

Investigation of E.B. Rosa's Round Wire Mutual Inductance Correction Formula

By Robert Weaver, July 2008
Saskatoon, Canada

Rosa's round wire correction factors for mutual inductance were tabulated in Rosa and Grover's 1911 paper [1](Table VIII, page 199), and in Grover's 1946 Inductance Formula's text [2](Table 39, page 150). The method used to determine these factors is not well explained. The generating function appears at the top of Rosa and Grover's Table VIII as:

$$B = \frac{2}{n} \sum_{m=1}^{n-1} m \operatorname{Ln} \frac{m}{R_m} \quad (1)$$

(Note that "B" is used in reference 1, and "H" is used in reference 2, for the same variable. I will use H in this discussion.)

Unfortunately, the meaning of this formula is not particular clear. The purpose of this investigation, was to determine the details of the original method used to generate the tabulated data, so that it could then be used to produce higher precision calculations than the original four decimal places, and also to gain a better understanding of the underlying science.

Although the above function is not well described, Grover provides a very brief discussion at the end of Chapter 2 which helps shed some light on the subject. Paraphrasing: *It is necessary to correct for the difference in mutual inductance between pairs of round wires from mutual inductance between pairs of current sheet segments.* This must be applied to each pair of turns in the coil. *Each pair of turns* means each turn paired with each other turn in the coil, not just adjacent turns. Hence, while a two turn coil has only one pair of turns, a three turn coil has three pairs of turns: turns 1 & 2 comprise one pair; turns 2 & 3 comprise a second pair; and turns 1 & 3 comprise the third pair. It follows then that a 4 turn coil will have six pairs, a five turn coil will have 10 pairs, and so on.

The correction for the difference between current sheet and round wire mutual inductance amounts to the ratio of the Geometric Mean Distance (GMD) between round wire pairs, to the GMD between current sheet segment pairs.

From Grover, page 24, the GMD between circular cross-sections is simply the distance between their centres. When expressed in terms of windings and winding pitch it is:

$$R_{RW} = mp$$

where R_{RW} is the GMD between round wires, m is the number of turns separating the two turns in question, and p is the centre to centre winding pitch.

Taking the natural logarithm of the expression, puts it in a form which will be more convenient later:

$$\operatorname{Ln}(R_{RW}) = \operatorname{Ln}(mp) \quad (2)$$

From Grover, page 20, the GMD between current sheet segments is given by:

$$\operatorname{Ln}(R_{CS}) = \operatorname{Ln}(mp) - \left[\frac{1}{12m^2} + \frac{1}{60m^4} + \frac{1}{168m^6} + \frac{1}{360m^8} + \frac{1}{660m^{10}} + \dots \right] \quad (3)$$

where R_{CS} is the GMD between current sheet segments, m is the number of turns separating the current sheet segments, and p is the winding pitch (and hence, the width of each current sheet segment). Referring back to Rosa's original formula (1), it appears that, inside the logarithm function, the numerator m accounts for the GMD of round wire pairs, and denominator R_m is the GMD of current sheet pairs. It also appears that the function has been normalized to a unit pitch factor, so that p disappears. Since m/R_m is inside a log function, and since the above GMD formulae are in logarithmic form, it is simpler to subtract the log of the denominator from the log of the numerator, rather than take the log of the ratio. Subtracting (3) from (2), the $\operatorname{Ln}(mp)$ portion neatly disappears and we are left with only the series portion of the

current sheet GMD formula which I will refer to as $\Delta\text{GMD}(\mathbf{m})$ in this discussion. (Admittedly, “ ΔLnGMD ” would be a more accurate term.)

The result is then:

$$\Delta\text{GMD}(\mathbf{m}) = \left[\frac{1}{12\mathbf{m}^2} + \frac{1}{60\mathbf{m}^4} + \frac{1}{168\mathbf{m}^6} + \frac{1}{360\mathbf{m}^8} + \frac{1}{660\mathbf{m}^{10}} + \dots \right] \quad (4)$$

This must then be applied to each combination of pair of turns in the coil. And:

A 2 turn coil has its one pair spaced $\mathbf{m}=1$ apart.

A 3 turn coil has two pairs spaced $\mathbf{m}=1$ apart, and one pair spaced $\mathbf{m}=2$ apart

A 4 turn coil has three pairs spaced $\mathbf{m}=1$ apart, two pairs spaced $\mathbf{m}=2$ apart, and one pair spaced $\mathbf{m}=3$ apart, and so on.

In general then, an \mathbf{N} turn coil will have:

$\mathbf{N}-1$ pairs spaced $\mathbf{m}=1$ apart;

$\mathbf{N}-2$ pairs spaced $\mathbf{m}=2$ apart;

$\mathbf{N}-3$ pairs spaced $\mathbf{m}=3$ apart;

...and so on, until finally:

1 pair spaced $\mathbf{m}=\mathbf{N}-1$ apart.

Using summation notation and combining the above, we get the correction factor \mathbf{H} for a coil of \mathbf{N} turns:

$$\mathbf{H}(\mathbf{N}) = \sum_{\mathbf{m}=1}^{\mathbf{N}-1} (\mathbf{N}-\mathbf{m}) \Delta\text{GMD}(\mathbf{m}) \quad (5)$$

Note that I have a coefficient of $(\mathbf{N}-\mathbf{m})$ while Rosa has only \mathbf{m} . I decided to stay with $(\mathbf{N}-\mathbf{m})$, at least for the time being, and I tested the function in a spreadsheet, to see how closely it matched the tabulated values, and to determine what other factors may need to be included. It turned out that the values produced from this formula were off by a factor of $2/\mathbf{N}$. Rosa’s formula shown at the top of Table VIII included a factor of $2/\mathbf{N}$. This factor is easily accounted for. The 2 accounts for the fact that we are counting each turn paired with each other turn as opposed to each pair of turns. Hence the pair consisting of turn 1 and 2, while being a single pair, must be considered as turn 1 paired with turn 2, and then subsequently turn 2 paired with turn one, so that the mutual inductance effect on every turn is accounted for. Hence the number of terms is exactly doubled. The $1/\mathbf{N}$ part of the factor is simply to account for the fact that Rosa’s overall correction formula multiplies both the \mathbf{G} and \mathbf{H} terms by \mathbf{N} , and therefore the \mathbf{H} correction must initially scaled down by that amount. Hence:

$$\mathbf{H}(\mathbf{N}) = \frac{2}{\mathbf{N}} \sum_{\mathbf{m}=1}^{\mathbf{N}-1} (\mathbf{N}-\mathbf{m}) \Delta\text{GMD}(\mathbf{m}) \quad (6)$$

This formula now reproduces the tabular data accurately to the required four decimal places for $\mathbf{N}<30$. However, the error increases for values of $\mathbf{N}>30$. This is understandable when considering that, in the calculation, the values of $\Delta\text{GMD}(\mathbf{m})$ for small values of \mathbf{m} are multiplied by very large coefficients when \mathbf{N} is large. It also appears to be related to the limited accuracy in the spreadsheet program. The value of $\Delta\text{GMD}(1)$ is especially troublesome since it converges very slowly, and Grover simply recommends that the value of 0.1137 be used, rather than attempt to calculate it with the series formula. This is clearly insufficient accuracy for $\Delta\text{GMD}(1)$ if we want to calculate $\mathbf{H}(\mathbf{N})$ to four or more decimal places for large values of \mathbf{N} . So, we need more terms in the series.

Grover only gives the first five terms of the series formula which, except for $\mathbf{m}=1$, is sufficiently convergent to calculate the data. It is therefore necessary to refer to the original reference source [3] for the complete definition of the series formula, or else deduce the subsequent coefficients in the series. Since I hadn’t been able to obtain a copy of reference 3, I

attempted to deduce the coefficients.

The pattern in the series: 12, 60, 168, 360, 660..., is not immediately evident. However, after looking at the factors of the coefficients for some time, I noticed that if each coefficient is written as the product of 3 particular factors, arranged in a particular way, then a pattern emerges:

$$\begin{aligned}12 &= 2 \cdot 2 \cdot 3 \\60 &= 3 \cdot 4 \cdot 5 \\168 &= 4 \cdot 6 \cdot 7 \\360 &= 5 \cdot 8 \cdot 9 \\660 &= 6 \cdot 10 \cdot 11\end{aligned}$$

From this, we can predict that the coefficient of the n^{th} term is given by:

$$(n+1) \cdot (2n) \cdot (2n+1)$$

or

$$4n^3+6n^2+2n$$

Looking at it another way, doing a least squares polynomial fit would produce exactly the same function. The only way we could come up with a different polynomial would be to find at least a fifth degree polynomial. Because it takes only a third degree polynomial with simple integer coefficients to exactly fit the five points, we can be confident that this is the correct function. Confidence increases further as we will find that this function correctly predicts the hundred series coefficients that are required to calculate $\Delta\text{GMD}(1)$ to four decimal places.

This formula gives the first nine coefficients as: 12, 60, 168, 360, 660, 1092, 1680, 2448, 3420...

As mentioned above, the series converges very slowly for $m=1$. In fact, to calculate $\Delta\text{GMD}(1)$ to eight decimal places requires thousands of terms. This is obviously a job for a computer, and that is how I proceeded. The value of $\Delta\text{GMD}(1)$ then came out to 0.11370563873392 using 29240 terms. At this point, the value was changing by only 10^{-14} per term. However, due to accumulated round off error when calculating this many terms, I estimated this value of $\Delta\text{GMD}(1)$ not to be any more accurate than about ten decimal places. It is important that $\Delta\text{GMD}(1)$ be accurate, because it is used in every subsequent calculation. Not happy with the calculated accuracy, I investigated the use of an acceleration technique, and tried using Wynn's Epsilon Algorithm to see if it would accelerate the convergence, but found that it was more accurate by a factor of only about 100 for up to 6000 terms, and then beyond that, it became less accurate than the brute force calculation, due to round off error again. At this point I decided to see if the value could be derived analytically. Ultimately, I was able to determine the sum of this series analytically, and found that this value is simply $\text{Ln}(1/4)^{+3/2}$, or 0.113705638880109. This derivation obviously deserves further explanation, but is somewhat involved. So, I have relegated its complete analysis to Appendix C.

Reference [3] was later located, and it was found to be none other than a reprint of reference [1], containing exactly the same information. However, on examination of the original reference to page 168, it was found to contain an alternative non-series formula (130) for the GMD for current sheet segments. Substituting it for the series formula, the ΔGMD formula becomes:

$$\Delta\text{GMD}(m) = (m^2+1) \text{Ln}(m) - \frac{(m+1)^2}{2} \text{Ln}(m+1) - \frac{(m-1)^2}{2} \text{Ln}(m-1) + \frac{3}{2}$$

This produces exactly the same value of 0.113705638880109 for $m=1$, as was derived in Appendix C, but with far less effort. Unfortunately, that appears to be the extent of its usefulness, because for large values of m , the subtraction of large nearly equal terms results in more severe roundoff error than series formula (4).

Using the exact value of $\Delta\text{GMD}(1)$, and just the original five terms in the series formula for other m values, the calculated values, rounded to four decimal places, now agree exactly with all of the original tabulated data.

Combining the Δ GMD function with the H(N) function, the complete mutual inductance correction formula becomes:

$$H(N) = \frac{2}{N} \sum_{m=1}^{N-1} \sum_{i=1}^{\infty} \frac{N-m}{(2i+1)(i+1)2^i m^{2i}} \quad (7)$$

Because the inner summation converges very quickly for $m > 1$, only the first few i -terms need be evaluated. For $m=1$, the pre-calculated value of 0.113705638880109 should be used.

(For the sake of completeness, the formula was retested using a coefficient of m rather than $(N-m)$ as per Rosa's formula (1), just in case I had overlooked something obvious. The results were incorrect, as expected, indicating that the meaning of the terms in his formula are somewhat different, or there is simply a typographical error.)

Formula (7) has been implemented into a BASIC computer program which has been used to generate the table of 10 digit H values shown in Appendix A. Program listings are included in Appendix B.

In addition, Wynn's Epsilon Algorithm which had been previously coded into a BASIC program in an unsuccessful attempt to determine the value of Δ GMD(1), was reused to estimate the limiting value of $H(\infty)$. The program listing for Wynn's algorithm is included in Appendix B. Only eight H(N) values were used, starting at $N=3$ and each one increasing thereafter by a factor of 6:

N	H(N)
3	0.2713724853
18	0.3283537742
108	0.3363266810
648	0.3376196662
3888	0.3378341940
23328	0.3378699218
139968	0.3378758757
839808	0.3378768680
∞	0.3378770664

As indicated, this algorithm produces a value $H(\infty)=0.3378770664$. The fact that it arrived at this value using only eight H(N) values, and rerunning it using additional values made no difference within ten decimal places, leads me to believe that this is very close to the true value. Despite this, I would be more confident having an $H(\infty)$ value that was derived analytically.

Fortunately, a later paper by Grover published in 1929 [4]), was discovered after the above derivations and calculations were done. In it, Grover revisits Rosa's mutual inductance corrections and elaborates on their calculation, as well as providing a new non-series formula for their calculation. Grover discusses the difficulty of generating the H values for large values of N ("large" meaning approximately $N > 30$). He specifically comments, "Furthermore, in the calculation of a table for different values of n , the fact that the calculation for a given value of n rests upon the calculation for smaller values of n , although seemingly an advantage, works to the end that any error made with a smaller value of n is carried through into the calculation for the larger values of n ."

The primary purpose of Grover's 1929 paper was to compare two popular inductance calculation methods: the current sheet method with Rosa's corrections, and a direct summation method; and to show that they are in fact equivalent. At the time, there were concerns by some individuals that the current sheet formula was not accurate. However, this was shown to be a result of incorrectly determining the effective current sheet dimensions, a problem which apparently persists to this day.

Having shown that the two inductance calculation methods were equivalent, Grover went on to identify the terms in each formula which corresponded to each other. He showed that the A_1/n term of Koga's formula was equivalent to Rosa's mutual inductance correction. Koga had previously derived a non-series version of the A_1/n expression by applying Euler's summation formula to an asymptotic series expression. (It is important to note, that Euler's summation formula is an

approximation, although the actual error is deterministic and can be made vanishingly small.) Grover refined Koga's expression slightly, and presented it as:

$$H(N) = \left[\ln(2\pi) - \frac{3}{2} \right] - \frac{1}{6N} \ln(N) - \frac{0.330842}{N} - \frac{1}{120N^3} + \frac{1}{504N^5} \quad (8)$$

From this, it is obvious that the limiting value of $H(\infty)$ is simply $\ln(2\pi)-3/2$, or 0.3378770664 which agrees exactly with the value produced by Wynn's Epsilon algorithm. Interestingly, the limiting value of $H(\infty)$ had never been derived analytically prior to the publication of Grover's 1929 paper. Prior to that, the value had only been estimated.

Use of formula (8) eliminates any accumulated roundoff errors when calculating $H(N)$ for large values of N . In comparing it with the series calculation derived above, I have found that for large values of N , both formulae (7) and (8) agree to ten decimal places. This is a good check to show that using double precision in the computer program calculations of formula (7) is sufficient to ensure ten decimal place accuracy. However, Grover notes that the coefficient 0.330842 for the $1/N$ term varies for different values of N , and the value given is "indicated as correct" for $N>3$. I have not investigated this any further, but have noted that using this formula for small values of N , it is in agreement with formula (7) to about five decimal places. For $N=1$, it is evident that the correction must be zero, while formula (8) produces a value of 0.0006858601 , and for $N=2$, the correct value is $\ln(1/4)+3/2$, or 0.1137056389 , while formula (8) produces a value of 0.1137141387 . The agreement improves however, as N increases:

- For $N>5$, the agreement is to 6 decimal places,
- For $N>10$, the agreement is to 7 decimal places,
- For $N>100$, the agreement is to 8 decimal places,
- For $N>500$, the agreement is to 9 decimal places,
- For $N>5000$, the agreement is to 10 decimal places.

For $5000<N<10^7$, the values continue to agree to at least ten decimal places. The largest N that was calculated for this investigation was 10^7 . This confirms that the accumulated roundoff error in the series calculation does not cause any problems with calculations to ten decimal places when using double precision operations. Formulae (7) and (8) complement each other nicely, as (7) is accurate for $N<5000$, and (8) is accurate and faster for $N>5000$. Used together, they can provide an accurate set of standard values to aid in producing a simpler fitting function.

In Grover's 1929 paper, he also discusses the limitations to the accuracy Rosa's corrections. Rosa's corrections neglect certain "curvature terms" [4](page 177), which in one example case of a 30 turn coil amounts to an error of 8×10^{-4} in the correction, which in turn affects the overall inductance calculation by 1.4×10^{-6} . From this, we may question the need to calculate these mutual inductance corrections to ten decimal places. However, it seems worthwhile to use the best accuracy readily available to minimize any other accumulated error that may later be compounded in the overall circuit design. Double precision operations are readily available in most popular programming languages.

Conclusion

Different methods have been presented for the calculation of $H(N)$ for both large and small values of N , including exact values for the important cases of $N=2$ and $N=\infty$. These independently derived formulae have provided a cross check to verify the accuracy of the calculations, allowing one to select the best method of calculation for any value of N . The table presented in Appendix A, and formula (7) used to calculate it, can be relied upon to give Rosa's correction accurate to ten decimal places for all values shown.

For maximum accuracy, when implementing the correction calculation in a computer program, the value 0 should be used for $N=1$, the value $\ln(1/4)+3/2$ should be used for $N=2$. The series formula (7) should be used for $N<5000$, and for speed and efficiency, formula (8) should be used for $N>5000$. And where required, the value $\ln(2\pi)-3/2$ should be used as the limiting value for $N=\infty$.

Appendix A

Using 18 i-terms in formula (7), the following data were generated from a BASIC program using double precision operations:

N	H(N)	N	H(N)	N	H(N)	N	H(N)
-----	-----	-----	-----	-----	-----	-----	-----
1	0.0000000000	31	0.3087421581	110	0.3277474630	3000	0.3373219874
2	0.1137056389	32	0.3094872828	120	0.3284707484	4000	0.3374487704
3	0.1662612544	33	0.3101921816	130	0.3290916932	5000	0.3375269915
4	0.1972758804	34	0.3108601179	140	0.3296309965	10000	0.3376904765
5	0.2179946299	35	0.3114940064	150	0.3301040761	20000	0.3377779952
6	0.2329272589	36	0.3120964588	160	0.3305226607	30000	0.3378087664
7	0.2442585094	37	0.3126698225	170	0.3308958371	40000	0.3378246427
8	0.2531838656	38	0.3132162133	180	0.3312307589	50000	0.3378343836
9	0.2604160704	39	0.3137375440	190	0.3315331373	100000	0.3378545698
10	0.2664081058	40	0.3142355480	200	0.3318075895	200000	0.3378652405
11	0.2714624996	41	0.3147118000	220	0.3322871556	300000	0.3378689572
12	0.2757894688	42	0.3151677343	240	0.3326925570	400000	0.3378708646
13	0.2795399058	43	0.3156046599	260	0.3330400561	500000	0.3378720306
14	0.2828250920	44	0.3160237741	280	0.3333414450	750000	0.3378736191
15	0.2857290007	45	0.3164261742	300	0.3336054904	1000000	0.3378744330
16	0.2883162583	46	0.3168128680	350	0.3341423106	1500000	0.3378752657
17	0.2906374592	47	0.3171847829	400	0.3345535170	2000000	0.3378756919
18	0.2927328100	48	0.3175427732	450	0.3348791769	2500000	0.3378759520
19	0.2946346882	49	0.3178876282	500	0.3351438457	3000000	0.3378761276
20	0.2963694754	50	0.3182200773	550	0.3353634385	3500000	0.3378762543
21	0.2979588959	55	0.3197182681	600	0.3355487377	4000000	0.3378763503
22	0.2994210101	60	0.3209898104	650	0.3357073161	4500000	0.3378764255
23	0.3007709619	65	0.3220835976	700	0.3358446535	5000000	0.3378764861
24	0.3020215503	70	0.3230352589	750	0.3359648160	7500000	0.3378766705
25	0.3031836696	75	0.3238713981	800	0.3360708861	8000000	0.3378766939
26	0.3042666549	80	0.3246122994	850	0.3361652448	8500000	0.3378767146
27	0.3052785531	85	0.3252737097	900	0.3362497611	9000000	0.3378767331
28	0.3062263403	90	0.3258680487	950	0.3363259233	9500000	0.3378767497
29	0.3071160952	95	0.3264052486	1000	0.3363949316	10 ⁷	0.3378767647
30	0.3079531406	100	0.3268933516	2000	0.3370782367	∞	0.3378770664

Appendix B

Following is the program code used to generate the table of $H(N)$ in Appendix A.

```
// Program 1
// Calculate and print the values of H(N)
Dim m,j,k,N,dN As Integer
Dim Hj,x, H(1,200) As Double
j=0
N=1
dN=1
//calculate 120 values
while j<120
  H(0,j)=N //Array indices are zero based
  Hj=0.0
  for m=1 to N-1
    Hj=Hj+dGMD(m)*(n-m)
  next
  H(1,j)=Hj*2/N
  Print right("      "+str(N),6)+" "+format(H(1,j),"0.0000000000")
  j=j+1
  //Change the increment for larger values of N
  if N >= 50 then dN=5
  if N >= 100 then dN=10
  if N >= 200 then dN=20
  if N >= 300 then dN=50
  if N >= 1000 then dN=1000
  if N >= 5000 then dN=5000
  if N >= 10000 then dN=10000
  if N >= 50000 then dN=50000
  if N >= 100000 then dN=100000
  if N >= 500000 then dN=250000
  if N >= 1000000 then dN=500000
  N=N+dN
wend
// At completion of this program, array H contains pairs of [N, H(N)]
// N values are included in the array, since the increment is not constant
End

// Function dGMD (used by Program 1 and Program 2)
Function dGMD(j As Integer) As Double
dim i,Nterms As Integer
dim k,x,N,Ni,Y As Double
//Nterms is Number of terms to use for series
Nterms=18
if j=1 then
  // Use the analytical expression for dGMD(1)
  return Log(1/4)+3/2
else
  N=j+0.0
  N=N*N
  Ni=N
  Y=0
  for i=1 to Nterms
    x=i+0.0
    k=(x+1)*(x+x)*(x+x+1)*Ni
    //Skip out of loop if term becomes vanishingly small
    if k>1e100 then exit
    Y=Y+1/k
    Ni=Ni*N
  next
  return Y
end if
End
```

Following is the program code using Wynn's Epsilon algorithm to estimate the limiting value of $H(\infty)$.

```
//Program 2
//Calculates the limiting value of  $H(\infty)$  using Wynn's Epsilon Algorithm
Dim i,m,j,k,N,dN,Nterms As Integer
Dim Hj,x As Double
dim S() As Double
//Number of terms to use in epsilon algorithm:
Nterms=8
j=0
N=3
dN=10
Window1.Refresh
//calculate number of terms given by Nterms, but don't go higher than 900000
while j<Nterms+1 and N<=900000
  if UserCancelled then exit
  H(0,j)=N
  Hj=0.0
  for m=1 to N-1
    x=dGMD(m)*(n-m)/n
    //skip out of loop if the next iteration is insignificant
    if x<1e-30 then exit
    if UserCancelled then exit
    Hj=Hj+x
  next
  H(1,j)=Hj*2
//Print out the values of n and H(n)
  Print right("          "+str(N),9)+" "+format(H(1,j),"0.0000000000")
  j=j+1
  if (j mod 20 =0) or (N>10000) then WindRefresh(j)
  N=N*6
wend
//call Epsilon to find the limit of the series
redim S(j)
Nterms=j
for i=0 to Nterms
  s(i)=h(1,i)
next
Hj=Epsilon(Nterms,S)
Print right("          "+str(N),9)+" "+format(Hj,"0.0000000000")
```



```

Function Epsilon(N As integer,S() As double) As Double
//Epsilon algorithm used by Program 2
//Finds the limit of a series when only the first N+1 terms are known
//This version has been ported to BASIC from a Fortran subroutine written by
//Todor Mishonov & Evgeni Penev
//and which appeared in an appendix to their technical paper:
//Thermodynamics of Gaussian fluctuations and paraconductivity in layered superconductors
//Int. J. Mod. Phys. B 14(32):3831--3879.
//
dim i,j,k As integer
dim rLimit,A_max As Double
dim i_Pade As integer,k_Pade As integer, err As double
dim A() As Double
// Check for invalid data
if N>UBound(s) or N<2 then return 0.0
Redim A(N) //Auxiliary row of e-table
rLimit=S(N) //S is sequential row of e-table
err=abs(s(N)-S(N-1)) //error value
i_Pade=N
k_Pade=0
for i=0 to N //zero array
    A(i)=0
next
A_max=0
k=1
Do
    if (n-2*k+1)<0 then exit
    //Update auxiliary row of e-table by applying "cross rule"
    for i=0 to N-2*k+1
        if s(i+1)<>S(i) then
            A(i)=A(i+1)+1/(s(i+1)-S(i))
        else
            A(i)=A(i+1)
        end if
    next
    if n-2*k<0 then exit
    //Update sequential row of e-table by applying "cross rule"
    for i=0 to n-2*k
        if A(i+1)<>A(i) then
            S(i)=S(i+1)+1/(A(i+1)-A(i))
        else
            S(i)=S(i+1)
        end if
    //Check for convergence based on A_max
    If abs(A(i))>A_max then
        A_max=abs(A(i))
        rLimit=S(i)
        k_Pade=k
        i_Pade=i+k_Pade
        err=1/A_max
        E_err=err
        if S(i+1)=S(i) then return rLimit
    end if
next
k=k+1 //increment row index
loop
return rLimit

```

Appendix C

Derivation of the Sum of the Series: $1/12 + 1/60 + 1/168 + 1/360 + 1/660 + \dots$

To sum an infinite series, one approach is to determine an expression for the partial sums S_n of the series, and then take the limit as $n \rightarrow \infty$ to find the sum.

Given a series $\sum a_k$, the partial sum $S_n(a)$ is defined as the sum of its first n terms or:

$$S_n(a) = \sum_{k=1}^n a_k$$

The letter in parentheses indicates which series we are summing. (There will be more than one, presently.) If the series is convergent, then the sum $s(a)$ is given by:

$$s(a) = \lim_{n \rightarrow \infty} S_n(a) = \lim_{n \rightarrow \infty} \sum_{k=1}^n a_k$$

The problem then, is to find an expression for $S_n(a)$.

In the subject series the terms are given by:

$$a_k = \frac{1}{(2k)(k+1)(2k+1)}$$

Using the method of partial fractions, this expression can be broken into three terms:

$$a_k = \frac{1}{2k} + \frac{1}{2(k+1)} - \frac{2}{2k+1}$$

and then the series written as the sum of three new sub-series:

$$\sum a_k = \sum b_k + \sum c_k + \sum d_k$$

where

$$\sum b_k = \sum \frac{1}{2k} = \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{6} + \frac{1}{8} + \dots \right)$$

$$\sum c_k = \frac{1}{2} \sum \frac{1}{k+1} = \frac{1}{2} \left(\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots \right)$$

$$\sum d_k = -2 \sum \frac{1}{2k+1} = -2 \left(\frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \dots \right)$$

Note that all three sub-series bear similarities to the harmonic series which is:

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots$$

The first sub-series, $\sum b_k$, is composed of the terms of the harmonic series divided by 2. The second sub-series, $\sum c_k$, (ignoring the constant $1/2$) is the harmonic series with the first term omitted. The last sub-series, $\sum d_k$, (ignoring the constant -2) is composed of the odd terms (omitting the first) of the harmonic series. (It is interesting to note that while the original series $\sum a_k$ converges, the three sub-series diverge individually, and the harmonic series also diverges.) We can define the partial sums of these sub-series in terms of partial sums of the harmonic series. The partial sums of the harmonic series are given by the harmonic numbers \mathcal{H}_n [5] where:

$$\mathcal{H}_n = \sum_{k=1}^n \frac{1}{k} = \text{Ln}(n) + \frac{1}{n} + \gamma_n$$

where γ_n is the n^{th} partial sum of the series which converges to Euler's *constant* γ as $n \rightarrow \infty$ (not to be confused with Euler's *number* e which is the base of natural logarithms).

Therefore, in terms of harmonic numbers, we can express $S_n(b)$, $S_n(c)$ and $S_n(d)$ as:

$$S_n(b) = 1/2 \mathcal{H}_n$$

$$S_n(c) = 1/2 (\mathcal{H}_{n+1} - 1)$$

$$S_n(d) = -2 (\mathcal{H}_{2(n+1)} - \mathcal{H}_{n+1} - 1)$$

Then, the partial sum of the original series $\sum a_k$ is:

$$\begin{aligned} S_n(a) &= S_n(b) + S_n(c) + S_n(d) = 1/2 \mathcal{H}_n + 1/2 (\mathcal{H}_{n+1} - 1) - 2 (\mathcal{H}_{2(n+1)} - \mathcal{H}_{n+1} - 1) \\ &= 1/2 \mathcal{H}_n + 3/2 \mathcal{H}_{n+1} - 2 \mathcal{H}_{2(n+1)} + 3/2 \end{aligned}$$

Replacing the harmonic numbers with their logarithmic form:

$$S_n(a) = 1/2 (\text{Ln}(n) + 1/n + \gamma_n) + 3/2 (\text{Ln}(n+1) + 1/(n+1) + \gamma_{n+1}) - 2 (\text{Ln}(2n+2) + 1/(2n+2) + \gamma_{2n+2}) + 3/2$$

Collecting terms:

$$S_n(a) = [1/2 \text{Ln}(n) + 3/2 \text{Ln}(n+1) - 2 \text{Ln}(2n+2)] + [1/n + 1/(2n+2)] + [\gamma_n + \gamma_{n+1} - 2\gamma_{2n+2}] + 3/2$$

Finally, the sum of the series $S(a)$ is the limit as $n \rightarrow \infty$ of $S_n(a)$. Hence:

$$S(a) = \lim_{n \rightarrow \infty} S_n(a) = \lim_{n \rightarrow \infty} [\frac{1}{2} \text{Ln}(n) + \frac{3}{2} \text{Ln}(n+1) - 2 \text{Ln}(2n+2)] + [\frac{1}{n} + \frac{1}{2n+2}] + [\gamma_n + \gamma_{n+1} - 2\gamma_{2n+2}] + \frac{3}{2}$$

Dealing with the easy parts first, at the limit, the $1/n$ terms inside the second set of brackets reduce to zeros. The gamma terms inside the third set of brackets converge to Euler's constant, and cancel each other out. This leaves only the log terms inside the first set of brackets plus the constant term:

$$S(a) = \lim_{n \rightarrow \infty} [\frac{1}{2} \text{Ln}(n) + \frac{3}{2} \text{Ln}(n+1) - 2 \text{Ln}(2n+2)] + \frac{3}{2}$$

Combining the log terms inside a single log function:

$$S(a) = \lim_{n \rightarrow \infty} \text{Ln} \left[\frac{\sqrt{n}(n+1)\sqrt{n+1}}{(2n+2)^2} \right] + \frac{3}{2}$$

$$S(a) = \text{Ln} \left[\lim_{n \rightarrow \infty} \frac{\sqrt{n}(n+1)\sqrt{n+1}}{(2n+2)^2} \right] + \frac{3}{2}$$

Taking out common factors:

$$S(a) = \text{Ln} \left[\lim_{n \rightarrow \infty} \frac{\sqrt{n}(n+1)\sqrt{n+1}}{4(n+1)^2} \right] + \frac{3}{2}$$

$$= \text{Ln} \left[\lim_{n \rightarrow \infty} \frac{\sqrt{n}\sqrt{n+1}}{4(n+1)} \right] + \frac{3}{2}$$

$$= \text{Ln} \left[\lim_{n \rightarrow \infty} \frac{\sqrt{n}}{4\sqrt{n+1}} \right] + \frac{3}{2}$$

$$= \text{Ln} \left[\lim_{n \rightarrow \infty} \frac{1}{4\sqrt{1+\frac{1}{n}}} \right] + \frac{3}{2}$$

$$= \text{Ln} \left[\frac{1}{4\sqrt{1}} \right] + \frac{3}{2}$$

and finally:

$$S(a) = \text{Ln} \left[\frac{1}{4} \right] + \frac{3}{2}$$

Appendix D

References

- 1 Rosa, E.B. & Grover, F.W., *Formulas and Tables for the Calculation of Mutual and Self Inductance*, Bulletin of the Bureau of Standards, Vol. 8, No. 1, (1911)
- 2 Grover, F.W., *Inductance Calculations, Working Formulas and Tables*, D. Van Nostrand Company, Inc. (1946)
- 3 Bureau of Standards, Scientific Paper, **169**, p. 166-170 (1912).
- 4 Grover, F.W., *A Comparison of the Formulas for the Calculation of the Inductance of Coils and Spirals Wound with Wire of Large Cross Section*, Paper No. **90**, Bureau of Standards Journal of Research, Vol. 3, p. 163-190 (1929)
- 5 Bonar, Daniel D. & Khoury, Michael, *Real Infinite Series*, The Mathematical Association of America, p. 72-73 (2006)

Note: It was later discovered that [3] is a reprint of [1], containing exactly the same information.