63-3-2

REPORT NO.
TDR-169(3305)TN-2

SSD-TDR-63-15

Derivation of Approximating Polynomials for Function $1-e^V$

21 JANUARY 1963

Prepared by S. HERBERT LEWIS

Prepared for COMMANDER SPACE SYSTEMS DIVISION
UNITED STATES AIR FORCE
Inglewood, California

ENGINEERING DIVISION • AEROSPACE CORPORATION
CONTRACT NO. AF 04(695)-169

SSD-TDR-63-15

Report No.
TDR-169(3305)TN-2

DERIVATION OF APPROXIMATING POLYNOMIALS FOR FUNCTION $1-e^{v}$

by

S. Herbert Lewis

AEROSPACE CORPORATION
El Segundo, California

Contract No. AF 04(695)-169

21 January 1963

Prepared for

COMMANDER SPACE SYSTEMS DIVISION
UNITED STATES AIR FORCE
Inglewood, California

SSD-TDR-63-15

Report No.
TDR-169(3305)TN-2

DERIVATION OF APPROXIMATING POLYNOMIALS FOR FUNCTION $1-e^{v}$

Prepared by S. H. Lewis

S. H. Lewis
Computation Department
Engineering Division

Approved by T. R. Parkin

T. R. Parkin, Head
Computation Department
Engineering Division

D. R. Mc Coll

D. R. Mc Coll, Director
Sensing and Information
Engineering Subdivision

AEROSPACE CORPORATION
El Segundo, California

ABSTRACT

Formulas and tables are developed for compressing the power series representation of a function to obtain minimal order approximating polynomials maintaining specified accuracy throughout an assigned range of arguments. Coefficients for polynomials approximating the function $1-e^{v}$ are derived and scaled for use in a fixed point computer.

CONTENTS

I. INTRODUCTION

In certain launch vehicle guidance equations (Ref. 1) there occurs a function of the form

$$u(v) = 1 - e^{v} \tag{1}$$

Values of the argument lie within a range $-0.3 < v < 2.1$. It is required that any approximation used shall yield $u = 0$ when $v = 0$. The accuracy desired is such that the least significant bit of u should be 2^{-18}.

At present, u(v) is approximated by an eighth order polynomial derived by truncating the Maclaurin series for the exponential.

$$u^*(v) = -\sum_{i=1}^{8} \frac{v^i}{i!} \tag{2}$$

The approximation (2), however, provides the desired accuracy only when $v \leq 1$. At the upper end of the range of arguments, the significance deteriorates to about 2^{-8}.

This report gives the derivation of coefficients for two new approximating polynomials. One of these is, and the other may be, sufficiently accurate over the entire pertinent range of arguments. Section II describes a general method by which a long power series can be compressed to yield the approximating polynomial of minimum order to maintain specified accuracy throughout a given range of arguments. In Section III, the method is applied to the function defined in Equation (1). Section IV introduces considerations of scaling, necessary to retain significance while avoiding overflows in a fixed point computer, and convenient to enable bit-by-bit simulation of the computer operations for purposes of checking and error analysis. In Section V, coefficients are evaluated for both seventh and eighth order polynomials to approximate function (1) over a limited range of arguments. The errors of these approximations are compared in Section VI with those of the currently used approximation (2). In Section VII it is shown that the approximating polynomials derived in this report are preferable to Equation (2) for numerical differentiation of Equation (1).

II. COMPRESSION OF POWER SERIES

In Reference 2 it is shown that although a function may be expressed by a convergent infinite power series, there are other expressions more suitable for approximating the function within specified tolerance for a limited range of the argument. Of particular importance are the Chebyshev polynomials $T_i(x)$ and $T_i^*(x)$. The former are useful when $-1 \leq x \leq 1$, and the latter when $0 \leq x \leq 1$. Both have the property that, within the range of applicability, their maxima are +1 and their minima are -1. A series of Chebyshev polynomials converges more rapidly than any other series representing a function when arguments are restricted to the appropriate range.

In this report, the polynomials $T_i^*(x)$ are used, with $0 \leq x \leq 1$. The normalization of the original independent variable to this range is discussed in Section III.

$$T_i^*(x) = \sum_{j=0}^{i} t_{ij} x^j \tag{3}$$

where

$$\left.\begin{aligned} t_{00} &= \frac{1}{2} \\ t_{i0} &= (-1)^i \qquad [i \geq 1] \\ t_{ij} &= (-1)^{i+j}\, 2^{2j-1} \frac{i}{j} \binom{i+j-1}{i-j} \qquad [i \geq j \geq 1] \end{aligned}\right\} \tag{4}$$

In Reference 2, the definition of $T_0^*(x)$ is given special attention. For present purposes, however, the value of 1/2 is appropriate. Table II-1 lists t_{ij} for $0 \leq j \leq i \leq 7$.

It is further shown in Reference 2 that any integral power of x can be expressed as a finite series of Chebyshev polynomials.

$$x^n = 2^{-(2n-1)} \sum_{i=0}^{n} h_i(n)\, T_i^*(x) \tag{5}$$

where

$$h_i(n) = \binom{2n}{n-i} \qquad [0 \le i \le n] \tag{6}$$

Table II-2 lists $h_i(n)$ for $0 \le i \le 13$ and $7 \le n \le 13$.

If m is chosen $\le n$, then substituting (3) into (5) for $0 \le i \le m - 1$ gives

$$x^n = 2^{-(2n-1)} \left[\sum_{j=0}^{m-1} g_j(m,n)\, x^j + \sum_{i=m}^{n} h_i(n)\, T_i^*(x) \right] \tag{7}$$

where

$$g_j(m,n) = \sum_{i=j}^{m-1} h_i(n)\, t_{ij} \qquad [0 \le j \le m - 1 < n] \tag{8}$$

$g_j(7,n)$ and $g_j(8,n)$ are listed in Tables II-3 and II-4 for $n \le 13$.

Equation (7) shows how any power $n \ge m$ of x can be represented by a polynomial in x of order $m - 1$ and a linear combination of $T_i^*(x)$ with

$m \leq i \leq n$. The coefficients $g_j(m,n)$ and $h_i(n)$ are all integers. It should be observed that

$$\sum_{j=0}^{m-1} g_j(m,n) + \sum_{i=m}^{n} h_i(n) = 2^{2n-1} \tag{9}$$

If $0 \leq x \leq 1$, then the series

$$\sum_{i=0}^{\infty} c_i x^i$$

can be expressed as the sum of a polynomial in x, of some order $m - 1$, and a residual series of Chebyshev polynomials of order $\geq m$.

$$\sum_{i=0}^{\infty} c_i x^i = \sum_{i=0}^{m-1} d_{mi} x^i + \sum_{i=m}^{\infty} e_i T_i^*(x) \tag{10}$$

From (7),

$$\sum_{i=0}^{\infty} c_i x^i = \sum_{i=0}^{m-1} c_i x^i + \sum_{n=m}^{\infty} 2^{-(2n-1)} c_n \left[\sum_{i=0}^{m-1} g_i(m,n) x^i + \sum_{i=m}^{n} h_i(n) T_i^*(x) \right] \tag{11}$$

Equating coefficients of x^i and $T_i^*(x)$ in (10) and (11) gives

$$d_{mi} = c_i + \sum_{n=m}^{\infty} 2^{-(2n-1)} c_n g_i(m,n) \qquad [0 \leq i \leq m-1] \tag{12}$$

$$e_i = \sum_{n=i}^{\infty} 2^{-(2n-1)} c_n h_i(n) \qquad [m \leq i \leq n] \tag{13}$$

Equation (13) provides a basis for selecting the order, $m - 1$, of an approximating polynomial whose error shall not exceed a specified magnitude when $0 \leq x \leq 1$. Since, in this range, $|T_i^*(x)| \leq 1$, the error of approximation is limited by

$$\sum_{i=m}^{\infty} |e_i| \leq E_m = \sum_{n=m}^{\infty} 2^{-(2n-1)} |c_n| \sum_{i=m}^{n} h_i(n) \tag{14}$$

By trial and error, a minimal m can be found such that E_m, computed by (14), is sufficiently small. With m thus determined, the coefficients of the approximating polynomial can be found from (12).

In practice, the infinite sum in (12) can be replaced by a finite sum. If the power series

$$\sum_{i=0}^{\infty} c_i x^i$$

converges when $0 \leq x \leq 1$, then for some r the terms beyond $c_r x^r$ can be neglected. Equation (10) then becomes

$$\sum_{i=0}^{\infty} c_i x^i = \sum_{i=0}^{m-1} d'_{mi} x^i + \sum_{i=m}^{r} e'_i T_i^*(x) + \sum_{i=r+1}^{\infty} c_i x^i \tag{15}$$

The error of approximation is limited, for $0 \leq x \leq 1$, by

$$E_{mr} = \sum_{n=m}^{r} 2^{-(2n-1)} |c_n| \sum_{i=m}^{n} h_i(n) + \sum_{n=r+1}^{\infty} |c_n| \tag{16}$$

The m coefficients of the approximating polynomial are then

$$d'_{mi} = c_i + \sum_{n=m}^{r} 2^{-(2n-1)} c_n g_i(m, n) \qquad [0 \leq i \leq m - 1] \tag{17}$$

Table II-1. t_{ij} from Equation (4) and Ref. 2, Table VII

i	t_{i0}	t_{i1}	t_{i2}	t_{i3}	t_{i4}	t_{i5}	t_{i6}	t_{i7}
0	1/2							
1	-1	2						
2	1	-8	8					
3	-1	18	-48	32				
4	1	-32	160	-256	128			
5	-1	50	-400	1,120	-1,280	512		
6	1	-72	840	-3,584	6,912	-6,144	2,048	
7	-1	98	-1,568	9,408	-26,880	39,424	-28,672	8,192

Table II-2. $h_i(n) = \binom{2n}{n-i}$ from Equation (6)

i	$h_i(13)$	$h_i(12)$	$h_i(11)$	$h_i(10)$	$h_i(9)$	$h_i(8)$	$h_i(7)$
0	10,400,600	2,704,156	705,432	184,756	48,620	12,870	3,432
1	9,657,700	2,496,144	646,646	167,960	43,758	11,440	3,003
2	7,726,160	1,961,256	497,420	125,970	31,824	8,008	2,002
3	5,311,735	1,307,504	319,770	77,520	18,564	4,368	1,001
4	3,124,550	735,471	170,544	38,760	8,568	1,820	364
5	1,562,275	346,104	74,613	15,504	3,060	560	91
6	657,800	134,596	26,334	4,845	816	120	14
7	230,230	42,504	7,315	1,140	153	16	1
8	65,780	10,626	1,540	190	18	1	
9	14,950	2,024	231	20	1		
10	2,600	276	22	1			
11	325	24	1				
12	26	1					
13	1						

Table II-3. $g_j\ (7,n)$ from Equation (8)

n	$g_0(7,n)$	$g_1(7,n)$	$g_2(7,n)$	$g_3(7,n)$
7	1	-98	1,568	-9,408
8	15	-1,440	22,400	-129,024
9	136	-12,852	195,840	-1,096,704
10	969	-90,440	1,356,600	-7,441,920
11	5,985	-553,014	8,192,800	-44,241,120
12	33,649	-3,083,472	45,224,256	-241,196,032
13	177,100	-16,096,100	234,416,000	-1,237,716,480

n	$g_4(7,n)$	$g_5(7,n)$	$g_6(7,n)$
7	26,880	-39,424	28,672
8	345,660	-450,560	245,760
9	2,820,096	-3,446,784	1,671,168
10	18,604,800	-21,829,632	9,922,560
11	108,345,600	-123,594,240	53,932,032
12	581,454,720	-649,752,576	275,652,608
13	2,946,944,000	-3,241,638,400	1,347,174,400

Table II-4. $g_j(8,n)$ from Equation (8)

n	$g_0(8,n)$	$g_1(8,n)$	$g_2(8,n)$	$g_3(8,n)$
8	-1	128	-2,688	21,504
9	-17	2,142	-44,064	342,720
10	-171	21,280	-430,920	3,283,200
11	-1,330	163,856	-3,277,120	24,578,400
12	-8,855	1,081,920	-21,422,016	158,681,600
13	-53,130	6,446,440	-126,584,640	928,287,360

n	$g_4(8,n)$	$g_5(8,n)$	$g_6(8,n)$	$g_7(8,n)$
8	-84,480	180,224	-212,992	131,072
9	-1,292,544	2,585,088	-2,715,648	1,253,376
10	-12,038,400	23,113,728	-22,763,520	9,338,880
11	-88,281,600	164,792,320	-155,803,648	59,924,480
12	-561,052,800	1,025,925,120	-943,022,080	348,192,768
13	-3,241,638,400	5,834,949,120	-5,253,980,160	1,886,044,160

III. APPLICATION TO $u(v) = 1 - e^v$

In order to apply the method developed above to the function defined in equation (1), it is first necessary to transform the variable to one lying in the range between 0 and 1. If $b \leq v \leq a + b$, then

$$x = \frac{v - b}{a} \tag{18}$$

Since it is required that the polynomial approximation $u_m(v)$ vanish when $v = 0$, and since it is further desirable that the approximation be most accurate in the neighborhood of $v = 0$, the process of Section II is applied to the function

$$\frac{u(v)}{-v} = \frac{1 - e^{(ax+b)}}{-(ax + b)} = \sum_{n=0}^{\infty} \frac{(ax + b)^n}{(n + 1)!} = \sum_{i=0}^{\infty} c_i x^i \qquad [0 \leq x \leq 1] \tag{19}$$

Expanding $(ax + b)^n$ by the binomial theorem, it follows that

$$c_i = \frac{a^i}{i!} \sum_{j=0}^{\infty} \frac{b^j}{(i + j + 1)(j!)} \tag{20}$$

A value of r is easily chosen such that

$$v \sum_{i=r+1}^{\infty} |c_i|$$

can be neglected for $b \leq v \leq a + b$. Application of equation (16) leads to a selection of m which keeps vE_{mr} within tolerance. m is the number of coefficients in the approximating polynomial. Then, using (17) to determine d'_{mi},

$$u_m(v) = -v \sum_{i=0}^{m-1} d'_{mi} \left(\frac{v - b}{a}\right)^i = -v \sum_{j=0}^{m-1} k_{mj} v^j \qquad (21)$$

From

$$(v - b)^i = \sum_{j=0}^{i} \binom{i}{j} (-b)^{i-j} v^j$$

It follows that

$$k_{mj} = (-b)^{-j} \sum_{i=j}^{m-1} \binom{i}{j} \left(-\frac{b}{a}\right)^i d'_{mi} \qquad [0 \leq j \leq m - 1] \qquad (22)$$

Note that if $b = 0$, then $c_i = a^i/(i + 1)!$ and $k_{mj} = a^{-j} d'_{mj}$.

It must be emphasized that the accuracy of approximating $u(v)$ by $u_m(v)$ deteriorates very rapidly when v exceeds the range whose limits have been used in determining the coefficients k_{mj}.

IV. COMPUTER SCALING

The polynomials $u_m(v)$ are to be evaluated in a fractional binary computer with word length of 23 magnitude bits and a sign bit. A rounded multiply operation is available.

In programming this computer, it is customary to assign a scale factor $s(N)$ to a number N such that $|N| < 2^s$. In a word representing N, the value of the least significant bit is 2^{s-23}. Numbers to be added must be scaled alike. The scale of a product is the sum of the scales of the factors.

In order to realize the advantage of rounded multiplication, there should be no shifting of intermediate products in a program for polynomial evaluation. It follows from (21) and the arithmetic of scale factors that

$$s(k_{mj}) = s(u_m) - (j+1)s(v) \tag{23}$$

Maximum precision and the avoidance of overflow in intermediate computations require, in the present application, that

$$\left.\begin{aligned} s(v) &= 2 \\ s(u_m) &= 5 \\ s(k_{mj}) &= 3 - 2j \end{aligned}\right\} \tag{24}$$

The bit configuration in a computer word for a number N with a scale factor s is the same as that for the number $2^{s'-s}$ N with a scale factor s'. Instead of computing k_{mj} from (22) and assigning individual scale factors from (24), it is preferable to transform the polynomial coefficients to integers which can all be scaled with $s = 23$. The advantage lies in the greater convenience with which the computer operations, including round-off, can be simulated bit by bit for purposes of error analysis and program checking.

Let

$$\left.\begin{array}{ll} K_{mj} = 2^{20+2j} k_{mj}, & s(K_{mj}) = 23 \\ U_m = 2^{18} u_m, & s(U_m) = 23 \\ w = 2^{-2} v, & s(w) = 0 \end{array}\right\} \qquad (25)$$

Then

$$U_m(w) = -w \sum_{j=0}^{m-1} K_{mj} w^j \qquad (26)$$

where U_m with $s = 23$ is the same configuration of digits as u_m with $s = 5$.

V. EVALUATION OF COEFFICIENTS

The range $-0.3 < v < 2.1$ includes values beyond all those to be expected in the current application. Therefore, $a = 2.4$; $b = -0.3$. With these parameters, the c_i are computed by equation (20) and listed in Table V-1 for $0 \leq i \leq 17$.

Accuracy to the order of 2^{-18} is desired in the approximation to u(v). Since

$$2.1 \sum_{i=14}^{\infty} c_i < 2^{-21} \quad ,$$

while

$$2.1 \sum_{i=13}^{\infty} c_i \text{ is nearly } 2^{-19} \quad ,$$

the value of $r = 13$ is used for termination of the sums in equations (16) and (17).

From (16),

$$2.1\ E_{7,13} = 754 \cdot 10^{-8} < 2^{-17}$$

$$2.1\ E_{8,13} = 78 \cdot 10^{-8} < 2^{-20}$$

It is evident that $m = 8$ keeps the error well below the specified tolerance, while $m = 7$ may be acceptable if slightly larger errors can be tolerated when $v > 1$. The difference between $r + 1 = 14$ and $m = 7$ or 8 measures the benefit gained by the series compression method of Section II in minimizing the order of an approximating polynomial satisfactory throughout the range of arguments.

Values of d'_{mi} computed by (17), with m = 7 and m = 8, are listed in Table V-2.

From (25) and (22), with a = 2.4, b = -0.3,

$$K_{mj} = 2^{20+2j} (0.3)^{-j} \sum_{i=j}^{m-1} 2^{-3i} \binom{i}{j} d'_{mi} \qquad [0 \leq j \leq m-1] \qquad (27)$$

Values of K_{mj} computed by (27), with m = 7 and m = 8 are listed in Table V-3.

Table V-1. c_i from Equation (20) with a = 2.4, b = -0.3

i	c_i
0	.86393 92644
1	.98496 83497
2	.76789 18791
3	.45365 10976
4	.21551 84202
5	.08557 59883
6	.02917 93574
7	.00871 64991
8	.00231 64848

i	c_i
9	.00055 44093
10	.00012 06818
11	.00002 40891
12	.00000 44398
13	.00000 07600
14	.00000 01214
15	.00000 00182
16	.00000 00026
17	.00000 00003

Table V-2. d'_{mi} from Equation (17) with r = 13, m = 7 and 8

i	d'_{8i}	d'_{7i}
0	.86393 90612	.86394 22776
1	.98499 39577	.98467 87486
2	.76736 44347	.77240 77891
3	.45776 39967	.42750 38699
4	.19992 35730	.28638 10778
5	.11713 94957	.00966 48447
6	-.00501 19813	.08720 93573
7	.02634 89539	

Table V-3. K_{mj} from Equation (27) with m = 7 and 8

j	K_{8j}	K_{7j}
0	1,048,576	1,048,577
1	2,097,148	2,097,259
2	2,796,363	2,795,331
3	2,796,485	2,779,653
4	2,214,973	2,433,572
5	1,645,497	751,671
6	405,518	1,959,997
7	986,971	

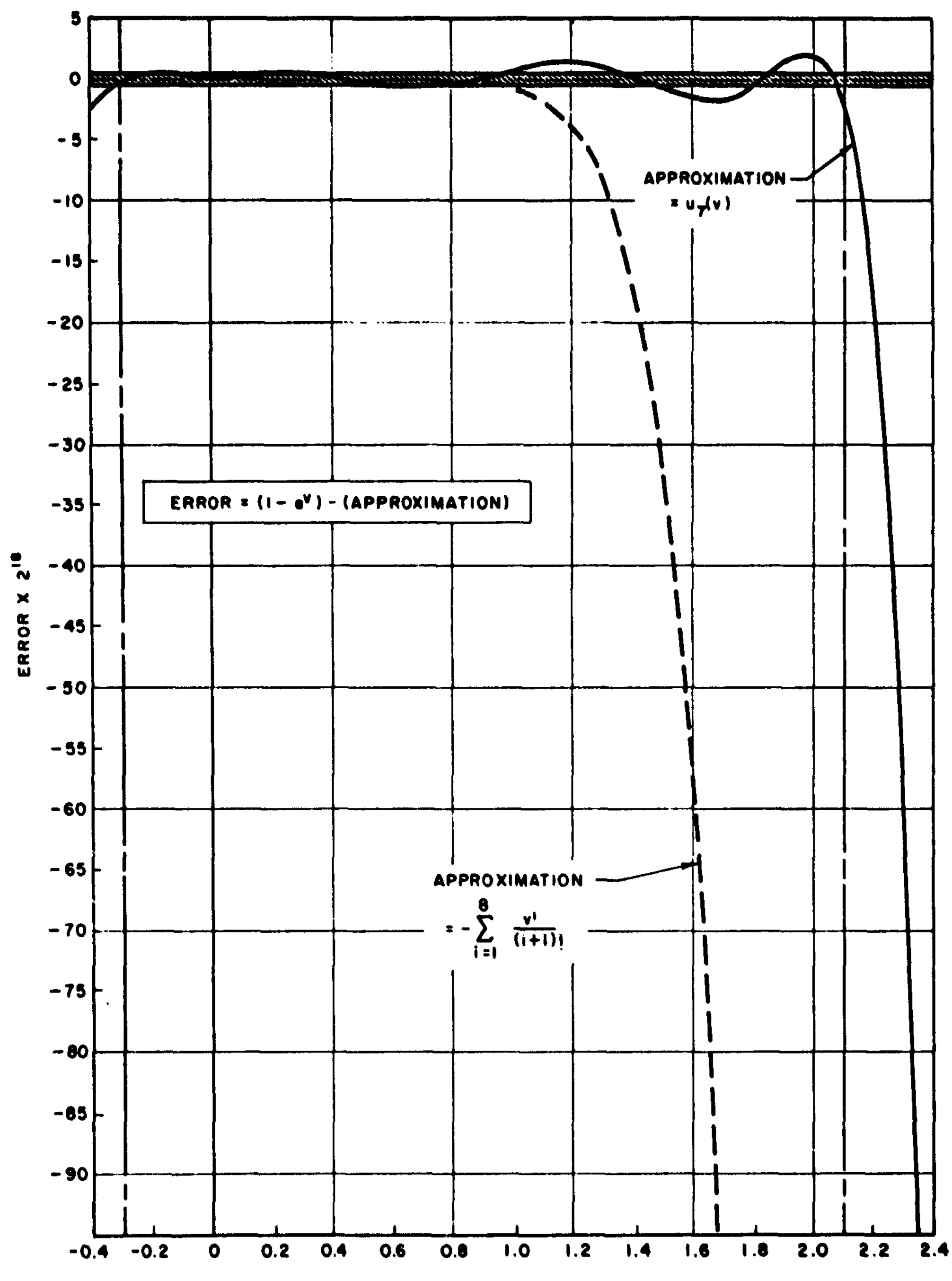

Figure VI-1. Approximation Errors

VI. ERROR ANALYSIS

Calculations of the seventh order polynomial U_7, equation (26), were made with K_{7j} from Table V-3. All multiplications were rounded, to simulate computer operation. The differences between these results and $2^{18}(1-e^v)$ are plotted in Figure VI-1. When $-0.3 \leq v < 1$, the errors lie virtually within the shaded region of round-off error, $\pm 1/2 \cdot 2^{-18}$. When $1 \leq v \leq 2.1$, the oscillatory error function never exceeds $2.5 \cdot 2^{-18}$ in magnitude.

In contrast, the eighth order truncated Maclaurin series approximation departs rapidly from u(v) when $v > 1$. The magnitude of its error reaches 2^{-16} when $v = 1.2$, 2^{-13} when $v = 1.5$, and 2^{-10} when $v = 1.9$.

Errors of the eighth order approximating polynomial U_8 are wholly obscured by round-off, and therefore are not shown in Figure VI-1.

The warning is repeated that approximating polynomials derived by the methods of this report cannot be expected to remain valid outside the range of arguments for which they are specified.

VII. NUMERICAL DIFFERENTIATION

Reference 1 recommends that, since du/dt is used in the application, u should be approximated by truncation of the exponential series. However, the polynomials derived in this report improve the approximation to the derivative as well as the approximation to the function itself.

If an approximation to u(v) is denoted by u*(v) and the error by $\epsilon(v)$, then

$$u(v_i) = u^*(v_i) + \epsilon(v_i) \tag{28}$$

$$u(v_2) - u(v_1) = u^*(v_2) - u^*(v_1) + [\epsilon(v_2) - \epsilon(v_1)] \tag{29}$$

In words, if the differentiation is to be approximated by differencing, then the error of a difference is given by the difference of the errors.

It is evident from Figure VI-1 that in the region $1 < v < 2.1$, the slope of the error curve for the truncated Maclaurin series is significantly greater in magnitude than the slope of the error curve for $u_7(v)$. Therefore, even the seventh order polynomial derived here gives a better approximation for the derivative than does the series truncated to eighth order.

REFERENCES

1. Mac Pherson, D. "An Explicit Solution to the Powered Flight Dynamics of a Rocket Vehicle." Aerospace Corporation, Rpt. TDR-169(3126)TN-2, 31 October 1962.

2. Lanczos, Cornelius. Applied Analysis, Chapter VII and Tables VII and VIII. Prentice-Hall, Inc., 1961.

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. DERIVATION OF APPROXIMATING POLYNOMIALS FOR FUNCTION $1-e^v$. Prepared by S. Herbert Lewis. 21 January 1963. [25]p. incl. illus.
(Report TDR-169(3305)TN-2)
(Contract AF 04(695)-169) Unclassified report

Formulas and tables are developed for compressing the power series representation of a function to obtain minimal order approximating polynomials maintaining specified accuracy throughout an assigned range of arguments. Coefficients for polynomials approximating the function $1-e^v$ are derived and scaled for use in a fixed point computer.

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. DERIVATION OF APPROXIMATING POLYNOMIALS FOR FUNCTION $1-e^v$. Prepared by S. Herbert Lewis. 21 January 1963. [25]p. incl. illus.
(Report TDR-169(3305)TN-2)
(Contract AF 04(695)-169) Unclassified report

Formulas and tables are developed for compressing the power series representation of a function to obtain minimal order approximating polynomials maintaining specified accuracy throughout an assigned range of arguments. Coefficients for polynomials approximating the function $1-e^v$ are derived and scaled for use in a fixed point computer.

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. DERIVATION OF APPROXIMATING POLYNOMIALS FOR FUNCTION $1-e^v$. Prepared by S. Herbert Lewis. 21 January 1963. [25]p. incl. illus.
(Report TDR-169(3305)TN-2)
(Contract AF 04(695)-169) Unclassified report

Formulas and tables are developed for compressing the power series representation of a function to obtain minimal order approximating polynomials maintaining specified accuracy throughout an assigned range of arguments. Coefficients for polynomials approximating the function $1-e^v$ are derived and scaled for use in a fixed point computer.

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. DERIVATION OF APPROXIMATING POLYNOMIALS FOR FUNCTION $1-e^v$. Prepared by S. Herbert Lewis. 21 January 1963. [25]p. incl. illus.
(Report TDR-169(3305)TN-2)
(Contract AF 04(695)-169) Unclassifed report

Formulas and tables are developed for compressing the power series representation of a function to obtain minimal order approximating polynomials maintaining specified accuracy throughout an assigned range of arguments. Coefficients for polynomials approximating the function $1-e^v$ are derived and scaled for use in a fixed point computer.

UNCLASSIFIED