THE MIGRATION OF THE POLES OF THE PADE APPROXIMANT

by

MICHAEL F. PAS

MATHEMATICS

Submitted in Partial Fulfillment of the Requirements of the

University Undergraduate Fellows Program

1981 - 1982

Approved by:

Dr. G. D. Allen

APRIL 1982

TABLE OF CONTENTS

ABSTRACT

A rational function is defined as the ratio of two polynomials. The Pade approximant (P[M,N]) is a special kind of rational function. Basically, the Pade approximant of the power series expansion of a function f(z), ($z \in \mathbb{C}$) is formed by approximating the power series via the ratio of two polynomials (degree M in numerator, degree N in the denominator). The coefficients of the two polynomials which forms the ratio are called Pade coefficients.

When working with approximations,it is necessary to determine how accurate the approximations are. This is expecially true when working with Pade approximants since the denominator of the approximant has N zeros (N is the degree of the polynomial in the denominator). There is not a formula which bounds the error like the remainder term of the Taylor approximation. Once the poles of the Pade approximant are found, it is important to know how they migrate in the complex plane as the degrees of the numerator and denominator increase. If, as the degrees increase, the poles do not move away from the origin rapidly, then the Pade approximant of the given function is of little use because, P[M,N] becomes unbounded for small value of z near the poles.

It is helpful to find the disk centered at the origin, which contains no zeros of the denominator polynomial. This is accomplished by using a minimum modulus theorem. Inside the disk of radius, ρ, (i.e., $|z| < \rho$) the theorem guarantees no zeros of the denominator. This assures that f(z) remains bounded inside the disk of the radius, ρ.

A Fortran program was written to accomplish this task. Using Gaussian elimination, the Pade coefficients are computed. Knowing the

Pade coefficients of the denominator, the subroutine MULLER (1) is used to find the roots of the polynomial (MULLER finds complex roots if they exist). The values of the zeros are evaluated in f(z) to determine the accuracy of MULLER. The modulus of the zero is computed to determine its distance from the origin of the complex plane. The minimum modulus theorem is incorporated into another subroutine. The output is printed in tabular form.

As is common in research, the mistakes and unexpected results are as important, if sometimes not more important, than the expected results. This research project was no exception. The many mathematical and computer-related mistakes are fully described in the second part of this report.

SUBJECT

This research report investigates numerically the Pade approximant of several functions (i.e., sin(z), cos (z), e^z, e^{-z}), focusing in particular on the migration of the poles of the Pade approximant of these functions. A Fortran program has been written to compute the Pade coefficients of a given function. The routine MULLER is used to compute the zeros of the polynomial in the denominator (i.e., the poles of the Pade approximant). By computing the modulus of the poles, the distance from the origin of the complex plane is computed. A minimum modulus theorem is used to determine the radius of the disk in which no zero of the denominator polynomial are found. The many problems which arise from analyzing the Pade approximant numerically are also discussed, such as round-off error, maximum number of iterations required in the MULLER routine, and the restrictions imposed by using the reciprocal of the factorial coefficients. Possible solutions to some of these problems are suggested, but are not thoroughly investigated.

PURPOSE

Since mathematics is an approximation of the physical world, it is important for one who is trying to apply a particular branch of mathematics to know how accurate their approximations are and how closely the mathematics models the physical world. The Pade approximant, which is frequently used in nuclear physics, mathematics, and engineering, does not have a theoretical method for determining the accuracy of the approximant (unlike the Taylor series approximation, which contains the remainder term, $R_n(x)$.This investigation uses a numerical approach to the Pade approximant, focusing in particular on the troubling problem of the poles of the Pade approximant and the migration of the poles as the degree of the Pade approximant increases. The ability to predict the migration of the poles is essential to anyone who wishes to understand Pade approximants.

HISTORICAL BACKGROUND

In 1892, H. Padé, in his thesis entitled "Sur la représentation approache'e d'une function par des fractiones rationelles", compiled the first tables of coefficients for the Pade approximant of several power series, originally attributed to Frobenius (2). Presently, little attention has been given by mathematicians to the subject of Pade approximants. For instance, today there are very few theorems which address the usual questions pertaining to series approximation of functions, such as the criteria for convergence or divergence and the accuracy of the approximation, which have been rigorously explored in other types of series approximations such as Taylor and Fourier series. Lack of this information pertaining to the Pade approximant is unfortunate, because Pade approximants have proved to be very useful in providing quantitative information for many interesting problems in physics and chemistry (3).

THEORETICAL BACKGROUND

The usual form of a power series is

$$f(z) = a_0 + a_1 z + \dots = \sum_{n=0}^{\infty} a_n z^n .$$

The [M,N] Pade approximant for a formal power series in terms of the power series coefficients is defined by the following:

$$[M,N]\,(z) = \frac{\det \begin{vmatrix} a_{M-N+1} & a_{M-N+2} & \cdots & a_{M+1} \\ \vdots & \vdots & \ddots & \vdots \\ a_M & a_{M+1} & & a_{M+N} \\ \sum_{j=N}^{M} a_{j-N} z^j & \sum_{j=N-1}^{M} a_{j-N+1} z^j & \cdots & \sum_{j=0}^{M} a_j z^j \end{vmatrix}}{\det \begin{vmatrix} a_{M-N+1} & a_{M-N+2} & \cdots & a_{M+1} \\ \vdots & \vdots & \ddots & \vdots \\ a_M & a_{M+1} & & a_{M+N} \\ z^N & z^{N-1} & \cdots & 1 \end{vmatrix}}$$

where the power series of the numerator has degree M, and the degree of the denominator is N (4).

Although this is the formal definition of the [M,N] Pade approximant, it is absent of any intuitive feeling of how the Pade approximates a power series expansion of a function by the ratio of two polynomials. Also, in the formal definition, a method to be used to compute the zeros of the denominator of the Pade, i.e., the poles of the Pade, is not readily evident. To solve these problems, an alternate definition has been used. As one will observe, the second definition is longer than the previous definition, but is mathematically more simple.

The second definition of the Pade approximant uses the fact that Pade approximants are special types of rational functions. A rational function, R(z) is defined as the ratio of two polynomials,

$$R(z) = \frac{\sum_{n=0}^{M} a_i z^i}{\sum_{j=0}^{N} b_j z^j},$$

where the a_i's and b_j's $\in \mathbb{C}$ and z is a complex variable. This definition is essential since the Pade approximant of a function expanded via a power series are special types of rational functions.

A Pade approximant of an arbitrary function f(z) is:

$$f(z) = \sum_{n=0}^{\infty} a_n z^n \doteq \frac{\sum_{n=0}^{M} p_n z^n}{\sum_{n=0}^{N} q_n z^n}$$, where the p_n, $q_n \in \mathbb{C}$ called

Pade coefficients.

The relationship between the Pade coefficients completes the definition of the Pade approximant. Assume all terms in which z has degree less than M+N+1 are zero. Thus,

$$f(z)Q(z) - P(z) = 0 + d_{M+N+1} z^{M+N+1} + d_{M+N+2} z^{M+N+2} + \ldots.$$

Then,

$$f(z)Q(z) = (\sum_{k=0}^{\infty} a_k z^k)(\sum_{k}^{N} q_k z^k) = \sum_{k=0}^{\infty} \sum_{i=0}^{k} a_i q_{k-i} z^k$$

(taking the Cauchy product).

So,

$$f(z)Q(z) - P(z) = \sum_{k=0}^{\infty} \sum_{i=0}^{k} (a_i q_{k-i}) z^k - \sum_{k=0}^{m} p_k z^k.$$

From this equation, the relationship between the Pade coefficients is derived. The equations are:

$$\sum_{i=0}^{k} q_i a_k - p_k = 0, \quad \text{for } 1 \leq k \leq M \qquad \text{i.}$$

$$\sum_{i=0}^{k} q_i a_{k-i} = 0, \quad \text{for } 1+M \leq k \leq M+N \quad (5) \text{ ii.}$$

Thus, if one can compute the power series of a given function, f(z), then using the coefficients of the power series and equations i. and ii. from above the Pade approximant of the function can be easily computed. An example of computing the [2,2] and [3,3] approximants of $f(z) = e^{-z}$ is found in Appendix A.

It should be noted that there is not any a priori reason to assume the P[M,N] Pade approximant of a function f(z) properly approximates the function. At present time, there are not any convergence theorems in the literature to guarantee the Pade approximant converges to the function.

A readily apparent problem involving Pade approximants to approximate a function f(z), is the position of the zeros in the denominator of the Pade approximant. An analytic function such as $f(z) = e^z$ does not have any poles, but clearly the Pade approximant of this (and every) function expanded via a Pade approximant has poles. Around the poles of the Pade, the approximation becomes unbounded and, therefore, cannot approximate the function.

Since the poles of the Pade approximant are of great importance, to determine how good the Pade approximation is, it was necessary to chart the migration of the poles. That is, at what"rate" did the poles of a function, $f(z)$, move from the origin of the complex plane. If, as the values of M and N increased, the poles migrated away from the complex origin, then a radius of convergence, ρ, could be established such that, inside the disk of radius ρ, i.e., $|z| < \rho$, it is guaranteed that no zeros of the denominator are found. Then, in this disk, it is a good assumption that the Pade approximant estimates the function. Of course, this estimation is subject to verification by analytically evaluating the function.

To find the radius of the disk, ρ, in which we are guaranteed no zeros of the denominator polynomial are found. This is accomplished by using a minimum modulus theorem. First, it is necessary to define the $\binom{n}{m}$ binomial coefficients;

$$\binom{n}{m} = \frac{n!}{(n-m)!m!} .$$

Given a polynomial of degree $n \geq 1$ with complex coefficients

$$p(z) := z^n + a_{n-1}z^{n-1} + \ldots + a_1 z + a_0$$

and given a point, z_0, in the complex plane, the disk constructed about z_0 (in this case $z_0=0$, the origin of the complex plane) which will not contain any zeros of $p(z)$, the polynomial in the denominator of the Pade approximant. Use the notation

$$b_m := \frac{1}{m!} p(m)(z_0), \quad m = 1,2,\ldots,n \quad (b_n=1)$$

so that $p(z_0+h) = b_0 + b_1 h + b_2 h^2 + \ldots + h^n$ for all complex h.

Now, let

$$B(z_0): \min_{1< m<n} \left[\binom{n}{m} \frac{|b_0|}{|b_m|} \right]^{1/m} \quad (6)$$

Then, no zero of p is contained in the disk $|z-z_0| \leq \frac{1}{(ne)}[B(z_0)]$.
The proof of this theorem is omitted here, but is contained in the reference cited.

With these tools the analytical work begins. First, the Pade coefficients of a function, f(z), are computed. Then, using a well known algorithm called MULLER, the zeros of the denominator of the Pade approximant are computed. As a check for the accuracy of zeros routine, the computed zeros are substituted into the denominator polynomial to insure the numbers computed are indeed the zeros. The modulus of the zeros is computed to determine the zeros' distance from the origin of the complex plane. Then, as a second check, the minimum modulus theorem is used to verify that no zeros of the denominator polynomial lie inside the minimum modulus.

ANALYSIS OF THE RESULTS

As is usually the rule in research one seldon obtains the intended results and is usually considered lucky if one can correctly interpret the incorrect results. There is no a priori reason to believe that this research would violate the above observation - in fact it has not, it is only hoped that in this research the incorrect results were correctly interpreted! Therefore, it is necessary to divide the results in three categories Expected Results, Unexpected Results and Areas to Further Study. The areas open to further study category is intended to leave open the door for more research into this interesting subject.

Expected Results

The main thrust of the research was to determine the migration of the poles of the Pade approximant. This is one method which can be used to determine the accuracy of the Pade approximant, that is as the poles of the Pade approximant get larger it is hoped that the Pade approximant will better estimate the function. To this end a FORTRAN Program was written to compute the Pade coefficients (M,N=<10) of the function f(z) given as input the coefficients of the power series expansion of f(z). The program then computed the zeros of the denominator polynomial, the modulus of these zeros and (using the minimum modulus theory discussed previously) minimum radius in which no zeros of the denominator polynomial can be found. Then using different functions (i.e., e^{-z}, e^{z}, $1000e^{-z}$) the poles were computed and chart the migration of the poles. The results of this program for the functions $f(z)=e^{z}$, e^{-z} for values of M and N between 2 and 4 and for $f(z)=1000e^{-z}$ for M and N between 2 and 6 is contained in Tables I. thru IV. following this setup. Because of the ensuing problems, which will be discussed in the next section, the only information which can be learned from this table is the fact that for values of M and N less than 5 there are much better methods (i.e., Taylor series) to approximate a function, f(z). This is because minimum modulus of the zeros is small, that is not greater than 10 for any of the examined functions.

It was expected that as the values of M and N increased the poles of the P[M,N] Pade approximant would migrate away from the origin of the complex plane. If this could have been concluded from the analytical approach it could have provided a large step in showing that the Pade approximant conveages to the function, f(z), since few convergence theorms are presently in the literature.

There is, however, a very important result of applying Pade approximants to viscosity equations contained in Appendix A. This application was discovered in a simultaneous research project carried out in the TAMU Chemistry Department under the direction of Professor Bruno J. Zwolinski and his graduate student Miss Dawn L. Wakefield.

Unexpected Results

From the results of the program little can be concluded about the migration of the poles. It was intended to analyize some common complex functions to determine where the poles of the Pade approximant migrate away from the complex origin as the values of M and N increase. There were quite a few difficulties with the program and the choice of functions which did not allow the calculation of the Pade approximants for values of M and N greater than 5.

The functions which were primarily examined were e^z, e^{-z}, and $1000e^z$. Only for the function $f(z)=1000e^z$ could the P[6,6] Pade approximant be calculated. The reason why M and N had to be limited to values less than 5 was the coefficients of the power series contained the reciprocal factorial term (i.e., $\frac{1}{n!}$). Since the program was in single precision the reciprocal functional seems to get small very fast (i.e., $(11!)^{-1}=2.5\times10^{-8}$ which is used for the [5,5] Pade approximant in all of these functions). This problem would have been aleviated if the program was written in double precision, allowing up to sixteen digits instead of the eight digits used in single precision. The entire program could not be transferred into double precision because the Gaussian elimination subroutine (see Appendix C) was initially in double precision. This problem was not discovered until late in the research period. Every effort was made to modify the single precision program, but due to the time constraints the transition could not be accomplished.

To solve the problem of precision it was decided to examine the complex function $f(z)=1000e^z$. The Pade approximant of this function is the same as $g(z)=e^z$, but the coefficients of the power series expansion of $f(z)=1000e^z$ are larger than those of $g(z)=e^z$ by a factor of one thousand. These results are contained in the following tables. It should be noted from the table for $f(z)=1000e^{-z}$ that there is the beginning of a trend. As M and N increase to six the modulus of the zeros of the denominator increases to approximately 8.0. That is with the exception of one zeros of the denominator polynomial in P[6,6] which has a zero at 4.0×10^{-3}, this it is believe is due to round off error. That is because when adding a number like 2.3×10^{-5} to 1.0×10^{3} the sum is 1.0×10^{3} still, this is especially true when using the single precision program, even if we used double precision could only use [10,10] because $\frac{1}{17!} < 10^{-16}$.

As a test to determine whether a slight change in the coefficients, A(I)'s, of the power series produces a change in the Pade coefficients, P(M)'s and Q(N)'s the fourth, ninth and tenth coefficients were changed in the fifth, seventh, and fifth decimal places respectively. This extremely small change in the coefficients produced a change in P(2) greater than one order of magnitude (note, P(1) is always chosen to be [1.0], the differences in the other P(M)'s and Q(N)'s were almost as drastic. This extreme change in the Pade coefficients is due to instability in the system. In other words when solving the equation Ax=B (where A is a nxn matrix, B and x are vectors) then the equation becomes $x=A^{-1}B$, the above results indicate A^{-1} is almost singular. If A^{-1} were almost singular it would account for the vast differences in the P(M)'s and Q(N)'s while the power series coefficients stayed relatively constant, that is constant in the first four decimal places.

As was mentioned earlier the functions cosz and sinz were also going to be discussed. However, using the Gaussian elimination subroutine on the matrix formed by the coefficients of the sine function resulted in a singular matrix and thus no information could be obtained. Thus, the sine function was not examined by further. Since both the cosine and sine function depend upon the $\frac{1}{n!}$ term in their power series expansion the cosine function was also not examined any further.

Areas For Further Study

Due to the lack of conclusive results pertaining to the migration of the poles of the Pade approximant and thus no information on accuracy of the approximation the areas of investigation on this problem are still open. In the following paragraphs a few of the ways are mentioned in which one might modify the program to obtain further information on the migration of the poles.

Probably the most obvious short term solution to the problem of not being able to have values of M and N greater than four or five is to convert the program into double precision. However, when working with power series coefficients which depend upon the reciprical factoral of n, i.e., $\frac{1}{n!}$, a double precision program could only handle coefficients as small as 1×10^{-16} (assuming 16 digit accuracy) which is approximately $\frac{1}{17!}$. So even with double precision using the same functions i.e., e^{z} and e^{-z} only the [8,8] Pade could be calculated for $1000e^{-z}$ the [10,10] Pade could be calculated. One possible solution to the factional problem is simply to choose a function like $\frac{1}{(2-z)}$ whose power series depends upon the term $\frac{1}{2n}$ not $\frac{1}{n!}$. It is clears that this function has a pole when Re(z)=2 so we assume as M and N yet large the poles of the Pade approximant of this function will be migrating towards 2.

More important than the double precision problem, since alternate functions can be examined, as was indicated above, is the problem of the instability of the system, one example of this is observed by comparing the coefficients of the power series, A(I)'s and the [6,6]

Pade coefficients of the function $f(z)=1000e^{-z}$. This information is contained in Tables III. and IV. It was determined above that probably the reason for the large deviation in the Pade coefficients was due to the inversion matrix (i.e., A^{-1}, where xA=B is a system of n equations with n unknowns, A is an nxn matrix, B and x vectors) being almost singular. Assuming the inversion matrix, A^{-1}, is almost singular then for even very small changes in the enteries in the matrix A, a very large difference $x=A^{-1}B$ will be noticed.

Another possible place for instability is in the zeros routine from examining the program which is contained in Appendix C one observes that the maximum number of interactions need was 500 and that EP1 was 1.0×10^{-4} and EP2 had the value 1.0×10^{-6}. As is observed in the tables which follow this section when the zero was re-evaluated in the polynomial, i.e., f(zero), the solution was very close to 0.00000. By altering the values of EP1 and EP2, that is setting their values too close together, could cause f(zero) to deviate from 0.000000 which would cause instability in the program. That is the zeros of the Pade denominator would be incorrect and in term the modulus would also be incorrect.

There several solutions to the problems of instability in the program. To prevent the instability incurred by the inversion matrix being almost singular one could use a subroutine in place of the Gaussian elimination algorithm. This new subroutine would be error sensitive. That is, the subroutine would be sensitive to the round off error of adding two numbers like 1.0×10^{3} and 2.3×10^{-5}. One possible answer to the instability brought into the program by the EP1 and EP2 being too close would be to monitor (like was done) the polynomial evaluated at the zero. If the value of f(zero) deviated from desired accuracy the EP1 and EP2 would need to be altered and if this did not solve the problem then possibly another algorithm would need to be used. From this summary it should be obvious that there are several possible problems with instability in this program and one must analize the results of the program in light of the possible problems with instability.

Table I. Pade Approximant of e^z

Degree: N=M=2

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1.0000000	1.0000000	1.0000000	3.000052	1.732204	-0.000044	0.000014	3.464223
1.0000000	0.5000002	-0.4999998	3.000004	-1.732051	0.000000	0.000000	3.464105
0.5000000	0.0833334	0.0833332					
0.1666667							
0.0416667							

Table I. cont.

Degree: N=M=3

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1.0000000	1.0000000	1.0000000	4.643845	0.000003	0.000001	-0.000000	4.643845
1.0000000	0.4999653	-0.5000347	3.677224	-3.508896	-0.000017	-0.000048	5.082747
0.5000000	0.0999819	0.1000166	3.677238	3.508659	-0.000000	0.000001	5.082595
0.1666667	0.0083301	-0.0083359					
0.0416667							
0.0083333							
0.0013889							

Table I. cont.

Degree: N=M=4

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1.0000000	1.0000000	1.0000000	5.828315	1.737275	0.000001	-0.000002	6.081724
1.0000000	0.5016597	-0.4983403	5.828270	-1.737303	0.000001	0.000003	6.081690
0.5000000	0.1079958	0.1063361	4.246865	-5.322587	0.000000	0.000001	6.809243
0.1666667	0.0120829	-0.0117498	4.246861	5.322603	0.000000	-0.000000	6.809254
0.0416667	0.0006114	0.0005831					
0.0083333							
0.0013889							
0.0001984							
0.0000248							

Table II. [2,2] Pade Approximant of $f(z)=e^{-z}$

Degree: N=M=2

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
	1.0000000	1.0000000	-3.000080	1.732163	-0.000032	-0.000022	3.464227
1.0000000	-0.5000002	0.4999998	-3.000003	-1.732053	-0.000000	-0.000000	3.464105
-1.0000000	0.0833334	0.0833332					
0.5000000							
-0.1666667							
0.0416667							

Table II. cont.

Degree: N=M=3

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1.0000000	1.0000000	1.0000000	-4.643825	-0.000000	0.000001	-0.000000	4.643825
-1.0000000	-0.4999653	0.5000347	-3.677245	-3.508872	-0.000012	0.000043	5.082747
0.5000000	0.0999819	0.1000166	-3.677244	3.508659	0.000001	0.000000	5.082599
-0.1666667	0.0083301	0.0083359					
0.0416667							
-0.0083333							
0.0013889							

Table II. cont.

Degree: N=M=4

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1.0000000	1.0000000	1.0000000	-5.828341	1.737273	0.000000	0.000000	6.081750
-1.0000000	-0.5016597	0.4983403	-5.828340	-1.737299	-0.000001	0.000000	6.081755
0.5000000	0.1079958	0.1063361	-4.246895	-5.322894	0.000045	0.000025	6.809502
-0.1666667	-0.0120829	0.0117498	-4.246861	5.322601	0.000000	0.000000	6.809252
0.0416667	0.0006114	0.0005831					
-0.0083333							
0.0013889							
-0.0001984							
0.0000248							

Table III. Pade Approximant of $f(z)=1000e^{-z}$

Degree: N=M=2

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1000.0000000	1.0000000	0.0010000	-3.000023	1.732029	0.000000	-0.000000	3.464110
-1000.0000000	-0.4999989	0.0005000	-2.999999	-1.732098	-0.000000	0.000000	3.464124
500.0000000	0.0833327	0.0000833					
-166.6667000							
41.6666700							

Table III. cont.

Degree: N=M=3

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
	1.0000000	0.0010000	-4.642144	0.001854	0.000000	0.000000	4.642144
1000.0000000	-0.5000038	0.0005000	-3.678567	3.505955	0.000000	0.000001	5.081690
-1000.0000000	0.1000022	0.0001000	-3.677855	-3.508872	-0.000000	0.000000	5.083187
500.0000000	-0.0083339	0.0000083					
-166.6667000							
41.6666700							
-8.3333330							
1.3888890							

Table III. cont.

Degree: N=M=4

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1000.0000000	1.0000000	0.0010000	5.793833	1.741264	-0.000000	0.000000	6.049834
-1000.0000000	-0.5000131	0.0005000	-5.790888	-1.737018	-0.000000	-0.000000	6.045793
500.0000000	0.1071495	0.0001071	-4.207914	5.314891	-0.000000	-0.000000	6.778982
-166.6667000	-0.0119062	0.0000119	-4.207948	-5.314940	0.000000	0.000000	6.779042
41.6666700	0.0005953	0.0000001					
-8.3333330							
1.3888890							
-0.1984126							
0.0248015							

Table III. cont.

Degree: N=M=5

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1000.0000000	1.0000000	0.0010000	-6.979667	3.475013	0.000000	0.000000	7.796887
-1000.0000000	-0.5072628	0.0049270	-6.976029	-3.487685	-0.000000	0.000000	7.799290
500.0000000	0.1147690	0.0001075	-7.579778	0.009500	-0.000000	0.000000	7.579783
-166.6667000	-0.0146803	0.0000131	-4.917559	-7.134132	0.000000	0.000000	8.664769
41.6666700	0.0010811	0.0000009	-4.917642	7.133582	-0.000000	0.000000	8.664363
-8.3333330	-0.0000376	0.0000000					
1.3888890							
-0.1984126							
0.0248015							
-0.0027560							
0.0002756							

Table III. cont.

Degree: N=M=6

Coefficients of the Power Series A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
1000.0000000	1.0000000	0.0010000	-0.004022	0.000000	0.000000	0.000000	0.004022
-1000.0000000	248.1139000	0.2491139	-6.974370	3.481656	0.000001	-0.000006	7.795112
500.0000000	-125.9964000	0.1226175	-6.974482	-3.481709	-0.000001	-0.000001	7.795235
-166.6667000	28.5166000	0.0267438	-7.564214	0.000003	0.000001	-0.000000	7.564214
41.6666700	-3.6481040	0.0032643	-4.917186	-7.133861	-0.000007	0.000000	8.664333
-8.3333330	0.2686726	0.0002259	-4.917161	7.133813	0.000016	0.000001	8.664280
1.3888890	-0.0093463	0.0000072					
-0.1984120							
0.0248015							
-0.0027560							
0.0002756							
-0.0000251							
0.0000230							

Table IV. Pade Approximant of $f(z)=1000e^{-z}$ Modified Coefficients

Degree: N=M=6

Coefficients of the Power Series,A(I)	P(M)	Q(N)	Zeros of P(M)		f(Zero)		Modulus
			Re(z)	Im(z)	Re(z)	Im(z)	
	1.0000000	0.0010000	0.099300	0.000000	-0.000000	0.000000	0.099300
1000.0000000	-11.1071700	-0.0101072	1.931770	0.000000	-0.000001	0.000000	1.931770
-1000.0000000	10.7940900	0.0001869	-6.249130	1.728072	0.000001	-0.000000	6.483661
500.0000000	-3.9251600	0.0014820	-6.249156	-1.728155	-0.000000	0.000000	6.483707
-166.6666000	0.7461440	0.0004085	-4.651896	-5.307741	-0.000000	-0.000000	7.057778
41.6666700	-0.0788800	0.0000492	-4.651895	5.307741	-0.000000	0.000000	7.057778
-8.3333330	0.0039083	0.0000025					
1.3888890							
-0.1984120							
0.0248016							
-0.0028000							
0.0002756							
-0.0000251							
0.0000230							

OUTLINE OF THE PROGRAM

The computer program written and used to evaluate the coefficients of Pade approximants is found in Appendix C. The program contains many explanatory comment statements but will be described briefly here. There are two main parts to the program. The first computes the Pade coefficients of a given power series. The second part computes the zeros of the Pade denominator, i.e., poles of the Pade approximant, and the modulus of the zeros, i.e., the distance from the origin of the complex plane.

The coefficients of the power series, i.e., A(I)'s, and the degree of the Pade approximant (M and N) are input as data. The coefficients of the power series are then placed in matrix form. This is denoted by the two-dimensional array, MATRX. Using Gaussian elimination on the array MATRX and the A(I)'s, the Q(N)'s are computed. Then, making the usual assumption that $p_0=1.0$, the P(M)'s are computed in a recursive fashion.

The second part of the program finds the zeros of the polynomial in the denominator of the Pade approximant. This is accomplished by the use of the MULLER subroutine (7). The MULLER algorithm is fully explained in Appendix C. Once the zeros are calculated, then the zeros are re-evaluated in the polynomial to determine the accuracy of the MULLER routine, i.e., how close f(zero) is to zero. The minimum modulus theorem is then used to determine the minimum radius in which no zeros of the Pade denominator are contained. The output is then printed in tabular form.

CONCLUSIONS

In mathematics an extremely important question is if an arbitary function, f(z), can be approximated via one of the many methods used to estimate functions. Most of the methods of approximating functions such as the Taylor and Fourier Series have theorems discribing when the approximation converges to the function. However, currently in the literature there are not any theorems discribing the conditions needed to insure the Pade approximant of a function converges.

Because of the lack of literature an analytical approached was used. It was hoped if the poles of the [M,N] Pade approximant migrated away from the origin of the complex plane, that is the modulus of the zeros of the Pade denominator became very large, then the Pade approximant would become very close to the value of the function. Due to the limitations of using the single percision program as well as the instability inherent in the program no conclusive data was obtained. So a similar program in double precision with built in checks for instability would be a great help to anyone trying to determine the migration of the poles of the Pade approximant and thus the convergence of the Pade approximant.

END NOTES

[1]S. D. Conte and Carl de Boor, Elementary Numerical Analysis (New York: McGraw-Hill), pp 77-80.

[2]H. Pade, Ann. El. Nom., Sup. 3, 5.14, (1892).

[3]B. J. Zwolinski, et al., Unpublished Paper, "Pade Approximants and Their Application to Liquid Viscosity".

[4]George A. Baker, Jr. and John L. Gammel, "The Pade Approximant Method; Related Generalizations", The Pade Approximant in Theoretical Physics (New York: Academic Press), p. 3.

[5]G. D. Allen, et al., "Pade Approximation and Gaussian Quadrature", Bulliten of the Australian Mathematical Society, 11, No. 1, (1974), 65.

[6]Peter Henrici, Applied and Computational Complex Analysis, vol. 1, (New York: John Wiley and Sons Ltd.), pp 450-452.

[7]S. D. Conte and Carl de Boor, pp 77-80.

[8]G. D. Allen, Personal Communication, "The Derivation of the [2,2] and [3,3] Pade Coefficients Equations", February, 1981.

[9]Ibid.

[10]G. H. Eduljee and Adrian P. Boyes, "Viscosity of Some Binary Liquid Mixtures of Olec Acid and Triolene with Selected Solvents", Journal of Chemical and Engineering Data, July 1980, p. 249.

[11]Ibid.

[12]Ibid.

[13]Ibid.

[14]B. J. Zwolinski, et al.

BIBLIOGRAPHY

Allen, G. D. Personal Communication. "The Derivation of the [2,2] and [3,3] Pade Coefficients Equations". February, 1981.

Allen, G. D., et al. "Pade Approximation and Gaussian Quadrature". Bulliten of the Australian Mathematical Society. Vol. 11, 1974, p. 63.

Baker, G. A., Jr. and J. L. Gammel. "The Pade Approximant Method; Related Generalizations". The Pade Approximant in Theoretical Physics. New York: Academic Press, 1970.

Clenshaw, G. W. and K. Lord. "Rational Approximations from Chebyshev Series". Studies in Numerical Analysis. Ed. B. K. P. Scaife. London: Academic Press, 1974.

Conte, S. D. and C. de Boor. Elementary Numerical Analysis. New York: McGraw-Hill, 1972, pp. 77-80.

Donnelly, J. D. P. "The Pade Table". Methods of Numerical Approximation. Ed. D. C. Hondscomb. Oxford: Pergamon Press, 1966.

Eduljee, G. H. and Adrain P. Boyes. "Viscosity of Some Binary Liquid Mixtures of Olec Acid and Tridene with Selected Solvents". Journal of Chemical and Engineering Data. July, 1980, p. 249.

Henrici, P. Applied and Computational Complex Analysis. New York: John Wiley and Sons Ltd., 1974, vol. 1, pp. 450-452.

McAllister, R. A. "The Viscosity of Liquid Mixtures". American Institute of Chemical Engineering Journal. September, 1960, p. 428.

Pade, H. Am. El. Nom. Sup. 3. 5.14. (1892).

Saff, E. B. and R. S. Varga. "Angular Overconvergence for Rational Functions Converging Geometrically on [o,+∞]". Theory of Approximations With Applications. Ed. Alan G. Law and Badri N. Sahney. New York: Academic Press, 1976.

Subach, Daniel J. "Equilibrium and Non-Equilibrium Studies on Certain Chemical Systems". Diss. Texas A&M University, 1974.

Wall, H. S. Continued Fractions. New York: Van Nostrand Press, 1948.

Watson, Fulks. _Advanced Calculus_. New York: John Wiley and Sons Ltd., 1978.

Watson, G. A. _Approximation Theory and Numerical Methods_. New York: John Wiley and Sons Ltd., 1980.

Zwolinski, B. J., _et al_. Unpublished Paper. "Pade Approximants and Their Application to Liquid Viscosity".

APPENDIX A

The [2,2] and [3,3] Pade Approximants of the Reciprocal Exponential Function

To give an example of how one could compute the Pade approximant from the power series expansion of an arbitrary function, this appendix is included which describes how the [2,2] and [3,3] Pade approximants of $f(z) = e^{-z}$ are calculated.

To determine the Pade coefficients for the reciprocal exponential function, the function must first be expressed as a power series. The method used to expand the reciprocal exponential function into a power series was the Taylor series expansion. The Taylor series (or power series) expansion of the reciprocal exponential function is a polynomial of the form

$$f(z) = e^{-z} = 1 - z + \frac{z^2}{2!} - \frac{z^3}{3!} + \ldots. = \sum_{i=0}^{\infty} \frac{(-1)^i z^i}{i!} ,$$

where, by definition, $0! = 1$ and

$$a_0 = 1,\ a_1 = -1,\ a_2 = \frac{1}{2!},\ a_3 = -\frac{1}{3!},\ a_4 = \frac{1}{4!},\ \ldots.$$

Now that the a_i's are known, the Pade coefficients need to be computed. Recall the two formulae,

$$\sum_{i=0}^{k} q_i a_{k-i} - p_k = 0 \text{ for } k \leq n \quad \text{and}$$

$$\sum_{i-0}^{n} q_i a_{k-i} = 0 \text{ for } k \geq m+n.$$

Using m = n = 2, that is the [2,2] Pade approximant for the reciprocal exponential function, the above equations reduce to a system of five equations and five unknowns. The five homogeneous equations, which were obtained for values of k ranging from zero to k=m+n=2+2=4, have the form

$$k=0 \quad q_0a_0 - p_0 = 0 \qquad \text{i.}$$

$$k=1 \quad q_0a_1 + q_1a_0 - p_1 = 0 \qquad \text{ii.}$$

$$k=2 \quad q_0a_2 + q_1a_1 + q_2a_0 - p_2 = 0 \qquad \text{iii.}$$

$$k=3 \quad q_0a_3 + q_1a_2 + q_2a_1 + 0 \qquad \text{iv.}$$

$$k=4 \quad q_0a_4 + q_1a_3 + q_2a_2 = 0 \qquad \text{v. (8)}$$

The five unkowns are q_0, q_1. q_2, p_1, and p_2. The Pade coefficient, p_0, is commonly given the value of one, so it is not considered an unknown. Recall the values of a_1, a_2, a_3, and a_4 are already known from the power series expansion of the reciprocal exponential function. To solve the system of equations, take equation i. and solve it for q_0; it becomes

$$q_0 = \frac{p_0}{a_0} \text{ , but } p_0=1 \text{ and } a_0=1\text{; therefore, } q_0=1.$$

Now, to find q_1 and q_2, solve equations iv. and v. simultaneously, so

$$\frac{1}{2}\left(\frac{1}{3!} + \frac{1}{2}q_2 + q_2\right) = 0 \qquad \text{iv.}$$

$$\frac{1}{4!} + \frac{1}{3!}q_2 + \frac{1}{2}q_2 = 0 \qquad \text{v.}$$

Thus, $q_1 = -\frac{1}{2}$ and $q_2 = \frac{1}{12}$

Using the values of q_1 and q_2 and equations ii. and iii., the values of p_1 and p_2 can be determined. The values are determined in a manner similar to the method used to compute q_1 and q_2, by solving a system of two equations and two unknowns. The values are

$$p_1 = \frac{1}{2} \text{ and } p_2 = \frac{1}{12} \quad .$$

So the [2,2] Pade approximant of the reciprocal exponential function takes the form:

$$e^{-z} = 1 - z + \frac{z^2}{2!} - \frac{z^3}{3!} + \ldots . = \frac{1 - \frac{1}{2}z + \frac{1}{12}z^2}{1 + \frac{1}{2}z + \frac{1}{12}z^2} .$$

The computation of the [3,3] Pade approximant for the reciprocal exponential function is calculated in a similar manner. First, the following system of equations is obtained.

k=0	$q_0a_0 - p_0 = 0$	i.
k=1	$q_0a_1 + q_1a_0 - p_1 = 0$	ii.
k=2	$q_0a_2 + q_1a_1 + q_2a_0 - p_2 = 0$	iii.
k=3	$q_0a_3 + q_1a_2 + q_2a_1 + q_3a_0 - p_3 = 0$	iv.
k=4	$q_0a_4 + q_1a_3 + q_2a_2 + q_3a_1 = 0$	v.
k=5	$q_0a_5 + q_1a_4 + q_2a_3 + q_3a_2 = 0$	vi.
k=6	$q_0a_6 + q_1a_5 + q_2a_4 + q_3a_3 = 0$	vii. (9)

Again, choose p_0=1, so equation i. reduces to

$$q_0 = \frac{p_0}{a_0} = 1$$

Now, solve equations v., vi., and vii. simultaneously for q_1, q_2, and q_3, as was done in the case of the [2,2] Pade approximant. Upon solving the system of three equations and three unknowns, the values of the Pade coefficients in the denominator are

$$q_1 = \frac{2}{5}, \quad q_2 = \frac{1}{5}, \text{ and } q_3 = \frac{1}{20}\frac{1}{3!}.$$

Finally, in a manner similar to that used in the [2,2] Pade approximant case, solve equations ii., iii., and iv. to determine p_1, p_2, and p_3. The values are

$$p_1 = -\frac{1}{2}, \; p_2 = \frac{1}{5}\frac{1}{2!}, \text{ and } p_3 = -\frac{1}{20}\frac{1}{3!}.$$

So, the form of the [3,3] Pade approximant for the reciprocal exponential function is

$$e^{-z} = 1 - z + \frac{z^2}{2!} - \ldots . = \frac{1 - \frac{1}{2}z + \frac{1}{5}\frac{z^2}{2!} - \frac{1}{20}\frac{z^3}{3!}}{1 - \frac{2}{5}z + \frac{1}{5}\frac{z^2}{2!} - \frac{1}{20}\frac{z^3}{3!}} .$$

A precalculated Pade table, which contains the Pade approximants for the reciprocal exponential function for values of m and n ranging from zero through four is on the following page.

TABLE V. PADE TABLE FOR THE RECIPROCAL EXPONENTIAL FUNCTION

The Pade table below contains the Pade table for the reciprocal exponential function. Along the horizontal axis, the value of m, the degree of the numerator, increases from zero to three. Along the vertical axis, the value of n, the degree of the denominator, increases from zero to four. It is of particular interest to look at the [2,2] and [3,3] Pade approximants. This table confirms the methods used in the text were correct to compute the [2,2] and [3,3] Pade approximants.

<u>Pade Table for e^{-x}</u>

$\frac{1}{1}$	$\frac{1-x}{1}$	$\frac{1-x+\frac{x^2}{2!}}{1}$	$\frac{1-x+\frac{x^2}{2!}-\frac{x^3}{3!}}{1}$
$\frac{1}{1+x}$	$\frac{1-\frac{1}{2}x}{1+\frac{1}{2}x}$	$\frac{1-\frac{2}{3}x+\frac{1}{3}\frac{x^2}{2!}}{1+\frac{1}{3}x}$	$\frac{1-\frac{3}{4}x+\frac{2}{4}\frac{x^2}{2!}-\frac{1}{4}\frac{x^3}{3!}}{1+\frac{1}{4}x}$
$\frac{1}{1+x+\frac{x^2}{2!}}$	$\frac{1-\frac{1}{3}x}{1+\frac{2}{3}x+\frac{1}{3}\frac{x^2}{2!}}$	$\frac{1-\frac{1}{2}x+\frac{1}{6}\frac{x^2}{2!}}{1+\frac{1}{2}x+\frac{1}{6}\frac{x^2}{2!}}$	$\frac{1-\frac{3}{5}x+\frac{3}{10}\frac{x^2}{2!}-\frac{1}{10}\frac{x^3}{3!}}{1+\frac{2}{5}x+\frac{1}{10}\frac{x^2}{2!}}$
$\frac{1}{1+x+\frac{x^2}{2!}+\frac{x^3}{3!}}$	$\frac{1-\frac{1}{4}x}{1+\frac{3}{4}x+\frac{2}{4}\frac{x^2}{2!}+\frac{1}{4}\frac{x^3}{3!}}$	$\frac{1-\frac{2}{5}x+\frac{1}{10}\frac{x^2}{2!}}{1+\frac{3}{5}x+\frac{3}{10}\frac{x^2}{2!}+\frac{1}{10}\frac{x^3}{3!}}$	$\frac{1-\frac{1}{2}x+\frac{1}{5}\frac{x^2}{2!}-\frac{1}{20}\frac{x^3}{3!}}{1+\frac{1}{2}x+\frac{1}{5}\frac{x^2}{2!}+\frac{1}{20}\frac{x^3}{3!}}$
$\frac{1}{1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+\frac{x^4}{4!}}$	$\frac{1-\frac{1}{5}x}{1+\frac{4}{5}x+\frac{3}{5}\frac{x^2}{2!}+\frac{2}{5}\frac{x^3}{3!}+\frac{1}{5}\frac{x^4}{4!}}$	$\frac{1-\frac{1}{3}x+\frac{1}{15}\frac{x^2}{2!}}{1+\frac{2}{3}x+\frac{2}{5}\frac{x^2}{2!}+\frac{1}{5}\frac{x^3}{3!}+\frac{1}{15}\frac{x^4}{4!}}$	$\frac{1-\frac{3}{7}x+\frac{1}{7}\frac{x^2}{2!}-\frac{1}{35}\frac{x^3}{3!}}{1+\frac{4}{7}x+\frac{2}{7}\frac{x^2}{2!}+\frac{4}{35}\frac{x^3}{3!}+\frac{1}{35}\frac{x^4}{4!}}$

APPENDIX B

VISCOSITY EQUATIONS

Pade approximants are currently used in many branches of science, including nuclear physics, engineering, and physical chemistry. In the latter field, a unique application of Pade approximants has been applied to liquid viscosity equations, expecially the Auslander and McAllister viscosity equations. The various liquid viscosity equations are believed to be rigorously derived from theory. This does not seem to be the case, since the Auslander equation is reducible to a Pade approximant and there is every indication that the McAllister equation is also reducible to a form of a Pade approximant. This interesting aspect of the Pade approximants has been investigated by Professor of Chemistry, Bruno J. Zwolinski and his graduate student Dawn Lee Wakefield at Texas A&M.

Since it appears that these two liquid viscosity equations are Pade approximants, the question arises as to the accuracy of the approximation to the true viscosity of the liquid, and thus the poles of the Pade approximant are important. Obviously, after the proper Pade approximant is derived for the viscosity equation, then one must be careful not to use this equation around the poles of the approximant to avoid erroneous results. Therefore, it is important to determine the Pade approximant of these liquid viscosity equations to find the poles to inform the engineer or chemist using such equation just exactly where these equations break down and are, thus, not useful.

The Auslander Equation as a Pade Approximant

The Auslander liquid mixture equation has the form

$$X_1(X_1 + B_{12}X_2)(v - v_1) - A_{21}X_2(B_{21}X_1 + X_2)\ (v - v_2) \quad \text{(B.1)}$$

where,

v, v_1, and v_2 are kinematic viscosities,

A_{21}, B_{12}, and B_{21} are constants representing binary interactions,

and X_1 and X_2 are mole fractions. (10)

Since the hydrodynamic property, viscosity, is assumed to be a thermodynamic property, the exact methods of thermodynamics may be used. This means for the Auslander equation to truly and precisely represent the viscosity of a liquid mixture, it must be an exact thermodynamic relation. The hypothesis is that the Auslander equation can be reduced to a ratio of two polynomials, a form of Pade approximant. So, the Auslander equation is only an approximation of the viscosity and not an exact thermodynamic relation as it is currently believed.

The Auslander equation reduces to a form similar to a [2,2] Pade approximant. The following are the algebraic steps required to reduce the Auslander equation to the ratio of two polynomials. The ratio of viscosities, $\frac{v - v_2}{v - v_1}$, is solved for,

The Auslander equation originally had the form

$$X_1(X_1 + B_{12}X_2)(v - v_1) + A_{21}X_2(B_{21}X_1 + X_2)(v - v_2) = 0.$$

Moving one term to the other side of the equation gives

$$X_1(X_1 + B_{12}X_2)(v - v_1) = -A_{21}X_2(B_{21}X_1 + X_2)(v - v_2)$$

and then solving for the ratio of viscosities gives

$$\frac{(v - v_2)}{(v - v_1)} = \frac{X_1(X_1 + B_{12}X_2)}{-A_{21}X_2(B_{21}X_1 + X_2)}$$

When working with solvents and solutes the usual convention is to let

$$X_1 = X \quad \text{and } X_2 = 1 - X$$

Then equation (B.1) becomes

$$\frac{(v - v_2)}{(v - v_1)} = - \frac{X(X + B_{12}(1 - X))}{A_{21}(1 - X)(B_{12}X + (1 - x))} ,$$

which reduces to

$$= - \frac{B_{12}X - (1 - B_{12})X^2}{A_{21} + (B_{21} + A_{21})X + (A_{21}B_{21} - A_{21})X^2} . \quad (B.2)$$

Comparing this to a [2,2] Pade approximant of the form

$$f(X) = \frac{p_0 + p_1X + p_2X}{q_0 + q_1X + q_2X^2} ,$$

the coefficients of the reduced Auslander equation (B.2) and the [2,2] Pade approximant can be equated. The coefficients are

$$q_0 = A_{21},\ q_1 = B_{21} + A_{21},\ q_2 = A_{21}(B_{21} - 1),$$

$$p_0 = 0,\ p_1 = B_{12},\ \text{and } p_2 = (1 - B_{12}). \quad (B.3)$$

Recall the coefficients A_{21}, B_{12}, and B_{21} are constants representing binary interactions. They are determined from a least-squares fit of the experimental data (11).

Practical Importance

The immediate application of the research is twofold. First, since the Auslander equation reduced to a Pade approximant, it is not an exact thermodynamic equation. This means that while the Auslander equation is a viable method for calculating the viscosity of a liquid mixture, given the correct constants, it is not based solely on theory. The Auslander equation must be based on some approximations. This leaves the door open for more research to determine an equation, based strictly on theory, which describes exactly the viscosity of a liquid mixture, given the correct constants.

Secondly, and probably the most important application, is the possibility of compiling Pade tables in which the user would have to supply the data and the approximant viscosity would be determined. This process will be made easier once the relation between the values of the binary interaction constants (A_{21}, B_{12}, and B_{21}) and the Pade coefficients are understood more clearly. At present it is believed that there is a direct relation between the Pade coefficients and the binary interaction constants, but it is still uncertain.

A Look at the McAllister Equation

The liquid viscosity McAllister equation has the form

$$\begin{aligned} \ln\nu &= X_1^3 \ln\nu_1 + 3X_1^2 X_3 \ln\nu_{12} + 3X_1 X_2^2 \ln\nu_{21} + X_2^3 \ln\nu_2 \\ &\quad - X_2^3 \ln\nu_2 - \ln(X_1 + X_2(M_2/M_1)) + 3X_1^2 X_2 \ln[(2 + M_1/M_2)/3] \\ &\quad + 3X_1 X_2^2 \ln[(1 + (2M_2/M_1))/3] + X_2^3 \ln(M_2/M_1), \end{aligned}$$

where, X_1 and X_2 are mole fractions,

ν_{12} and ν_{21} are fitting constants,

M_1 and M_2 are molecular weight,

and ν is the kinematic viscosity. (12)

The McAllister equation considers the interactions between like and unlike molecules developed from a correlation for three-body interactions. The molecular size was restricted to a ratio of 1.5 (13). Recall the reciprocal exponential function is the inverse function of the natural logarithm function. Since the McAllister equation involves the natural logarithm of the viscosity, that is $\ln\nu$, it is required to use the reciprocal exponential function to find the viscosity. Because the exponential function is not distributive, the McAllister equation is much more difficult to manipulate algebraically than the Auslander equation. By not being distributive, it is meant that the exponential of the sum of the two numbers is not equal to the exponential of each of the two numbers spearately added together. Current research efforts involve showing how the McAllister equation reduces to a Pade approximant, if this is possible.

The liquid mixture viscosity Auslander equation is reducible to a form of a Pade approximant. This indicates that the Auslander equation is not exact, but rather is an approximation. This approximation relates the kinematic viscosity (ν, ν_1, and ν_2), the constants representing binary interactions (A_{21}, B_{12}, and B_{21}), and the mole fractions (X_1 and X_2). So the viscosity of liquid mixtures computed by the Auslander equation could also be computed via the proper Pade table.

All evidence indicates the liquid viscosity McAllister equation can be reduced to a Pade approximant. This means the McAllister equation is also not an exact equation. So the viscosity calculated through the McAllister equation probably could be calculated through a Pade approximant with the proper choice of coefficients (14).

APPENDIX C

Program Listing

```
//MODF JOB (E405,500,502,001,MP),'COEFS'
//*XBM WATFIV
COMMENT:    THIS PROGRAM COMPUTES THE PADE COEFFICIENTS
C        OF A FUNCTION THE ZEROS AND THE MINIMUM MODULUS OF THE
C        PADE DENOMINATOR.  THE COEF'S OF THE POWER SERIES ARE INPUT
C        IN THE DATA.
C
C        THIS PARTICULAR PROGRAM COMPUTES THE (M,N)
C         PADE COEFICIENTS.
C
        EXTERNAL DECOMP, SOLVE1, IMPRUV
C       READ THE DEGREE OF THE NUMERATOR, M, AND THE DEGREE OF THE
C       DENOMERATOR, N (THE MAXIMUM DEGREE IS M = N = 18).
C       READ THE COEFFICIENTS OF THE POWER SERIES, A(I)'S.
        CALL DATA
C
C       PLACE THE POWER SERIES COEFFICIENTS, A(I)'S
C       INTO A N BY N MATRIX.
        CALL MATRX
C
C       COMPUTE Q(1) WITH THE COMMON PADE ASSUMPTION THAT P(1) = 1.
        CALL COMPUT
C       MULTIPLY THE A(I)'S BY -Q(1) TO PUT THE A(I)'S IN THE
C       PROPER FORM.
        CALL ADJUST
C
C       USING GAUSSIAN ELIMINATION DETERMINE THE Q(N)'S.
C       THIS IS ACCOMPLISHED BY USING THE SUBROUTINES DECOMP, SOLVE1,
C       & IMPRUV WHICH ARE DECLARED AS EXTERNALS AND ARE IN THE FILE
C       COURTSEY OF PROF. NAUGLE.  THESE SUBPROGRAMS ARE CONTAINED
C       IN THE BOOK "COMPUTER SOLUTIONS OF LINEAR ALGERBRAIC SYSTEMS",
C       G. FORSYTHE AND C. B. MOLER; PUBLISHED BY PRENTICE-HALL
C       INC. (1967).
        CALL CONVRT
```

```
C
C      NOW USING THE VALUES OF THE Q(N)'S, SOLVE FOR THE P(M)'S.
       CALL SOLVP
C
C      PRINT OUT THE P(M)'S AND THE Q(N)'S.
       CALL OUTPT
C
C     THE SECOND PART OF THIS PROGRAM USES THE Q(N) COEFFICIENTS TO
C     DETERMINE THE ZEROES OF THE POLYNOMINAL IN THE DENOMINATEOR.
C     THIS IS ACCOMPLISHED AGAIN BY THE HELP DR. NAUGLE AND THE
C     MULLER METHOD SUBROUTINE CONTAINED IN THE NUMERICAL TEXT
C     BY S. D. CONTE AND CARL DE BOOR TITLED "ELEMENTARY NUMERICAL
C     ANALYSIS", SECOND EDITION, PUBLISHED BY MCGRAW-HILL (1972).
C
C     COMPUTE THE ZEROS OF THE POLYNOMIAL IN THE DENOMINATOR
C     USING THE SUBROUTINE ZEROS.
      CALL ZEROS
C      COMPUTE THE MINIMUM MODULUS OF THE POLYNOMIAL IN
C      THE DENOMINATOR.  THIS IS COURTESY OF PROF. G. D. ALLEN.
       CALL MINMOD
C      PLACE DIAGNOSTICS ON A SEPRATE PAGE.
       WRITE (6,1000)
  1000 FORMAT ('1')
       STOP
       END
       SUBROUTINE DATA
C      THIS SUBROUTINE READS THE DEGREE OF M AND N (IN THAT
C      ORDER) AND THE N + M + 1 COEFFICIENT OF THE POWER SERIES.
       REALA(36)
       INTEGER I, M, N, X
       COMMON /AREA 1/ A /AREA 5/ M,N
```

```
          WRITE (6,3000)
    3000  FORMAT ('1', 45X, 'THE DEGREE OF M AND N RESPECTIVELY.')
          READ (5,1000) M, N
    1000  FORMAT (2I4)
          WRITE (6,4000) M, N
    4000  FORMAT (' ', 50X, 2I4)
          X = M + N + 1
          DO 10 I=1,X
             READ (5,2000) A(I)
    2000     FORMAT (F10.7)
    10    CONTINUE
          RETURN
          END
          SUBROUTINE MATRX
   C      THIS SUBROUTINE PLACES THE POWER SERIES INTO A MATRIX
   C      TO USE GAUSSIAN ELIMINATION TO SOLVE FOR THE Q(N)'S.
          REAL A(36), MATX(36,36)
          INTEGER I, J, K, M, N, NN
          COMMON /AREA 1/ A /AREA 3/ MATX /AREA 5/ M, N
          NN = N + 1
          K = M - 1
          DO 10 I= 2, NN
              DO 20 J= 2, NN
                  MATX(J,I) = A(J+K)
    20        CONTINUE
              K = K - 1
    10    CONTINUE
   C      WRITE THE 2-D ARRAY BY ROWS.
         WRITE (6,2000)
    2000 FORMAT (' ', 40X, 'THE MATRIX DENOTED BY THE ARRAY MATX.')
          DO 40 I=2,NN
```

```
                DO 30 J=2,NN
                     WRITE (6,1000) MATX(I,J)
        1000         FORMAT (' ', 50X, F15.7)
          30    CONTINUE
          40 CONTINUE
             RETURN
             END
             SUBROUTINE COMPUT
      C      THIS SUBROUTINE COMPUTES Q(1) USING THE ASSUMPTION P(1) = 1.
             REAL P(36), Q(36), A(36)
             COMMON /AREA1/ A  /AREA 2/ P, Q
             P(1) = 1.0
             Q(1) = P(1) / A(1)
             RETURN
             END
             SUBROUTINE ADJUST
      C      THIS SUBROUTINE ADJUSTS THE A(I)'S TO BE USED IN THE
      C      GAUSSIAN ELIMINATION.
             REAL P(36), Q(36), A(36), DA(36)
             INTEGER I, M, N, NN
             COMMON /AREA1/ A /AREA 2/ P,Q  /AREA 4/ DA
             COMMON /AREA 5/ M, N
      C      WRITE THE ADJUSTED A(I)'S.
             WRITE (6,1000)
        1000 FORMAT (' ', 40X, 'WRITE THE ADJUSTED A(I)S.')
             NN = M + 1
             DO 10 I = 1, N
                DA(I) = -Q(1) * A(I+NN)
                WRITE (6,2000) DA(I)
        2000    FORMAT (' ', 50X, F15.7)
        10   CONTINUE
             RETURN
             END
```

```
      SUBROUTINE CONVRT
C     THIS SUBROUTINE APPLIES GAUSSIAN ELIMINATION TO THE SYSTEM
C     OF EQUATIONS  MATX(I,J) * Q(N) = DA(N).
      EXTERNAL DECOMP, SOLVE1, IMPRUV
      REAL MATX(36,36), A(36), P(36), Q(36), DIGETS
      REAL UL(36,36), DMATX(36,36), DQ(36), DA(36)
      INTEGER NN, I, J, K, M, N
      COMMON /AREA1/ A /AREA2/ P,Q /AREA3/ MATX
      COMMON /AREA 4/ DA /AREA 5/ M, N
C     RE-INDEX THE MATRIX 'MATX' BECAUSE IT CURRENTLY STARTS
C     AT (2,2) INSTEAD OF THE USUAL (1,1). THIS IS ACCOMPLISHED
C     BY THE USE OF A DUMMY MATRIX 'DMATX'.
      NN = N + 1
      DO 10 I = 2, NN
         DO 20 J = 2, NN
            DMATX(I,J) = MATX(I,J)
            MATX(I-1,J-1) = DMATX(I,J)
   20      CONTINUE
   10 CONTINUE
      CALL DECOMP (N, MATX, UL)
      CALL SOLVE1 (N, UL,DA, DQ)
      CALL IMPRUV (N, MATX, UL, DA, DQ, DIGETS)
C     MODIFY THE Q(N)'S TO BE COMPATABLE WITH THE SUBROUTINE USING
C     THE DUMMY ARRAY 'DQ(N)'.
      DO 30 K = 1, N
         Q(K+1) = DQ(K)
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SOLVP
C     THIS SUBROUTINE COMPUTES THE P(M)'S.
       REAL A(36), P(36), Q(36), SUM
       INTEGER I, J, K, M, N, MM, X
       COMMON /AREA1/ A /AREA2/ P, Q /AREA 5/ M, N
C      WRITE THE A(I)'S.
       WRITE (6,1000)
 1000 FORMAT (' ', 40X, 'WRITE THE COEFS OF THE POWER SERIES, A(I)S.
     +')
       MM = M + 1
       X = M + N + 1
        DO 30 I = 1, X
            WRITE (6,2000) A(I)
 2000       FORMAT (' ', 50X, F15.7)
   30   CONTINUE
        DO 10 K = 2, MM
C            COMPUTE P(K).
             SUM = 0.
             DO 20 I = 1 , K
                 J = K - I
                   SUM = Q(I) * A(J+1) + SUM
   20        CONTINUE
             P(K) = SUM
   10  CONTINUE
       RETURN
       END
```

```
      SUBROUTINE OUTPT
C     THIS SUBROUTINE WRITES THE P(M)'S AND Q(N)'S.
      REAL P(36), Q(36)
      INTEGER I, M, N, X
      COMMON /AREA2/ P, Q /AREA 5/ M, N
C     WRITE THE P(M)'S.
       WRITE (6,2000)
 2000  FORMAT (' ', 45X, 'THE P(M) PADE COEFFICIENTS.')
       X = M + 1
      DO 10 I = 1, X
          WRITE (6,1000) P(I)
 10   CONTINUE
C     WRITE THE Q(N)'S.
      WRITE (6,3000)
 3000 FORMAT (' ', 45X, 'THE Q(N) PADE COEFFICIENTS.')
      X = N + 1
      DO 20 I = 1, X
          WRITE (6,1000) Q(I)
  20  CONTINUE
 1000 FORMAT (' ', 50X, F15.7)
      RETURN
      END
```

```
      SUBROUTINEDECOMP(NN,A,UL);DIMENSIONA(36,36),UL(36,36),SCALES(36),I
     *PS(36);COMMONIPS,S1;N=NN;S1=1
C      INITIALIZE IPS, UL, AND SCALES
      DO5I=1,N;IPS(I)=I;ROWNRM=0.0;DO2J=1,N;UL(I,J)=A(I,J);IF(ROWNRM-ABS
     *(UL(I,J)))1,2,2;1:ROWNRM=ABS(UL(I,J));2:CONTINUE;IF(ROWNRM)3,4,3;
    3 SCALES(I)=1.0/ROWNRM;GOTO5;4:CALLSING(1);SCALES(I)=0.0;5:CONTINUE
C      GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING
      NMI=N-1;DO17K=1,NMI;BIG=0.0;DO11I=K,N;IP=IPS(I);SIZE=ABS(UL(IP,K))
     **SCALES(IP);IF(SIZE-BIG)11,11,10;10:BIG=SIZE;IDXPIV=I;11:CONTINUE
      IF(BIG)13,12,13;12:CALLSING(2);GOTO17;13:IF(IDXPIV-K)14,15,14;
   14 J=IPS(K);IPS(K)=IPS(IDXPIV);IPS(IDXPIV)=J;S1=-S1;15:KP=IPS(K);PIVO
     *T=UL(KP,K);KPI=K+1;DO16I=KPI,N;IP=IPS(I);EM=-UL(IP,K)/PIVOT;UL(IP,
     *K)=-EM;DO16J=KPI,N;UL(IP,J)=UL(IP,J)+EM*UL(KP,J);16:CONTINUE;
   17 CONTINUE;KP=IPS(N);IF(UL(KP,N))19,18,19;18:CALLSING(2);19:RETURN;E
     *ND
      SUBROUTINESOLVE1(NN,UL,B,X);DIMENSIONUL(36,36),B(36),X(36),IPS(36)
      COMMONIPS,S1;N=NN;NPI=N+1;IP=IPS(1);X(1)=B(IP);DO2I=2,N;IP=IPS(I)
      IM1=I-1;SUM=0.0;DO1J=1,IM1;1:SUM=SUM+UL(IP,J)*X(J);2:X(I)=B(IP)-SU
     *M;IP=IPS(N);X(N)=X(N)/UL(IP,N);DO4IBACK=2,N;I=NPI-IBACK
C      I GOES (N-1), ..., 1
      IP=IPS(I);IP1=I+1;SUM=0.0;DO3J=IP1,N;3:SUM=SUM+UL(IP,J)*X(J);
    4 X(I)=(X(I)-SUM)/UL(IP,I);RETURN;END
      SUBROUTINEIMPRUV(NN,A,UL,B,X,DIGITS);DIMENSIONA(36,36),UL(36,36),B
     *(36),X(36),R(36),DX(36)
C      USES LIBRARY ROUTINES ABS, AMAX1, ALOG10
      DOUBLEPRECISIONSUM,AIJ,XJ,XB;DATAEPS/Z3BFFFFFF/;N=NN;ITMAX=14
C      EPS IS LARGEST NUMBER FOR WHICH 1.0 + EPS = 1.0
C      ITMAX IS TAKEN TO BE TWICE THE NUMBER OF DECIMAL DIGITS
C      IN A FLOATING POINT NUMBER.
```

```
      XNORM=0.0;DO1I=1,N;1:XNORM=AMAX1(XNORM,ABS(X(I)));IF(XNORM)3,2,3;
    2 DIGITS=-ALOG10(EPS);GOTO10;3:DO9ITER=1,ITMAX;DO5I=1,N;SUM=0.0D0;DO
     *4J=1,N;AIJ=A(I,J);XJ=X(J);4:SUM=SUM+AIJ*XJ;XB=B(I);SUM=XB-SUM;
    5 R(I)=SUM
C     IT IS ESSENTIAL THAT A(I,J)*X(J) YIELD A DOUBLE PRECISION
C     RESULT AND THAT THE ABOVE + AND - BE DOUBLE PRECISION.
      CALLSOLVE1(N,UL,R,DX);DXNORM=0.0;DO6I=1,N;T=X(I);X(I)=X(I)+DX(I);D
     *XNORM=AMAX1(DXNORM,ABS(X(I)-T));6:CONTINUE;IF(ITER-1)8,7,8;7:DIGIT
     *S=-ALOG10(AMAX1(DXNORM/XNORM,EPS));8:IF(DXNORM-EPS*XNORM)10,10,9;
    9 CONTINUE
C     ITERATION DID NOT CONVERGE
      CALLSING(3);10:RETURN;END
      SUBROUTINE SING (IWHY)
   11 FORMAT ('0MATRIX WITH ZERO IN DECOMPOSE.'  )
   12 FORMAT ('0SINGULAR MATRIX IN DECOMPOSE.  ZERO DIVIDE
     1IN SOLVE1.'  )
   13 FORMAT ('0NO CONVERGENCE IN IMPRUV.  MATRIX IS NEARLY
     1SINGULAR.'  )
      GO TO (1,2,3), IWHY
    1 WRITE (6,11)
      GO TO 10
    2 WRITE (6,12)
      GO TO 10
    3 WRITE (6,13)
   10 RETURN
      END
      SUBROUTINE MULLER(KN,N,RTS,MAXIT,EP1,EP2,FN,FNREAL)
      COMPLEX RTS(1)
      LOGICAL FNREAL
      COMPLEX RT,H,DELFPR,FRTDEF,LAMBDA,DELF,DFPRLM,NUM,
     +DEN,G,SQR,FRT,FRTPRV
C
C     INITIALIZATION
      EPS1=AMAX1(EP1,1.E-12)
```

```
      EPS2=AMAX1(EP2,1.E-20)
      IBEG=KN+1
      IEND=KN+N
C
      DO 100 I=IBEG,IEND
      KOUNT=0
C     COMPUTE FIRST THREE ESTIMATES FOR ROOTS AS
C     RTS(I)+.5,RTS(I)-.5,RTS(I).
    1 H=.5
      RT=RTS(I)+H
      ASSIGN 10 TO NN
                                   GO TO 70
   10 DELFPR=FRTDEF
      RT=RTS(I)-H
      ASSIGN 20 TO NN
                                   GO TO 70
   20 FRTPRV=FRTDEF
      DELFPR=FRTPRV-DELFPR
      RT=RTS(I)
      ASSIGN 30 TO NN
                                   GO TO 70
   30 ASSIGN 80 TO NN
      LAMBDA=-0.5
C     COMPUTE NEXT ESTIMATE FOR ROOT
   40 DELF=FRTDEF-FRTPRV
      DFPRLM=DELFPR*LAMBDA
      NUM=-FRTDEF*(1.+LAMBDA)*2
      G=(1.+LAMBDA*2.)*DELF-LAMBDA*DFPRLM
      SQR=G*G+2.*NUM*LAMBDA*(DELF-DFPRLM)
      IF(FNREAL.AND.REAL(SQR).LT.0.)SQR=0.
      SQR=CSQRT(SQR)
      DEN=G+SQR
      IF(REAL(G)*REAL(SQR)+AIMAG(G)*AIMAG(SQR).LT.0.)
     +DEN=G-SQR
      IF(CABS(DEN).EQ.0.) DEN=1.
```

```
      LAMBDA=NUM/DEN
      FRTPRV=FRTDEF
      DELFPR=DELF
      H=H*LAMBDA
      RT=RT+H
      IF(KOUNT.GT.MAXIT)          GO TO 100
C
   70 KOUNT=KOUNT+1
      CALL FN(RT,FRT)
      FRTDEF=FRT
      IF(I.LT.2)                 GO TO 75
      DO 71 J=2,I
      DEN = RT- RTS(J-1)
      IF(CABS(DEN).LT.EPS2)       GO TO 79
   71 FRTDEF=FRTDEF/DEN
   75                             GO TO NN,(10,20,30,80)
   79 RTS(I)=RT+.001
                                  GO TO 1
C     CHECK FOR CONVERGENCE
   80 IF (CABS(H).LT.EPS1+CABS(RT)) GO TO 100
      IF (AMAX1(CABS(FRT),CABS(FRTDEF)).LT.EPS2) GO TO 100
C
C     CHECK FOR DIVERGENCE
      IF (CABS(FRTDEF).LT.10.+CABS(FRTPRV)) GO TO 40
       H=H/2.
       LAMBDA=LAMBDA/2.
       RT=RT-H
                                      GO TO 70
  100 RTS(I)=RT
      RETURN
      END
```

```
      SUBROUTINE ZEROS
C     THE FOLLOWING IS A LEDGEND OF THE ARGUMENT OF THE SUBROUTINE
C     MULLER.
C         KN = NUMBER OF ROOTS PREVIOUSLY COMPUTED, NORMALLY ZERO.
C         N = NUMBER OF ROOTS DESIRED.
C         RTS = AN ARRAY OF INITIAL ESTIMATES OF ALL DESIRED ROOTS;
C               SET TO ZERO IF NO BETTER ESTIMATES ARE AVAILABLE.
C         MAXIT = MAXIMUM NUMBER OF ITERATIONS PER ROOT ALLOWED.
C         EP1 = RELATIVE ERROR TOLERANCE ON X(I).
C         EP2 = ERROR TOLERANCE ON F(X(I)).
C         FN = FN(Z,FZ) IS A SUBROUTINE WHICH, FOR GIVEN Z, RETURNS
C               FZ = F(Z).
C         FNREAL = A LOGICAL VARIBLE; IF TRUE, THE SUBROUTINE FORCES
C         ALL APPROXIMATIONS TH ALL THE ROOTS TO BE REAL.  THIS MAKES
C         IT POSSIBLE TO USE THIS ROUTINE EVEN IF F(X) IS DEFINED
C         ONLY FOR REAL X.
      EXTERNAL MULLER, FN
      REAL EP1, ETS, MODRT, REALC, IMAGC
      INTEGER M, N
      INTEGER KN, MAXIT
      COMPLEX RTS(18), FZ
      COMMON /AREA 5/ M, N
      LOGICAL FNREAL
      FNREAL = .FALSE.
      DO 10 I = 1, N
           RTS(I) = 0.
   10 CONTINUE
      EP1 = 1.E-04
      EP2 = 1.E-06
      KN = 0
      MAXIT = 500
      WRITE (6,4000)
 4000 FORMAT ('0', 40X, 'MAXIUM NUMBER OF ITERATIONS', 9X, 'EP1', 9X,'EP
     +2')
      WRITE (6,5000) MAXIT, EP1, EP2
 5000 FORMAT ('0', 52X, I4, 14X, E10.4, 5X, E10.4)
```

```
      CALL MULLER(KN,N,RTS,MAXIT,EP1,EP2,FN,FNREAL)
      WRITE (6,2000)
 2000 FORMAT ('0', 25X, 'ROOTS OF DENOMINATOR.', 20X, 'F(ZERO)', 20X,'MO
     +DULUS')
      DO 20 I = 1, N
          CALL FN( RTS(I), FZ)
          MODRT = CABS(RTS(I))
          REALC = REAL(RTS(I))
          IMAGC = AIMAG(RTS(I))
          WRITE (6,1000) REALC, IMAGC, FZ, MODRT
 1000     FORMAT ('0', 25X,2F10.6, 15X, 2F10.6, 10X, F10.6)
   20 CONTINUE
      RETURN
      END
      SUBROUTINE FN(Z,FZ)
      REAL P(36), Q(36)
      COMPLEX Z, FZ, SUM
      INTEGER I, M, N, NN
      COMMON /AREA 2/ P, Q /AREA 5/ M, N
      NN = N + 1
      SUM = Q(1)
      DO 10 I = 1, N
         SUM = Q(I+1) + (Z**I) + SUM
   10 CONTINUE
      FZ = SUM
      RETURN
      END
```

```
      SUBROUTINE MINMOD
C     THIS PROGRAM COMPUTES AND PRINTS THE MINIMUM MODULUS
C     IN WHICH NO ZEROS OF THE Q(N) POLYNOMIAL WILL BE FOUND.
      REAL P(36), Q(36), COEF(36), B(36), BETA0, MIMOD
      REAL II, KK, MINB, NN
      INTEGER M, N, I, K
      COMMON /AREA 2/ P, Q /AREA 5/ M, N
C     COMPUTE  N!/(N-1)!*I! BY REFERING TO THE FUNCTION FACT.
      DO 10 I=1, N
           K = N - I
           KK = FLOAT(K)
           II = FLOAT(I)
           NN = FLOAT(N)
           COEF(I) = FACT(NN)/(FACT(KK)*FACT(II))
   10 CONTINUE
C     COMPUTE BETA0.
      K = N
      DO 20 I=1, K
          B(I)=(COEF(I)*(Q(1)/Q(I)))**(1/I)
   20 CONTINUE
      MINB = B(1)
      DO 30 I=2, N
          IF (B(I).LT.MINB) MINB = B(I)
   30 CONTINUE
      BETA0 = MINB
C     COMPUTE MINIMUM MODULUS.
      MIMOD = BETA0*((NN *EXP(1.0))**(-1.0))
      WRITE (6,1000) MIMOD
 1000 FORMAT ('0',45X, 'THE MINIMUM MODULUS OF THE DENOMINATOR IS ',2X,F
     +10.2)
      RETURN
      END
```

```
      REAL FUNCTION FACT(X)
      REAL X, SUM, XX
      INTEGER I, N
      IF (X .LT. 0.9) THEN DO
              FACT = 1.0
      ELSE DO
           IF (X .EQ. 1.0) THEN DO
                   FACT = X
           ELSE DO
                SUM = X
                XX = X - 1
                N = INT(XX)
                DO 10  I=1, N
                    SUM = SUM*(X-I)
                    FACT = SUM
10              CONTINUE
           END IF
      END IF
      RETURN
      END
```