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** ⇒ (07-15-2018 11:03 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

07-16-2018, 12:16 AM **Post: #24**

Gerson W. Barbosa
Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **John Keith Wrote:** ⇨ (07-15-2018 11:22 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, 02:31 AM (This post was last modified: 07-16-2018 02:53 AM by Gerson W. Barbosa.) **Post: #25**

Gerson W. Barbosa

Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** ⇨ (07-15-2018 08:26 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
'182650875111521033/36893488147419103232*π' '44670833701814021/9223372036854775808*π' 0 0
'335205518724079925/73786976294838206464*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, 05:27 AM **Post: #26**

Thomas Klemm
Senior Member

Posts: 1,449
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **John Keith Wrote:** (07-15-2018 11:22 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 02:31 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 all 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, 03:18 PM **Post: #27**

John Keith
Senior Member

Posts: 389
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** ⇒ (07-16-2018 05:27 AM)
>
> > **John Keith Wrote:** ⇒ (07-15-2018 11:22 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, 02:36 AM **Post: #28**

Valentin Albillo
Senior Member

Posts: 347
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.
.

PM FIND EDIT ✕ QUOTE REPORT

07-18-2018, 05:07 AM **Post: #29**

Gerson W. Barbosa
Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-18-2018 02:36 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, 08:17 AM **Post: #30**

Werner
Senior Member

Posts: 346
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, 07:32 PM (This post was last modified: 07-18-2018 07:35 PM by ijabbott.) **Post: #31**

ijabbott
Senior Member

Posts: 533
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, 01:39 AM **Post: #32**

Valentin Albillo
Senior Member

Posts: 347
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:** ⇨ (07-18-2018 05:07 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 at least 4,000 decimal digits of precision, 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 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)
```

Regards.
V.
.

PM FIND EDIT QUOTE REPORT

07-19-2018, 02:27 AM **Post: #33**

Thomas Klemm
Senior Member

Posts: 1,449
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **ijabbott Wrote:** (07-18-2018 07:32 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 * 2338542934955268900652384642482357512452401001319567607991874253759362772469052222287756100400313058490646700422464606790012714357771908383676317500319933152618246357256959031783393818502766544769519367557819335776078670054986664044830878858579421236476481386740895972986819419273322159044663387 08 / 2**998
0.00027425815360837894
```

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

Code:

```
(defn zeros [k]
  (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)))]
```

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, 03:22 AM **Post: #34**

Gerson W. Barbosa
Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Thomas Klemm Wrote:** (07-19-2018 02:27 AM)
>
> > **ijabbott Wrote:** (07-18-2018 07:32 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, 03:31 AM **Post: #35**

Gerson W. Barbosa
Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-19-2018 01:39 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, 11:07 PM **Post: #36**

Valentin Albillo
Senior Member

Posts: 347
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#001 - Spiky Integral

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:** (07-19-2018 03:31 AM)
>
> > **Valentin Albillo Wrote:** (07-19-2018 01:39 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     0.000319732675251708806710904860429290813827540315
1     0.000293664662073707698824357734758191408742496989
2     0.000272327835887749444493527986927062286291843477
3     0.000274034638824020953737856637483929708504259670
4     0.000274259426644445675375095867663504951685809235
5     0.000274258154414552357469096400753858778167613322
6     0.000274258153608375557632839196888728683604929730
7     0.000274258153608378926807741119028301299681276600
```

```
        8     0.000274258153608378926807734432669808007752908528
        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.

PM FIND EDIT QUOTE REPORT

07-20-2018, 04:30 PM (This post was last modified: 07-20-2018 11:41 PM by Albert Chan.) **Post: #37**

Albert Chan
Senior Member

Posts: 623
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

> **Valentin Albillo Wrote:** (07-11-2018 12:10 AM)
>
> **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, 10:06 PM (This post was last modified: 07-21-2018 05:41 AM by Albert Chan.) **Post: #38**

Albert Chan
Senior Member

Posts: 623
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

> **Gerson W. Barbosa Wrote:** (07-18-2018 05:07 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) ~ 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.

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, 06:06 AM **Post: #39**

Gerson W. Barbosa
Senior Member

Posts: 1,135
Joined: Dec 2013

RE: [VA] SRC#001 - Spiky Integral

> **Albert Chan Wrote:** (07-20-2018 10:06 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, 06:00 PM **Post: #40**

Albert Chan
Senior Member

Posts: 623
Joined: Jul 2018

RE: [VA] SRC#001 - Spiky Integral

> **Gerson W. Barbosa Wrote:** (07-21-2018 06:06 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

« Next Oldest | Next Newest »

Enter Keywords Search Thread

Pages (3): « Previous 1 **2** 3 Next »

NEW REPLY

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