Rosa's mutual inductance correction for the round-wire solenoid.

By David Knight

Version 1.03, (minor update) 3rd August 2012.

1st pdf version with machine-optimised empirical formula (version 1.00): 3rd April 2010.
Prior HTML article with optimised version of Grover's semi-empirical formula: 11th April 2009.
Please check the author's website to ensure that you have the most recent version:
http://www.g3ynh.info/ .

Abstract

Rosa's mutual inductance correction, used for obtaining round-wire solenoid inductance from the corresponding current-sheet inductance, is difficult to calculate and is traditionally given in tabular form. This article describes accurate approximation formulae which have been fitted to the tabulated data and to a precise machine-calculation given by Robert Weaver. A formula originally given in 2006, before the new calculations became available, is found to fulfil its original accuracy claims but is improved by further parameter adjustment. A truncated series formula given by Grover in 1929 is accurate for large N, but inaccurate for small N. Restoration of the N=1 boundary condition and extension of the series provides a one-line formula which is accurate to within ±0.000 000 013.

Introduction

The inductance of a helical coil is most readily calculated by using the hypothetical current-sheet solenoid as a basis, and by applying corrections for the difference between realistic wire and a conducting sheet of infinitesimal thickness. This approach was developed by Edward B Rosa of the American National Bureau of Standards (NBS)[1] in 1906 and remains applicable in situations in which it is sufficient to calculate low-frequency inductance to an accuracy of better than about 1 parts in 1000 (i.e., almost universally). Rosa's expression for the inductance of a round-wire solenoid, written in the SI form (i.e., rationalised mks with permeability μ shown explicitly) becomes:

$$L = L_s - \mu\, r\, N\, (k_s + k_m) \qquad \text{[Henrys]}$$

where L_s is the inductance of the corresponding current-sheet, r is the solenoid radius, N is the number of turns, and k_s and k_m are Rosa's correction coefficients. Note however, that the symbols used here differ from those used in the references cited. Specifically, k_s is elsewhere given the symbol A or G, and k_m is given the symbol B or H (the notation has been changed, because A, B, G and H now have fixed meanings in an electrical context). Notice also that k_s and k_m are dimensionless. They are converted into Henrys by the factor μr outside the bracket.

k_s (A, G) is a correction for the difference between the self-inductance of a round-wire loop and that of a single-turn current-sheet. It can be obtained, to a good first-order approximation, using relatively simple formulae. Rosa's expression for the low-frequency case (including internal

1 **Formulas and Tables for the Calculation of Mutual and Self Induction**. E B Rosa & F W Grover, 3rd edition 1916, with 1948 corrections. Bureau of Standards Scientific Paper No. 169 [**BS Sci.169**]. p122. [available from http://g3ynh.info/zdocs/magnetics/].

inductance) is:

$k_{s(dc)} = (5/4) - \ln(2p/d)$

where p is the winding pitch distance and d is the wire diameter (and ln means $\log_e$). Correction of this formula for high frequencies is discussed in another article[2].

k_m (B, H) is a correction for the difference between the mutual inductance of pairs of round-wire loops and the mutual inductance of pairs of current-sheet loops, and depends only on the number of turns N . In contrast to the self-inductance term however; the calculation of k_m is complicated, because the coefficient is the sum of incremental contributions obtained by considering *every* pair of turns in a coil (i.e., adjacent and non-adjacent). In the days before electronic computers, such calculations were laborious; and so the NBS issued a table, and corrected it from time to time. An early version (4 decimal places) is given in BS Science Paper 169 [3]. A re-calculation using improved techniques (to 5 places) was given in by Grover in 1929 [4], and proves to be accurate to within one count in the last place. The most widely known version however is given in Grover's 1946 monograph[5], and appears to have been obtained by rounding the 1929 table to 4 places. In some cases, the rounding in Grover's 1946 table has gone the wrong way (because it requires at least 6 digits to produce data rounded to 4 digits without bias). Hence Grover inadvertently produced a table with errors of 1 count in the 4th place. More recently (2008) the calculation procedure has been re-investigated by Robert Weaver[6], leading to a set of tabulated values precise to 10 decimal places. The latter study is the reason why we now know the accuracies of the other tables.

The information provided by Bob Weaver includes example algorithms, an Open Office Basic macro function, and a description of the geometric-mean-distance (GMD) method. This might encourage programmers and spreadsheet users to calculate k_m for themselves (i.e., it renders the practice of interpolating Grover's table obsolete), but a warning about execution speed is in order. When running code using a program interpreter, the calculation time for coils with many thousands of turns can range from minutes to hours using a fast modern computer. Hence reproduction of Bob's full table, using (say) a spreadsheet, is not a task to be undertaken lightly.

In view of the computationally intensive nature of the GMD method, and the fact that k_m represents only a small correction to the overall inductance of a coil, there is an obvious need for a fast and simple calculation method. A one-line empirical formula, capable of reproducing the values in Grover's 1946 table within one count in the 4th place, was made available by this author (DWK) in 2006 and is known to have been used in some programs and calculations. This study shows that the original accuracy claims were correct, and that there is consequently no need for remedial action (although further optimisation is possible). That formula however, is now superseded due to a re-investigation of a series formula given in Grover's 1929 paper.

Bob Weaver, incidentally, has also found an analytical expression for the N=2 case[7]. This may be of theoretical interest, or as an alternative to the tabulated value:

When $N = 2$, $k_m = \ln(1/4) + 3/2 = 0.113\ 705\ 6389$

---

2 **Solenoid inductance and impedance calculation**, D W Knight. section 7. http://g3ynh.info/zdocs/magnetics/

3 **BS Sci. 169** (already cited), p199.

4 **Comparison of Formulas for the Calculation of Inductance of Coils and Spirals Wound with Wire of Large Cross-Section**. F W Grover, JBS Vol 3. 1929. [**RP90**]. p190. [available from g3ynh.info/zdocs/magnetics/]

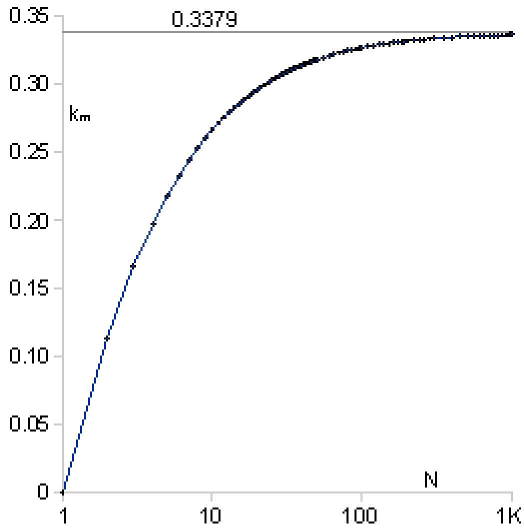
5 **Inductance Calculations: Working Formulas and Tables,** F W Grover, 1946 and 1973. Dover Phoenix Edition 2004. ISBN: 0 486 49577 9. [**Grover 1946**] p150, table 39.

6 **Investigation of E.B. Rosa's Round Wire Mutual Inductance Correction Formula**. Robert Weaver, July 2008. [**Weaver 2008**] [available from g3ynh.info/zdocs/magnetics/]

7 **Weaver 2008**, Appendix C.

Fit to the tabulated data

A formula for calculating k_m was first produced by this author (DWK) in 2006, before Bob Weaver's study was carried out. Grover's 1929 paper (RP90) was also not known to this author at the time. The generating function for the tabulated data is moreover not well described in the 1946 monograph or the 1916 paper, and Bob's later reconstruction of it is a commendable piece of detective work. The approach adopted therefore, was to obtain a curve corresponding to the available data. The values in the 1916 and 1946 tables (given to 4 decimal places in both instances) were found to differ by as much as 0.0011; but the later document was assumed to be authoritative.



A graph of the data in Grover's 1946 table is shown on the right. The actual numbers are given in the accompanying Open-Document spreadsheet **Rosa_km.ods** (sheet 3) The curve tends to a limit of 0.3379 for very large N , but if we regard the asymptotic value as a matter of normalisation, it appears that the overall shape is that of a relatively simple function. Various candidates were tried, and it was found that the best crude fit was given by a function having the (un-normalised) form:

$f_0(N) = 1 - 1/N$

This expression gives half of its maximum value when N=2 , whereas the value in the table for N=2 is about 34% of the maximum, and so the gradient of the function is obviously incorrect for small N. The gradient is however changing rapidly around N=1 ; and so a correction can be obtained by including an offset, i.e., by sliding the starting point along the candidate curve until a suitable gradient is found (analogous to the process of using a French-curve to join-up the points on a hand-drawn graph). Hence:

$f_1(N) = 1 - 1/(N+k)$

This new function goes to 1-1/(1+k) when N=1 , whereas a function which goes to zero is required, and so the zero-crossing point must be restored by subtracting the value of the function at N=1 , i.e.;

$f_2(N) = [1 - 1/(N+k)] - [1 - 1/(1+k)]$

which simplifies to:

$f_2(N) = [1/(1+k)] - [1/(N+k)]$

This function goes to 1/(1+k) as $N\rightarrow\infty$, but it can be normalised to vary between 0 and 1 by multiplying it by 1+k , i.e.;

$f_3(N) = 1 - (1+k)/(N+k)$

Multiplying by $k_{m\infty} = 0.3379$ (the $N\rightarrow\infty$ value in the table) then gives a first candidate for a fitting function:

$k_m' = k_{m\infty} [1-(1+k)/(N+k)]$

With $k=1.07$, this function was found to fit the data within about 3%. This is remarkable for an expression with a single adjustable parameter, but nowhere-near good enough to reproduce the table. Further improvement requires additional parameters, and there a various options in this respect. The approach chosen was to add a correcting function which is fixed to zero at the extreme ends of its range (i.e., when $N=1$ and when $N\rightarrow\infty$). A suitable candidate is:

$\Delta f(N) = [(k_1/N) + (k_2/N^2) + (k_3/N^3) + ...] \ln(N)$

where the ln(N) multiplier forces the function to zero when $N=1$, and the terms in the series in square brackets all go to zero as $N\rightarrow\infty$. Hence:

$k_m' = k_{m\infty} [1-(1+k)/(N+k)] + [(k_1/N) + (k_2/N^2) + (k_3/N^3) + ...] \ln(N)$

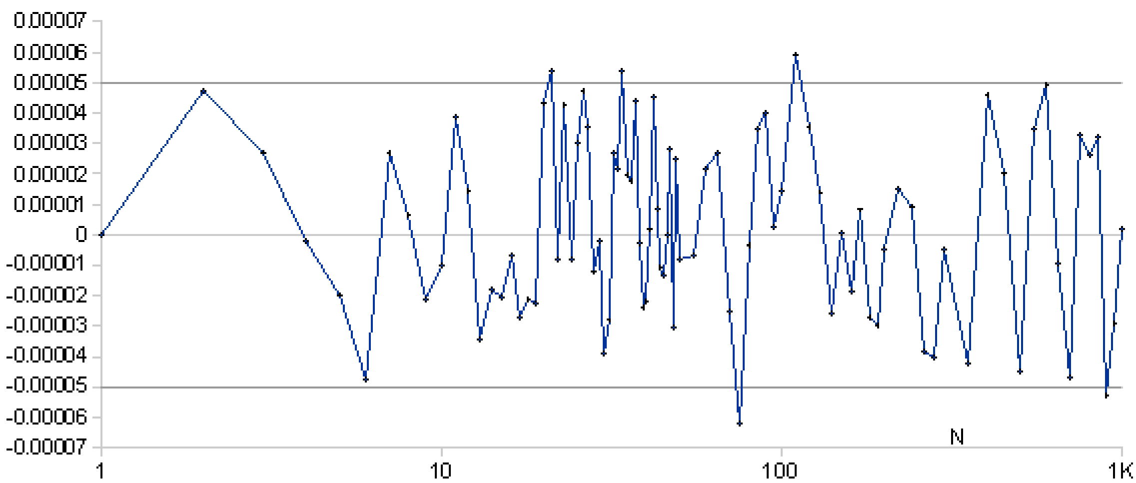
With the N^2 and higher terms excluded by setting $k_2=0$, $k_3=0$, etc.; by varying only k and k_1 , it was easily possible to fit the table with no residual (observed-calculated) greater than 0.0002. Then, by allowing k_2 to deviate from zero, it was found that the data could be fitted with no residual greater than 0.00007. Thus the formula acquired the form:

$k_m = k_{m\infty} [1-(1+k)/(N+k)] + [(k_1/N) + (k_2/N^2)] \ln(N)$

Finally it was allowed that the value of $k_{m\infty}$ could lie between 0.337850 and 0.337949 , since anything in this range will round to 0.3379 , and this small latitude was used to distribute the residuals evenly above and below the value calculated by the function. The resulting formula is shown below (the parameter 0.9754 is $1+k$, where $k = -0.0246$):

$k_m = 0.337883[1 - 0.9754/(N-0.0246)] + [(-0.16725/N)+(0.0033/N^2)] \ln(N)$

This reproduces the values in Grover's table 39 [8] with a maximum difference of ±0.000 062. Details of the fit are given in the spreadsheet file **Rosa_km.ods** (sheet 3) and the graph of residuals (tabulated-calculated) is shown below.



The difference consists entirely of noise, this being due to the rounding errors incurred by truncation of the data to 4 decimal places. The bars at ±0.00005 mark the limits for perfect

8 **Grover 1946**, p150.

reproduction of the table; i.e., any residual which falls outside this range corresponds to a discrepancy of 1 in the last place when the value returned by the function is rounded to 4 places. As can be seen, there are 5 such discrepancies. It was argued at the time that, since the fitting function is smooth and of low order (i.e., it cannot change direction suddenly); then its effect will have been to average the noise. Thus it was deduced that there must be some errors of 1 digit in the last place of the data, which means that that the fitting function is at least as reliable as the table. With no means of arbitration (and bearing in mind that such esoteric issues have negligible effect on actual inductance calculations), the 2006 work came to its natural conclusion.

When Bob Weaver's calculations and Grover's 1929 paper became available, it was discovered that the three discrepancies for N of 75, 110 and 900 were due to errors in the data, and the discrepancies at 21 and 34 were due to the function. Bearing in mind that the fitting criterion is that of minimum runout, rather than least-squares, the error at 75 had led to an adjustment which caused the residuals at 21 and 34 to creep above the limit. Interestingly, the $k_{m\infty}$ value of 0.337883 , averaged from the data, is remarkably close to the analytical result[9] :

$$k_{m\infty} = \ln(2\pi) - 3/2 = 0.3378770664$$

By using the analytical value for $k_{m\infty}$ and adding an extra term, the formula can be improved. The version shown below has been fitted to data produced by the exact GMD calculation method, and has a maximum absolute error of ±0.000 0011 for integer N (see spreadsheet: **Rosa_km.ods**, sheet 2).

$$k_m = [\ln(2\pi)-3/2] \left[1 - \frac{1-0.017111}{N-0.017111} \right] + \ln(N) \left[\frac{-0.16641}{N} + \frac{0.00479}{N^2} + \frac{0.001772}{N^3} \right]$$

The formula is given below as a function, written in Open Office Basic. Note that N is dimensioned as double-precision because there is no requirement that it should be an integer.

```
Function Kmeo (Byval N as double) as double
'Rosa's round-wire solenoid mutual inductance correction. D W Knight, April 2010.
'Optimised empirical formula. Max error is +/-0.000 0011
if N<1 then
  Kmeo=0
else
  Kmeo=(log(2*pi)-1.5)*(1-0.982889/(N-0.017111))   _
  +(-0.16641/N +0.00479/N^2 +0.001772/N^3)*log(N)
endif
end function
```

9 **RP90**, p176.

Optimisation of Grover's 1929 Series formula

In RP90, Grover gives a series formula for k_m which may be written as follows[10]:

$$k_m = [\ln(2\pi)-3/2] - \frac{\ln(N)}{6N} - \frac{0.330842}{N} - \frac{1}{120N^3} + \frac{1}{504N^5}$$

(notice that the coefficient of ln(N)/N is -1/6 =-0.16667 , which is very close to the value obtained from the fitting procedure). This formula is extremely accurate for large N (it converges with the 10-place GMD calculations), but it is inaccurate for $N < 4$ and does not return zero for $N=1$. Given the simplicity of the expression, the grounds for using it are compelling; with perhaps a reversion to tabulated values for smaller N . Piecewise and spot solutions are unsatisfactory however; particularly because there is no reason why the number of turns on a solenoid should have to be an integer, and the function above is valid for any positive non-integer argument. It is preferable therefore to optimise the function, and this turns out to be remarkably straightforward. Firstly; note the general form, which is:

$$k_m = [\ln(2\pi)-3/2] - \frac{\ln(N)}{6N} + \frac{k_1}{N} + \frac{k_3}{N^3} + \frac{k_5}{N^5} + \dots\dots$$

It is clear that terms of N^{-7}, N^{-9} and so on can be added, and since such terms will vanish rapidly with increasing N , they will only affect the expression in its region of greatest inaccuracy. Also, by fitting the 10-place calculated data for $N=1$ to 4 or so to a simple smooth curve and comparing this against the formula; it can be shown that the series, as truncated, oscillates for non-integer values of N in this region. This, without revisiting the underlying theory, is evidence to the effect that the formula can be made accurate to an arbitrary degree by including extra terms. We do not know the theoretical values for the coefficients of these new terms; but then again, for the purpose of obtaining a convenient expression, we would not necessarily want to use them. Instead, we can allow one or more adjustable coefficients with a view to truncating the series once an acceptable degree of accuracy has been obtained.

Now observe what happens when we put $N=1$ into the formula. We get:

$$k_{m1} = [\ln(2\pi)-3/2] + k_1 + k_3 + k_5 + \dots.$$

This should add up to zero, but for the original formula it does not. This boundary error can be eliminated by ending the series with a closing term; i.e., we first extend the series up to some order p (say). Thus:

$$k_{m1} = \ln(2\pi)-3/2 + k_1 + k_3 + k_5 + \dots. + k_p$$

We then define $k_{m1} = 0$, so that:

$$k_p = -[\ \ln(2\pi)-3/2 + k_1 + k_3 + k_5 + \dots.\ ]$$

Adjustable terms between 5 and p can now be inserted until the required accuracy is achieved.

10 **RP90**, P176

Using the strategy just outlined; it was found that the inclusion of just one variable term, and an adjustment of k_1 from 0.330842 to 0.33084236 , brought the maximum absolute error to within ±0.000 000 013 . The resulting formula is given below, and the comparison between it and the 10-place data is given in the spreadsheet file: **Rosa_km.ods** (sheet 1). The stated accuracy is only guaranteed for integer N , because there is still some residual oscillation on the interval between N=1 and N=5 ; but the error is unlikely to extend into the sixth decimal place for non-integer N, and since accuracy to 4 places is generally considered adequate, any further extension of the series appears unwarranted. Thus we have an expression which is both simple enough for direct spreadsheet entry, and good enough to be acceptable in general-purpose computer programs.

Formula for Rosa's mutual inductance correction parameter:
Maximum error: ±0.000 000 013 for integer N.

$$k_m = \ln(2\pi) - 3/2 - \ln(N)/(6N) - 0.33084236/N - 1/(120N^3) + 1/(504N^5) - 0.0011925/N^7 + 0.000507/N^9$$

Where:

$$0.000507000 = -[\ln(2\pi) - 3/2 - 0.33084236 - 1/120 + 1/504 - 0.0011925]$$

A Basic implementation is as follows:

```
Function KMGO (Byval N as double) as double
'Rosa's round-wire solenoid mutual inductance correction. D W Knight, April 2010
'Optimised version of Grover's 1929 formula. Max error is +/-0.000 000 013
if N<1 then
  KMGO=0
else
  KMGO=log(2*pi)-1.5 -log(N)/(6*N) -0.33084236/N -1/(120*N^3) +1/(504*N^5) _
   -0.0011925/N^7 +0.000507/N^9
endif
end function
```

* * *

The Basic functions shown can be copied from the accompanying spreadsheet: **Rosa_km.ods**. To open the macro editor (assuming the use of Open Office version 3), use the 'Tools > Macros > Organise Macros > OOo Basic' menu and navigate to the folder ' L_funcs ' .