Numerical Methods Preliminaries

Professor PhD Henry Arguello Fuentes

Universidad Industrial de Santander
Colombia

May 19, 2020

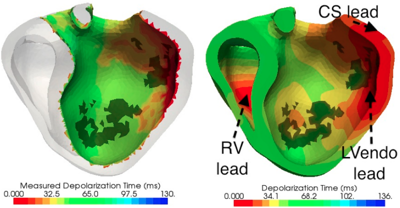
High Dimensional Signal Processing Group
www.hdspgroup.com
henarfu@uis.edu.co
LP 304

Universidad Industrial de Santander

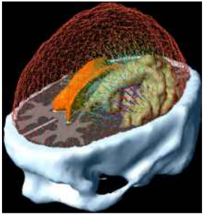
Outline

Introduction: numerical methods applications

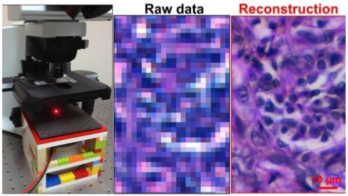
(a) Model the probable evolution of a pathology



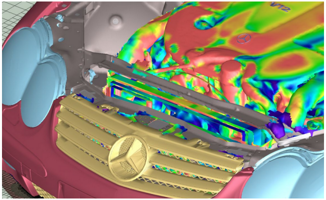
(b) Model and simulate the growth of a tumor



(c) Microscopy super-resolution



(d) Thermal management



Contents

Base 10 numbers: Expanded form of the number 1563

$$1563 = (1 \times 10^3) + (5 \times 10^2) + (6 \times 10^1) + (3 \times 10^0).$$

Base 2 numbers

Base 10 numbers: Expanded form of the number 1563

$$1563 = (1 \times 10^3) + (5 \times 10^2) + (6 \times 10^1) + (3 \times 10^0).$$

Let N denote a positive integer; then the digits $a_0, a_1, ..., a_k$ exist so that N has the base 10 expansion

Base 10 expansion

$$N = (a_k \times 10^k) + (a_{k-1} \times 10^{k-1}) + \cdots + (a_1 \times 10^1) + (a_0 \times 10^0), \quad (1)$$

Where the digits a_k are chosen from $0, 1, ..., 8, 9$.

Base 2 numbers: Expanded form of the number 1563

$$\begin{aligned}1563 =&(1\times 2^{10})+(1\times 2^{9})+(0\times 2^{8})+(0\times 2^{7})+(0\times 2^{6})+(0\times 2^{5})+\\&(1\times 2^{4})+(1\times 2^{3})+(0\times 2^{2})+(1\times 2^{1})+(1\times 2^{0}).\end{aligned}$$

So that:

$$1563 = 1024 + 512 + 16 + 8 + 2 + 1.$$

Base 2 numbers

Let N denote a positive integer; the digits $b_0, b_1, ..., b_J$ exist so that N has the base 2 expansion

Base 2 expansion

$$N = (b_J \times 2^J) + (b_{J-1} \times 2^{J-1}) + \cdots + (b_1 \times 2^1) + (b_0 \times 2^0), \quad (2)$$

Where each digit b_j is either a 0 or 1. Thus N is expressed in binary notation as

$$N = b_J b_{J-1} \cdots b_2 b_1 b_{0two}. \quad (3)$$

Contents

Base 2 representation of the integer N

Process: Generate sequences Q_k and R_k of quotients and remainders, respectively. End the process when $Q_k = 0$, for some integer $k = J$.

Process: Generate sequences Q_k and R_k of quotients and remainders, respectively. End the process when $Q_k = 0$, for some integer $k = J$.

Example:

k	1563	Q_k	R_k
0	1563/2=	781	1
1	781/2=	390	1
2	390/2=	195	0
3	195/2=	97	1
4	97/2=	48	1
5	48/2=	24	0
6	24/2=	12	0
7	12/2=	6	0
8	6/2=	3	0
9	3/2=	1	1
10	1/2=	0	1

1	1	0	0	0	0	1	1	0	1	1
b_{10}	b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0

Most Significant Bit -**MSB**

Least Significant Bit - **LSB**

Base 2 representation of the integer N

Exercise 1: Find the base 2 representation of 697

Base 2 representation of the integer N

Exercise 1: Find the base 2 representation of 697

- Start by dividing the integer N from 2 to calculate Q_0 and R_0.

$$697/2 = 348.5 \rightarrow Q_0 = 348 \text{ and } R_0 = 1$$

Exercise 1: Find the base 2 representation of 697

- Start by dividing the integer N from 2 to calculate Q_0 and R_0.

$$697/2 = 348.5 \rightarrow Q_0 = 348 \text{ and } R_0 = 1$$

- Continue the process until finding $Q_k = 0$, for some integer $k = J$.

$$Q_k = Q_{k-1}/2$$

k	697	Q_k	R_k
0	697/2=	348	1
1			
2			
3			
4			
5			
6			
7			
8			
9			

b_9 b_8 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0

Base 2 representation of the integer N

Solution

k	697	Q_k	R_k
0	697/2=	348	1
1	348/2=	174	0
2	174/2=	87	0
3	87/2=	43	1
4	43/2=	21	1
5	21/2=	10	1
6	10/2=	5	0
7	5/2=	2	1
8	2/2=	1	0
9	1/2=	0	1

1	0	1	0	1	1	1	0	0	1
b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0

Then, $697_{10} = 1010111001_2$

Contents

Commonly, when you express a rational number in decimal form, you require infinitely many digits.

For example, in $\dfrac{1}{3} = 0.\overline{3}$, the symbol $\overline{3}$ means that the digit 3 is repeated forever to form an infinite repeating decimal.

But, the number $\dfrac{1}{3}$ is the shorthand notation for the infinite series S

$$S = (3 \times 10^{-1}) + (3 \times 10^{-2}) + \cdots + (3 \times 10^{-\infty})$$

$$S = \sum_{k=1}^{\infty} 3(10)^{-k} = \frac{1}{3}.$$

Definition 1.

The infinite series S

$$S = \sum_{n=0}^{\infty} cr^n = c + cr + cr^2 + \cdots + cr^n + \cdots , \tag{4}$$

where $c \neq 0$ and $r \neq 0$, is called a *geometric series* with ratio r.

Definition 1.

The infinite series S

$$S = \sum_{n=0}^{\infty} cr^n = c + cr + cr^2 + \cdots + cr^n + \cdots , \tag{4}$$

where $c \neq 0$ and $r \neq 0$, is called a *geometric series* with ratio r.

Theorem 1. (Geometric Series)

The geometric series has the following properties:

$$\begin{aligned} &\text{If } \; |r| < 1, \;\; \text{then } \; \sum_{n=0}^{\infty} cr^n = \frac{c}{1-r}. \\ &\text{If } \; |r| > 1, \;\; \text{then the series diverges.} \end{aligned} \tag{5}$$

Example: The series S is given by

$$S = (7)\left(\frac{1}{7}\right)^1 + (7)\left(\frac{1}{7}\right)^2 + \cdots + (7)\left(\frac{1}{7}\right)^\infty = \sum_{n=1}^{\infty} 7\left(\frac{1}{7}\right)^n,$$

Example: The series S is given by

$$S = (7)\left(\frac{1}{7}\right)^1 + (7)\left(\frac{1}{7}\right)^2 + \cdots + (7)\left(\frac{1}{7}\right)^\infty = \sum_{n=1}^{\infty} 7\left(\frac{1}{7}\right)^n,$$

which is equal to $-7 + \sum_{n=0}^{\infty} 7\left(\frac{1}{7}\right)^n,$

Example: The series S is given by

$$S = (7)\left(\frac{1}{7}\right)^1 + (7)\left(\frac{1}{7}\right)^2 + \cdots + (7)\left(\frac{1}{7}\right)^\infty = \sum_{n=1}^{\infty} 7\left(\frac{1}{7}\right)^n,$$

which is equal to $-7 + \sum_{n=0}^{\infty} 7\left(\frac{1}{7}\right)^n,$

and acording with (5) $S = -7 + \dfrac{7}{1 - \dfrac{1}{7}} = \dfrac{7}{6} = 1.1\overline{6},$

Then, $\dfrac{7}{6}$ is the shorthand notation for the infinite series S

Contents

Binary Fractions

A binary fraction is a serie of sums with negative powers of 2, which is used to express a real number R that lies in the range $0 < R < 1$.

Binary Fractions

A binary fraction is a serie of sums with negative powers of 2, which is used to express a real number R that lies in the range $0 < R < 1$.

Binary fractions

$$R = (d_1 \times 2^{-1}) + (d_2 \times 2^{-2}) + \cdots + (d_n \times 2^{-n}) + \cdots , \tag{6}$$

where $d_j \in 0, 1$ and $0 < R < 1$.

Binary fraction

$$R = 0.d_1 d_2 \cdots d_n \cdots_{two}$$

Representation of R

$$R = \sum_{j=1}^{\infty} d_j (2)^{-j}$$

Binary Fractions-Decimal to binary

Process: Generate sequences d_k and F_k multiplying by two.

Binary Fractions-Decimal to binary

Process: Generate sequences d_k and F_k multiplying by two.
Example:

	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	
0.	1	0	1	1	0	0	1	1	0	...

j	0.7	F_j	d_j	$frac$
1	$(0.7)(2) =$	1.4	1	0.4
2	$(0.4)(2) =$	0.8	0	0.8
3	$(0.8)(2) =$	1.6	1	0.6
4	$(0.6)(2) =$	1.2	1	0.2
5	$(0.2)(2) =$	0.4	0	0.4
6	$(0.4)(2) =$	0.8	0	0.8
7	$(0.8)(2) =$	1.6	1	0.6
8	$(0.6)(2) =$	1.2	1	0.2
9	$(0.2)(2) =$	0.4	0	0.4
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

$$0.7 = 0.1\overline{0110}_2$$

Exercise 2: Calculate the binary fraction for 0.6.

- Start by multiplying 0.6 by 2, to generate sequences d_j and F_j

d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9 ...

j	0.6	F_j	d_j	$frac$
1	$(0.6)(2) =$	1.2	1	0.2
2				
3				
4				
5				
6				
7				
8				
9				
⋮	⋮	⋮	⋮	⋮

Solution

$d_1 \quad d_2 \quad d_3 \quad d_4 \quad d_5 \quad d_6 \quad d_7 \quad d_8 \quad d_9 \quad \ldots$

j	0.6	F_j	d_j	$frac$
1	$(0.6)(2) =$	1.2	1	0.2
2	$(0.2)(2) =$	0.4	0	0.4
3	$(0.4)(2) =$	0.8	0	0.8
4	$(0.8)(2) =$	1.6	1	0.6
5	$(0.6)(2) =$	1.2	1	0.2
6	$(0.2)(2) =$	0.4	0	0.4
7	$(0.4)(2) =$	0.8	0	0.8
8	$(0.8)(2) =$	1.6	1	0.6
9	$(0.6)(2) =$	1.2	1	0.2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

$$0.6 = 0.\overline{1001}$$

Binary Fractions-Binary to decimal

The base 10 rational number R_{10} associated to a base 2 binary fraction R_2 can be found using geometric series.

Binary Fractions-Binary to decimal

The base 10 rational number R_{10} associated to a base 2 binary fraction R_2 can be found using geometric series.

Example:

$$0.\overline{01}_2 = (0 \times 2^{-1}) + (1 \times 2^{-2}) + (0 \times 2^{-3}) + (1 \times 2^{-4}) \cdots$$

the expression above is writted as

Binary Fractions-Binary to decimal

The base 10 rational number R_{10} associated to a base 2 binary fraction R_2 can be found using geometric series.

Example:

$$0.\overline{01}_2 = (0 \times 2^{-1}) + (1 \times 2^{-2}) + (0 \times 2^{-3}) + (1 \times 2^{-4}) \cdots$$

the expression above is writted as

$$= \sum_{k=1}^{\infty} (2^{-2})^k = -1 + \sum_{k=0}^{\infty} (2^{-2})^k$$

$$= -1 + \frac{1}{1 - \frac{1}{4}} = -1 + \frac{2}{3} = \frac{1}{3}.$$

Binary Fractions-Binary to decimal

The base 10 rational number R_{10} associated to a base 2 binary fraction R_2 can be found using geometric series.

Example:

$$0.\overline{01}_2 = (0 \times 2^{-1}) + (1 \times 2^{-2}) + (0 \times 2^{-3}) + (1 \times 2^{-4}) \cdots$$

the expression above is writted as

$$= \sum_{k=1}^{\infty} (2^{-2})^k = -1 + \sum_{k=0}^{\infty} (2^{-2})^k$$

$$= -1 + \frac{1}{1 - \frac{1}{4}} = -1 + \frac{2}{3} = \frac{1}{3}.$$

then, $\frac{1}{3}$ is the 10 rational number associated to $0.\overline{01}_2$

Contents

Binary shifting

Let R be

$$R = 0.00000\overline{11000}_2. \tag{7}$$

Binary shifting

Let R be

$$R = 0.00000\overline{11000}_2. \tag{7}$$

Multiplying both sides of (7) by $2^5 = 32$ will **shift** the binary point **5** places to the right	Multiplying both sides of (7) by $2^{10} = 1024$ will **shift** the binary point **10** places to the right
$32R = 0.\overline{11000}_2.$	$1024R = 11000.\overline{11000}_2.$

Binary shifting

Let R be

$$R = 0.00000\overline{11000}_2. \tag{7}$$

Multiplying both sides of (7) by $2^5 = 32$ will **shift** the binary point **5** places to the right	Multiplying both sides of (7) by $2^{10} = 1024$ will **shift** the binary point **10** places to the right
$32R = 0.\overline{11000}_2.$	$1024R = 11000.\overline{11000}_2.$

Taking the difference $1024R - 32R = 11000.\overline{11000}_2 - 0.\overline{11000}_2,$

we obtain $992R = 11000_2,$

given that $11000_2 = 24_{10}$ we find that,

$$992R = 24, \text{ Therefore } R = \frac{3}{124}_{10}.$$

Contents

Scientific Notation

The scientific notation is a standard way to present a real number. It is obtained by properly shifting the decimal point.

Scientific Notation

The scientific notation is a standard way to present a real number. It is obtained by properly shifting the decimal point.

Examples

- $0.0000747 = 7.47 \times 10^{-5}$
- $31.4159265 = 3.14159265 \times 10$
- $9,700,000.000 = 9.7 \times 10^{9}$
- The Avogadro's constant used in chemistry $= 6.02252 \times 10^{23}$.
- The quantity $1K = 1.024 \times 10^{3}$ used in computer science.

Contents

Machine Numbers

A mathematical quantity x is stored in a computer as a binary approximation given by

$$x \approx \pm q \times 2^n. \tag{8}$$

- The finite binary number q is the **mantissa**, where $1/2 \leq q \leq 1$.
- The integer n is the **exponent**.

Floating-point format

A real number is stored in a computer as a set of binary numbers expressing:

- The sign
- The exponent
- The mantissa

Sign	*Exponent*	*Mantissa*

- The sign is always one bit where, $S = 0$ if, $x > 0$ and $S = 1$, if $x < 0$.
- The amount of bits for the exponent and the mantissa depends on the precision of the machine.

Floating-point format-IEEE 754 standard

Precision	Total	Sign	Exponent	Mantissa	Exponent bias
Single	32 bits	1 bit	8 bits	23 bits	127
Double	64 bits	1 bit	11 bits	52 bits	1023

Floating-point format-IEEE 754 standard

Precision	Total	Sign	Exponent	Mantissa	Exponent bias
Single	32 bits	1 bit	8 bits	23 bits	127
Double	64 bits	1 bit	11 bits	52 bits	1023

Note: Biasing is done because exponents have to be signed values to be able to represent both tiny and huge values, but two's complement.

Floating-point format-IEEE 754 standard

Precision	Total	Sign	Exponent	Mantissa	Exponent bias
Single	32 bits	1 bit	8 bits	23 bits	127
Double	64 bits	1 bit	11 bits	52 bits	1023

Note: Biasing is done because exponents have to be signed values to be able to represent both tiny and huge values, but two's complement.

Then, the exponent is biased by adjusting its value.

Floating-point format-IEEE 754 standard

Precision	Total	Sign	Exponent	Mantissa	Exponent bias
Single	32 bits	1 bit	8 bits	23 bits	127
Double	64 bits	1 bit	11 bits	52 bits	1023

Note: Biasing is done because exponents have to be signed values to be able to represent both tiny and huge values, but two's complement.

Then, the exponent is biased by adjusting its value.

The exponent bias is calculated as $bias = 2^{exp-1} - 1$, where exp indicates the amount of bits for the exponent.

Example:
if $exp = 15$ bits, then, $bias = 2^{15-1} - 1 = 16383$

Possible cases:

Sign (S)	Exponent (E)	Mantissa (M)	Value
0-1	All 0 < E < All 1	M	$(-1)^S\,(1.M)(2^{E\text{-bias}})$
0	E=all 1	M=0	$+\infty$
1	E=all 1	M=0	$-\infty$
0-1	E=all 1	$M\neq0$	NaN
0-1	E=all 0	M=0	0
0-1	E=all 0	$M\neq0$	$(-1)^S\,(0.M)(2^{1\text{-bias}})$

Floating-point format

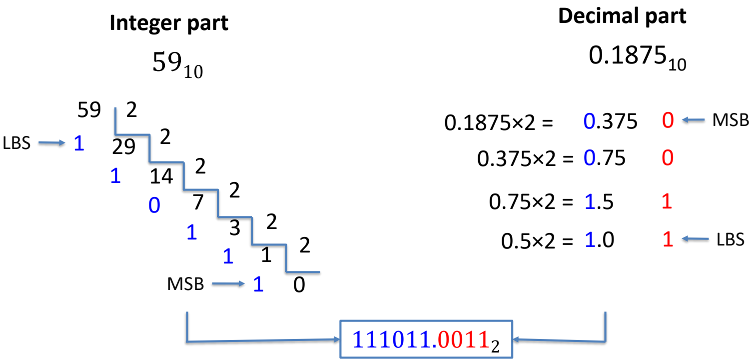
Example: Determine the floating point format to stored the number 59.1875_{10} in a computer with 32 bits of precision.

Floating-point format

Example: Determine the floating point format to stored the number 59.1875_{10} in a computer with 32 bits of precision.

- 1. Find the binary representation of the number 59.1875_{10}

$$59.1875_{10}$$



Floating-point format

- 2. Do the proper binary shifting

$$111011.0011_2 = 1.1101100011_2 \times 2^5$$

Floating-point format

- 2. Do the proper binary shifting

$$111011.0011_2 = 1.1101100011_2 \times 2^5$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

Floating-point format

- 2. Do the proper binary shifting

$$111011.0011_2 = 1.110110011_2 \times 2^5$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

- 4. Determine the mantissa

$$\text{Mantissa} = 110110011_2$$

Floating-point format

- 2. Do the proper binary shifting

$$111011.0011_2 = 1.110110011_2 \times 2^5$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

- 4. Determine the mantissa

$$\text{Mantissa} = 110110011_2$$

- 5. Determine the exponent

$$exp = 5 + bias = 5 + 127 = 132_{10} = 10000100_2$$

Floating-point format

- 2. Do the proper binary shifting

$$111011.0011_2 = 1.110110011_2 \times 2^5$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

- 4. Determine the mantissa

$$\text{Mantissa} = 110110011_2$$

- 5. Determine the exponent

$$exp = 5 + bias = 5 + 127 = 132_{10} = 10000100_2$$

S	E	M
0	10000100	11011001100000000000000

Floating-point format

Example: Determine the floating point format to stored the number 132.28125_{10} in a computer with 32 bits of precision.

Floating-point format

Example: Determine the floating point format to stored the number 132.28125_{10} in a computer with 32 bits of precision.

- 1. Find the binary representation of the number 132.28125_{10}

$$132.28125_{10}$$

Integer part

132_{10}

Dividend	Divisor	Remainder
132	2	LBS → 0
66	2	0
33	2	1
16	2	0
8	2	0
4	2	0
2	2	0
1	2	MSB → 1
0		

Decimal part

0.28125_{10}

$0.28125 \times 2 = 0.5625$ 0 ← MSB

$0.375 \times 2 = 1.125$ 1

$0.125 \times 2 = 0.25$ 0

$0.25 \times 2 = 0.5$ 0

$0.5 \times 2 = 1.0$ 1 ← LBS

$$\boxed{10000100.01001_2}$$

Floating-point format

- 2. Do the proper binary shifting

$$10000100.01001_2 = 1.000010001001_2 \times 2^7$$

Floating-point format

- 2. Do the proper binary shifting

$$10000100.01001_2 = 1.000010001001_2 \times 2^7$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

Floating-point format

- 2. Do the proper binary shifting

$$10000100.01001_2 = 1.000010001001_2 \times 2^7$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

- 4. Determine the mantissa

$$\text{Mantissa} = 000010001001_2$$

Floating-point format

- 2. Do the proper binary shifting

$$10000100.01001_2 = 1.000010001001_2 \times 2^7$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

- 4. Determine the mantissa

$$\text{Mantissa} = 000010001001_2$$

- 5. Determine the exponent

$$exp = 7 + bias = 7 + 127 = 134_{10} = 10000110_2$$

Floating-point format

- 2. Do the proper binary shifting

$$10000100.01001_2 = 1.000010001001_2 \times 2^7$$

- 3. Calculate the bias

$$bias = 2^{8-1} - 1 = 127$$

- 4. Determine the mantissa

$$\text{Mantissa} = 000010001001_2$$

- 5. Determine the exponent

$$exp = 7 + bias = 7 + 127 = 134_{10} = 10000110_2$$

S	E	M
0	10000110	00001000100100000000000

Floating-point format

The real value associated with a given 32 bit binary is calculated as

$$value = (-1)^S \left(1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i}\right) \times 2^{(E-127)}$$

Where,

- S = The sign
- E = Exponent
- 127 = Bias
- d_j = Bits of the mantissa

Floating-point format

Exercise: Find the real value for the binary data:

S	E	M
0	01010010	01101000000100100000000

Floating-point format

Exercise: Find the real value for the binary data:

S	E	M
0	01010010	01101000000100100000000

$$value = (-1)^S \left(1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i}\right) \times 2^{(E-127)}$$

Floating-point format

Exercise: Find the real value for the binary data:

S	E	M
0	01010010	01101000000100100000000

$$value = (-1)^S \left(1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i}\right) \times 2^{(E-127)}$$

In this example:

- $S = 0$
- $1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i} = 1 + 2^{-2} + 2^{-3} + 2^{-5} + 2^{-12} + 2^{-15} = 1.4065246582$
- $2^{(E-127)} = 2^{((2^1+2^4+2^6)-127)} = 2^{82-127} = 2^{-45}$

Floating-point format

Exercise: Find the real value for the binary data:

S	E	M
0	01010010	01101000000100100000000

$$value = (-1)^S \left(1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i}\right) \times 2^{(E-127)}$$

In this example:

- $S = 0$
- $1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i} = 1 + 2^{-2} + 2^{-3} + 2^{-5} + 2^{-12} + 2^{-15} = 1.4065246582$
- $2^{(E-127)} = 2^{((2^1+2^4+2^6)-127)} = 2^{82-127} = 2^{-45}$

Thus

$$value = 1.4065246582 \times 2^{-45}$$

Floating-point format

Example: Find the real value for the binary data:

S	E	M
1	10000100	01000000000000000000000
31	30 — 23	22 — 0

In this example:

- $S = 1$
- $1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i} = 1 + 2^{-2} = 1.25$
- $2^{(E-127)} = 2^{(132-127)} = 2^5$

Example: Find the real value for the binary data:

S	E	M
1	10000100	01000000000000000000000
31	30 … 23	22 … 0

In this example:

- $S = 1$
- $1 + \sum_{i=1}^{23} d_{(23-i)} 2^{-i} = 1 + 2^{-2} = 1.25$
- $2^{(E-127)} = 2^{(132-127)} = 2^{5}$

Thus

$$value = 1.25 \times 2^{5} = -40.$$

Contents

Absolute and relative error

Definition 2.

Suppose that $\widehat{p}$ is an approximation to p. The ***absolute error*** is $E_p = |p - \widehat{p}|$, and the ***relative error*** is $R_p = |p - \widehat{p}|/|p|$, provided that $p \neq 0$.

Absolute and relative error

Definition 2.

Suppose that $\widehat{p}$ is an approximation to p. The ***absolute error*** is $E_p = |p - \widehat{p}|$, and the ***relative error*** is $R_p = |p - \widehat{p}|/|p|$, provided that $p \neq 0$.

- The **absolute error** is the difference between the true value and the approximate value.
- The **relative error** expresses the error as a percentage of the true value.

Example: Find the absolute and relative error in the following three cases:

Absolute and relative error

Example: Find the absolute and relative error in the following three cases:

Real $\|p\|$	$x = 3.141592$	$y = 1,000,000$	$z = 0.000012$
Approximation $\hat{p}$	$\widehat{x} = 3.14$	$\widehat{y} = 999,996$	$\widehat{z} = 0.000009$
Absolute Error E_p	$E_x = \|x - \widehat{x}\| = 0.001592$	$E_y = \|y - \widehat{y}\| = 4$	$E_z = \|z - \widehat{z}\| = 0.000003$
Relative Error R_p	$R_x = E_x/\|x\| = 5.067_{\times 10^{-4}}$	$R_y = E_y/\|y\| = 0.000004$	$R_z = E_z/\|z\| = 0.25$

Absolute and relative error

Example: Find the absolute and relative error in the following three cases:

Real $\lvert p \rvert$	$x = 3.141592$	$y = 1,000,000$	$z = 0.000012$
Approximation $\hat{p}$	$\widehat{x} = 3.14$	$\widehat{y} = 999,996$	$\widehat{z} = 0.000009$
Absolute Error E_p	$E_x = \lvert x - \widehat{x} \rvert$ $= 0.001592$	$E_y = \lvert y - \widehat{y} \rvert$ $= 4$	$E_z = \lvert z - \widehat{z} \rvert$ $= 0.000003$
Relative Error R_p	$R_x = E_x / \lvert x \rvert$ $= 5.067_{\times 10^{-4}}$	$R_y = E_y / \lvert y \rvert$ $= 0.000004$	$R_z = E_z / \lvert z \rvert$ $= 0.25$

Observe that as $|p|$ moves away from 1 (greater than or less than) the relative error R_p is a better indicator than E_p of the accuracy of the approximation.

Absolute and relative error

Definition 3.

The number $\widehat{p}$ is said to ***approximate*** p to d significant digits if d is the **largest** nonnegative integer for which

$$\frac{|p-\widehat{p}|}{|p|} < \frac{10^{1-d}}{2}.$$

Absolute and relative error

Example:

Let $\hat{w}$ be the approximation for $w = 2.1645$, then

$$\frac{|2.1645 - 2.16|}{|2.1645|} = 2.07900 \times 10^{-}3$$

Example:
Let $\hat{w}$ be the approximation for $w = 2.1645$, then

$$\frac{|2.1645 - 2.16|}{|2.1645|} = 2.07900 \times 10^{-}3$$

if $d = 0$: $2.07900 \times 10^{-}3 < \frac{10^{1-0}}{2} = 5$ ✓satisfies. However, as we need to find the largest integer d, we need to continue..

Absolute and relative error

Example:

Let $\hat{w}$ be the approximation for $w = 2.1645$, then

$$\frac{|2.1645 - 2.16|}{|2.1645|} = 2.07900 \times 10^{-}3$$

if $d = 0$: $2.07900 \times 10^{-}3 < \frac{10^{1-0}}{2} = 5$ ✓satisfies. However, as we need to find the largest integer d, we need to continue..

if $d = 1$: $2.07900 \times 10^{-}3 < \frac{10^{1-1}}{2} = 0.5$ ✓satisfies

Absolute and relative error

Example:
Let $\hat{w}$ be the approximation for $w = 2.1645$, then

$$\frac{|2.1645 - 2.16|}{|2.1645|} = 2.07900 \times 10^-3$$

if $d = 0$: $2.07900 \times 10^-3 < \frac{10^{1-0}}{2} = 5$ ✓satisfies. However, as we need to find the largest integer d, we need to continue..

if $d = 1$: $\quad 2.07900 \times 10^-3 < \frac{10^{1-1}}{2} = 0.5$ ✓satisfies

if $d = 2$: $\quad 2.07900 \times 10^-3 < \frac{10^{1-2}}{2} = 0.05$ ✓satisfies

Absolute and relative error

Example:

Let $\hat{w}$ be the approximation for $w = 2.1645$, then

$$\frac{|2.1645 - 2.16|}{|2.1645|} = 2.07900 \times 10^{-}3$$

if $d = 0$: $2.07900 \times 10^{-}3 < \frac{10^{1-0}}{2} = 5$ ✓satisfies. However, as we need to find the largest integer d, we need to continue..

if $d = 1$: $\quad 2.07900 \times 10^{-}3 < \frac{10^{1-1}}{2} = 0.5$ ✓satisfies

if $d = 2$: $\quad 2.07900 \times 10^{-}3 < \frac{10^{1-2}}{2} = 0.05$ ✓satisfies

if $d = 3$: $\quad 2.07900 \times 10^{-}3 < \frac{10^{1-3}}{2} = 0.005$ ✓satisfies

Absolute and relative error

Example:
Let $\hat{w}$ be the approximation for $w = 2.1645$, then

$$\frac{|2.1645 - 2.16|}{|2.1645|} = 2.07900 \times 10^{-}3$$

if $d = 0$: $2.07900 \times 10^{-}3 < \frac{10^{1-0}}{2} = 5$ ✓satisfies. However, as we need to find the largest integer d, we need to continue..

if $d = 1$: $2.07900 \times 10^{-}3 < \frac{10^{1-1}}{2} = 0.5$ ✓satisfies

if $d = 2$: $2.07900 \times 10^{-}3 < \frac{10^{1-2}}{2} = 0.05$ ✓satisfies

if $d = 3$: $2.07900 \times 10^{-}3 < \frac{10^{1-3}}{2} = 0.005$ ✓satisfies

if $d = 4$: $2.07900 \times 10^{-}3 < \frac{10^{1-4}}{2} = 0.0005$ X **does not satisfy**

Then, $\hat{w}$ approximate w to 3 significant digits.

Other examples:

- If $x = 3.141592$ and $\widehat{x} = 3.14$, then $|x - \widehat{x}|/|x| = 0.000507 < 10^{-2}/2$. Therefore, $\widehat{x}$ approximates x to three significant digits.
- If $y = 1,000,000$ and $\widehat{y} = 999,996$, then $|y - \widehat{y}|/|y| = 0.000004 < 10^{-5}/2$. Therefore, $\widehat{y}$ approximates y to six significant digits.
- If $z = 0.000012$ and $\widehat{z} = 0.000009$, then $|z - \widehat{z}|/|z| = 0.25 < 10^{-0}/2$. Therefore, $\widehat{z}$ approximates z to one significant digits.

Contents

Truncation Error

Truncation error refers to errors introduced when a more complicated mathematical expression is "replaced" with a more elementary formula.

Truncation Error

Truncation error refers to errors introduced when a more complicated mathematical expression is "replaced" with a more elementary formula.

For example, the infinite Taylor series

$$e^{x^2} = 1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!} + \cdots + \frac{x^{2n}}{n!} + \cdots$$

might be replaced with just the first five terms $1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}$. Then a truncation error appears.

Example: Given $p = \int_0^{1/2} e^{x^2} dx = 0.544987104184$. Determine the accuracy of the approximation obtained by replacing the integrand $f(x) = e^{x^2}$ with the truncated Taylor series $P_8(x) = 1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}$.

- Determine $\int_0^{1/2} P_8(x)dx$:

Example: Given $p = \int_0^{1/2} e^{x^2} dx = 0.544987104184$. Determine the accuracy of the approximation obtained by replacing the integrand $f(x) = e^{x^2}$ with the truncated Taylor series $P_8(x) = 1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}$.

- Determine $\int_0^{1/2} P_8(x)dx$:

$$\int_0^{1/2} \left(1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!}\right) dx = \left(x + \frac{x^3}{3} + \frac{x^5}{5(2!)} + \frac{x^7}{7(3!)} + \frac{x^9}{9(4!)}\right)\Bigg|_{x=0}^{x=1/2}$$
$$= \frac{1}{2} + \frac{1}{24} + \frac{1}{320} + \frac{1}{5376} + \frac{1}{110592}$$
$$= \frac{2109491}{3870720} = 0.544986720817 = \widehat{p}$$

Since

$$\frac{|p - \widehat{p}|}{|p|} = 7.03442 \times 10^{-7} < \frac{10^{1-6}}{2} = 5 \times 10^{6}$$

then, the approximation $\widehat{p}$ agrees with the true value to 6 significant digits.

Contents

Round-off Error

- The accuracy of the representation of a real number stored in a computer is determined by the precision of the mantissa.

Round-off Error

- The accuracy of the representation of a real number stored in a computer is determined by the precision of the mantissa.
- The error occurred due to the mantissa precision is the ***round-off error.***

Round-off Error

- The accuracy of the representation of a real number stored in a computer is determined by the precision of the mantissa.
- The error occurred due to the mantissa precision is the ***round-off error.***
- The actual number that is stored in the computer may be **chopping** or **rounding** of the last digit.

Round-off Error

- The accuracy of the representation of a real number stored in a computer is determined by the precision of the mantissa.
- The error occurred due to the mantissa precision is the ***round-off error.***
- The actual number that is stored in the computer may be **chopping** or **rounding** of the last digit.
- The computer hardware works with a limited number of digits in machine numbers, errors are introduced and **propagated** in successive computations.

Chopping Off versus Rounding Off

Example:
Consider p expressed in *normalized decimal form*:

$$p = \pm 0.d_1 d_2 d_3 \cdots d_k d_{k+1} \cdots \times 10^n,$$

where $1 \leq d_1 \leq 9$ and $0 \leq d_j \leq 9$ for $j > 1$.

Example:
Consider p expressed in *normalized decimal form*:

$$p = \pm 0.d_1 d_2 d_3 \cdots d_k d_{k+1} \cdots \times 10^n,$$

where $1 \leq d_1 \leq 9$ and $0 \leq d_j \leq 9$ for $j > 1$.

If k is the maximum number of decimal digits; then the real number p is represented by $fl_{chop}(p)$, which is given by

$$fl_{chop}(p) = \pm 0.d_1 d_2 d_3 \cdots d_k \times 10^n, \tag{9}$$

Where $1 \leq d_1 \leq 9$ and $0 \leq d_j \leq 9$ for $1 < j \leq k$. The number $fl_{chop}(p)$ is called the ***chopped floating-point representation*** of p.

Chopping Off versus Rounding Off

On the other hand, the ***rounded floating-point representation*** $fl_{round}(p)$ is given by

$$fl_{round}(p) = \pm 0.d_1 d_2 d_3 \cdots r_k \times 10^n, \tag{10}$$

where $1 \le d_1 \le 9$ and $0 \le d_j \le 9$ for $1 < j < k$ and the last digit, r_k, is obtained by rounding the number $d_k d_{k+1} d_{k+2} \cdots$ to the nearest integer.

Chopping Off versus Rounding Off

Example:

The real number $p = \dfrac{22}{7} = 3.142857142857142857...$ has the following six-digit representations:

$$fl_{chop}(p) = 0.314285 \times 10^1,$$
$$fl_{round}(p) = 0.314286 \times 10^1.$$

For common purposes the chopping and rounding would be written as 3.14285 and 3.14286, respectively.

Contents

Loss of Significance

- Consider $p = 3.14155926536$ ans $q = 3.1415957341$, which are nearly equal and both carry 11 decimal digits of precision.
- Their difference is formed: $p - q = -0,0000030805$. Since the first six digits of p and q are the same, their difference $p - q$ contains only five decimal digits of precision.
- This phenomenon is called ***loss of significance***.

Loss of Significance

Example:

Compare the results of calculating $f(500)$ and $g(500)$ using six digits and rounding. Where, $f(x) = x(\sqrt{x+1} - \sqrt{x})$ and $g(x) = \dfrac{x}{\sqrt{x+1} + \sqrt{x}}$.

Example:

Compare the results of calculating $f(500)$ and $g(500)$ using six digits and rounding. Where, $f(x) = x(\sqrt{x+1} - \sqrt{x})$ and $g(x) = \dfrac{x}{\sqrt{x+1} + \sqrt{x}}$.

For the first function,

$$\begin{aligned} f(500) =& 500\left(\sqrt{501} - \sqrt{500}\right) \\ & 500(22.3830 - 22.3607) = 500(0.0223) = 11.1500 \end{aligned}$$

Example:

Compare the results of calculating $f(500)$ and $g(500)$ using six digits and rounding. Where, $f(x) = x(\sqrt{x+1} - \sqrt{x})$ and $g(x) = \dfrac{x}{\sqrt{x+1} + \sqrt{x}}$.

For the first function,

$$
\begin{aligned}
f(500) =& 500\left(\sqrt{501} - \sqrt{500}\right) \\
& 500(22.3830 - 22.3607) = 500(0.0223) = 11.1500
\end{aligned}
$$

For $g(x)$

$$
\begin{aligned}
g(500) =& \frac{500}{\sqrt{501} + \sqrt{500}} \\
& \frac{500}{22.3830 + 22.3607} = \frac{500}{44.7437} = 11.1748.
\end{aligned}
$$

Example:

Compare the results of calculating $f(500)$ and $g(500)$ using six digits and rounding. Where, $f(x) = x(\sqrt{x+1} - \sqrt{x})$ and $g(x) = \dfrac{x}{\sqrt{x+1} + \sqrt{x}}$.

For the first function,

$$\begin{aligned} f(500) =& 500\left(\sqrt{501} - \sqrt{500}\right) \\ & 500(22.3830 - 22.3607) = 500(0.0223) = 11.1500 \end{aligned}$$

For $g(x)$

$$\begin{aligned} g(500) =& \frac{500}{\sqrt{501} + \sqrt{500}} \\ & \frac{500}{22.3830 + 22.3607} = \frac{500}{44.7437} = 11.1748. \end{aligned}$$

The second function, $g(x)$, is algebraically equivalent to $f(x)$, but the answer, $g(500) = 11.1748$, involves less error and it is the same as that obtained by rounding the true $11.174755300747198...$ to six digits.

Loss of Significance

Example: Compare the results of calculating $f(0.01)$ and $P(0.01)$ using six digits and rounding, where

$$f(x) = \frac{e^x - 1 - x}{x^2} \qquad \text{and} \qquad P(x) = \frac{1}{2} + \frac{x}{6} + \frac{x^2}{24}$$

The function $P(x)$ is the Taylor polynomial of degree $n = 2$ for $f(x)$ expanded about $x = 0$.

Loss of Significance

Example: Compare the results of calculating $f(0.01)$ and $P(0.01)$ using six digits and rounding, where

$$f(x) = \frac{e^x - 1 - x}{x^2} \qquad \text{and} \qquad P(x) = \frac{1}{2} + \frac{x}{6} + \frac{x^2}{24}$$

The function $P(x)$ is the Taylor polynomial of degree $n = 2$ for $f(x)$ expanded about $x = 0$.

For the first function

$$f(0.01) = \frac{e^{0.01} - 1 - 0.01}{(0.01)^2} = \frac{1.010050 - 1 - 0.01}{0.001} = 0.5.$$

Loss of Significance

Example: Compare the results of calculating $f(0.01)$ and $P(0.01)$ using six digits and rounding, where

$$f(x) = \frac{e^x - 1 - x}{x^2} \quad \text{and} \quad P(x) = \frac{1}{2} + \frac{x}{6} + \frac{x^2}{24}$$

The function $P(x)$ is the Taylor polynomial of degree $n = 2$ for $f(x)$ expanded about $x = 0$.

For the first function

$$f(0.01) = \frac{e^{0.01} - 1 - 0.01}{(0.01)^2} = \frac{1.010050 - 1 - 0.01}{0.001} = 0.5.$$

For the second function

$$P(0.01) = \frac{1}{2} + \frac{0.01}{6} + \frac{0.001}{24} = 0.5 + 0.001667 + 0.000004 = 0.501671.$$

Loss of Significance

Example: Compare the results of calculating $f(0.01)$ and $P(0.01)$ using six digits and rounding, where

$$f(x) = \frac{e^x - 1 - x}{x^2} \quad \text{and} \quad P(x) = \frac{1}{2} + \frac{x}{6} + \frac{x^2}{24}$$

The function $P(x)$ is the Taylor polynomial of degree $n = 2$ for $f(x)$ expanded about $x = 0$.

For the first function

$$f(0.01) = \frac{e^{0.01} - 1 - 0.01}{(0.01)^2} = \frac{1.010050 - 1 - 0.01}{0.001} = 0.5.$$

For the second function

$$P(0.01) = \frac{1}{2} + \frac{0.01}{6} + \frac{0.001}{24} = 0.5 + 0.001667 + 0.000004 = 0.501671.$$

The answer $P(0.01) = 0.501671$ contains less error and it is the same as that obtained rounding the true answer $0.5016708416805...$ to six digits.

Contents

For functions

Definition 4.

The function $f(h)$ is said to be ***big Oh*** of $g(h)$, denoted $f(h) = \mathbf{O}(g(h))$, if there exist constants C and c such that:

$$|f(h)| \leq C|g(h)| \quad \text{whenever} \quad h \geq c. \tag{11}$$

$O(h^n)$ Order of Approximation

For functions

Definition 4.

The function $f(h)$ is said to be ***big Oh*** of $g(h)$, denoted $f(h) = \mathbf{O}(g(h))$, if there exist constants C and c such that:

$$|f(h)| \leq C|g(h)| \quad \text{whenever} \quad h \geq c. \tag{11}$$

Example: Consider $f(x) = x^2 + 1$ and $g(x) = x^3$.

$O(h^n)$ Order of Approximation

For functions

Definition 4.

The function $f(h)$ is said to be ***big Oh*** of $g(h)$, denoted $f(h) = \mathbf{O}(g(h))$, if there exist constants C and c such that:

$$|f(h)| \leq C|g(h)| \quad \text{whenever} \quad h \geq c. \tag{11}$$

Example: Consider $f(x) = x^2 + 1$ and $g(x) = x^3$.

- Since $x^2 \leq x^3$ and $1 \leq x^3$ for $x \geq 1$

For functions

Definition 4.

The function $f(h)$ is said to be ***big Oh*** of $g(h)$, denoted $f(h) = \mathbf{O}(g(h))$, if there exist constants C and c such that:

$$|f(h)| \leq C|g(h)| \quad \text{whenever} \quad h \geq c. \tag{11}$$

Example: Consider $f(x) = x^2 + 1$ and $g(x) = x^3$.

- Since $x^2 \leq x^3$ and $1 \leq x^3$ for $x \geq 1$
- it follows that $x^2 + 1 \leq 2x^3$ for $x \geq 1$.

$O(h^n)$ Order of Approximation

For functions

Definition 4.

The function $f(h)$ is said to be ***big Oh*** of $g(h)$, denoted $f(h) = \mathbf{O}(g(h))$, if there exist constants C and c such that:

$$|f(h)| \leq C|g(h)| \quad \text{whenever} \quad h \geq c. \tag{11}$$

Example: Consider $f(x) = x^2 + 1$ and $g(x) = x^3$.

- Since $x^2 \leq x^3$ and $1 \leq x^3$ for $x \geq 1$
- it follows that $x^2 + 1 \leq 2x^3$ for $x \geq 1$.
- Therefore, $f(x) = \mathbf{O}(g(x))$, whenever $h \geq 1$.

The big Oh notation provides an useful way of describing the rate of growth of a function in terms of the well-known elementary function (x^n, $x^{1/n}$, a^x, $log_a(x), etc.$).

For sequences

Definition 5.

Let $x_n = 1^\infty$ and $y_n = 1^\infty$ be two sequences. The sequence x_n is said to be of order big Oh of y_n, denoted $x_n = \mathbf{O}(y_n)$, if there exist constants C and N such that

$$|x_n| \leq C|y_n| \quad \text{whenever} \quad n \geq N. \tag{12}$$

For sequences

Definition 5.

Let $x_n = 1^\infty$ and $y_n = 1^\infty$ be two sequences. The sequence x_n is said to be of order big Oh of y_n, denoted $x_n = \mathbf{O}(y_n)$, if there exist constants C and N such that

$$|x_n| \leq C|y_n| \quad \text{whenever} \quad n \geq N. \tag{12}$$

Example:

$$\frac{n^2-1}{n^3} = \mathbf{O}\left(\frac{1}{n}\right), \text{ since } \frac{n^2-1}{n^3} \leq \frac{n^2}{n^3} = \frac{1}{n} \text{ whenever } n \geq 1.$$

$O(h^n)$ Order of Approximation

Definition 6.

Assume that $f(h)$ is approximated by the function $p(h)$ and there exist a real constant $M > 0$ and a positive integer n so that

$$\frac{|f(h) - p(h)|}{h^n} \leq M \quad \text{for sufficiently small } h. \tag{13}$$

We say that $p(h)$ ***approximates*** $f(h)$ with order of approximation $\mathbf{O}(h^n)$ and write

$$f(h) = p(h) + \mathbf{O}(h^n) \tag{14}$$

When relation (13) is rewritten in the form $|f(h) - p(h)| \leq M|h^n|$, we see that the notation $\mathbf{O}(h^n)$ stands in place of the error bound $M|h^n|$.

Theorem 2. Order of approximation for basic operations

Assume that $f(h) = p(h) + \mathbf{O}(h^n)$, $g(h) = q(h) + \mathbf{O}(h^m)$, and $r = \min(m, n)$. Then

$$f(h) + g(h) = p(h) + q(h) + \mathbf{O}(h^r), \tag{15}$$

$$f(h)g(h) = p(h)q(h) + \mathbf{O}(h^r), \tag{16}$$

and

$$\frac{f(h)}{g(h)} = \frac{p(h)}{q(h)} + \mathbf{O}(h^r) \quad \text{provided that } g(h) \neq 0 \text{ and } q(h) \neq 0. \tag{17}$$

$O(h^n)$ Order of Approximation

Theorem 3. (Taylor's Theorem).

Assume $f \in C^{n+1}[a,b]$. If both x_0 and $x = x_0 + h$ lie in $[a,b]$, then

$$f(x_0+h) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!} h^k + \mathbf{O}(h^{n+1}). \tag{18}$$

Additional properties:

(i) $\mathbf{O}(h^p) + \mathbf{O}(h^p) = \mathbf{O}(h^p)$,

(ii) $\mathbf{O}(h^p) + \mathbf{O}(h^q) = \mathbf{O}(h^r)$, where $r = \min(p,q)$, and

(iii) $\mathbf{O}(h^p)\mathbf{O}(h^q) = \mathbf{O}(h^s)$, where $s = p + q$.

$O(h^n)$ Order of Approximation

Example:
Consider the Taylor polynomial expansions

$$e^h = 1+h+\frac{h^2}{2!}+\frac{h^3}{3!}+\mathbf{O}(h^4) \qquad \text{and} \qquad \cos(h) = 1 - \frac{h^2}{2!} + \frac{h^4}{4!} + \mathbf{O}(h^6).$$

Determine the order of approximation for their sum and product.

Example:
Consider the Taylor polynomial expansions

$$e^h = 1+h+\frac{h^2}{2!}+\frac{h^3}{3!}+\mathbf{O}(h^4) \qquad \text{and} \qquad \cos(h) = 1-\frac{h^2}{2!}+\frac{h^4}{4!}+\mathbf{O}(h^6).$$

Determine the order of approximation for their sum and product.

- For the sum we have

$$\begin{aligned} e^h+\cos(h) =& 1+h+\frac{h^2}{2!}+\frac{h^3}{3!}+\mathbf{O}(h^4)+1-\frac{h^2}{2!}+\frac{h^4}{4!}+\mathbf{O}(h^6) \\ =& \, 2+h+\frac{h^3}{3!}+\mathbf{O}(h^4)+\frac{h^4}{4!}+\mathbf{O}(h^6) \end{aligned}$$

$O(h^n)$ Order of Approximation

Since $\mathbf{O}(h^4) + \frac{h^4}{4!} = \mathbf{O}(h^4)$ and $\mathbf{O}(h^4) + \mathbf{O}(h^6) = \mathbf{O}(h^4)$, this reduces to

$$e^h + \cos(h) = 2 + h + \frac{h^3}{3!} + \mathbf{O}(h^4),$$

and the order of approximation is $\mathbf{O}(h^4)$.

- The product is treated similarly:

$$
\begin{aligned}
e^h\cos(h) =& \left(1+h+\frac{h^2}{2!}+\frac{h^3}{3!}+\mathbf{O}(h^4)\right)\left(1-\frac{h^2}{2!}+\frac{h^4}{4!}+\mathbf{O}(h^6)\right)\\
=& \left(1+h+\frac{h^2}{2!}+\frac{h^3}{3!}\right)\left(1-\frac{h^2}{2!}+\frac{h^4}{4!}\right)+\\
& \left(1+h+\frac{h^2}{2!}+\frac{h^3}{3!}\right)\mathbf{O}(h^6)+\left(1-\frac{h^2}{2!}+\frac{h^4}{4!}\right)\mathbf{O}(h^4)+\mathbf{O}(h^4)\mathbf{O}(h^6)\\
=& 1+h-\frac{h^3}{3}-\frac{5h^4}{24}-\frac{h^5}{24}+\frac{h^6}{48}+\frac{h^7}{144}+\mathbf{O}(h^6)+\mathbf{O}(h^4)+\mathbf{O}(h^4)\mathbf{O}(h^6).
\end{aligned}
$$

$O(h^n)$ Order of Approximation

Since $\mathbf{O}(h^4)\mathbf{O}(h^6) = \mathbf{O}(h^{10})$ and

$$\frac{-5h^4}{24} - \frac{h^5}{24} + \frac{h^6}{48} + \frac{h^7}{144} + \mathbf{O}(h^6) + \mathbf{O}(h^4) + \mathbf{O}(h^{10})$$

Since $\mathbf{O}(h^6) + \mathbf{O}(h^4) + \mathbf{O}(h^{10}) = \mathbf{O}(h^4)$, the preceding equation is simplified to yield

$$e^h \cos(h) = 1 + h + \frac{h^3}{3} + \mathbf{O}(h^4),$$

and the order of approximation is $\mathbf{O}(h^4)$.

Convergence of a sequence

Definition 7.

Suppose that $\lim_{n\longrightarrow\infty} x_n = x$ and $\{r_n\}_{n=1}^{\infty}$ is a sequence with $\lim_{n\longrightarrow\infty} r_n = 0$. We say that $\{x_n\}_{n=1}^{\infty}$ ***converges*** to x with the order of convergence $\mathbf{O}(r_n)$, if there exists a constant $K \geq 0$ such that

$$\frac{|x_n - x|}{|r_n|} \leq K \quad \text{for } n \text{ sufficiently large.} \tag{19}$$

This is indicated by writing $x_n = x + \mathbf{O}(\mathbf{r_n})$, or $x_n \longrightarrow x$ with order of convergence $\mathbf{O}(r_n)$

Definition 7.

Example:

Let $x_n = \cos(n)/n^2$ and $r_n = 1/n^2$ then,

$$\lim_{n \longrightarrow \infty} x_n = 0$$

with a rate of convergence $\mathrm{O}(1/n^2)$. This follows immediately from the relation

$$\frac{|\cos(n)/n^2|}{|1/n^2|} = |\cos(n) \leq 1| \text{ for all } n.$$

Contents

Propagation of Error

- **Addition** consider two numbers p and q (the true values) with the approximate values $\widehat{p}$ and $\widehat{q}$, which contains errors ϵ_p and ϵ_q, respectively. Starting with $p = \widehat{p} + \epsilon_p$ and $q = \widehat{q} + \epsilon_q$, the sum is

$$p + q = (\widehat{p} + \epsilon_p) + (\widehat{q} + \epsilon_q) = (\widehat{p} + \widehat{q}) + (\epsilon_p + \epsilon_q). \tag{20}$$

- Hence, for addition, the error in the sum is the **sum** of the errors in the addends.

$$\epsilon_s = \epsilon_p + \epsilon_q.$$

Propagation of Error

The propagation of error in **multiplication** is more complicated. The product is

$$pq = (\widehat{p} + \epsilon_p)(\widehat{q} + \epsilon_q) = \widehat{p}\widehat{q} + \widehat{p}\epsilon_p + \widehat{q}\epsilon_p + \epsilon_p\epsilon_q. \quad (21)$$

Propagation of Error

The propagation of error in **multiplication** is more complicated. The product is

$$pq = (\widehat{p} + \epsilon_p)(\widehat{q} + \epsilon_q) = \widehat{p}\widehat{q} + \widehat{p}\epsilon_p + \widehat{q}\epsilon_p + \epsilon_p\epsilon_q. \tag{21}$$

Hence, if $\widehat{p}$ and $\widehat{q}$ are larger than 1 in absolute value, the terms $\widehat{p}\epsilon_q$ and $\widehat{q}\epsilon_p$ show that there is a possibility of magnification of the original errors ϵ_p and ϵ_q. Insights are gained if we look at the relative error. Rearrange the terms in (21) to get

$$pq - \widehat{p}\widehat{q} = \widehat{p}\epsilon_q + \widehat{q}\epsilon_p + \epsilon_p\epsilon_q. \tag{22}$$

Propagation of Error

The propagation of error in **multiplication** is more complicated. The product is

$$pq = (\widehat{p} + \epsilon_p)(\widehat{q} + \epsilon_q) = \widehat{p}\widehat{q} + \widehat{p}\epsilon_p + \widehat{q}\epsilon_p + \epsilon_p\epsilon_q. \tag{21}$$

Hence, if $\widehat{p}$ and $\widehat{q}$ are larger than 1 in absolute value, the terms $\widehat{p}\epsilon_q$ and $\widehat{q}\epsilon_p$ show that there is a possibility of magnification of the original errors ϵ_p and ϵ_q. Insights are gained if we look at the relative error. Rearrange the terms in (21) to get

$$pq - \widehat{p}\widehat{q} = \widehat{p}\epsilon_q + \widehat{q}\epsilon_p + \epsilon_p\epsilon_q. \tag{22}$$

Suppose that $\widehat{p} \neq 0$ and $\widehat{q} \neq 0$; then we can divide (22) by pq to obtain the relative error in the product pq:

$$R_{pq} = \frac{pq - \widehat{p}\widehat{q}}{pq} = \frac{\widehat{p}\epsilon_q + \widehat{q}\epsilon_p + \epsilon_p\epsilon_q}{pq} = \frac{\widehat{p}\epsilon_q}{pq} + \frac{\widehat{q}\epsilon_p}{pq} + \frac{\epsilon_p\epsilon_q}{pq}. \tag{23}$$

Furthermore, suppose that $\widehat{p}$ and $\widehat{q}$ are good approximations for $\widehat{p}$ and $\widehat{q}$; then $\widehat{p}/p \approx 1, \widehat{q}/q \approx 1$, and $R_pR_q = (\epsilon_p/p)(\epsilon_q/q) \approx 0$ (R_p and R_q are the relative errors in the approximations $\widehat{p}$ and $\widehat{q}$). Then making these substitutions yields the simplified relationship

$$R_{pq} = \frac{pq - \widehat{p}\widehat{q}}{pq} \approx \epsilon_q/q + \epsilon_p/p + 0 = R_q + R_p. \tag{24}$$

Propagation of Error

Furthermore, suppose that $\widehat{p}$ and $\widehat{q}$ are good approximations for $\widehat{p}$ and $\widehat{q}$; then $\widehat{p}/p \approx 1, \widehat{q}/q \approx 1$, and $R_pR_q = (\epsilon_p/p)(\epsilon_q/q) \approx 0$ (R_p and R_q are the relative errors in the approximations $\widehat{p}$ and $\widehat{q}$). Then making these substitutions yields the simplified relationship

$$R_{pq} = \frac{pq - \widehat{\widehat{p}\widehat{q}}}{pq} \approx \epsilon_q/q + \epsilon_p/p + 0 = R_q + R_p. \tag{24}$$

This shows that the relative error in the product $\mathbf{pq}$ is approximately **the sum of the relative errors** in the approximations $\widehat{\mathbf{p}}$ and $\widehat{\mathbf{q}}$.

A quality that is desirable for any numerical process is that a small error in the initial conditions will produce small changes in the final result. An algorithm with this feature is called ***stable;*** otherwise, it is called ***unstable***.

Propagation of Error

Definition 8.

Suppose that ϵ represents an initial error and $\epsilon(n)$ represents the growth of the error after n steps. If $|\epsilon(n)| \approx n\epsilon$, the growth of error is said to be ***linear***. If $|\epsilon(n)| \approx K^n\epsilon$, the growth of error is called ***exponential***. If $K > 1$, the exponential error gronws without bound as $n \longrightarrow \infty$, and if $0 < K < 1$, the exponential error diminishes to zero as $n \longrightarrow \infty$.

Example: Show that the following three schemes can be used with finite-precision arithmetic to recursively generate the terms in the sequence $\{1/3^n\}_{n=0}^{\infty}$.

$$r_0 = 1 \quad \text{and} \quad r_n = \frac{1}{3} r_{n-1} \quad \text{for} \quad n = 1, 2, \cdots, \tag{25}$$

$$p_0 = 1, p_1 = \frac{1}{3}, \quad \text{and} \quad p_n = \frac{4}{3} p_{n-1} - \frac{1}{3} p_{n-2} \quad \text{for} \quad n = 1, 2, \cdots, \tag{26}$$

$$q_0 = 1, q_1 = \frac{1}{3}, \quad \text{and} \quad q_n = \frac{10}{3} q_{n-1} - q_{n-2} \quad \text{for} \quad n = 1, 2, \cdots, \tag{27}$$

Propagation of error

Formula (25) is obvious. In (26) the difference equation has the general solution $p_n = A(1/3^n) + B$. This can be verified by direct substitution:

$$\begin{aligned}\frac{4}{3}p_{n-1} - \frac{1}{3}p_{n-2} =& \frac{4}{3}\left(\frac{A}{3^{n-1}} + B\right) - \frac{1}{3}\left(\frac{A}{3^{n-2}} + B\right)\\ &= \left(\frac{4}{3^n} - \frac{3}{3^n}\right)A - \left(\frac{4}{3} - \frac{1}{3}\right)B = A\frac{1}{3^n} + B = p_n\end{aligned}$$

Setting $A = 1$ and $B = 0$ will generate the sequence desired.

Propagation of error

Formula (25) is obvious. In (26) the difference equation has the general solution $p_n = A(1/3^n) + B$. This can be verified by direct substitution:

$$\begin{aligned} \frac{4}{3}p_{n-1} - \frac{1}{3}p_{n-2} =& \frac{4}{3}\left(\frac{A}{3^{n-1}} + B\right) - \frac{1}{3}\left(\frac{A}{3^{n-2}} + B\right) \\ =& \left(\frac{4}{3^n} - \frac{3}{3^n}\right)A - \left(\frac{4}{3} - \frac{1}{3}\right)B = A\frac{1}{3^n} + B = p_n \end{aligned}$$

Setting $A = 1$ and $B = 0$ will generate the sequence desired. In (27) the difference equation has the general solution $q_n = A(1/3^n) + B3^n$. This too verified by substitution:

$$\begin{aligned} \frac{10}{3}q_{n-1} - q_{n-2} =& \frac{10}{3}\left(\frac{A}{3^{n-1}} + B3^{n-1}\right) - \left(\frac{A}{3^{n-2}} + B3^{n-2}\right) \\ =& \left(\frac{10}{3^n} - \frac{9}{3^n}\right)A - (10-1)3^{n-1}B = A\frac{1}{3^n} + B3^n = q_n \end{aligned}$$

Example:

Generate approximations to the sequences $\{x_n\} = 1/3^n$ using hemes

$$r_0 = 0.99996 \quad \text{and} \quad r_n = \frac{1}{3} r_{n-1} \quad \text{for} \quad n = 1, 2, \cdots, \tag{28}$$

$$p_0 = 1, p_1 = 0.33332, \quad \text{and} \quad p_n = \frac{4}{3} p_{n-1} - \frac{1}{3} p_{n-2} \quad \text{for} \quad n = 1, 2, \cdots, \tag{29}$$

$$q_0 = 1, q_1 = 0.33332, \quad \text{and} \quad q_n = \frac{10}{3} p_{n-1} - p_{n-2} \quad \text{for} \quad n = 1, 2, \cdots, \tag{30}$$

In (28) the initial error in r_0 is 0.00004, and in (29) and (30) the initial errors in p_1 and q_1 are $0.00001\overline{3}$. Investigate the propagation of error for each scheme.

Propagation of error

Table: Sequence $x_n = 1/3^n$ and the approximations r_n, p_n, and q_n

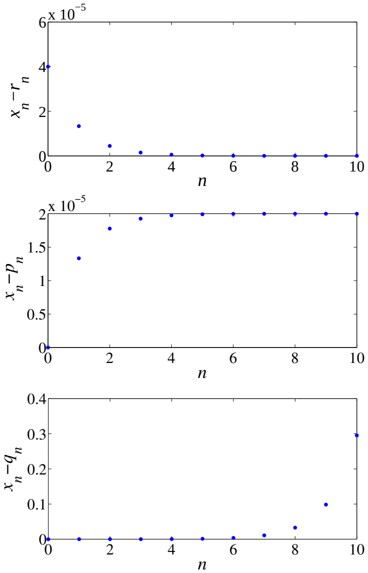
n	x_n	r_n	p_n	q_n
0	1.0000000000	0.9999600000	1.0000000000	1.0000000000
1	0.3333333333	0.3333200000	0.3333200000	0.3333200000
2	0.1111111111	0.1111066667	0.1110933333	0.1110666667
3	0.0370370370	0.0370355556	0.0370177778	0.0369022222
4	0.0123456790	0.0123451852	0.0123259259	0.0119407407
5	0.0041152263	0.0041150617	0.0040953086	0.0029002469
6	0.0013717421	0.0013716872	0.0013517695	-0.0022732510
7	0.0004572474	0.0004572291	0.0004372565	-0.0104777503
8	0.0001524158	0.0001524097	0.0001324188	-0.0326525834
9	0.0000508053	0.0000508032	0.0000308063	-0.0983641945
10	0.0000169351	0.0000169344	-0.0000030646	-0.2952280648

Propagation of error

Table: Error sequences $x_n - r_n$, $x_n - p_n$, and $x_n - q_n$

n	$x_n - r_n$	$x_n - p_n$	$x_n - q_n$
0	0.0000400000	0.0000000000	0.0000000000
1	0.0000133333	0.0000133333	0.0000133333
2	0.0000044444	0.0000177778	0.0000444444
3	0.0000014815	0.0000192593	0.0001348148
4	0.0000004938	0.0000197531	0.0004049383
5	0.0000001646	0.0000199177	0.0012149794
6	0.0000000549	0.0000199726	0.0036449931
7	0.0000000183	0.0000199909	0.0109349977
8	0.0000000061	0.0000199970	0.0328049992
9	0.0000000020	0.0000199990	0.0984149997
10	0.0000000007	0.0000199997	0.2952449999

Propagation of error



- The error for $\{r_n\}$ is stable and decreases in an exponential manner.
- The error $\{p_n\}$ is stable.
- The error for $\{q_n\}$ is unstable and grows at an exponential rate.

Although the error for $\{p_n\}$ is stable, the terms $p_n \longrightarrow 0$ as $n \longrightarrow \infty$, so that the error eventually dominates and the terms past p_8 have no significant digits.