CSE 209: Numerical Methods

Dr. M. Jahirul Islam
Associate Professor, CSE Dept., SUST

Email: jahir-cse@sust.edu

Engineering Problem Solving

- Requires understanding of engineering systems
  - –By observation and experiment
  - –Theoretical analysis and generalization

- Computers are great tools, however, without fundamental understanding of engineering problems, they will be useless.

How do We Solve an Engineering Problem?

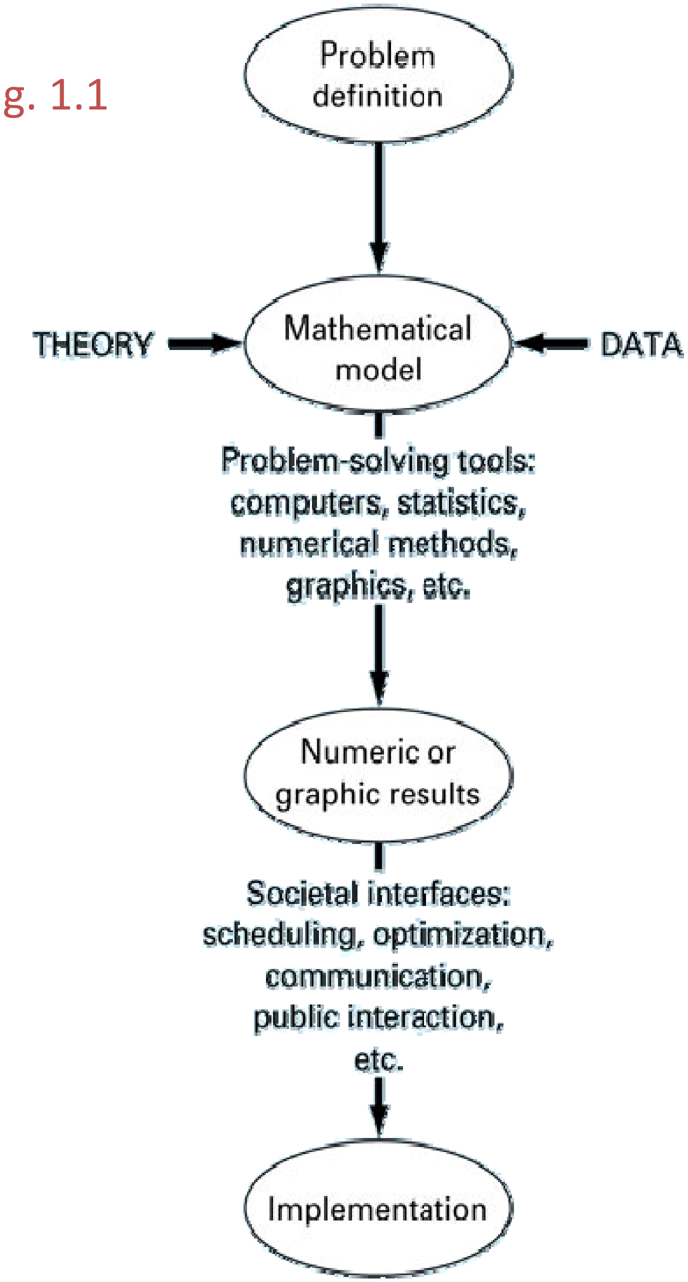
Problem Description

Mathematical Model

Solution of Mathematical Model

Using the Solution

Fig. 1.1


Problem definition
THEORY
Mathematical model
DATA
Problem-solving tools: computers, statistics, numerical methods, graphics, etc.
Numeric or graphic results
Societal interfaces: scheduling, optimization, communication, public interaction, etc.
Implementation

Mathematical Modeling

- A mathematical model is represented as a functional relationship of the form

$$\textbf{Dependent Variable} = f\left(\text{independent variables},\ \text{parameters},\ \text{forcing functions}\right)$$

- *Dependent variable*: Characteristic that usually reflects the state of the system
- *Independent variables*: Dimensions such as time and space along which the systems behavior is being determined
- *Parameters*: reflect the system's properties or composition
- *Forcing functions*: external influences acting upon the system

Newton’s 2nd law of Motion

- States that *“the time rate change of momentum of a body is equal to the resulting force acting on it*.”
- The model is formulated as

$$\mathbf{F = m\ a} \quad (1.2)$$

F=net force acting on the body (N)

m=mass of the object (kg)

a=its acceleration (m/s^2)

$$\mathbf{a = F/\ m} \quad (1.3)$$

Observations

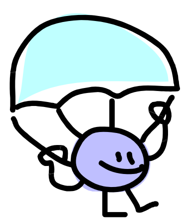
- . Formulation of Newton's 2nd law has several characteristics that are typical of mathematical models of the physical world:
  - It describes a natural process or system in mathematical terms
  - It represents an idealization and simplification of reality
  - Finally, it yields reproducible results, consequently, can be used for predictive purposeS

Observations (2)

- Some mathematical models of physical phenomena may be much more complex.

- Complex models may not be solved exactly or require more sophisticated mathematical techniques than simple algebra for their solution

  – Example, modeling of a falling parachutist:



Determine the Terminal Velocity of a Free Falling Body Near Earth's Surface

$$\frac{dv}{dt} = \frac{F}{m}$$

$$F = F_D + F_U$$

$$F_D = mg$$

$$F_U = -cv$$

$$\frac{dv}{dt} = \frac{mg - cv}{m}$$

- F→+ve: Accelerate
- F→-ve: decelerate
- F→0: Velocity will remain at a constant level
- F_D: Downward pull of gravity
- F_U: Upward force of air resistance
- C= drag coefficients depends of the shape or surface roughness
- g: gravitational constant 9.8 m/s^2

Modeling of a Falling Parachutist

$$\frac{dv}{dt} = g - \frac{c}{m} v$$

- This is a differential equation and is written in terms of the differential rate of change dv/dt of the variable that we are interested in predicting.
- If the parachutist is initially at rest (v=0 at t=0), using calculus

Independent variable

$$v(t) = \frac{gm}{c}\left(1 - e^{-(c/m)t}\right)$$

Dependent variable

Forcing function

Parameters

Conclusions

- Analytical or exact solution because it satisfies the original differential equation
- Unfortunately there are many mathematical models that can not be solved exactly
- Only alternative is to develop the numerical solution that approximates the exact solution

- **New Value= Old Value + Slope* Step Size**

$$v(t_{i+1}) = v(t_i) + \left[g - \frac{c}{m} v(t_i) \right] (t_{i+1} - t_i)$$

$$v_{i+1} = v_i + \frac{dv_i}{dt} \Delta t$$

Euler's Method

Conservation Laws and Engineering

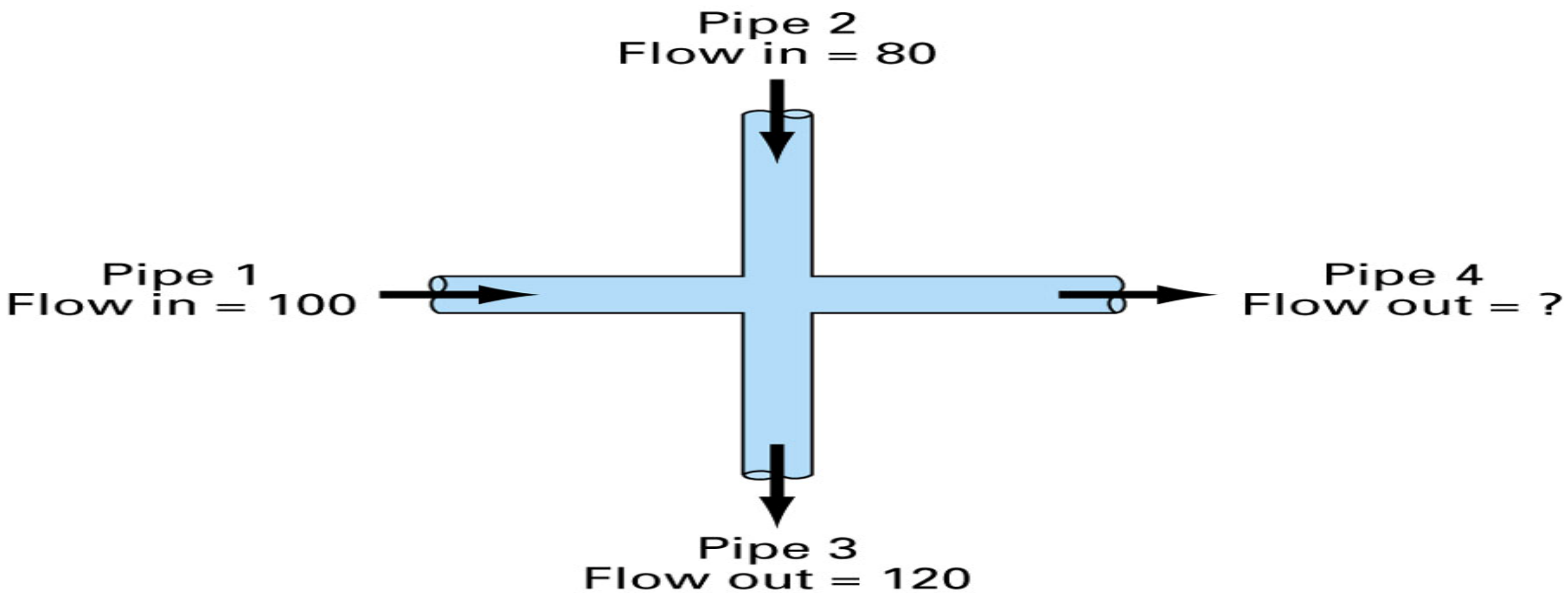
- Conservation laws are the most important and fundamental laws that are used in engineering.

Change = increases – decreases (1.13)

- Change implies changes with time (transient). If the change is nonexistent (steady-state), Eq. 1.13 becomes

Increases =Decreases

Fig 1.6



- For steady-state incompressible fluid flow in pipes:

$$\text{Flow in} = \text{Flow out}$$

or

$$100 + 80 = 120 + \text{Flow}_4$$

$$\text{Flow}_4 = 60$$

Approximations and Round off Errors

Chapter 3

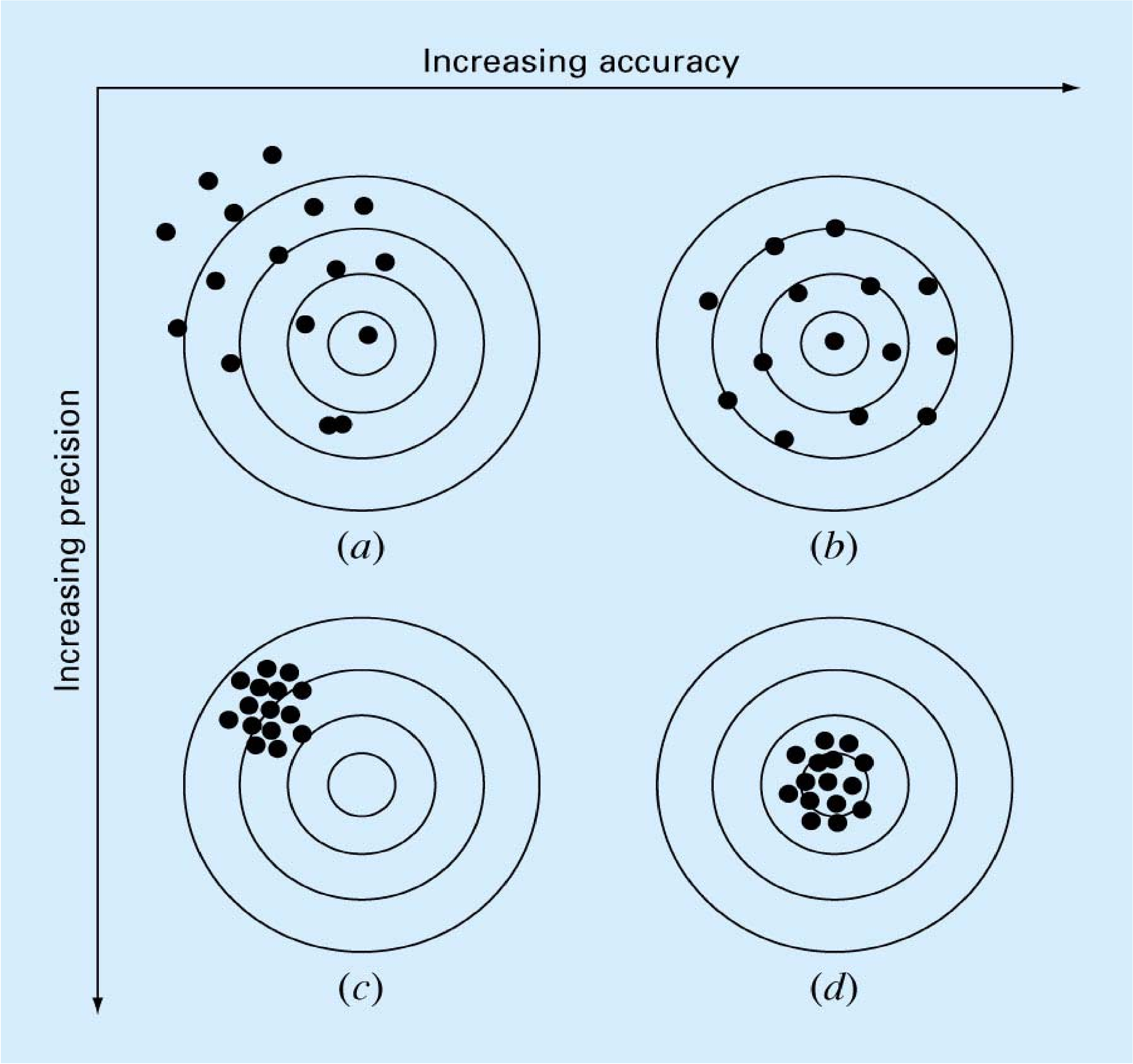
Approximations and Round-Off Errors

- For many engineering problems, we cannot obtain analytical solutions.
- Numerical methods yield approximate results, results that are close to the exact analytical solution. We cannot exactly compute the errors associated with numerical methods.
  - Only rarely given data are exact, since they originate from measurements. Therefore there is probably error in the input information.
  - Algorithm itself usually introduces errors as well, e.g., unavoidable round-offs, etc …
  - The output information will then contain error from both of these sources.
- How confident we are in our approximate result?
- The question is "*how much error is present in our calculation and is it tolerable?*"

Terminology

- Accuracy. How close is a computed or measured value to the true value
- Precision (or *reproducibility*). How close is a computed or measured value to previously computed or measured values.
- Inaccuracy (or *bias*). A systematic deviation from the actual value.
- Imprecision (or *uncertainty*). Magnitude of scatter.

Fig. 3.2



Significant Figures

- Number of significant figures indicates precision. Significant digits of a number are those that can be *used* with *confidence,* e.g., the number of certain digits plus one estimated digit.

53,8<u>00</u> How many significant figures?

5.38×10^4	3
5.380×10^4	4
5.3800×10^4	5

Zeros are sometimes used to locate the decimal point not significant figures.

0.00001753	4
0.0001753	4
0.001753	4

Error Definitions

True Value = Approximation + Error

E_t = True value – Approximation (+/-)

True error

$$\text{True fractional relative error} = \frac{\text{true error}}{\text{true value}}$$

$$\text{True percent relative error, } \varepsilon_t = \frac{\text{true error}}{\text{true value}} \times 100\%$$

- For numerical methods, the true value will be known only when we deal with functions that can be solved analytically (simple systems). In real world applications, we usually not know the answer a priori. Then

$$\varepsilon_{\mathrm{a}} = \frac{\text{Approximate error}}{\text{Approximation}} \times 100\%$$

- *Iterative approach,* example Newton's method

$$\underset{(+\,/\,-)}{\varepsilon_{\mathrm{a}}} = \frac{\text{Current approximation - Previous approximation}}{\text{Current approximation}} \times 100\%$$

- Use absolute value.
- Computations are repeated until stopping criterion is satisfied.

$$\left|\varepsilon_a\right| < \varepsilon_s$$

Pre-specified % tolerance based on the knowledge of your solution

- If the following criterion is met

$$\varepsilon_s = (0.5 \times 10^{(2-n)})\%$$

you can be sure that the result is correct to at least n significant figures.

Round-off Errors

- Numbers such as p, e, or $\sqrt{7}$ cannot be expressed by a fixed number of significant figures.
- Computers use a base-2 representation, they cannot precisely represent certain exact base-10 numbers.
- Fractional quantities are typically represented in computer using “floating point” form, e.g.,

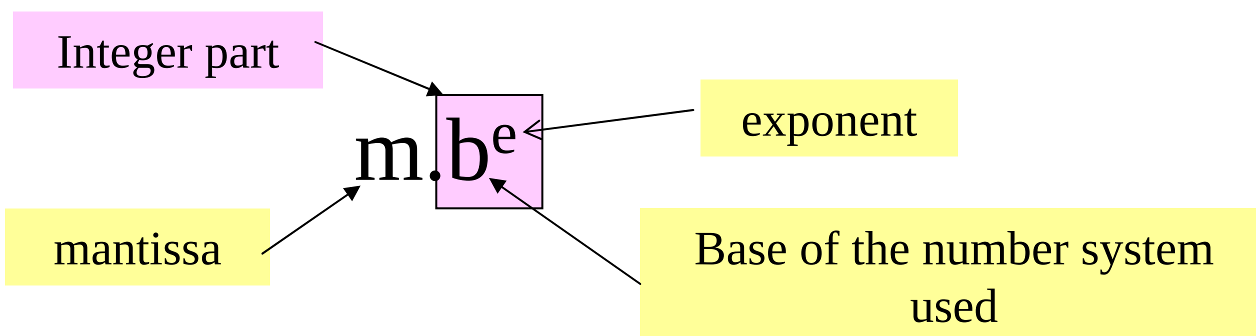
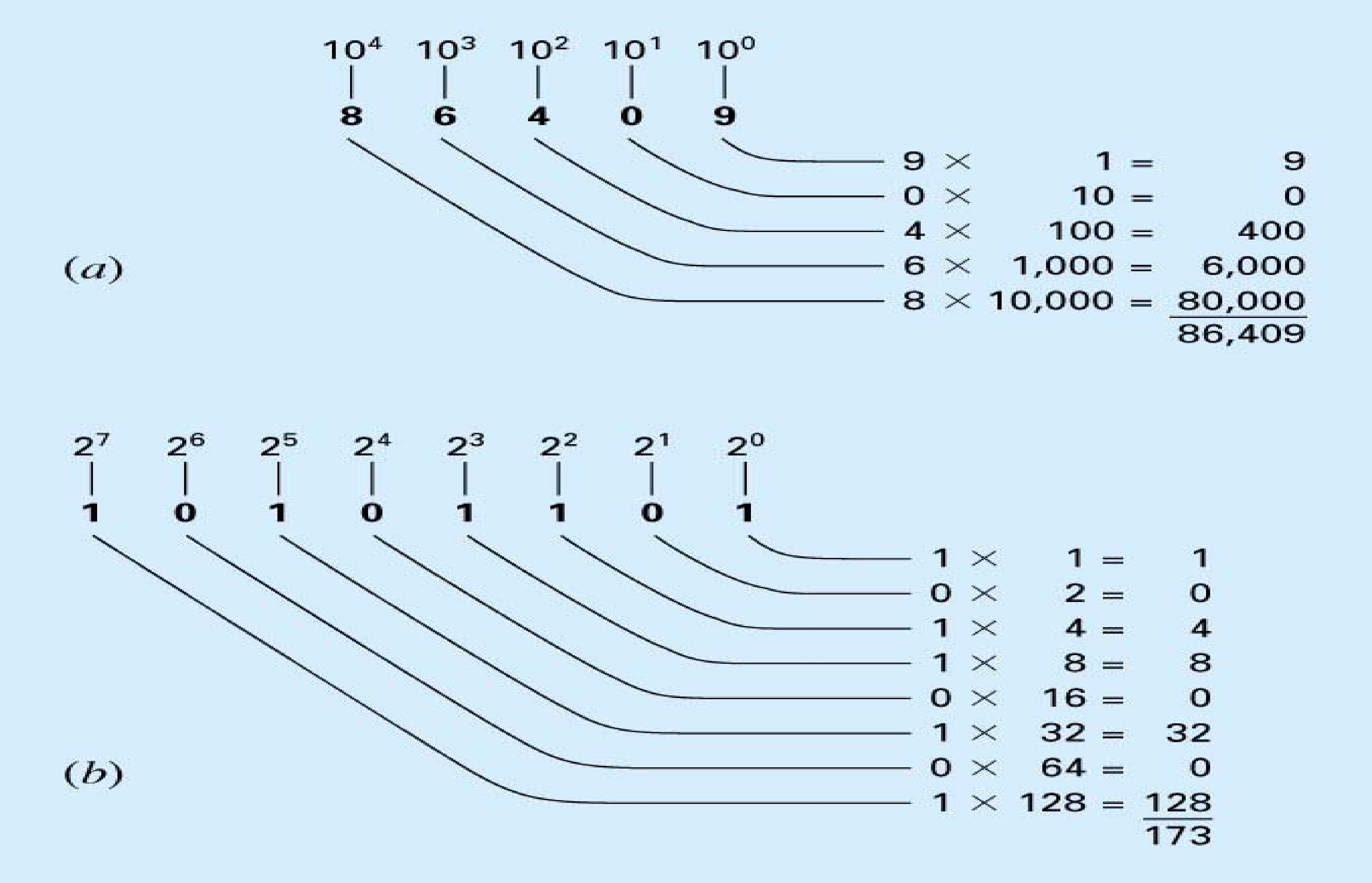


Figure 3.3



10^4 10^3 10^2 10^1 10^0
8 6 4 0 9
9 × 1 = 9
0 × 10 = 0
4 × 100 = 400
6 × 1,000 = 6,000
8 × 10,000 = 80,000
86,409
(a)
2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0
1 0 1 0 1 1 0 1
1 × 1 = 1
0 × 2 = 0
1 × 4 = 4
1 × 8 = 8
0 × 16 = 0
1 × 32 = 32
0 × 64 = 0
1 × 128 = 128
173
(b)

Figure 3.4

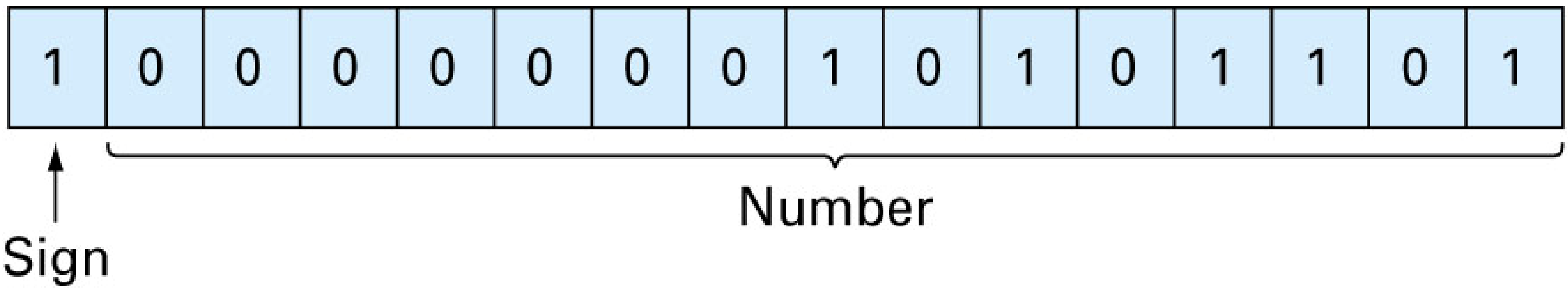
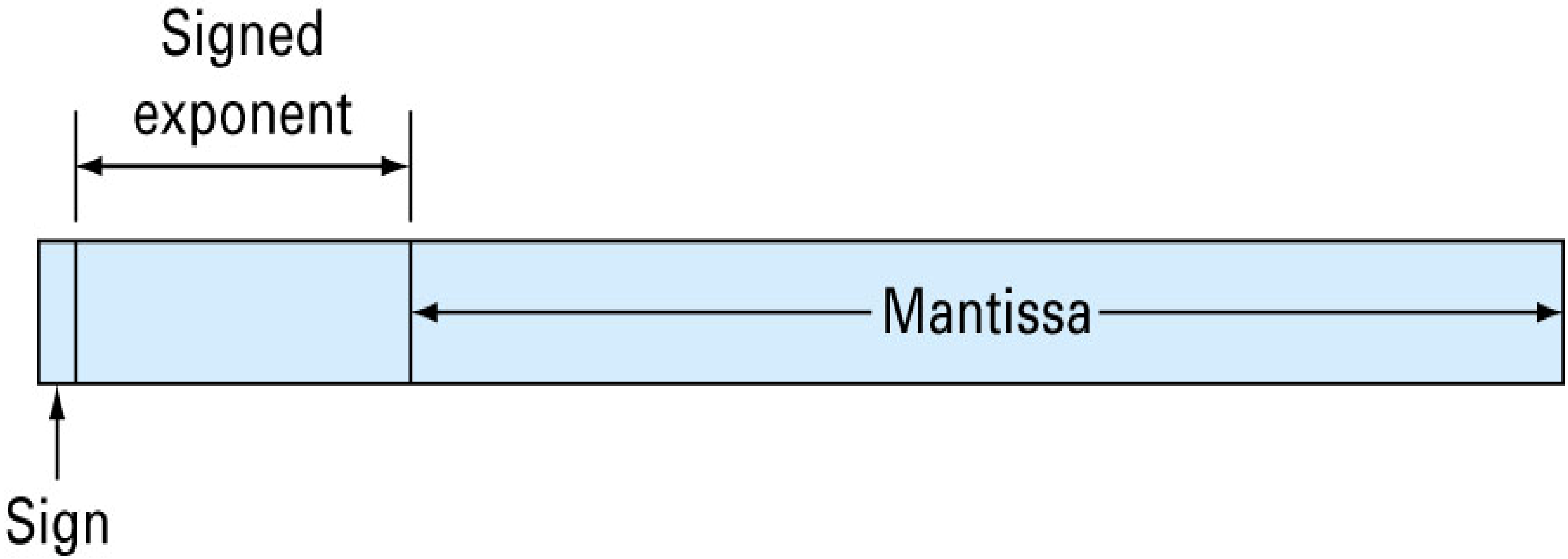


Figure 3.5



156.78 ⏩ 0.15678×10^3 in a floating point base-10 system

$$\frac{1}{34} = 0.029411765$$

Suppose only 4 decimal places to be stored

$$0.0294 \times 10^0 \qquad \frac{1}{2} \le |m| < 1$$

- Normalized to remove the leading zeroes. Multiply the mantissa by 10 and lower the exponent by 1

$0.294\underline{1} \times 10^{-1}$

Additional significant figure is retained

$$\frac{1}{b} \le |m < 1|$$

Therefore

for a base-10 system 0.1 ≤m<1

for a base-2 system 0.5 ≤m<1

- Floating point representation allows both fractions and very large numbers to be expressed on the computer. However,
  - Floating point numbers take up more room.
  - Take longer to process than integer numbers.
  - Round-off errors are introduced because mantissa holds only a finite number of significant figures.

Chopping

Example:

π=3.14159265358 to be stored on a base-10 system carrying 7 significant digits.

π=3.141592 chopping error ε_t=0.00000065

If rounded

π=3.141593 ε_t=0.00000035

- Some machines use chopping, because rounding adds to the computational overhead. Since number of significant figures is large enough, resulting chopping error is negligible.