Numerical Heat Transfer

(数值传热学)

Chapter 7 Mathematical Characteristics of Discretized Equations and Verification & Validation of the Numerical Solutions



Instructor Li Chen

Key Laboratory of Thermo-Fluid Science & Engineering
Int. Joint Research Laboratory of Thermal Science & Engineering
Xi'an Jiaotong University
Innovative Harbor of West China, Xian

2026-Nov.-2

Contents

7.1 Consistency, Convergence and Stability of Discretized Equations

7.2 von Neumann Method for Analysing Stability of Initial Problems

7.3 Conservation of Discretized Equations

7.4 Classification and V&V of Numerical Solution Errors

7.5 Analysis of Truncation Error of Discretization Schemes

7.6 Analysis of Grid Convergence Index and Incomplete Iteration Error

7.1 Consistency, Convergence and Stability of Discretized Equations

7.1.1 Truncation error and consistency (相容性)

7.1.2 Discretization error（离散误差） and convergence(收敛性)

7.1.3 Round off error （舍入误差）and stability （稳定性） of initial problems（初值问题）

7.1.4 Examples for discretization error and stability

7.1 Consistency, Convergence and Stability of Discretized Equations

7.1.1 Truncation error (截断误差) and consistency (相容性)

1) Truncation error (T.E. 截断误差) of the discretized equation

Let us consider the original differential (微分) equation with accurate solution as $\phi(i,n)$

$$\rho\frac{\partial\phi}{\partial t}+\rho u\frac{\partial\phi}{\partial x}=\Gamma\frac{\partial^2\phi}{\partial x^2}$$

which is a 1-D partial differential equation (PDE) with unsteady, convection and diffusion terms.

Original differential (微分） equation:

$$\rho\frac{\partial\phi}{\partial t}+\rho u\frac{\partial\phi}{\partial x}=\Gamma\frac{\partial^2\phi}{\partial x^2}$$

Using **forward time scheme for unsteady term （向前差分）** and **central space scheme （中心差分）for the convective and diffusive terms**, the following **discretized equation of 1-D transient model is obtained:**

$$\rho\frac{\phi_i^{n+1}-\phi_i^n}{\Delta t}+\rho u\frac{\phi_{i+1}^n-\phi_{i-1}^n}{2\Delta x}=\Gamma\frac{\phi_{i+1}^n-2\phi_i^n+\phi_{i-1}^n}{\Delta x^2},O(\Delta t,\Delta x^2)$$

Comment on ϕ_i^n;

It refers to the accurate numerical solution of the discretized equation **（离散方程的精确解）** without any round-off **(舍入)** error or incomplete iteration error **(不完全迭代）** introduced.

Original differential (微分) equation:

$$\rho\frac{\partial\phi}{\partial t}+\rho u\frac{\partial\phi}{\partial x}=\Gamma\frac{\partial^2\phi}{\partial x^2}+S$$

Using **forward time scheme for unsteady term** and **central space scheme for the convective and diffusive terms** to discretize the original equation

Transient term of FD form

$$\rho\frac{\phi_i^{n+1}-\phi_i^n}{\Delta t}=\rho\frac{\phi_i^n+(\frac{\partial\phi}{\partial t})_{i,n}\Delta t+(\frac{\partial^2\phi}{\partial t^2})_{i,n}\frac{\Delta t^2}{2!}+\ldots-\phi_i^n}{\Delta t}$$

i.e.

$$\rho\frac{\phi_i^{n+1}-\phi_i^n}{\Delta t}-\rho(\frac{\partial\phi}{\partial t})_{i,n}=\frac{1}{2}\frac{\partial^2\phi}{\partial t^2}\Delta t+\ldots=O(\Delta t)$$

For the convection term

$$\rho u\frac{\phi_{i+1}^n-\phi_{i-1}^n}{2\Delta x}=\rho u[\frac{\phi_i^n+\Delta x\frac{\partial\phi}{\partial x}+\frac{1}{2}\Delta x^2\frac{\partial^2\phi}{\partial x^2}+O(\Delta x^3)}{2\Delta x}$$

$$-\frac{(\phi_i^n-\Delta x\frac{\partial\phi}{\partial x}+\frac{1}{2}\Delta x^2\frac{\partial^2\phi}{\partial x^2}+O(\Delta x^3))}{2\Delta x}]$$

$$=\frac{2\rho u\frac{\partial\phi}{\partial x}\Delta x+O(\Delta x^3)}{2\Delta x}=\rho u\frac{\partial\phi}{\partial x}+O(\Delta x^2)$$

$$\rho u\frac{\phi_{i+1}^n-\phi_{i-1}^n}{2\Delta x}-\rho u(\frac{\partial\phi}{\partial x})_{i,n}=O(\Delta x^2)$$

For the diffusion term

$$\Gamma\frac{\phi_{i+1}^n-2\phi_i^n+\phi_{i-1}^n}{\Delta x^2}-\Gamma\frac{d^2\phi}{dx^2}=O(\Delta x^2)$$

Finally, the following discretized equation is obtained:

$$\rho\frac{\phi_i^{n+1}-\phi_i^n}{\Delta t}+\rho u\frac{\phi_{i+1}^n-\phi_{i-1}^n}{2\Delta x}=\Gamma\frac{\phi_{i+1}^n-2\phi_i^n+\phi_{i-1}^n}{\Delta x^2},O(\Delta t,\Delta x^2)$$

$O(\Delta t, \Delta x^2)$ actually is the **T.E. of the discretized equation**.

2) Consistency (相容性) of discretized equations

If the **T.E. of a discretized equation approaches 0** when $\Delta t\to 0,\Delta x\to 0$ then: **the discretized equation is said to be consistent with the original PDE.**

Comment: when T.E. is in the form of $O(\Delta t^n,\Delta x^m)(n,m>0)$, the discretized equations possess(具有) consistency.

7.1.2 Discretization error and convergence

1. Discretization error(离散误差) ρ_i^n

$$\rho_i^n = \phi(i,n) - \phi_i^n$$

Accurate solution of **PDE**

Accurate solution of **discretized equation**

2. Factors affecting discretization error

(1) T.E.: The higher the order, the smaller the value of ρ_i^n for the same grid system; $O(\Delta t^n, \Delta x^m)(n,m>0)$

(2) Size of grid or time step: For the same order of accuracy, a finer grid system leads to less numerical error. $O(\Delta t^n, \Delta x^m)(n,m>0)$

7.1.3 Round-off error(舍入误差) and stability of initial problems

1）Round-off error ε_i^n

The round-off error originates from finite-precision floating-point arithmetic operations （有限精度浮点运算）

$$\varepsilon_i^n = \phi_i^n - \phi_i^n$$

ϕ_i^n -- actual solution from computer we can obtain.

Length of computer word such as double precision and single precision (字长, 如双精度，单精度) will affect the round-off error.

Errors of numerical solutions:

Dominant

$$\phi(i,n) - \phi_i^n = \underline{\phi(i,n) - \phi_i^n} + \underline{\phi_i^n - \phi_i^n} = \rho_i^n + \varepsilon_i^n$$

2) Stability of initial problems (初值问题)

Definition: An initial problem means **an unsteady problem (非稳态问题)**. To solve an unsteady problem, not only boundary condition but also initial condition should be provided.

Stable or unstable (稳定或不稳定): the solution procedure of an initial problem is of marching (**步进**) type; if errors introduced at any time level are enlarged (**放大**) in the subsequent (**随后的**) simulation such that the solutions become infinite (**无限**), this scheme is called **unstable (不稳定)**; Otherwise, the scheme for the initial problem is **stable**.

$$T_i^{n+1} = \frac{a\Delta t}{\Delta x^2} T_{i+1}^n + (1 - 2\frac{a\Delta t}{\Delta x^2}) T_i^n + \frac{a\Delta t}{\Delta x^2} T_{i-1}^n$$

Stability is an **inherent (固有的)** character of a **numerical scheme (格式)**, no matter what kind of error is introduced. **(Example 7-3 of the textbook)**

7.1.4 Example

Factors affecting discretization error (例7-1)

(1) T.E.: The higher the order, the smaller the value of ρ_i^n for the same grid system; $O(\Delta t^n, \Delta x^m)(n, m > 0)$

(2) Size of grid or time step: For the same order of accuracy, a finer grid system leads to less numerical error. $O(\Delta t^n, \Delta x^m)(n, m > 0)$

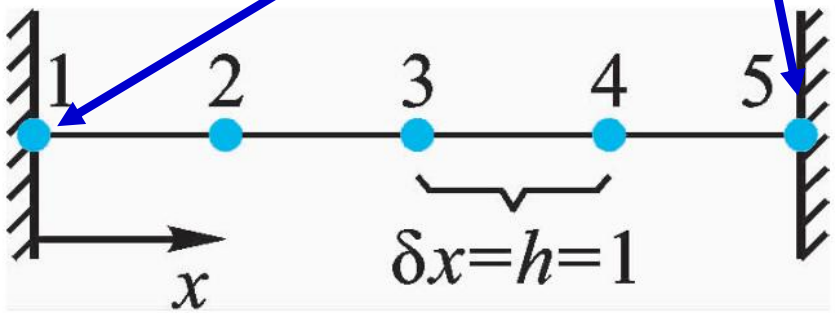
Stability of initial problems (初值问题, 例7-3)

Stability is an **inherent** (固有的) character of a **numerical scheme** (格式), no matter what kind of error is introduced.

7.1.4 Example

[Example 7-1 of Textbook] Effects of T.E. and grid number

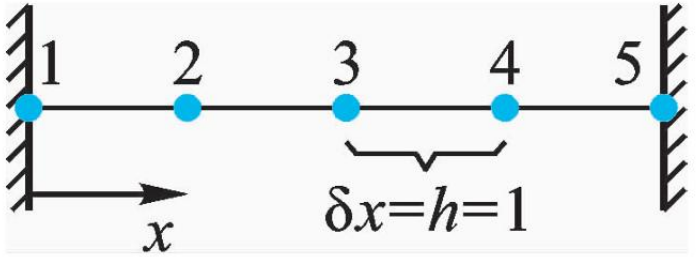
$$\frac{d^2\phi}{dx^2}+\frac{d\phi}{dx}-2\phi=0\,,\ \phi(0)=0;\ \ \phi(L)=1$$



Find: Using schemes with 2-order accuracy or 4-order accuracy to discretize the equation, and solve the values of 5 uniformly distributed nodes.

Table 3-2

导数	差分表示式	格式图案	截差
$\left(\frac{\partial\phi}{\partial x}\right)_{i,n}$	$\frac{\phi_{i+1}^n-\phi_i^n}{\Delta x}$	i $i+1$ x	$O(\Delta x)$
	$\frac{\phi_i^n-\phi_{i-1}^n}{\Delta x}$	$i-1$ i x	$O(\Delta x)$
	$\frac{\phi_{i+1}^n-\phi_{i-1}^n}{2\Delta x}$	$i-1$ i $i+1$ x	$O(\Delta x^2)$
	$\frac{-3\phi_i^n+4\phi_{i+1}^n-\phi_{i+2}^n}{2\Delta x}$	i $i+1$ $i+2$ x	$O(\Delta x^2)$
	$\frac{3\phi_i^n-4\phi_{i-1}^n+\phi_{i-2}^n}{2\Delta x}$	$i-2$ $i-1$ i x	$O(\Delta x^2)$
	$\frac{4\phi_{i+1}^n+6\phi_i^n-12\phi_{i-1}^n+2\phi_{i-2}^n}{12\Delta x}$	$i-2$ $i-1$ i $i+1$ x	$O(\Delta x^3)$
	$\frac{-2\phi_{i+2}^n+12_{i+1}^n-6\phi_i^n-4\phi_{i-1}^n}{12\Delta x}$	$i-1$ i $i+1$ $i+2$ x	$O(\Delta x^3)$
	$\frac{\phi_{i-2}^n-8\phi_{i-1}^n+8\phi_{i+1}^n-\phi_{i+2}^n}{12\Delta x}$	$i-2$ $i-1$ i $i+1$ $i+2$ x	$O(\Delta x^4)$
$\left(\frac{\partial^2\phi}{\partial x^2}\right)_{i,n}$	$\frac{\phi_i^n-2\phi_{i+1}^n+\phi_{i+2}^n}{\Delta x^2}$	i $i+1$ $i+2$ x	$O(\Delta x)$
	$\frac{\phi_i^n-2\phi_{i-1}^n+\phi_{i-2}^n}{\Delta x^2}$	$i-2$ $i-1$ i x	$O(\Delta x)$
	$\frac{\phi_{i+1}^n-2\phi_i^n+\phi_{i-1}^n}{\Delta x^2}$	$i-1$ i $i+1$ x	$O(\Delta x^2)$
	$\frac{(-\phi_{i-2}^n+16\phi_{i-1}^n-30\phi_i^n+16\phi_{i+1}^n-\phi_{i+2}^n}{12\Delta x^2}$	$i-2$ $i-1$ i $i+1$ $i+2$ x	$O(\Delta x^4)$



If the 2-order scheme is adopted, then two neighbor nodes are needed. This is satisfied by nodes 2, 3 and 4.

However, for 4-order scheme, only node 3 satisfies, while 2 and 4 have to use 2-order.

With the **2-order schemes**, the following discretized equations are obtained:

$$-(2-h)\phi_{i-1}+4(1+h^2)\phi_i-(2+h)\phi_{i+1}=0;\quad i=2,\ 3,\ 4$$

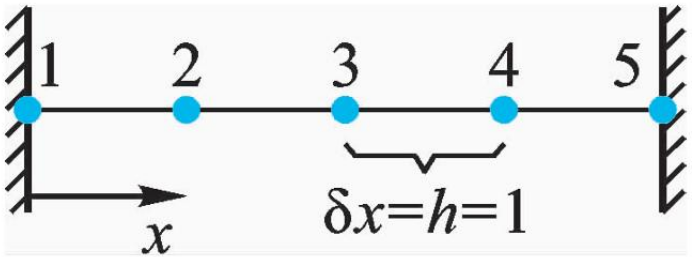
With the **4-order schemes**, the following discretized equations are obtained for only node 3:

$$\left(\frac{1}{12h}-\frac{1}{12h^2}\right)\phi_{i-2}+\left(\frac{16}{12h^2}-\frac{8}{12h}\right)\phi_{i-1}-\left(\frac{30}{12h^2}+2\right)\phi_i+\left(\frac{16}{12h^2}+\frac{8}{12h}\right)\phi_{i+1}-\left(\frac{1}{12h^2}+\frac{1}{12h}\right)\phi_{i+2}=0;\quad i=3$$

$$-(2-h)\phi_{i-1}+4(1+h^2)\phi_i-(2+h)\phi_{i+1}=0;\quad i=2,\qquad 4$$

The analytical solution

$$\phi=\frac{e^x-e^{2x}}{e^4-e^{-8}}$$



格式	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5
精确解	0	0.0473	0.1350	0.3679	1
(i=2，3，4)二阶格式	0	0.0582	0.1552	0.3944	1
(i=3)四阶格式，其余二阶格式	0	0.0505	0.1348	0.3918	1

Even with only one node using the 4-order accuracy scheme, the solution is more accurate.

区间数	4	8	16	32	64	精确解
$\phi_{x=1}$	0.0582	0.0502	0.0480	0.0475	0.0473	0.0473
$\phi_{x=2}$	0.1552	0.1404	0.1364	0.1353	0.1350	0.1350
$\phi_{x=3}$	0.3944	0.3752	0.3697	0.3683	0.3679	0.3679

With the grid size decreases, or the Δx decreases, the TE decreases.

[Ex. 7-3 of Textbook] Instability of explicit scheme

$$\frac{\partial T}{\partial t}=a\frac{\partial^2 T}{\partial x^2};\quad 0<x<1$$

$$t\leqslant 0,\ T=2x,\ 0\leqslant x\leqslant 0.5;\quad T=2(1-x),\ 0.5\leqslant x\leqslant 1$$

$$t>0,\quad T(0,\ t)=T(1,\ t)=0$$

For $\frac{a\Delta t}{\Delta x^2}$ as 0.48 and 0.52, solving the equations and compare with the analytical solution of the PDE.

$$\frac{T_i^{n+1}-T_i^n}{\Delta t}=a\frac{T_{i+1}^n-2T_i^n+T_{i-1}^n}{\Delta x^2},O(\Delta t,\Delta x^2)$$

$$T_i^{n+1}=\frac{a\Delta t}{\Delta x^2}T_{i+1}^n+(1-2\frac{a\Delta t}{\Delta x^2})T_i^n+\frac{a\Delta t}{\Delta x^2}T_{i-1}^n$$

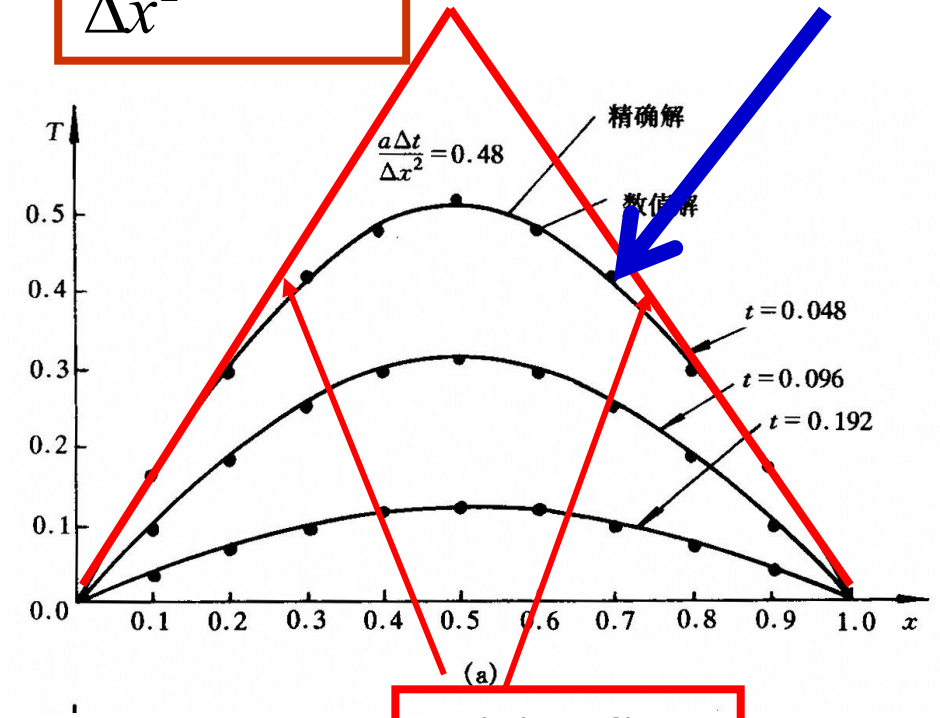
$$\frac{a\Delta t}{\Delta x^2}<0.5$$

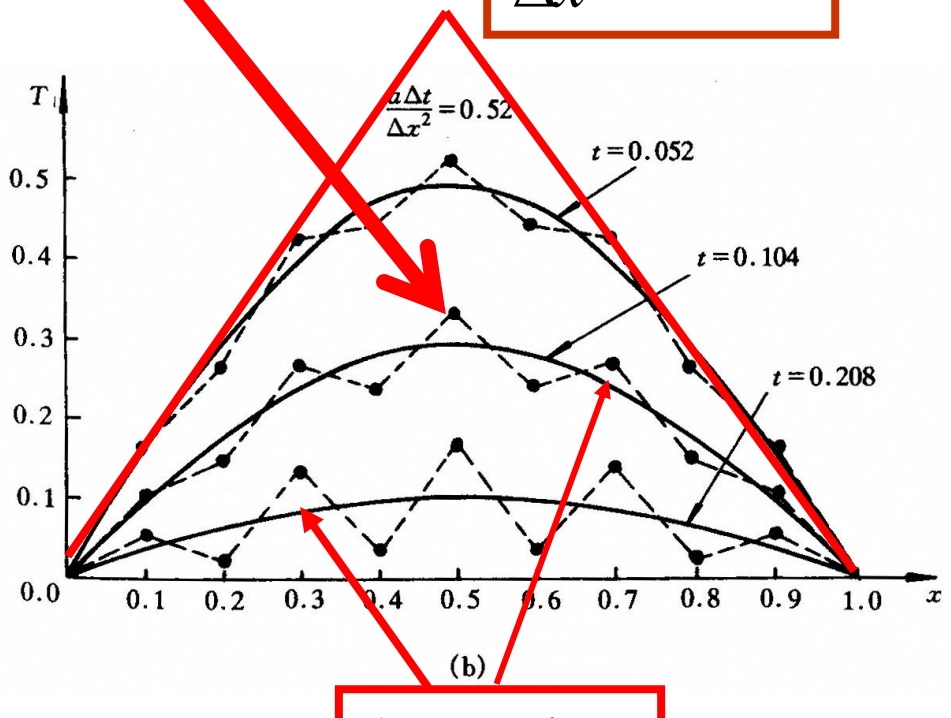
$t \leqslant 0,\ T=2x,\ 0 \leqslant x \leqslant 0.5;\ T=2(1-x),\ 0.5 \leqslant x \leqslant 1$

$$\frac{a\Delta t}{\Delta x^2}=0.48$$

Numerical solution

$$\frac{a\Delta t}{\Delta x^2}=0.52$$





Initial field

Analytical

Numerical solutions are converged, but oscillating(振荡的) . This is because of the physically meaningless coefficients of the algebraic equations .

Contents

7.1 Consistency, Convergence and Stability of Discretized Equations

7.2 von Neumann Method for Analysing Stability of Initial Problems

7.3 Conservation of Discretized Equations

7.4 Classification and V&V of Numerical Solution Errors

7.5 Analysis of Truncation Error of Discretization Schemes

7.6 Analysis of Grid Convergence Index and Incomplete Iteration Error

7.2 von Neumann Method for Analyzing Stability of Initial Problems

7.2.1 Propagation of error vector with time

7.2.2 Fourier expansion

7.2.3 Basic idea of von Neumann analysis

7.2.4 Examples of von Neumann analysis

Review: Section 7-1: Stability of initial problems (初值问题)

Stable or unstable (稳定或不稳定): the solution procedure of an initial problem is of marching (步进) type; if errors introduced at any time level are enlarged (放大) in the subsequent (随后的) simulation such that the solutions become infinite (无限), this scheme is called **unstable** (不稳定); Otherwise, the scheme for the initial problem is **stable**.

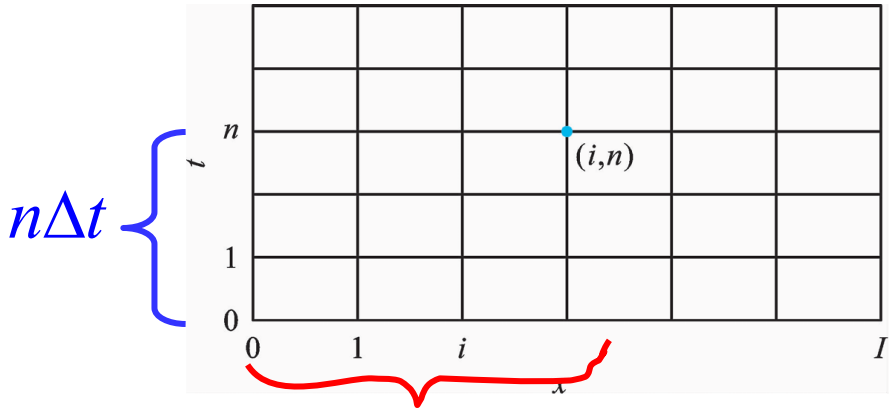
Question 1: how does the error propagate (传递) as time step proceeds? (Marching)

7.2 von Neumann Method for Analyzing Stability of Initial Problems

7.2.1 Propagation (传递) of error vector with time

1 Matrix expression of discretized equations

$$\begin{cases} \dfrac{\partial T}{\partial t} = a\dfrac{\partial^2 T}{\partial x^2}, \quad 0 < x < L \\ T(x,0) = F(x) \\ T(0,t) = f_1(t),\ T(L,t) = f_2(t) \end{cases}$$



$$\begin{cases} \dfrac{T_i^{n+1} - T_i^n}{\Delta t} = a\dfrac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2},\ i = 1,2,,3,....(I-1) \\ T_i^0 = F(x_i),\ i = 0,1,2,3.....I \quad \text{Initial condition} \\ T_0^n = f_1(n\Delta t),\ T_I^n = f_2(n\Delta t),\ n = 1,2,..... \end{cases} \quad \text{(a)}$$

Boundary condition

Set $\frac{a\Delta t}{\Delta x^2} = r$, the difference eqs. can be expressed as

$$T_i^{n+1} = T_i^n(1-2r) + r(T_{i+1}^n + T_{i-1}^n),\ i = 1,2,....(I-1)$$

For a fixed time level n, the above eqs. can be rewritten for each inner point as follows:

$$\begin{cases} i=1, & T_1^{n+1} = T_1^n(1-2r) + r(T_0^n + T_2^n) \\ i=2, & T_2^{n+1} = T_2^n(1-2r) + r(T_1^n + T_3^n) \\ i=3, & T_3^{n+1} = T_3^n(1-2r) + r(T_2^n + T_4^n) \\ \cdots\cdots\cdots\cdots & \\ i=I-2, & T_{I-2}^{n+1} = T_{I-2}^n(1-2r) + r(T_{I-1}^n + T_{I-3}^n) \\ i=I-1, & T_{I-1}^{n+1} = T_{I-1}^n(1-2r) + r(T_I^n + T_{I-2}^n) \end{cases}$$

Expressed in matrix (矩阵)form:

$$T_1^{n+1} = T_1^n(1-2r) + r(T_0^n + T_2^n)$$

$$\begin{bmatrix} T_1^{n+1} \\ T_2^{n+1} \\ T_3^{n+1} \\ \cdot \\ \cdot \\ \cdot \\ T_{I-2}^{n+1} \\ T_{I-1}^{n+1} \end{bmatrix} = \begin{bmatrix} (1-2r) & r & & & & \\ r & (1-2r) & r & & \mathbf{0} & \\ & r & (1-2r) & r & & \\ & & \cdot & \cdot & \cdot & \\ & \mathbf{0} & & \cdot & \cdot & \\ & & & & r & (1-2r) \end{bmatrix} \begin{bmatrix} T_1^n \\ T_2^n \\ T_3^n \\ \cdot \\ \cdot \\ \cdot \\ T_{I-2}^n \\ T_{I-1}^n \end{bmatrix} + \begin{bmatrix} rT_0^n \\ 0 \\ 0 \\ \cdot \\ \cdot \\ 0 \\ rT_I^n \end{bmatrix}$$

$\vec{T}^{n+1}$ $\vec{A}$ Matrix(矩阵) $\vec{T}^{n}$ $\vec{g}$

Simply:

$$\begin{aligned} \vec{T}_0^{n+1} &= \vec{A}\vec{T}^n + \vec{g} \\ \vec{T}^0 &= \vec{F} \end{aligned} \quad \text{(b)}$$

$$i = I-1, \quad T_{I-1}^{n+1} = T_{I-1}^n(1-2r) + r(T_I^n + T_{I-2}^n)$$

$\vec{A}$ represents a transformation(变换) from $\vec{T}^{n}$ to $\vec{T}^{n+1}$

2 Propagation (传递) of error vector with time

Assuming that no error is introduced at the boundary, while it is introduced at the initial condition . Then the error components at each node form(形成) an error vector, denoted by $\vec{\varepsilon}^{0}$:

For the exact solution:

$$\left.\begin{aligned}\vec{T}^{n+1} &= \vec{A}\vec{T}^{n} + \vec{g}\\ \vec{T}^{0} &= \vec{F}\end{aligned}\right\} \text{(b)}$$

Denoting the solution with error by $\tilde{\vec{T}}$:

then

$$\left.\begin{aligned}\tilde{\vec{T}}^{n+1} &= \vec{A}\tilde{\vec{T}}^{n} + \vec{g}\\ \tilde{\vec{T}}^{0} &= \vec{F} + \vec{\varepsilon}^{0}\end{aligned}\right\} \text{(c)}$$

(c)−(b) →

$$\begin{cases}\tilde{\vec{T}}^{n+1} - \vec{T}^{n+1} = \vec{A}\left(\tilde{\vec{T}}^{n} - \vec{T}^{n}\right)\\ \tilde{\vec{T}}^{0} - \vec{T}^{0} = \vec{\varepsilon}^{0}\end{cases}$$

That is:

$$\vec{\varepsilon}^{\,n+1} = \vec{A}\vec{\varepsilon}^{\,n},$$

with $\vec{\varepsilon}^{\,0}$ being specified (给定)

(d)

The marching (步进) process of ***T***

$$\vec{T}^{\,n+1} = \vec{A}\vec{T}^{\,n} + \vec{g}$$

$$\vec{T}^{\,0} = \vec{F}$$

(b)

Thus the propagation of error vector can be described by the same matrix $\vec{A}$ under the condition that:

No error is introduced at the boundary!

$$\vec{\varepsilon}^{\,n+1} = \vec{\vec{A}}\,\vec{\varepsilon}^{\,n},$$

with $\vec{\varepsilon}^{\,0}$ being specified (给定)

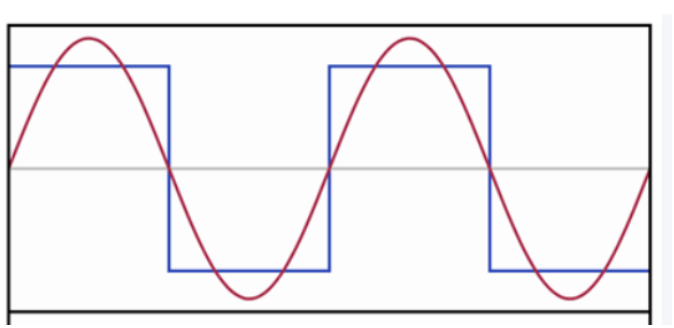
The above equation tells us how the error vector propagates with time

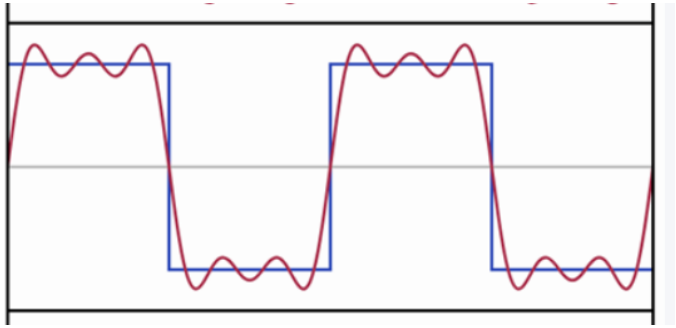
Question 2: How to mathematically describe the error? Or what is the expression of error?

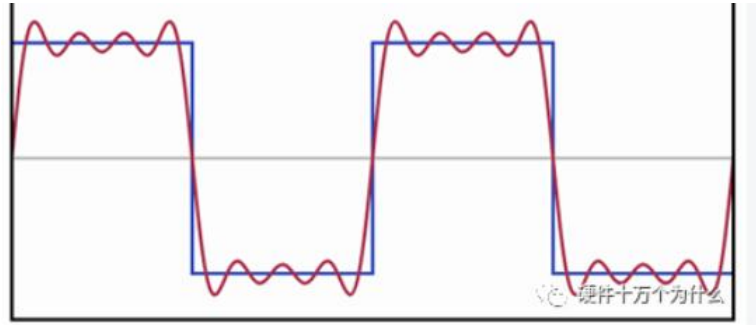
7.2.2 Discrete Fourier expansion(离散傅里叶展开)

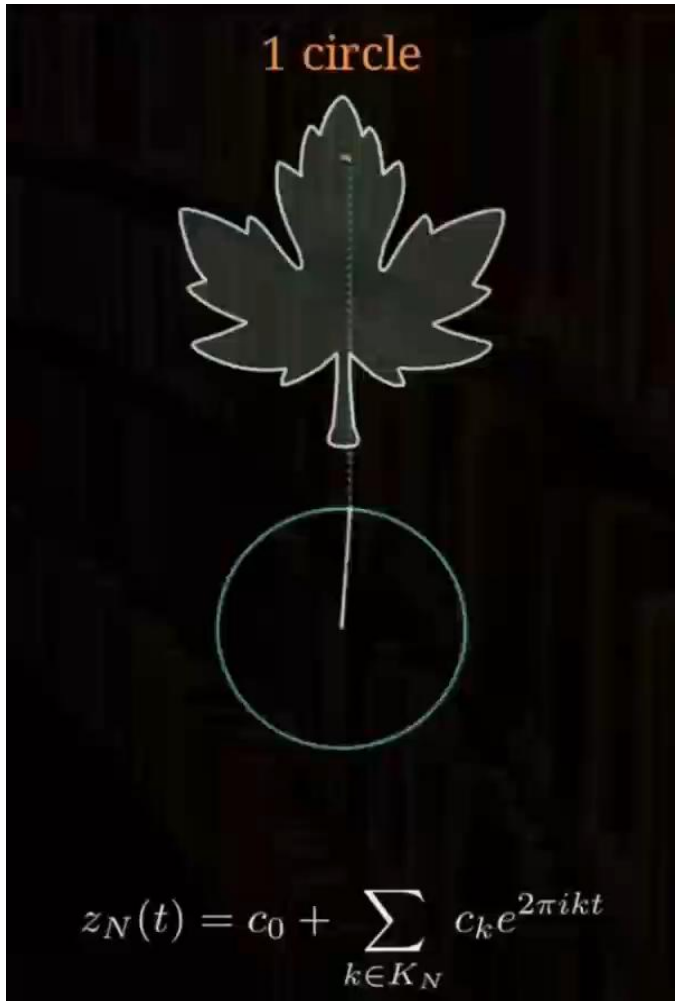
1) Expression of error vector (误差矢量的表示方法)

- Any **error vector, or in general, any curve,** can be regarded as consisting of a finite number of **components with different frequencies (不同频率的分量)** .
- To analyze the propagation of the error vector during computation, one of its components can be selected for investigation.
- The **discrete Fourier expansion is a powerful tool** to obtain the expression of different components.

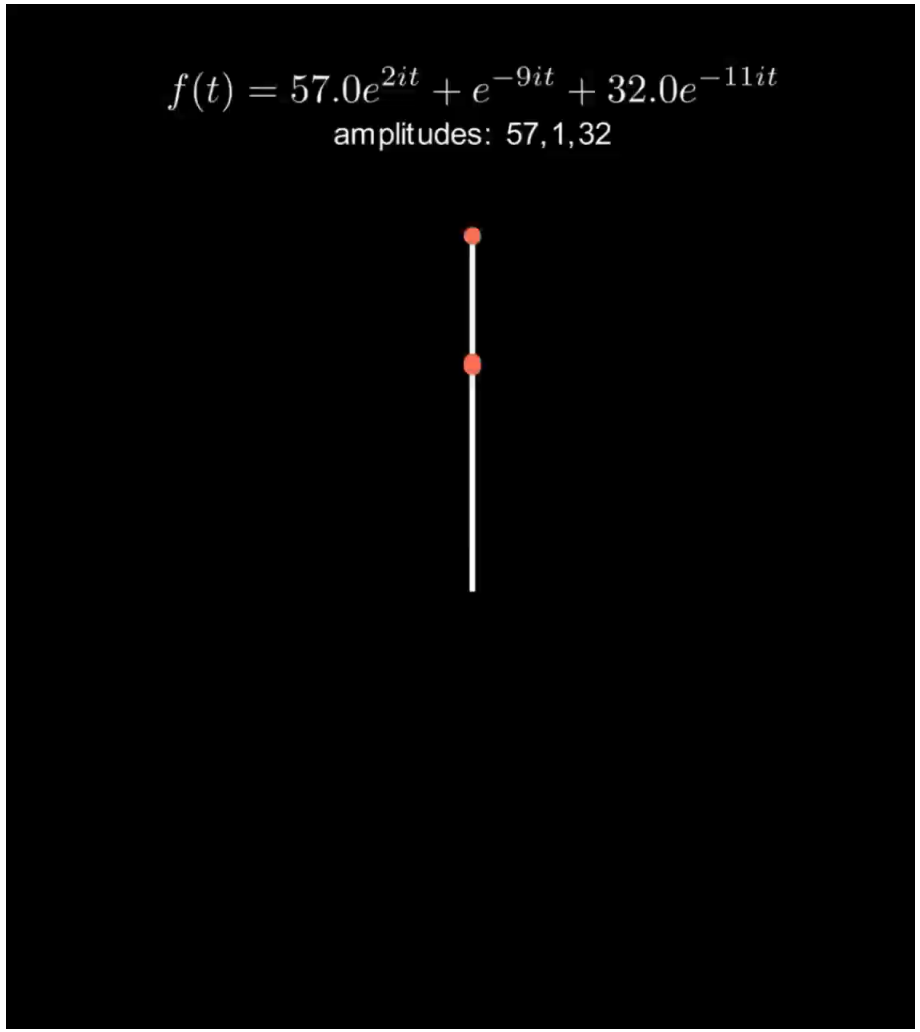




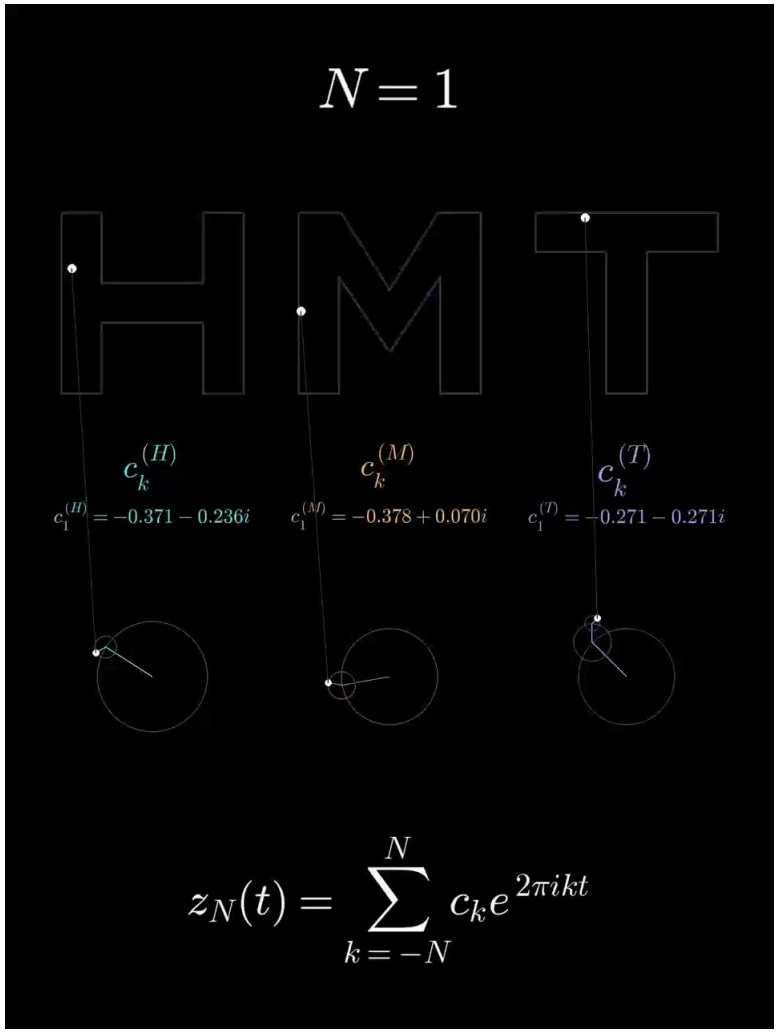




The power of Fourier expansion



The power of Fourier expansion



The power of Fourier expansion

The Fourier series (傅里叶级数)

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{2\pi n}{L} x\right) + b_n \sin\left(\frac{2\pi n}{L} x\right) \right]$$

—$f(x)$, the curve to be analyzed or decomposed (分解）, or the error vector to be analyzed.

—L, period of the function (周期)

—x, independent spatial variable (空间坐标)

—n, (n=1,2,3,…) index of harmonic wave (谐波序号)

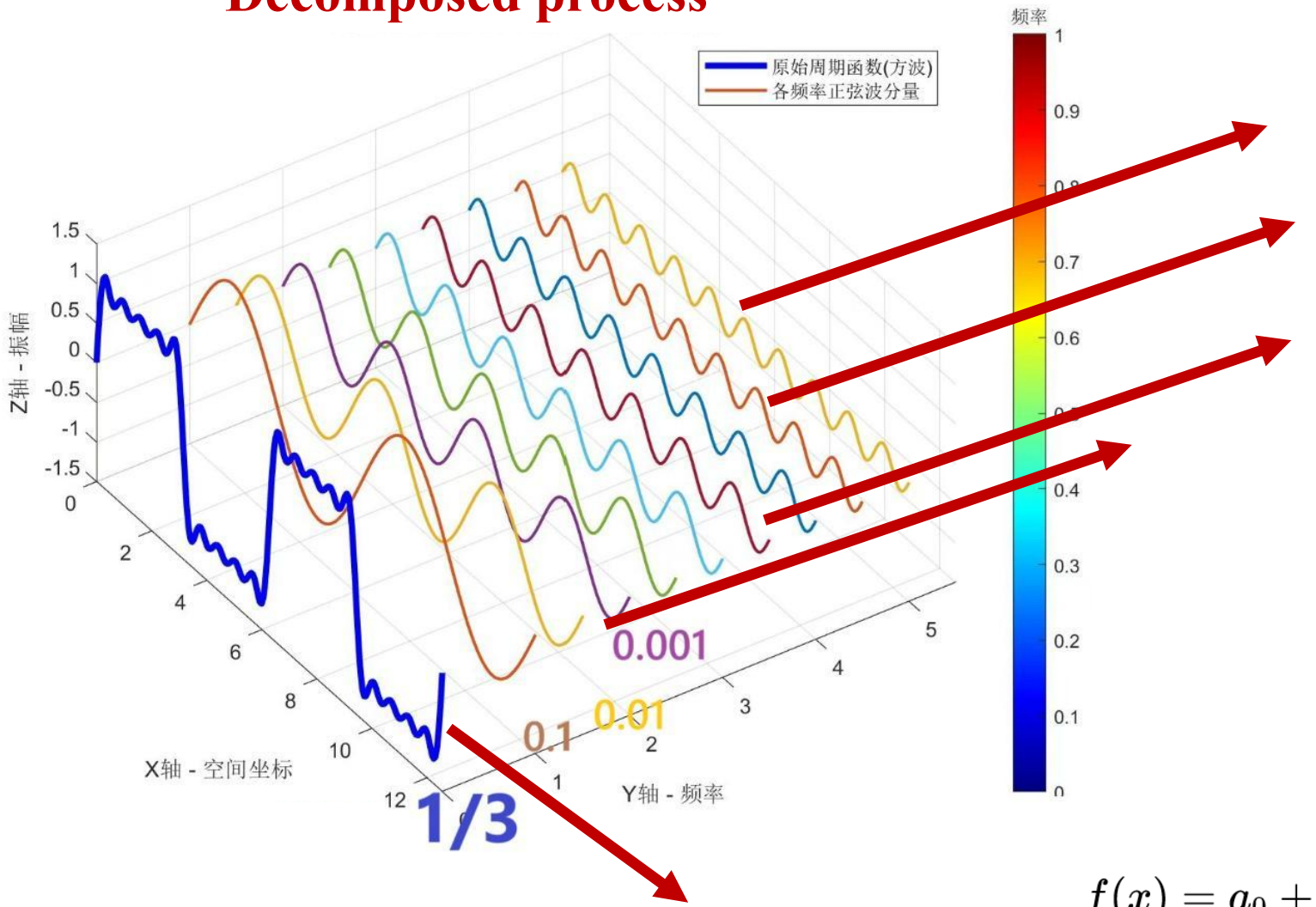
—a_0, DC constant component (直流分量), mean value of $f(x)$ over one period

—a_n, Fourier coefficient for cosine terms （余弦分量的幅值系数）

—b_n, Fourier coefficient for sine terms （正弦分量的幅值系数）

—$2\pi n/L$, angular wavenumber k_n, the larger the k_n, the denser the waveform. (波形越密)

Decomposed process



The curves with different frequency

The original curve

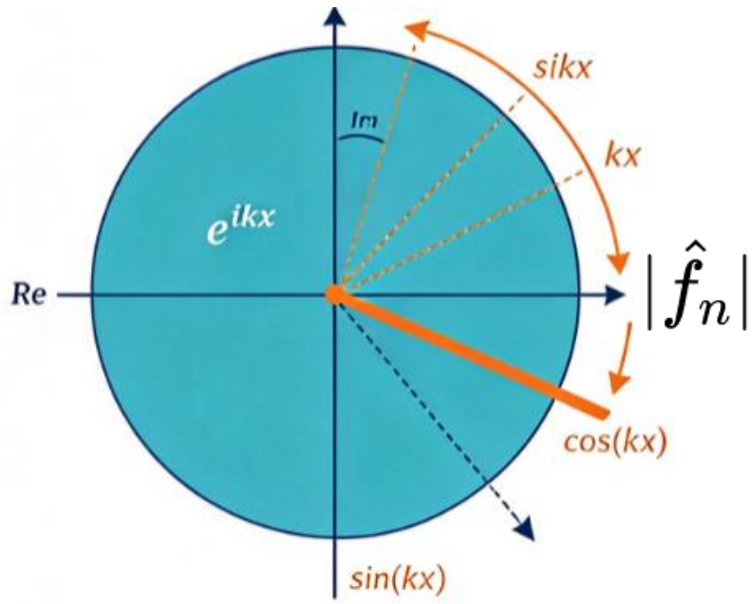
$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{2\pi n}{L} x\right) + b_n \sin\left(\frac{2\pi n}{L} x\right) \right]$$

The Fourier series (傅里叶级数) : real trigonometric form实三角函数形式

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{2\pi n}{L} x\right) + b_n \sin\left(\frac{2\pi n}{L} x\right) \right]$$

The complex exponential form (复指数形式)

$$f(x) = \sum_{n=-\infty}^{+\infty} \hat{f}_n e^{ik_n x}, \quad k_n = \frac{2\pi n}{L}$$



with the Euler's formula (欧拉公式)

$$e^{ikx} = \cos(kx) + i\sin(kx) \quad i = \sqrt{-1}$$

$kx = \theta$, radian angle imaginary unit

The point undergoes rotation about the origin on the unit circle with radius as $|\hat{f}_n|$

The Fourier series (傅里叶级数）： real trigonometric form实三角函数形式

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{2\pi n}{L} x\right) + b_n \sin\left(\frac{2\pi n}{L} x\right) \right]$$

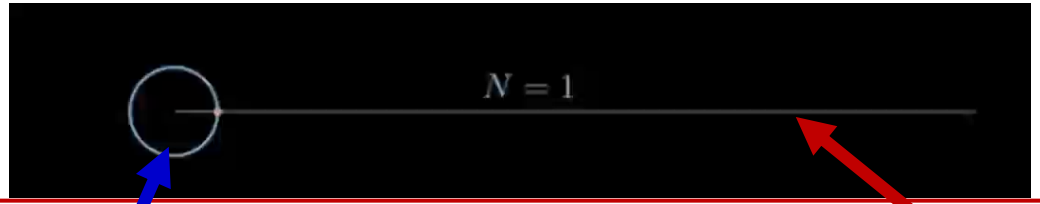
Meaning: A periodic function is decomposed into the superposition of infinitely many real sine and cosine waves.

The complex exponential form (复指数形式)

$$f(x) = \sum_{n=-\infty}^{+\infty} \hat{f}_n e^{ik_n x}, \quad k_n = \frac{2\pi n}{L}$$

Meaning: A periodic function equals the superposition of infinitely many rotating vectors with different rotating speeds, magnitudes and initial phases.

Complex plane



Rectangular coordinates

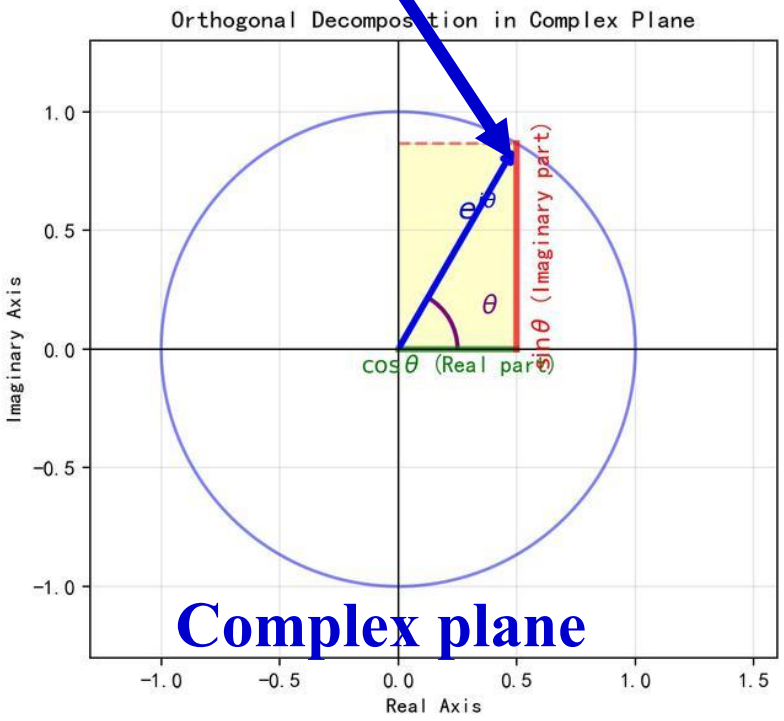
Trigonometric form

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{2\pi n}{L} x\right) + b_n \sin\left(\frac{2\pi n}{L} x\right) \right]$$

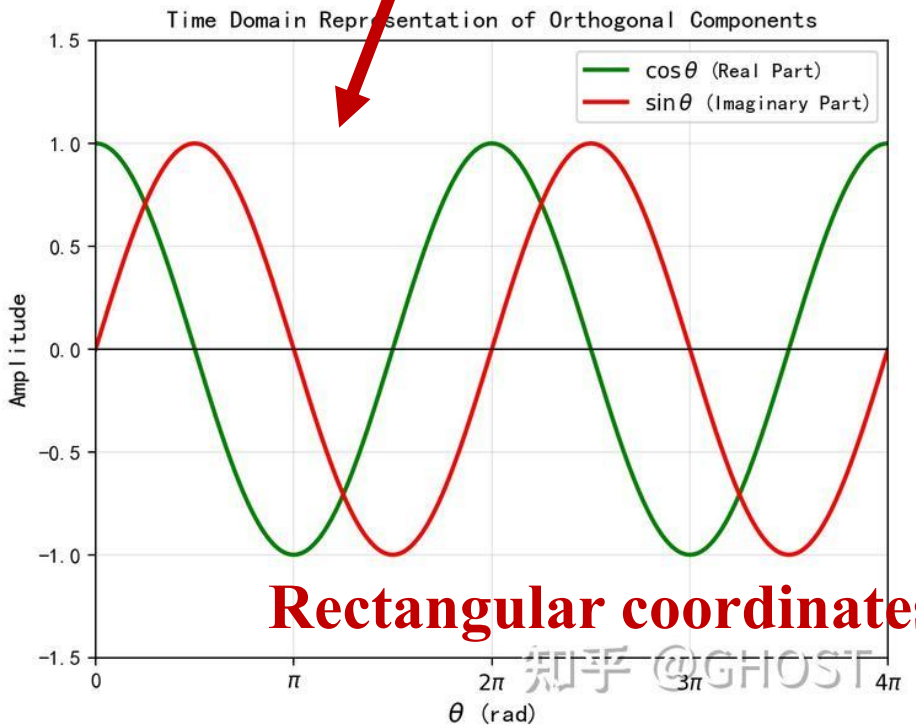
Complex Plane Form

$$f(x) = \sum_{n=-\infty}^{+\infty} \hat{f}_n e^{i k_n x}, \quad k_n = \frac{2\pi n}{L}$$

Mathematically equivalent; only the basis functions differ.

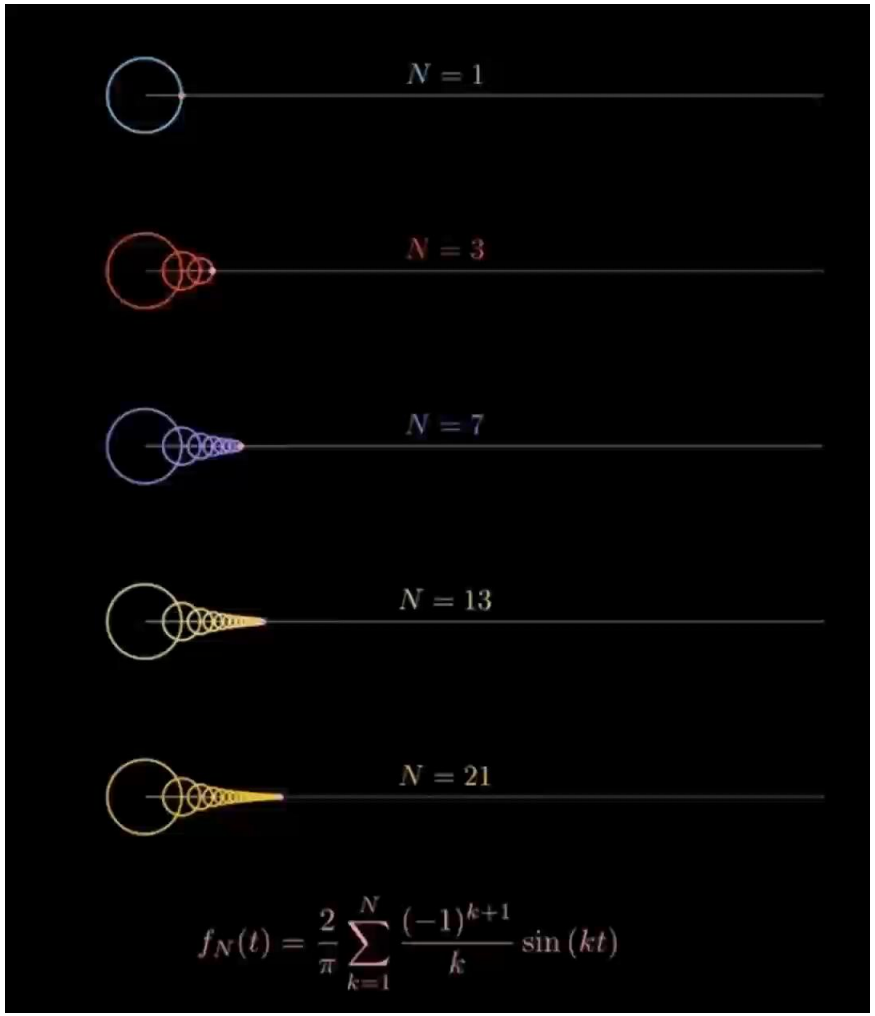


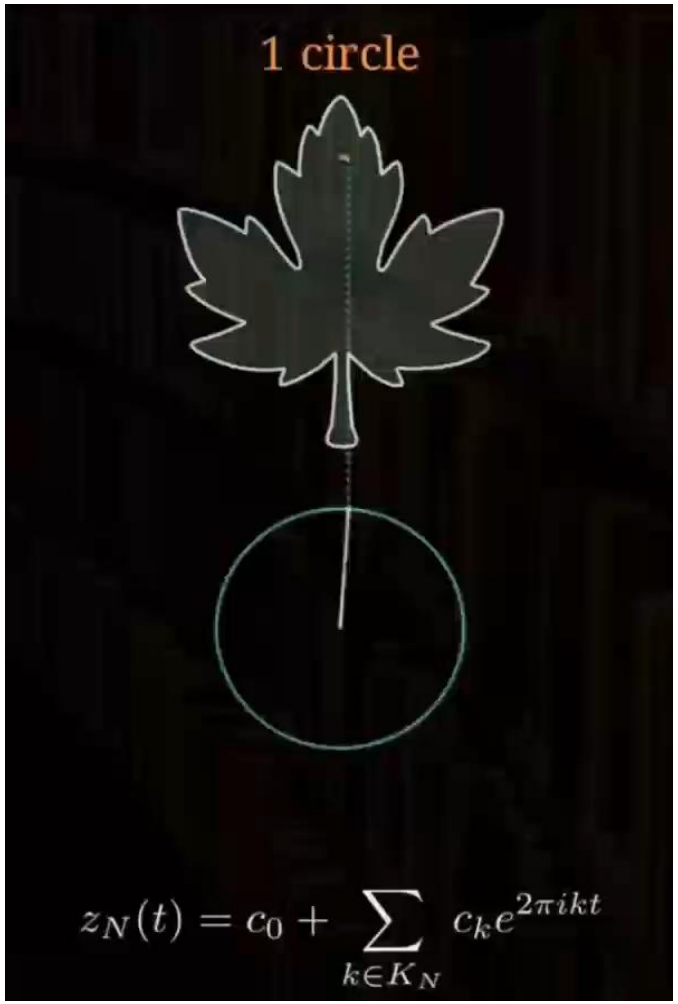
Complex plane



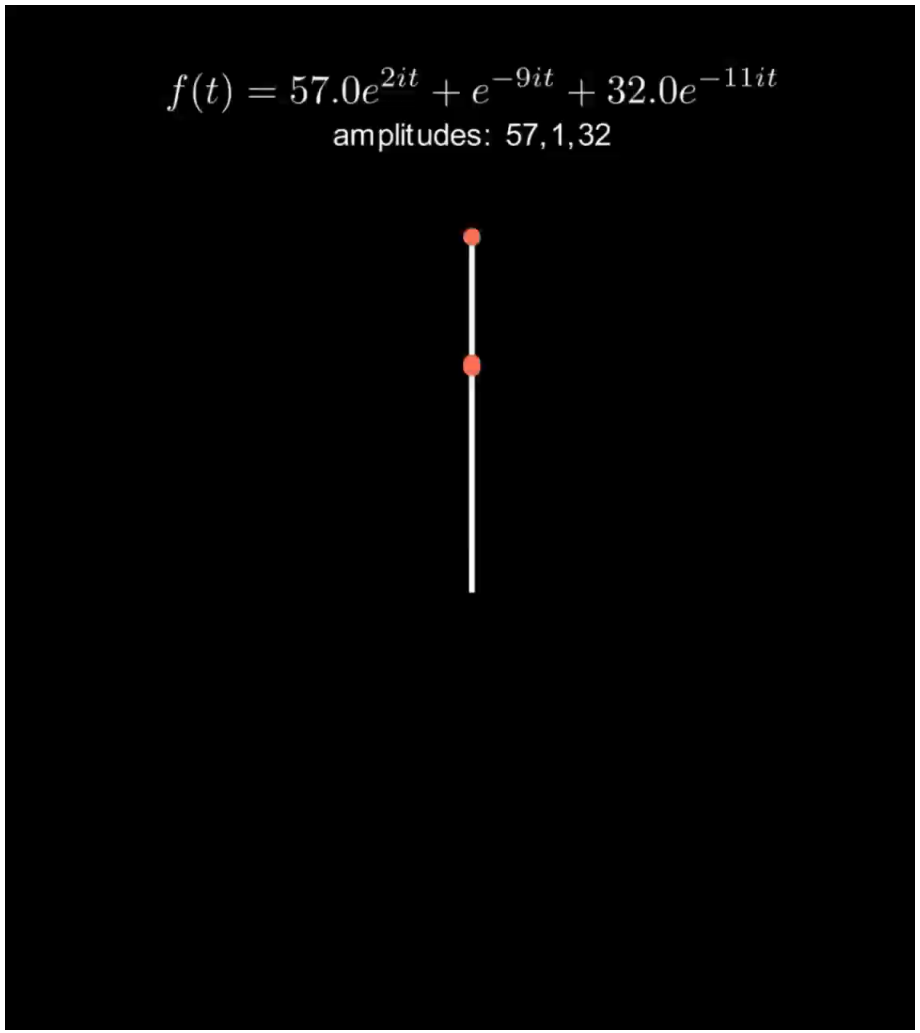
Rectangular coordinates

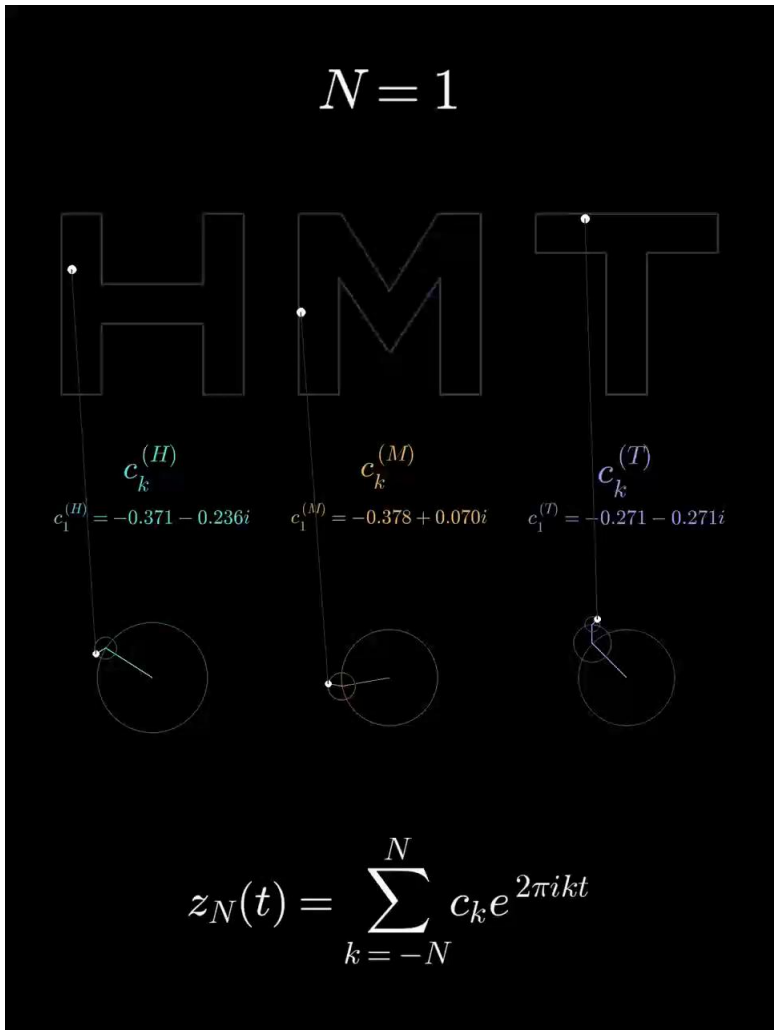
知乎 @GHOST





The power of Fourier expansion





The power of Fourier expansion

Let us go back to the textbook, a function $f(x)$ satisfying the Dirichlet conditions on the domain ($[-l, l]$) can be expanded into a Fourier series in the complex form

$$f(x) = \sum_{n=-\infty}^{+\infty} C_n e^{In\pi x/l}; I = \sqrt{-1}$$

imaginary unit

Similarly, given ($2N+1$) data pairs (x_i, y_i) (i=0,1,2,…$2N$), y_i can be expressed as a function of x_i by a trigonometric polynomial

$$y_i = \sum_{k=-N}^{+N} C_k e^{I\left(\frac{2\pi k}{2N+1}\right)x_i}; I = \sqrt{-1}, k = 0,1,2,\cdots,2N$$

which is the Fourier expansion of y_i.

Comment:

The harmonic index k=$2N$+1 equals to the data pairs 2N+1, therefore neither overfitted nor underfitted (**谐波数量等于数对，既不过拟合，也不欠拟合**)

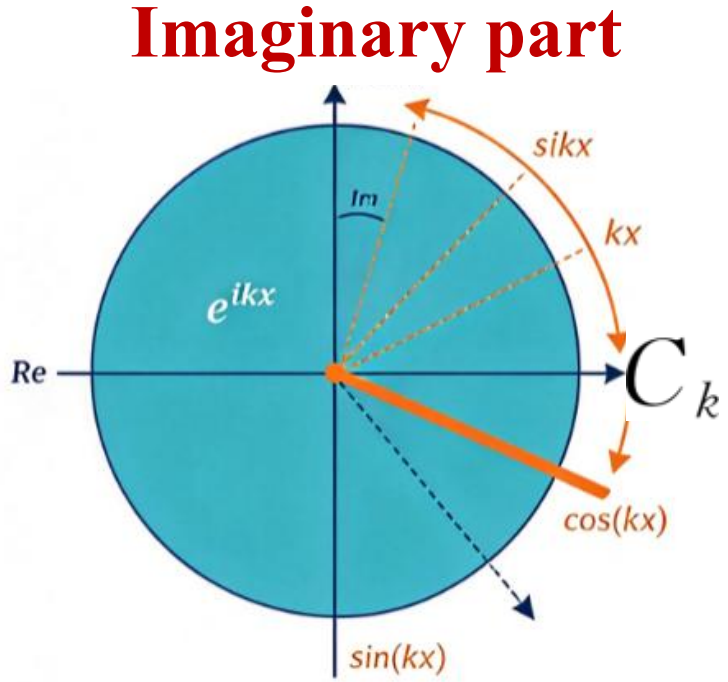
If x_i is uniformly distributed, then

$$y_i = \sum_{k=-N}^{+N} C_k \mathrm{e}^{\mathrm{I}\left(\frac{2\pi k}{2N+1}\right)x_i} ; \mathrm{I} = \sqrt{-1}, k = 0,1,2,\cdots,2N$$

$$\frac{2\pi k}{2N+1}x_i = \frac{2\pi k}{2N+1}(i\Delta x) = i\,\frac{2\pi k}{2N+1}\Delta x = ik_x\Delta x = i\theta$$

Compare with the original Euler's formula

$$e^{ikx} = \cos(kx) + i\sin(kx)$$



It can be found that:

—k, (k=1,2,3,…) harmonic wavy index (谐波序号)
—θ, the radian angle (相位角)

Therefore:

$$y_i = \sum_{k=-N}^{+N} C_k \mathrm{e}^{\mathrm{I}\left(\frac{2\pi k}{2N+1}\right)x_i};\mathrm{I} = \sqrt{-1}, k = 0,1,2,\cdots,2N$$

$$\frac{2\pi k}{2N+1}x_i = \frac{2\pi k}{2N+1}(i\Delta x) = i\,\frac{2\pi k}{2N+1}\Delta x = ik_x\Delta x = i\theta$$

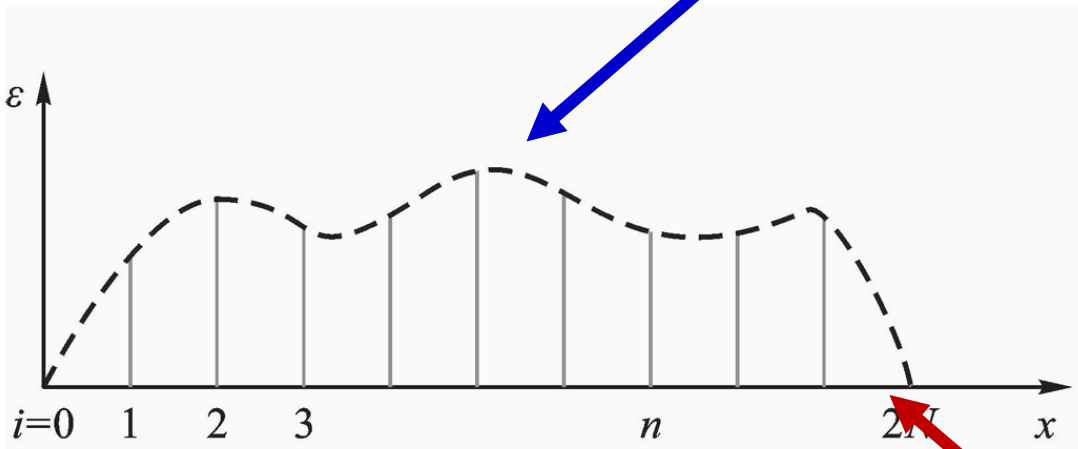
$$C_k e^{I\left(\frac{2k\pi}{2N+1}\right)x_i} = C_k e^{I\,i\,\theta_k}$$

C_k – amplitude (**振幅**) In transient problem it is a function of time, $\psi(t)$

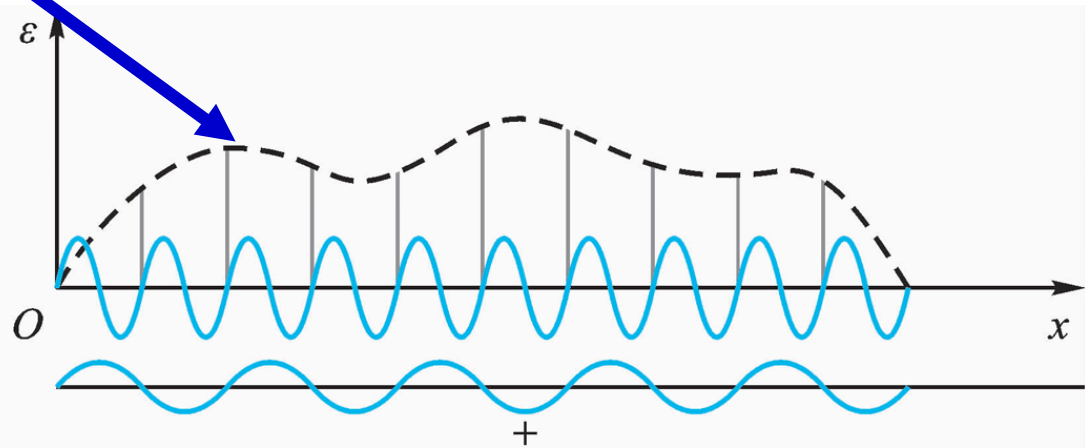
The general expression of harmonic component is then $\psi(t)e^{I\,i\,\theta}$

* Getting this form of harmonic components is the purpose of discussion on discrete Fourier-expansion

Error curve (误差曲线)



(a) Discrete-component representation
(离散分量形式)



(b) Harmonic-component representation
(谐波分量形式)

$$y_i = \sum_{k=-N}^{+N} C_k \mathrm{e}^{\mathrm{I}\left(\frac{2\pi k}{2N+1}\right)x_i};\mathrm{I} = \sqrt{-1}, k = 0,1,2,\cdots,2N$$

The general expression of harmonic component: $\psi(t)e^{Ii\theta}$

$$\vec{\varepsilon}^{\,n+1} = \vec{A}\,\vec{\varepsilon}^{\,n},$$

with $\vec{\varepsilon}^{\,0}$ being specified (给定)

The above equation tells us how the error vector propagates with time

$$\psi(t)e^{I\,i\,\theta}$$

The above equation tells us the expression of the error

Question 3: How to analyze the stability of the initial problem?

7.2.3 Basic idea of von Neumann analysis

1）Basic idea

The **numerical error** can be decomposed(**分解**) into a finite number of **harmonic components** (**谐波分量**) ; If a discretized scheme can guarantee that the amplitude of any component will be decreased (**衰减**) or at least be kept unchanged in the calculation procedure then the scheme is stable; Otherwise, it is unstable.

2）Analysis method

How to implement (**实施**) this idea? Replacing the dependent variable by the expression of a harmonic component , finding the ratio of amplitude of two subsequent time levels, and demanding (**要求**) that

$$\left|\frac{\psi(t+\Delta t)}{\psi(t)}\right| \le 1$$

$$\left|\frac{\psi(t+\Delta t)}{\psi(t)}\right| \le 1$$

The condition of this inequality is the criterion of scheme stability.

The ratio is called magnified factor (放大因子)

7.2.4 Examples

Stability analysis for FTCS of 1-D conduction eq.

$$\frac{T_i^{n+1}-T_i^n}{\Delta t} = a\frac{T_{i+1}^n-2T_i^n+T_{i-1}^n}{\Delta x^2}$$

Replacing T in the discretized eq. by $\varepsilon(t)=\psi(t)e^{Ii\theta}$

yields $$\frac{\psi(t+\Delta t)-\psi(t)}{\Delta t}e^{Ii\theta} = a\psi(t)\frac{e^{I(i+1)\theta}-2e^{Ii\theta}+e^{I(i-1)\theta}}{\Delta x^2}$$

Divided by $e^{Ii\theta}$ and from Euler Eq. $e^{I\theta}=\cos\theta+I\sin\theta$

Rearranging, $$\frac{\psi(t+\Delta t)}{\psi(t)} = 1-2\left(\frac{a\Delta t}{\Delta x^2}\right)(1-\cos\theta)$$

$$1-\cos\theta = 2(\sin\frac{\theta}{2})^2 \rightarrow \frac{\psi(t+\Delta t)}{\psi(t)} = 1-4(\frac{a\Delta t}{\Delta x^2})\sin^2(\frac{\theta}{2})$$

Stability condition requires:

$$-1 \le \frac{\psi(t+\Delta t)}{\psi(t)} \le 1 \quad \textbf{i.e.,} \quad -1 \le 1-4(\frac{a\Delta t}{\Delta x^2})\sin^2(\frac{\theta}{2}) \le 1$$

Automatically satisfied

Thus, it is required:

$$-1 \le 1-4(\frac{a\Delta t}{\Delta x^2})\sin^2(\frac{\theta}{2}) \rightarrow 4(\frac{a\Delta t}{\Delta x^2})\sin^2(\frac{\theta}{2}) \le 2$$

This requirement should be satisfied for all possible values of θ , the most severe case is $\sin^2(\theta/2)=1$

$$4\left(\frac{a\Delta t}{\Delta x^2}\right)\sin^2\left(\frac{\theta}{2}\right)\le 2 \quad \text{if } \sin^2\left(\frac{\theta}{2}\right)=1 \longrightarrow \frac{a\Delta t}{\Delta x^2}\le\frac{1}{2}$$

The above analysis method is called von Neumann method: concept is clear , and its implementation is easy !

Question 1: how does the error propagate (传递） as time step proceeds? (Marching)

Question 2: How to mathematically describe the error? Or what is the expression of error?

Question 3: How to analyze the stability of the initial problem?

7.3 Conservation of Discretized Equations

7.3.1 Definition and analyzing model

7.3.2 Direct summation method

7.3.3 Conditions for guaranteeing conservation of discretized equations

7.3 Conservation （守恒性） of Discretized Equations

7.3.1 Definition and analyzing model

1. Definition

If the summation of a certain number of discretized equations over a finite volume （有限大小的体积） satisfies conservation requirement , these discretized equations are said to have conservation **(离散方程具有守恒性).**

2. Analyzing model （用于分析的模型）

It is easy to show that central difference (CD) of diffusion term possesses conservation. Thus, discussion is only performed for the equation which only has transient term and convective term (Pure advection equation, **纯对流方程**).

Advection equation

$$\begin{cases} \dfrac{\partial \phi}{\partial t} + \dfrac{\partial (u\phi)}{\partial x} = 0 \\ \dfrac{\partial \phi}{\partial t} + u\dfrac{\partial \phi}{\partial x} = 0 \end{cases}$$

(Conservative，守恒型的)

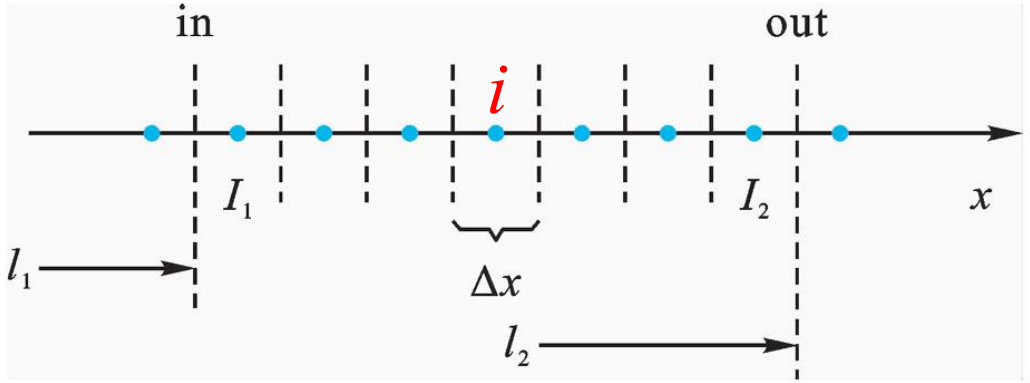
(Non-conservative)

7.3.2 Direct summation method (直接求和法)

Summing up **FTCS** scheme of advection eq. of conservative form over the region of $[l_1, l_2]$:

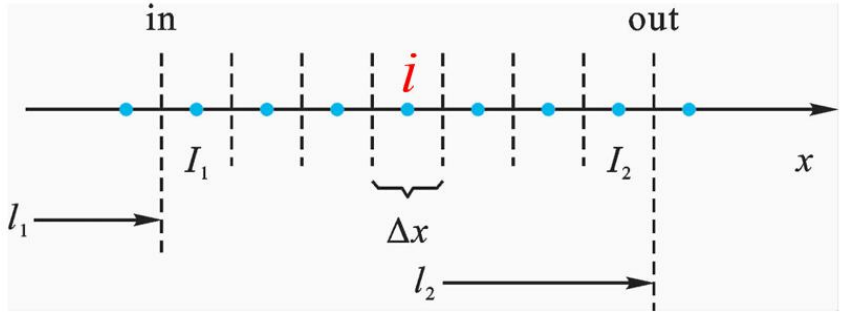
$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = -\frac{u_{i+1}\phi_{i+1} - u_{i-1}\phi_{i-1}}{2\Delta x}$$

Time level of the spatial terms are not shown



$$\sum_{I_1}^{I_2}\frac{\phi_i^{n+1}-\phi_i^n}{\Delta t}=-\sum_{I_1}^{I_2}\frac{u_{i+1}\phi_{i+1}-u_{i-1}\phi_{i-1}}{2\Delta x}=-\sum_{I_1}^{I_2}\frac{(u\phi)_{i+1}-(u\phi)_{i-1}}{2\Delta x}$$

$$\sum_{I_1}^{I_2}(\phi_i^{n+1}-\phi_i^n)\Delta x=-\Delta t\sum_{I_1}^{I_2}\frac{(u\phi)_{i+1}-(u\phi)_{i-1}}{2}$$



Increment(增值) of ϕ within Δt and $[l_1, l_2]$

Is it equal to the net amount of ϕ entering the space region by convection within the same time period?

Analyzing should be made for the right-hand terms of the equation to see whether this is true:

$$-\Delta t\sum_{I_1}^{I_2}\frac{(u\phi)_{i+1}-(u\phi)_{i-1}}{2}=\frac{\Delta t}{2}\sum_{I_1}^{I_2}[(u\phi)_{i-1}-(u\phi)_{i+1}]$$

For the term $\sum_{I_1}^{I_2}[(u\phi)_{i-1}-(u\phi)_{i+1}]$ directly summing up: for the left end, we have:

$i=I_1$ $(u\phi)_{I_1-1}$ $-(u\phi)_{I_1+1}$

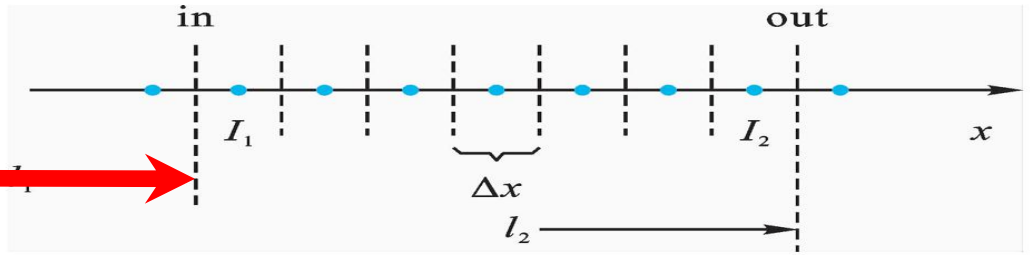
$i=I_1+1$ $(u\phi)_{I_1}$ $(u\phi)_{I_1+1}$ $-(u\phi)_{I_1+2}$

$i=I_1+2$ $(u\phi)_{I_1+2}$ $-(u\phi)_{I_1+3}$

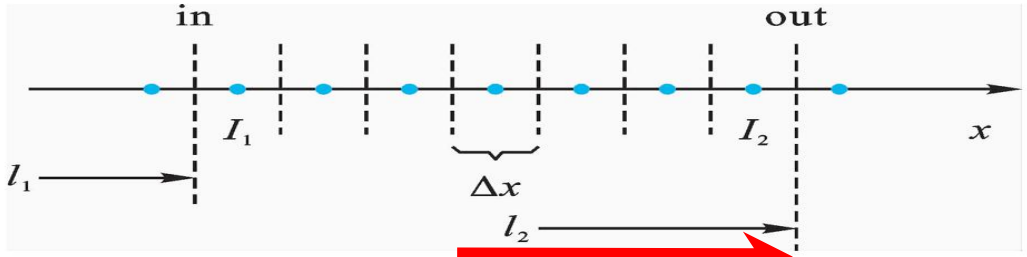
$i=I_1+3$ $(u\phi)_{I_1+3}$ $-(u\phi)_{I_1+4}$

$i=I_1+4$

$$(u\phi)_{I_1-1}+(u\phi)_{I_1}$$



For the right end: $\sum_{I_1}^{I_2}[(u\phi)_{i-1}-(u\phi)_{i+1}]$



$i=I_2-3$ $(u\phi)_{I_2-4}$ $-(u\phi)_{I_2-2}$

$i=I_2-2$ $(u\phi)_{I_2-3}$ $-(u\phi)_{I_2-1}$

$i=I_2-1$ $(u\phi)_{I_2-2}$ $-(u\phi)_{I_2}$

$i=I_2$ $(u\phi)_{I_2-1}$ $-(u\phi)_{I_2+1}$

$-[(u\phi)_{I_2}+(u\phi)_{I_2+1}]$

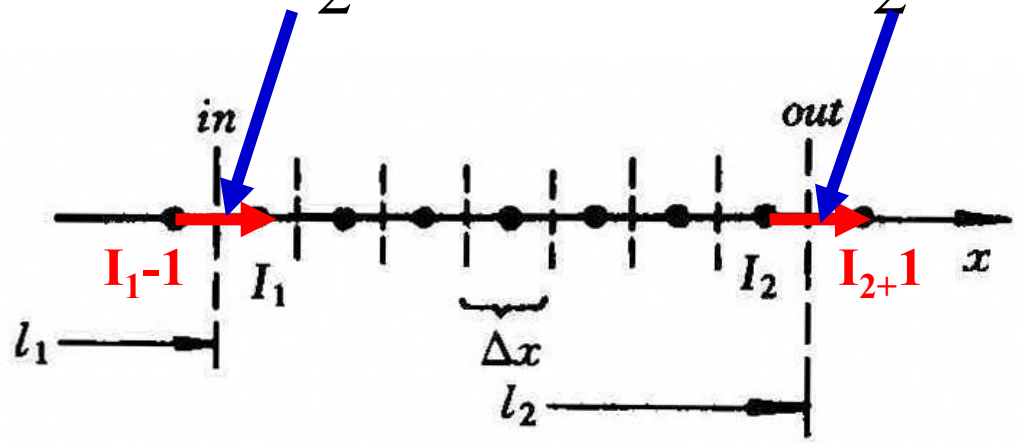
Then: $\frac{\Delta t}{2}\sum_{I_1}^{I_2}[(u\phi)_{i-1}-(u\phi)_{i+1}]$

$=\frac{\Delta t}{2}\{[(u\phi)_{I_1-1}+(u\phi)_{I_1}]-[(u\phi)_{I_2}+(u\phi)_{I_2+1}]\}$

Left end of domain　　Right end of domain

Further: $$\frac{\Delta t}{2}\{[(u\phi)_{I_1-1}+(u\phi)_{I_1}]-[(u\phi)_{I2}+(u\phi)_{I_2+1}]\}=$$

$$\Delta t\{[\frac{(u\phi)_{I_1-1}+(u\phi)_{I_1}}{2}]-[\frac{(u\phi)_{I2}+(u\phi)_{I_2+1}}{2}]\}$$ **CD-uniform grid**



$$=\Delta t(\phi_{flowin}-\phi_{flowout})$$

The interpolation scheme of the interface flow rate is in consistence with the central difference scheme of the convective term.

Thus, the central difference discretization of the convective term possesses conservative feature.

7.3.3 Conditions for guaranteeing conservation

1.Governing equation should be conservative

For non-conservative form: $$\frac{\partial \phi}{\partial t}+u\frac{\partial \phi}{\partial x}=0$$

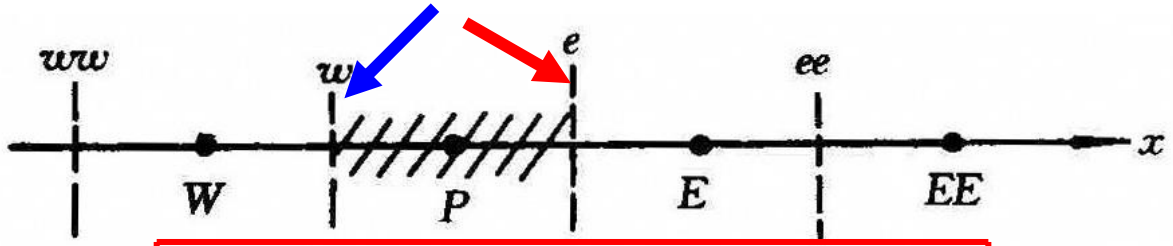
Its FTCS scheme is $$\frac{\phi_i^{n+1}-\phi_i^{n}}{\Delta t}=-u_i\frac{\phi_{i+1}-\phi_{i-1}}{2\Delta x}$$

By direct summation, the above results do not possess conservation because of no cancellation (**抵消**) can be made for the product terms. Only when u and ϕ have the same subscript , the cancellation of inner terms can be done.

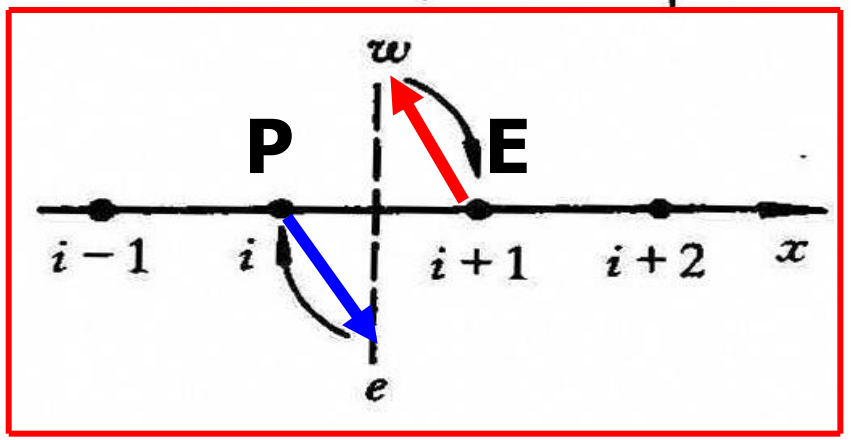
2. Variable and its 1st derivative are continuous at interface

Meaning of "Continuous"

Different interfaces viewed from point P



The same interface viewed from two points P and E



By "Continuous" we mean:

$$(\phi_e)_P = (\phi_w)_E;$$

$$[(\frac{\delta\phi}{\delta x})_e]_P = [(\frac{\delta\phi}{\delta x})_w]_E$$

The piecewise linear profile （分段线性） can meet this condition.

Contents

7.1 Consistency, Convergence and Stability of Discretized Equations

7.2 von Neumann Method for Analysing Stability of Initial Problems

7.3 Conservation of Discretized Equations

7.4 Classification and V&V of Numerical Solution Errors

7.5 Analysis of Truncation Error of Discretization Schemes

7.6 Analysis of Grid Convergence Index and Incomplete Iteration Error

7.4 Classification and V&V of Numerical Solutions

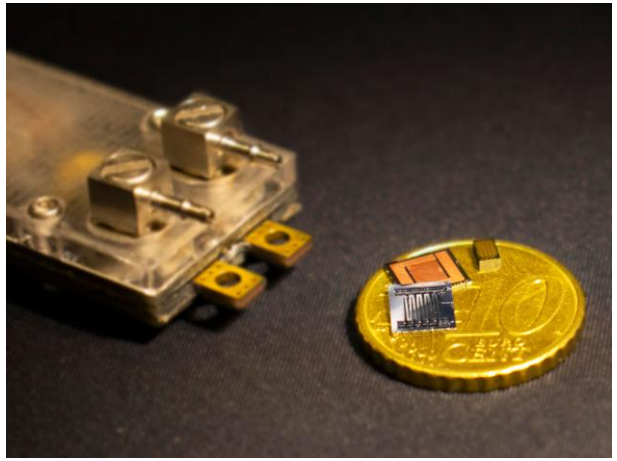
7.4.1 Classification of Numerical errors

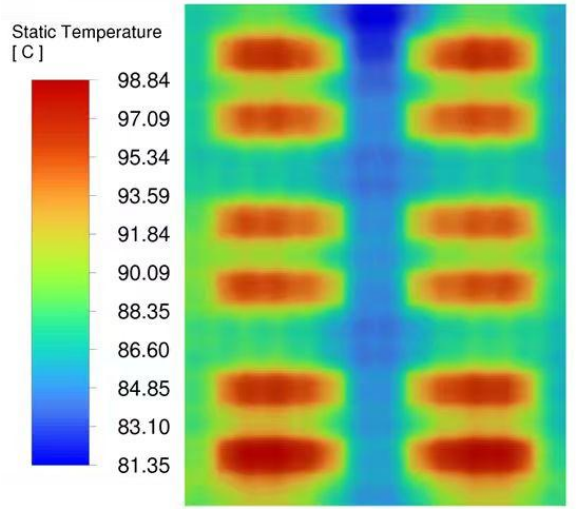
7.4.2 Three Types of Data for V&V of Numerical Solutions

7.4.3 Exact Solutions for Verification of Numerical Solutions

7.4.5 Benchmark Solutions for Verification of Numerical Solutions

7.4.1 Classification of Numerical errors





Practical physical process
真实物理问题

Final temperature distribution from simulation
最终数值仿真的结果

There are several kinds of errors during this process

7.4.1 Classification of Numerical solution errors

A. Practical physical process
真实物理问题

↓

B. Mathematical model
数学物理模型

↓

C. Discretization of governing equations and boundary conditions

↓

D. Analysis of numerical solutions

Model Error

- **(1) Correct governing equations?**
- **(2) Correct boundary conditions?**
- **(3) Ccorrect thermophysical properties?**
- **(4) Correct source terms?**

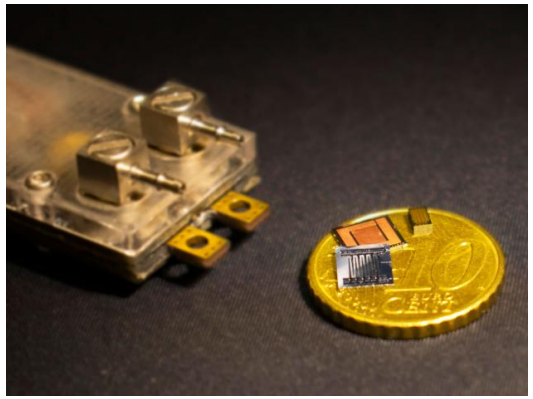
Discretization Error

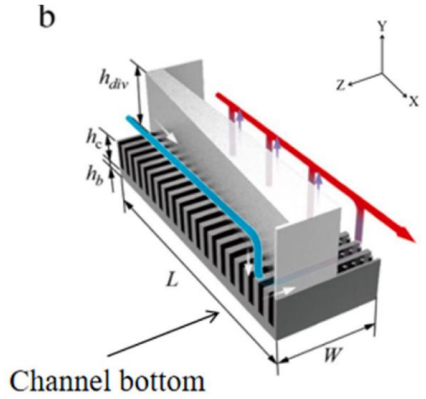
- **(5) Truncation error**
  (Depending on the Discretization Scheme)
- **(6) Numerical treatment of boundary conditions**
- **(7) Mesh quality**
- **(8) Temporal discretization of time-dependent terms**

Computational Error

- **(9) Round-off error**
- **(10) Error due to incomplete iteration**

Building a 3D numerical model for chip cooling





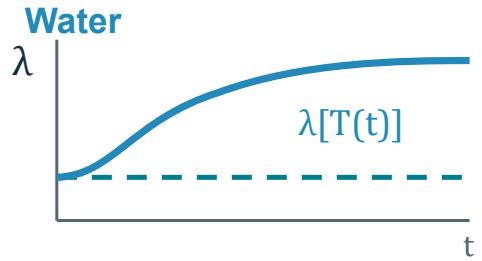
(1) Correct governing equations?

(2) Correct boundary conditions?

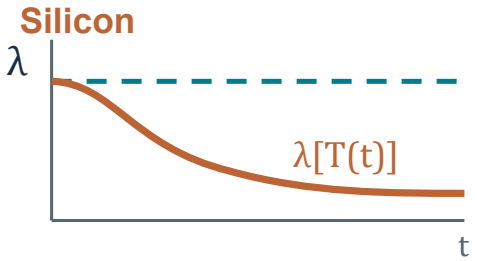
(3) Ccorrect thermophysical properties?

(4) Correct source terms?

Assumption: steady, incompressible, laminar flow



$\lambda = 0.6028\ W/(m\cdot K)$



$\lambda_s = 148\ W/(m\cdot K)$

The thermal properties change with temperature.

Source terms: no body-force or dissipation term.

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0$$

$$\rho\left(\frac{\partial u}{\partial \tau}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)=F_x-\frac{\partial p}{\partial x}+\eta\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)$$

$$\rho\left(\frac{\partial v}{\partial \tau}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right)=F_y-\frac{\partial p}{\partial y}+\eta\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)$$

$$\rho c_p\left(\frac{\partial t}{\partial \tau}+u\frac{\partial t}{\partial x}+v\frac{\partial t}{\partial y}\right)=\lambda\left(\frac{\partial^2 t}{\partial x^2}+\frac{\partial^2 t}{\partial y^2}\right)$$

2) Basic requirements for writing a numerical simulation paper (10 steps, a little different from the textbook, 撰写一个数值仿真论文的基本要求)

(1) **Describe the physical processes** to be numerically simulated as comprehensively and accurately as possible. (尽可能全面准确描述所要研究的物理过程）

(2) **State the assumptions** adopted when **formulating the governing equations** for the investigated physical process（给出构建方程时做的假设;） and **present the governing equations (给出控制方程）**.

(3) Clearly **specify the order of truncation error （截差） for each term of the discretized governing equations**, especially for the convection term.

2) Basic requirements for writing a numerical simulation paper (10 steps, a little different from the textbook)

(4) The **discretization scheme applied to internal nodes** within the computational domain shall have **at least second-order** spatial truncation error (内部节点至少二阶精度). The hybrid scheme, power-law scheme and exponential scheme are all regarded as first-order schemes, unless it can be demonstrated that the inherent numerical diffusion of these schemes does not overwhelm the diffusion term of the simulated physical process.

(5) The **numerical implementation of boundary conditions and initial conditions** shall be clearly described (边界及初始条件的数值实施).

(6) Clearly present the **numerical method for solving the governing equations** （求解离散方程的数值方法，如SIMPLE）.

2) Basic requirements for writing a numerical simulation paper (10 steps, a little different from the textbook)

(7) **Grid independence** （网格独立性考核） of computational results shall be checked. For unsteady problems, **time-step independence** （时间独立性考核）shall also be examined.

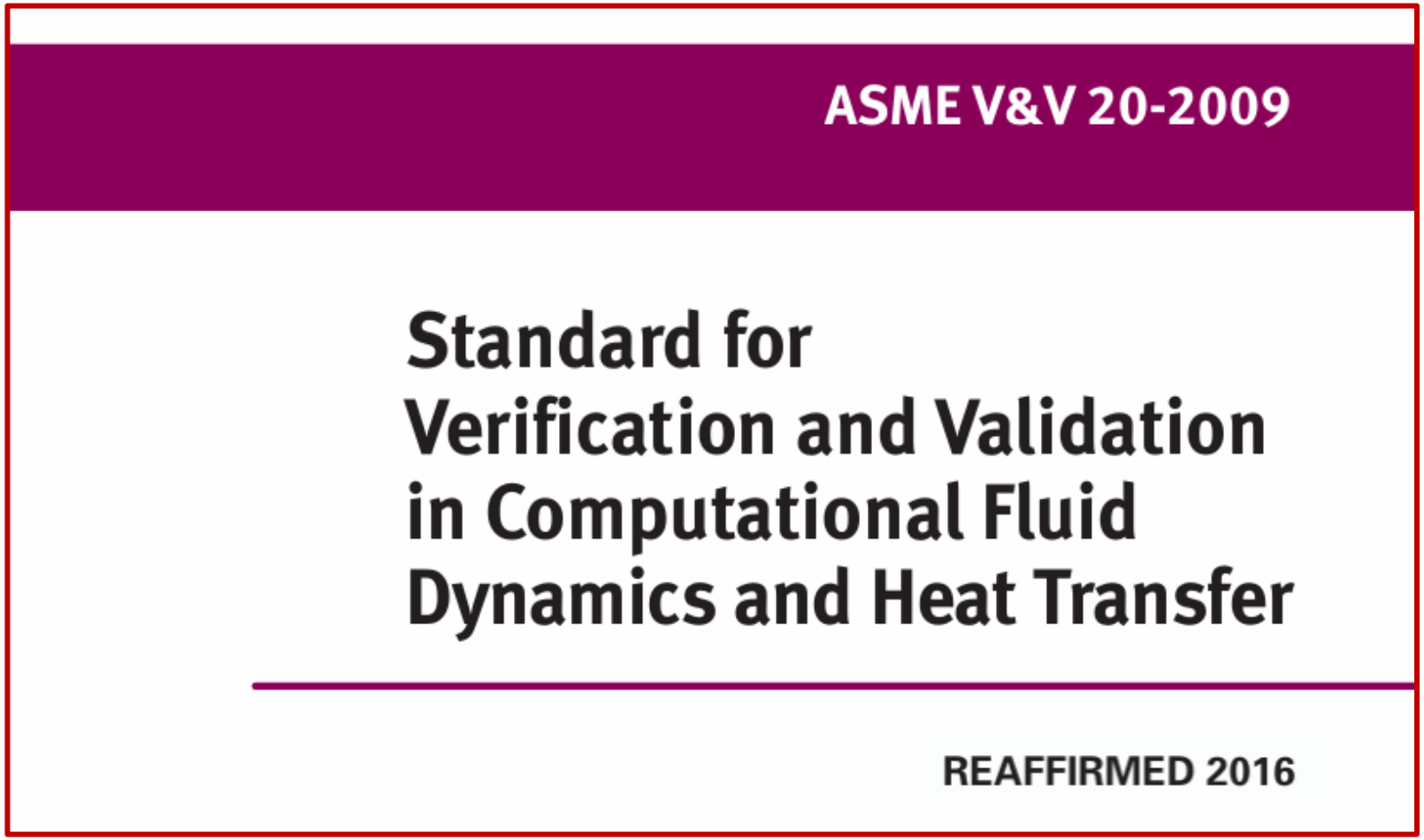
(8) **The convergence criterion (**收敛标准） for numerical iteration shall be clearly specified.

(9) **Verify the analytical solutions with well-established numerical benchmark solutions.**（用精确解或基准解验证并进一步推广到其他问题）

(10) **Validate the numerical solutions with reliable experimental data whose uncertainty is provided.**（与给定不确定度的实验结果比较）

7.4.2 Three types of data for validation and verification of numerical solutions

1) Validation and verification (V&V)



Validation: The equations to be solved accurately describe the physical processes. (求解的方程正确地描述了物理过程)

Verification: The equations is accurately solved. (方程被正确地求解了)

This Standard addresses verification and validation (V&V) in computational fluid dynamics (CFD) and computational heat transfer (CHT). The concern of V&V is to assess the accuracy of a computational simulation. The V&V procedures presented in this Standard can be applied to engineering and scientific modeling problems ranging in complexity from simple lumped masses, to 1-D steady laminar flows, to 3-D unsteady turbulent chemically reacting flows. In V&V, the ultimate goal of engineering and scientific interest is validation, which is defined as the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model. However, validation must be preceded by code verification and solution verification. Code verification establishes that the code accurately solves the mathematical model incorporated in the code, i.e. that the code is free of mistakes for the simulations of interest. Solution verification estimates the numerical accuracy of a particular calculation.

The equations are correct, the numerical schemess to solve the equations are with low error, and the code wrote is correct. (方程正确，求解方程的方法正确，编写的程序正确)

2) Three levels understanding of V&V

(1) Verify the accuracy of the numerical schemes (software or the computational program, 软件或者代码程序).

- The scheme may be not accurate. (数值方法是否正确？)
- The scheme is accurate, but the code for implementing the scheme maybe not right. (数值方法正确，程序是否正确？)
- In this level, usually relatively simple problems with analytical solution or benchmark solution will be simulated. (具有解析解或基准解的相对简单问题)

(2) Validate the numerical models using problems (with experimental results) similar to the original practical problems studied.

- The problem is relatively complex.

2) Three levels understanding of V&V

(3) Validate the numerical model with the available experimental results of the original practical problem.

Comments:

(1) The first and second level of V&V of the numerical model is indirect V&V.

(2) The third level of V&V is direct V&V, which is however, usually expensive as conducting experiments for the original problem is usually expensive and time consuming.

（3）Usually, only limited reliable experimental results （可靠实验数据有限）are available.

ASME V&V standard

CONTENTS

Foreword vi
Committee Roster viii
Correspondence With the V&V 20 Committee ix

Section 1 Introduction to Validation Methodology 1
1-1 General 1
1-2 Objective and Scope 1
1-3 Errors and Uncertainties 1
1-4 Example for Validation Nomenclature and Approach 2
1-5 Validation Approach 3
1-6 Overview of Subsequent Sections 5
1-7 References 5

Section 2 Code Verification and Solution Verification 7
2-1 General 7
2-2 Introduction 7
2-3 Code Verification 7
2-4 Solution Verification 11
2-5 Special Considerations 16
2-6 Final Comment 17
2-7 References 17

Section 3 Effect of Input Parameter Uncertainty on Simulation Uncertainty 19
3-1 Introduction 19
3-2 Sensitivity Coefficient (Local) Method for Parameter Uncertainty Propagation 19
3-3 Sampling (Global) Methods for Parameter Uncertainty Propagation 23
3-4 Importance Factors 25
3-5 Special Considerations 25
3-6 Final Comment on Parameter Uncertainty 26
3-7 References 26

Section 4 Uncertainty of an Experimental Result 27
4-1 Overview 27
4-2 Experimental Uncertainty Analysis 27
4-3 Uncertainty of Validation Experiment 28
4-4 Summary 28
4-5 References 28

Section 5 Evaluation of Validation Uncertainty 30
5-1 Overview 30
5-2 Estimating u_{val} When the Experimental Value, D, of the Validation Variable is Directly Measured (Case 1) 30
5-3 Estimating u_{val} When the Experimental Value, D, of the Validation Variable is Determined From a Data Reduction Equation (Cases 2 and 3) 31
5-4 Estimating u_{val} When the Experimental Value, D, of the Validation Variable is Determined From a Data Reduction Equation That Itself Is a Model (Case 4) 36
5-5 Assumptions and Issues 37
5-6 References 39

Section 6 Interpretation of Validation Results 40
6-1 Introduction 40
6-2 Interpretation of Validation Results Using E and u_{val} With No Assumptions Made About Error Distributions 40
6-3 Interpretation of Validation Results Using E and u_{val} With Assumptions Made About Error Distributions 40
6-4 References 41

Section 7 Examples 42
7-1 Overview 42
7-2 Code Verification Example 42
7-3 Validation Example 48
7-4 References 65

Figures
1-4-1 Schematic of Finned-Tube Assembly for Heat Transfer Example 2
1-5-1 Schematic Showing Nomenclature for Validation Approach 3
1-5-2 Overview of the Validation Process With Sources of Error in Ovals 4
2-4-1 Sample Uncertainty Analysis: Explosive Detonation in a Fluid Filled Box 15
3-2-1 Relative Error in Finite Difference Computation of $k\partial T/\partial k$ Using a Backwards Difference 21
3-2-2 Estimated Uncertainty in Model Temperature Due to Uncertainty in q, k, and ρc_p 22
3-3-1 Representative Probability Distribution Function for Thermal Conductivity 23
3-3-2 Standard Deviation in Temperature at $z/L = 0$ and 1 for Constant Heat Flux Example Using 10 LHS Runs and Mean Value Method (With $u_y/X = 0.05$) 25
5-1-1 Schematic for Combustion Gas Flow Through a Duct With Wall Heat Flux Being the Validation Variable (Case 4) 30
5-2-1 Sensitivity Coefficient Propagation Approach for Estimating u_{val} When the Validation Variable (T_o) Is Directly Measured (Case 1) 32
5-2-2 Monte Carlo Approach for Estimating u_{val} When the Validation Variable (T_o) Is Directly Measured (Case 1) 32
5-3-1 Sensitivity Coefficient Propagation Approach for Estimating u_{val} When the Validation Variable Is Defined by a Data Reduction Equation That Combines Variables Measured in the Experiment (Case 2) 34
5-3-2 Monte Carlo Approach for Estimating u_{val} When the Validation Variable Is Defined by a Data Reduction Equation That Combines Variables Measured in the Experiment (Case 2) 34
5-3-3 Sensitivity Coefficient Propagation Approach for Estimating u_{val} When the Validation Variable Is Defined by a Data Reduction Equation That Combines Variables Measured in the Experiment and Two Measured Variables Share an Identical Error Source (Case 3) 35
5-3-4 Monte Carlo Propagation Approach for Estimating u_{val} When the Validation Variable Is Defined by a Data Reduction Equation That Combines Variables Measured in the Experiment and Two Measured Variables Share an Identical Error Source (Case 3) 36
5-4-1 Sensitivity Coefficient Propagation Approach for Estimating u_{val} When the Validation Variable Is Defined by a Data Reduction Equation That Itself Is a Model (Case 4) 37
5-4-2 Monte Carlo Propagation Approach for Estimating u_{val} When the Validation Variable Is Defined by a Data Reduction Equation That Itself Is a Model (Case 4) 38
7-2-1 Problem Domain With (x, y) Coordinates Shown for Domain Corners 42
7-2-2 Finite Element Meshes Used in the Code Verification Refinement Study 44
7-2-3 Error as a Function of Characteristic Mesh Size 47
7-3-1 Schematic of Fin-Tube Heat Exchanger Assembly 48
7-3-2 Experimental Total Heat Transfer Rate and Its Standard Uncertainty, u_D 51
7-3-3 Heat Transfer Model for the Fin-Tube Assembly 52
7-3-4 Mesh Refinement Study for Solution Verification 55
7-3-5 Simulation Values of Total Heat Transfer Rate and Its Uncertainty, u_{input} 57
7-3-6 LHS Samples of Simulated and Experimental Values of Total Heat Transfer Rate 60
7-3-7 Interval for δ_{model} ($E \pm 2u_{val}$) Assuming a Gaussian Distribution for the Errors and 95% Probability 61
7-3-8 Interval for δ_{model} ($E \pm 2u_{val}$) Assuming a Gaussian Distribution for the Errors and 95% Probability for the Model With Contact Conductance at the Fin/Tube Interface 65

Tables
2-4-1 Sample Uncertainty Analysis: Backward Facing Step 15
2-4-2 Sample Uncertainty Analysis: Explosive Detonation 16
3-3-1 Matrix Representation of Number of LHS Samples (n_{LHS}) and Number of Parameters (n_p) 24
3-3-2 LHS Samples for the Three Parameters q, k, and C 24
7-2-1 Parameter Values Used for the Code Verification Example 45
7-2-2 Code Verification Results 45
7-2-3 Error (E_h) in the Code Simulation During Mesh Refinement 46
7-2-4 Observed Order of Convergence (p^{obs}) From Mesh Refinement 47
7-3-1 Details of the Fin-Tube Assembly and Flow Conditions 49
7-3-2 Measured Flow Conditions and Calculated Total Heat Transfer Rate 50
7-3-3 Estimates of the Experimental Measurement Standard Uncertainties 50
7-3-4 Sensitivity Coefficients for Average Conditions 50
7-3-5 Experimental Values of Total Heat Transfer Rate and Its Standard Uncertainties 51
7-3-6 Simulation Model Input Parameters and Standard Uncertainties 53
7-3-7 Simulation Values of Total Heat Transfer Rate 54
7-3-8 Solution Verification Results for Total Heat Transfer Rate 54
7-3-9 Measures of the Numerical Error and Numerical Uncertainty for Total Heat Transfer Rate 55
7-3-10 Partial Derivatives of the Total Heat Transfer Rate for the Simulation Model With Respect to Uncertain Model Inputs for the Average of Measured Experimental Conditions and Standard Uncertainty for the Inputs 56
7-3-11 Simulation Values of Total Heat Transfer Rate and Its Standard Uncertainty From Input Parameter Uncertainty 56
7-3-12 Parameters Included in Evaluating u_{val}, Parameter Standard Uncertainty Estimates, and Parameter Sensitivity Coefficients 58
7-3-13 Experimental and Simulation Values of Total Heat Transfer Rate and Associated Standard Uncertainties 58
7-3-14 Parameter Standard Uncertainty and Example Latin Hypercube Samples 59
7-3-15 LHS Samples for the Simulated and Experimental Values of the Total Heat Transfer Rate 60
7-3-16 Comparison of Nominal Values and Standard Uncertainties Computed With the Propagation and LHS Approaches 60
7-3-17 Simulation Values of the Total Heat Transfer Rate for the Model With Contact Conductance 62
7-3-18 Simulation Values of the Total Heat Transfer Rate and the Standard Uncertainty for the Model With Contact Conductance 63
7-3-19 Solution Verification Results for Total Heat Transfer Rate for the Model With Contact Conductance 63
7-3-20 Measures of the Numerical Error and Numerical Uncertainty for Total Heat Transfer Rate for the Model With Contact Conductance 63
7-3-21 Partial Derivatives of the Total Heat Transfer Rate for the Simulation Model With Respect to Uncertainty Model Inputs for Model With Contact Conductance for the Average Measured Conditions 64
7-3-22 Parameters Included in Evaluating u_{val}, Parameter Standard Uncertainty Estimates, and Parameter Sensitivity Coefficients for the Model With Contact Conductance 64
7-3-23 Experimental and Simulation Values of Total Heat Transfer Rate and Associated Uncertainties 65

Mandatory Appendices
I Detailed Development of Simulation Equations for Example Problem 67
II Nomenclature 70

Nonmandatory Appendices
A Method of Manufactured Solutions for the Sample Problem 72
B Importance Factors 78
C Additional Topics 82

ASME V&V standard

Section 1 Introduction to Validation Methodology

1-1 General
1-2 Objective and Scope
1-3 Errors and Uncertainties
1-4 Example for Validation Nomenclature and Approach
1-5 Validation Approach
1-6 Overview of Subsequent Sections
1-7 References

Section 2 Code Verification and Solution Verification

2-1 General
2-2 Introduction
2-3 Code Verification
2-4 Solution Verification
2-5 Special Considerations
2-6 Final Comment
2-7 References

ASME V&V standard

Section 3 Effect of Input Parameter Uncertainty on Simulation Uncertainty

3-1 Introduction
3-2 Sensitivity Coefficient (Local) Method for Parameter Uncertainty Propagation
3-3 Sampling (Global) Methods for Parameter Uncertainty Propagation
3-4 Importance Factors
3-5 Special Considerations
3-6 Final Comment on Parameter Uncertainty
3-7 References

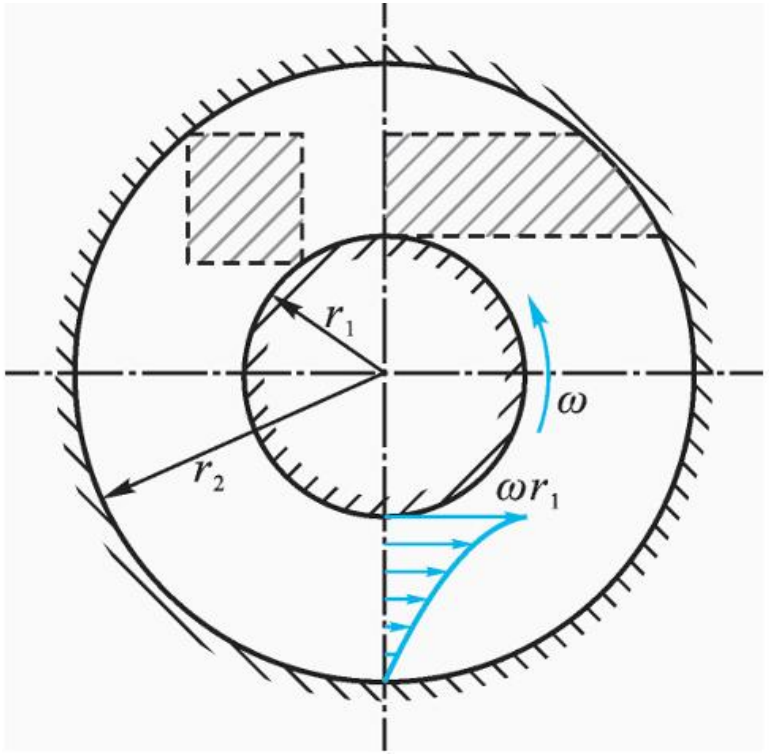
Section 4 Uncertainty of an Experimental Result

4-1 Overview
4-2 Experimental Uncertainty Analysis
4-3 Uncertainty of Validation Experiment
4-4 Summary
4-5 Reference

7.4.3 Exact Solutions for **Verification** of Numerical Solutions

1) Exact solutions of NS equations with given boundary conditions

Only a few exact solutions exist. Fluid flow in annual space is an example.

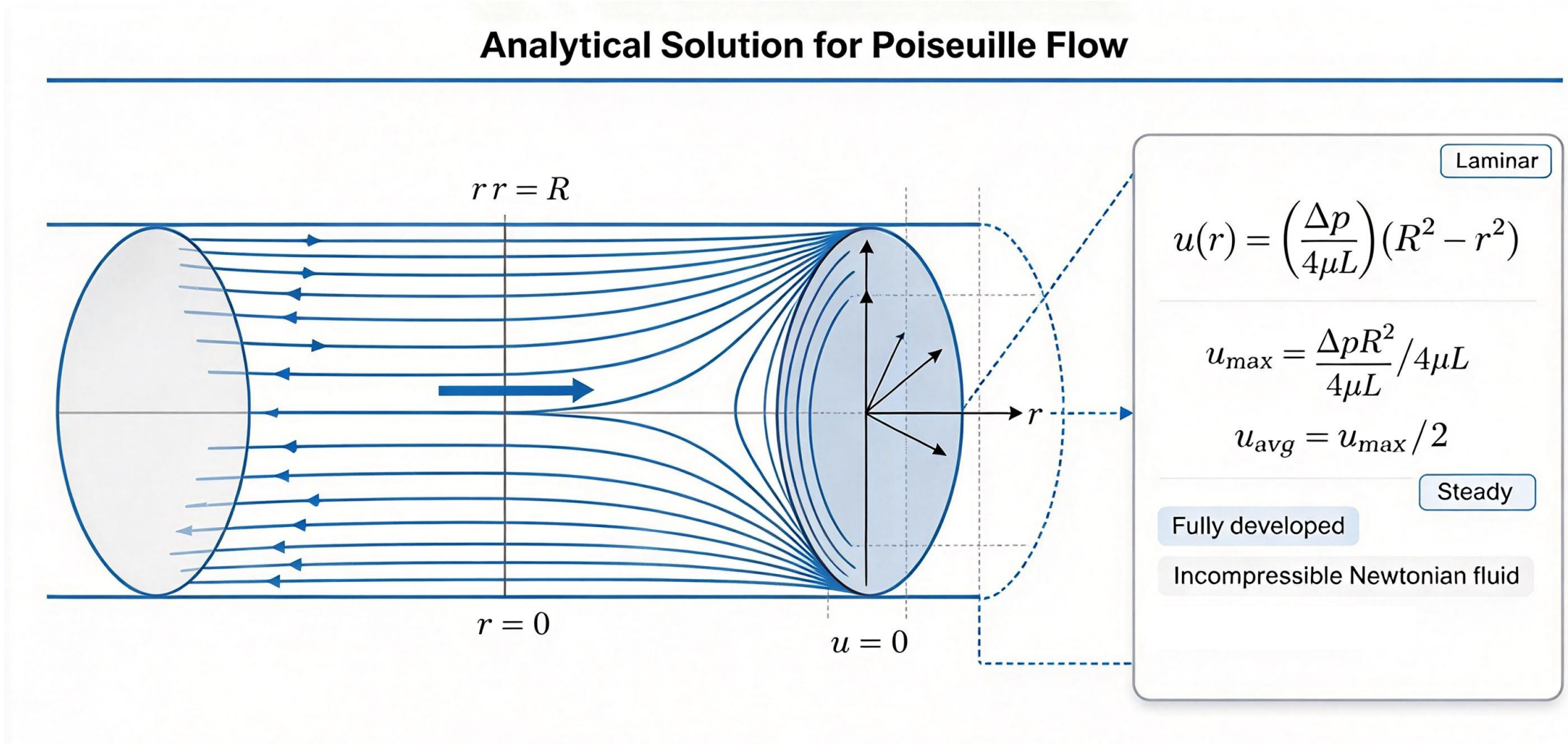


- **Annual space with two circles.**
- **The outer circle is stable while the inner circle is rotating.**
- **The exact solution is as follows:**

$$\frac{u}{u_1}=\frac{r_1/r_2}{1-(r_1/r_2)^2}\frac{1-(r/r_2)^2}{r/r_2}$$

The whole annual space or part of it can be simulated and to verify the numerical solution.

Poiseuille flow in a circular pipe with exact solution



7.4.3 Exact Solutions for Verification of Numerical Solutions

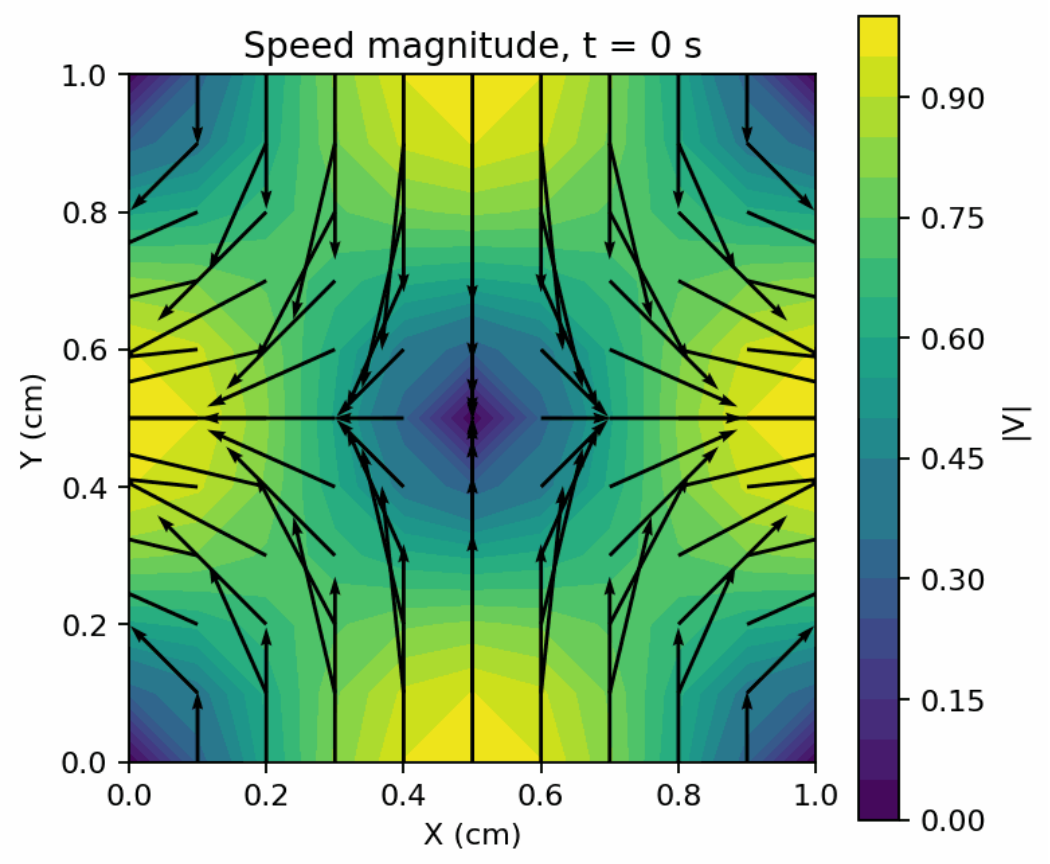
2) Exact solutions of NS equations without given boundary conditions

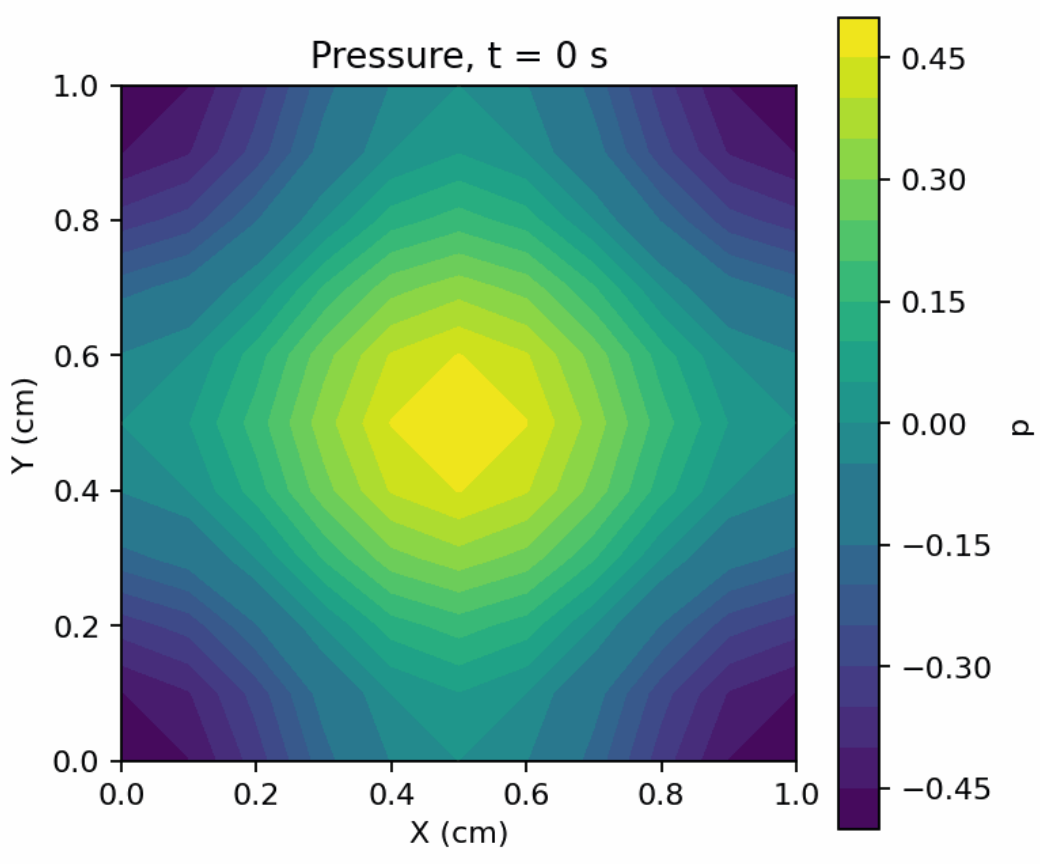
2D Taylor-Green Vortex, a famous exact solution of NS equation

$$\frac{\partial \boldsymbol{u}}{\partial t}+Re\,\boldsymbol{u}\cdot\nabla\boldsymbol{u}=-Re\,\nabla p+\nabla^2\boldsymbol{u};\quad \nabla\cdot\boldsymbol{u}=0$$

$$\begin{cases} u=-\cos(\pi x)\sin(\pi y)\,\mathrm{e}^{-2\pi^2\nu t} \\ v=\sin(\pi x)\cos(\pi y)\,\mathrm{e}^{-2\pi^2\nu t} \\ p=-\dfrac{\cos(2\pi x)+\cos(2\pi y)}{4}\mathrm{e}^{-4\pi^2\nu t} \end{cases}$$

2D Taylor-Green Vortex, a famous exact solution of NS equation





Comments:

- It is merely a solution to the governing equations and does not satisfy engineering boundary conditions such as solid walls or inlet/outlet boundaries（没有任何有工程意义的边界如固体边界或进出口边界）.
- It is thus categorized as the second type of exact solution. Its key application is to verify the accuracy of numerical schemes , the algorithm and/or the code (用于验证数值格式、算法及程序）.
- A computational domain is selected for numerical simulation, with the velocity and pressure at the boundaries calculated using the exact solution. The distribution of velocity and pressure in the inner domain is compared with the exact solution to verify the numerical models.

7.4.3 Exact Solutions for Verification of Numerical Solutions

3) Manually constructed exact solutions (人工构造精确解)

The method assumes a smooth analytical field in advance, then substitutes it into the governing equations to back-calculate the required source terms, so that the assumed field satisfies the governing equations with source terms for code **verification**.

For example, for a 1D unsteady heat transfer problem

$$\frac{\partial T}{\partial t}=a\,\frac{\partial^2 T}{\partial x^2};\quad 0\leqslant x\leqslant L$$

Assume the following unsteady temperature distribution

$$T=T_0\,\mathrm{e}^{t/t_0}\sin(\pi x/L)$$

For example, for a 1D unsteady heat transfer problem

$$\frac{\partial T}{\partial t}=a\frac{\partial^2 T}{\partial x^2};\quad 0\leqslant x\leqslant L$$

Substituting the following unsteady temperature distribution

$$T=T_0\,\mathrm{e}^{t/t_0}\sin(\pi x/L)$$

Into the original equation

$$\frac{\partial T}{\partial t}-a\frac{\partial^2 T}{\partial x^2}=\left[\frac{1}{t_0}+a\left(\frac{\pi}{L}\right)^2\right]T_0\sin(\pi x/L)\,\mathrm{e}^{t/t_0}$$

Therefore, the following 1D unsteady equation with source terms is obtained

$$\begin{cases}\dfrac{\partial T}{\partial t}-a\dfrac{\partial^2 T}{\partial x^2}=S & 0\leqslant x\leqslant L\\[2ex] S=\left[\dfrac{1}{t_0}+a\left(\dfrac{\pi}{L}\right)^2\right]T_0\sin(\pi x/L)\,\mathrm{e}^{t/t_0}\end{cases}$$

7.4.4 Benchmark solutions for verifying numerical solutions

1. Definition

Benchmark solutions (基准解): Well-recognized numerical solutions with high accuracy.

There are lots of classical fluid flow and heat transfer processes with benchmark solutions:

1) **2D Lid-driven flow in a square cavity (二维顶盖驱动流)**
2) **3D Lid-driven flow in a square cavity (三维顶盖驱动流)**
3) **Natural convection in a square cavity (二维方腔中自然对流)**
4) **Two-dimensional annular lid-driven flow (二维环腔中驱动流)**
5) **Natural convection around a horizontal circular cylinder (二维环腔中自然对流)**
6) **….**

Table 7–4 Summary of Benchmark Problems for Flow and Heat Transfer

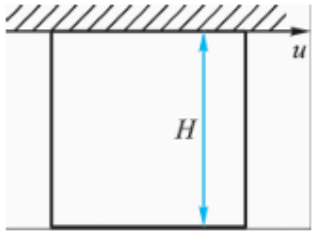
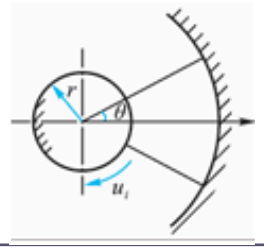
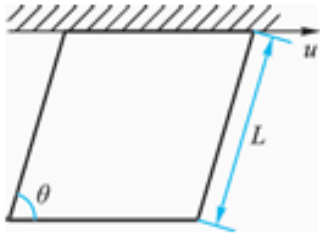
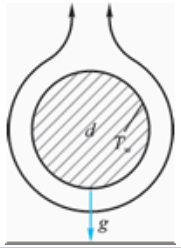
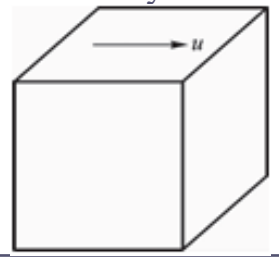
No.	Physical problem	Parameter range	Numerical method	References
1	Two-dimensional lid-driven flow in a square cavity 	$Re = uH/\nu$ $Re = 10^2 \sim 10^4$	Vorticity–streamfunction method and multigrid; finest grid: 257 × 257. Flow bifurcation occurs =at Re 10,000.	[21] [22]
2	Two-dimensional natural convection in a square cavity 	$Pr = 0.7$ $Re = 10^3 \sim 10^{10}$	Vorticity–streamfunction method; primitive-variable method; k–ε model; wall-function method.	[23] [24]
3	Two-dimensional annular lid-driven flow 	$Re = u_i r/\nu$ $Re = 60 \sim 350$ $2\theta = 1$ rad	Vorticity–streamfunction method; multigrid; 128 × 128 grid; primitive-variable method; adaptive grid.	[25] [26]

Table 7–4(continued) Summary of Benchmark Problems for Flow and Heat Transfer

No.	Physical problem	Parameter range	Numerical method	References
4	Two-dimensional lid-driven flow in an inclined rectangular cavity 	$Re = 100, 1000$ $\theta = 30^\circ, 45^\circ$ $Re = uL/v$	Body-fitted coordinate method; multigrid.	[27] [28]
5	Natural convection around a horizontal circular cylinder 	$Pr = 0.7$ $Ra = 10^3 \sim 10^5$	Outer boundary of the computational domain is 2000–20000d from the cylinder. Mean Nusselt number is accurate to three significant figures.	[29]
6	Three-dimensional lid-driven cavity flow 	$Re = 1000, 2000, 3200$	Primitive-variable method; least-squares finite-element method.	[30]

Contents

7.1 Consistency, Convergence and Stability of Discretized Equations

7.2 von Neumann Method for Analysing Stability of Initial Problems

7.3 Conservation of Discretized Equations

7.4 Classification and V&V of Numerical Solution Errors

7.5 Analysis of Truncation Error of Discretization Schemes

7.6 Analysis of Grid Convergence Index and Incomplete Iteration Error

7.5 Analysis of Truncation Error of Discretization Schemes and Discretization Error of Numerical Solutions

7.5.1 FDM and FVM methods for analyzing the truncation error

7.5.2 Richardson extrapolation for analysing truncation error of numerical solutions

7.4.3 Exact Solutions for Verification of Numerical Solutions

7.4.5 Benchmark Solutions for Verification of Numerical Solutions

7.5.1 FDM and FVM methods for analyzing the truncation error T.E.

1) FDM method (or Taylor expansion method)

$$\left(\frac{\partial \phi}{\partial x}\right)_i = \left(\frac{\partial \phi}{\partial x}\right)_{\text{model}} + (\text{TE})_{\text{FD}}$$

The first term on the right-hand side is the numerical scheme to discretize the first derivative term （一阶导数项）.

Let us expand ϕ_{i+1} and ϕ_{i-1} around ϕ_i

$$\phi_{i+1} = \phi_i + h\left(\frac{\partial \phi}{\partial x}\right)_i + \frac{h^2}{2}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_i + \frac{h^3}{6}\left(\frac{\partial^3 \phi}{\partial x^3}\right)_i + \frac{h^4}{24}\left(\frac{\partial^4 \phi}{\partial x^4}\right)_i + \frac{h^5}{120}\left(\frac{\partial^5 \phi}{\partial x^5}\right)_i + O(h^6)$$

$$\phi_{i-1} = \phi_i - h\left(\frac{\partial \phi}{\partial x}\right)_i + \frac{h^2}{2}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_i - \frac{h^3}{6}\left(\frac{\partial^3 \phi}{\partial x^3}\right)_i + \frac{h^4}{24}\left(\frac{\partial^4 \phi}{\partial x^4}\right)_i - \frac{h^5}{120}\left(\frac{\partial^5 \phi}{\partial x^5}\right)_i + O(h^6)$$

The following formula is obtained, which is the central difference scheme

$$\left(\frac{\partial\phi}{\partial x}\right)_{\text{model}} = \frac{\phi_{i+1}-\phi_{i-1}}{2h} = \left(\frac{\partial\phi}{\partial x}\right)_i + \frac{h^2}{6}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i + \frac{h^4}{120}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i + O(h^6)$$

The T.E. is as follows:

$$(\text{TE})_{CD} = -\frac{1}{6}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i h^2 - \frac{1}{120}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i h^4 + O(h^6)$$

2) The FVM method

$$\frac{1}{h}\int_w^e \frac{\partial\phi}{\partial x}\,dx = \frac{\phi_e-\phi_w}{h} = \frac{\phi_e^{\text{model}}-\phi_w^{\text{model}}}{h} + (\text{TE})_{\text{FV}}$$

Let us expand ϕ_{i+1} and ϕ_i around ϕ_e

$$\phi_{i+1} = \phi_e + \frac{h}{2}\left(\frac{\partial \phi}{\partial x}\right)_e + \frac{h^2}{8}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_e + \frac{h^3}{48}\left(\frac{\partial^3 \phi}{\partial x^3}\right)_e + \frac{h^4}{384}\left(\frac{\partial^4 \phi}{\partial x^4}\right)_e + O(h^5)$$

$$\phi_i = \phi_e - \frac{h}{2}\left(\frac{\partial \phi}{\partial x}\right)_e + \frac{h^2}{8}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_e - \frac{h^3}{48}\left(\frac{\partial^3 \phi}{\partial x^3}\right)_e + \frac{h^4}{384}\left(\frac{\partial^4 \phi}{\partial x^4}\right)_e + O(h^5)$$

Then

$$\phi_e^{\text{model}} = \frac{\phi_{i+1} + \phi_i}{2} = \phi_e + \frac{h^2}{8}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_e + \frac{h^4}{384}\left(\frac{\partial^4 \phi}{\partial x^4}\right)_e + O(h^6)$$

Similarly,

$$\phi_w^{\text{model}} = \frac{\phi_i + \phi_{i-1}}{2} = \phi_w + \frac{h^2}{8}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_w + \frac{h^4}{384}\left(\frac{\partial^4 \phi}{\partial x^4}\right)_w + O(h^6)$$

The TE for FVM is obtained

$$(\mathrm{TE})_{\mathrm{FV}} = \frac{(\phi_e - \phi_e^{\mathrm{model}}) - (\phi_w - \phi_w^{\mathrm{model}})}{h} = -\frac{h}{8}\left[\left(\frac{\partial^2\phi}{\partial x^2}\right)_e - \left(\frac{\partial^2\phi}{\partial x^2}\right)_w\right] - \frac{h^3}{384}\left[\left(\frac{\partial^4\phi}{\partial x^4}\right)_e - \left(\frac{\partial^4\phi}{\partial x^4}\right)_w\right] + O(h^5)$$

However, the 2-order and 4-order derivative at the interface in the formula is not known

Expanding the 2-order derivative at the interface around that at node *i*

$$\left(\frac{\partial^2\phi}{\partial x^2}\right)_e = \left(\frac{\partial^2\phi}{\partial x^2}\right)_i + \left(\frac{h}{2}\right)\left(\frac{\partial^3\phi}{\partial x^3}\right)_i + \frac{1}{2!}\left(\frac{h}{2}\right)^2\left(\frac{\partial^4\phi}{\partial x^4}\right)_i + \frac{1}{3!}\left(\frac{h}{2}\right)^3\left(\frac{\partial^5\phi}{\partial x^5}\right)_i + O(h^4)$$

$$= \left(\frac{\partial^2\phi}{\partial x^2}\right)_i + \frac{h}{2}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i + \frac{h^2}{8}\left(\frac{\partial^4\phi}{\partial x^4}\right)_i + \frac{h^3}{48}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i + O(h^4)$$

Similarly,

$$\left(\frac{\partial^2\phi}{\partial x^2}\right)_w=\left(\frac{\partial^2\phi}{\partial x^2}\right)_i-\frac{h}{2}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i+\frac{h^2}{8}\left(\frac{\partial^4\phi}{\partial x^4}\right)_i-\frac{h^3}{48}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i+O(h^4)$$

Thus, we have

$$\left(\frac{\partial^2\phi}{\partial x^2}\right)_e-\left(\frac{\partial^2\phi}{\partial x^2}\right)_w=h\left(\frac{\partial^3\phi}{\partial x^3}\right)_i+\frac{h^3}{24}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i+O(h^6)$$

Following the same way, we have

$$\left(\frac{\partial^4\phi}{\partial x^4}\right)_e-\left(\frac{\partial^4\phi}{\partial x^4}\right)_w=h\left(\frac{\partial^5\phi}{\partial x^5}\right)_i+O(h^6)$$

Finally, we have

$$(\mathrm{TE})_{\mathrm{FV}}=\frac{(\phi_e-\phi_e^{\mathrm{model}})-(\phi_w-\phi_w^{\mathrm{model}})}{h}=-\frac{h}{8}\left[\left(\frac{\partial^2\phi}{\partial x^2}\right)_e-\left(\frac{\partial^2\phi}{\partial x^2}\right)_w\right]-\frac{h^3}{384}\left[\left(\frac{\partial^4\phi}{\partial x^4}\right)_e-\left(\frac{\partial^4\phi}{\partial x^4}\right)_w\right]+O(h^5)$$

$$(\mathrm{TE})_{\mathrm{FV}}=-\frac{h^2}{8}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i-\frac{h^4}{128}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i+O(h^6)$$

FDM method

$$(\mathrm{TE})_{\mathrm{CD,\ FD}} = -\frac{1}{6}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i h^2 - \frac{1}{120}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i h^4 + O(h^6)$$

The FVM method

$$(\mathrm{TE})_{\mathrm{CD,\ FV}} = -\frac{1}{8}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i h^2 - \frac{1}{128}\left(\frac{\partial^5\phi}{\partial x^5}\right)_i h^4 + O(h^6)$$

It can be found that the although with the same accuracy order, **coefficient of $(\mathrm{TE})_{\mathrm{FV}}$ is smaller than that of the $(\mathrm{TE})_{\mathrm{FD}}$**.

This means for the same grid size, for CD scheme, the error of the former one is lower.

7.5.2 Richardson extrapolation (外推法) for analyzing truncation error of numerical solutions

1) Basic idea of Richardson extrapolation

In principle, the converged solution can be obtained by continuously refining (细化） the meshes to mesh size approaching zero.However, this approach is impractical due to the limitations of computer memory and computational cost.

The **extrapolation method developed by Richardson uses results from grids of different resolutions to estimate the converged solution at zero grid size.**

The following assumption:

(1) The numerical solution converges monotonically (单调地) toward its converged solution.

(2) The grid is sufficiently fine such that higher-order terms in the truncation error are negligible compared with the leading term （首项）.

(3) Round-off errors and incomplete iteration errors can be neglected.

The **extrapolation method developed by Richardson uses results from grids of different resolutions to estimate the converged solution at zero grid spacing.**

$$\phi = \phi_{\alpha h} + C(\alpha h)^n$$

—ϕ is the converged solution;

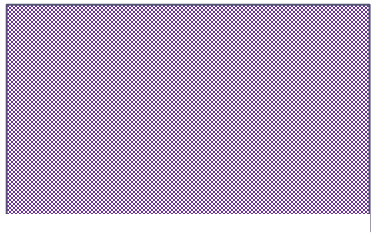
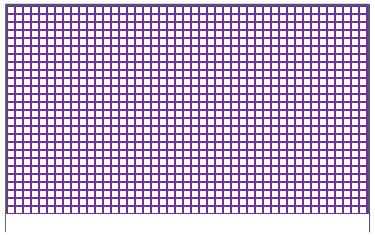
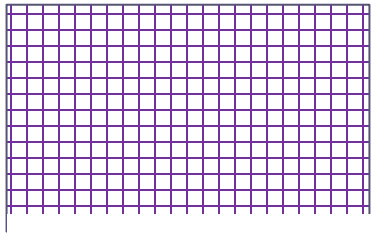
—h is size of the basic mesh size. α is grid refinement parameter, defined as the ratio of a certain mesh size to the basic mesh size;

—C and n depend on the discretization scheme used, and n denotes the order of the truncation error.

For a variable at the same position, three sets of mesh size are adopted

$$\begin{cases} \phi = \phi_h + C(h)^n \\ \phi = \phi_{2h} + C(2h)^n \\ \phi = \phi_{4h} + C(4h)^n \end{cases}$$

Mesh

	h	2h	4h
ϕ	ϕ1	ϕ2	ϕ3

The solution for ϕ, C and n can be determined:

$$\begin{cases} \phi = \phi_h + C(h)^n \\ \phi = \phi_{2h} + C(2h)^n \\ \phi = \phi_{4h} + C(4h)^n \end{cases}$$

$$n = \frac{\ln[(\phi_{2h} - \phi_{4h})/(\phi_h - \phi_{2h})]}{\ln 2}$$

$$\phi = \frac{2^n \phi_h - \phi_{2h}}{2^n - 1}$$

$$C = \frac{\phi - \phi_h}{h^2}$$

Therefore, the truncation error for mesh size as h is as follows:

$$\varepsilon_n = \phi - \phi_h = \frac{\phi_h - \phi_{2h}}{2^n - 1}$$

Comment:

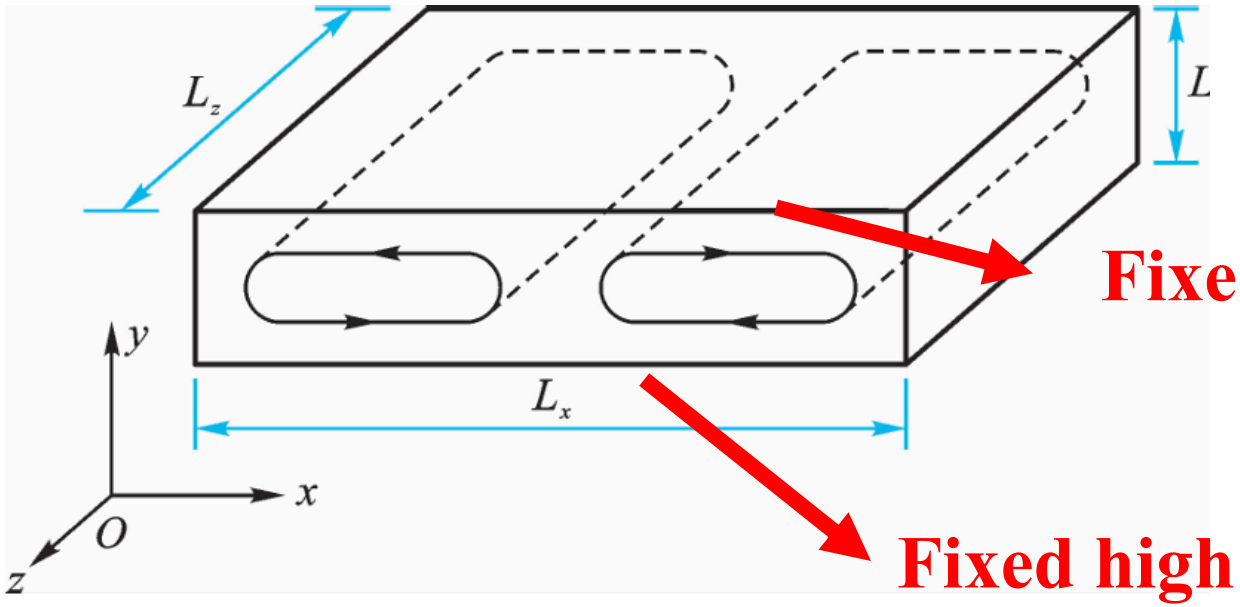
- ϕ can be a physical variable of a certain position in the domain such as velocity, temperature and pressure. It also can be a statical variable (统计量）of the entire domain such as the *Nu* number of the friction factor *f*.
- If the discretizing scheme is with second order, namely *n*=2, then

$$\phi = \frac{2^n \phi_h - \phi_{2h}}{2^n - 1} \longrightarrow \phi = \frac{4\phi_h - \phi_{2h}}{3}$$

- All the values of ϕ in the above equation should at the same position. However, if α is not an integer such as 2, 4,…, Then

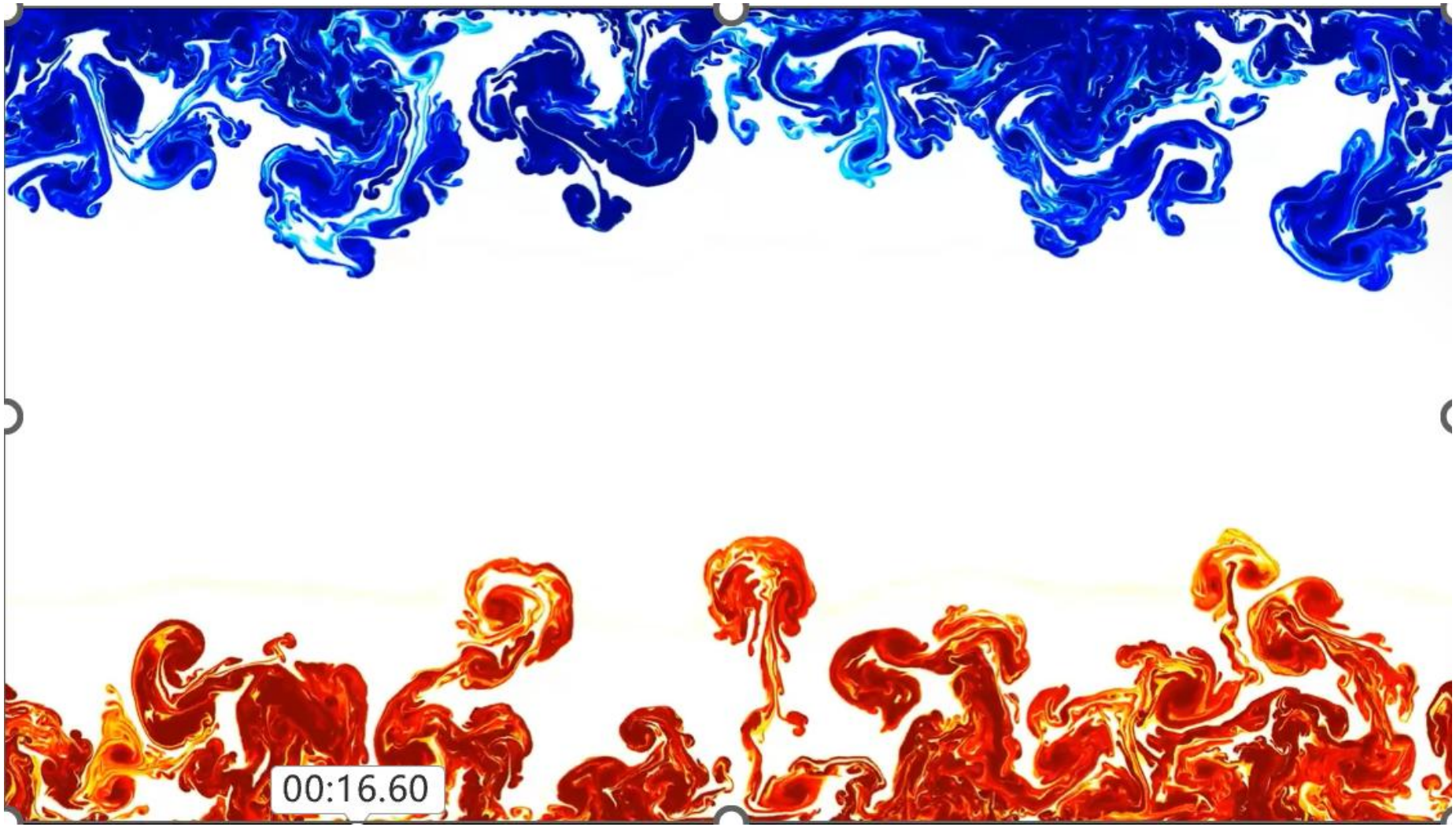
$$\phi = \frac{\alpha \phi_h - \phi_{\alpha h}}{\alpha^n - 1}$$

Application of Richardson extrapolation

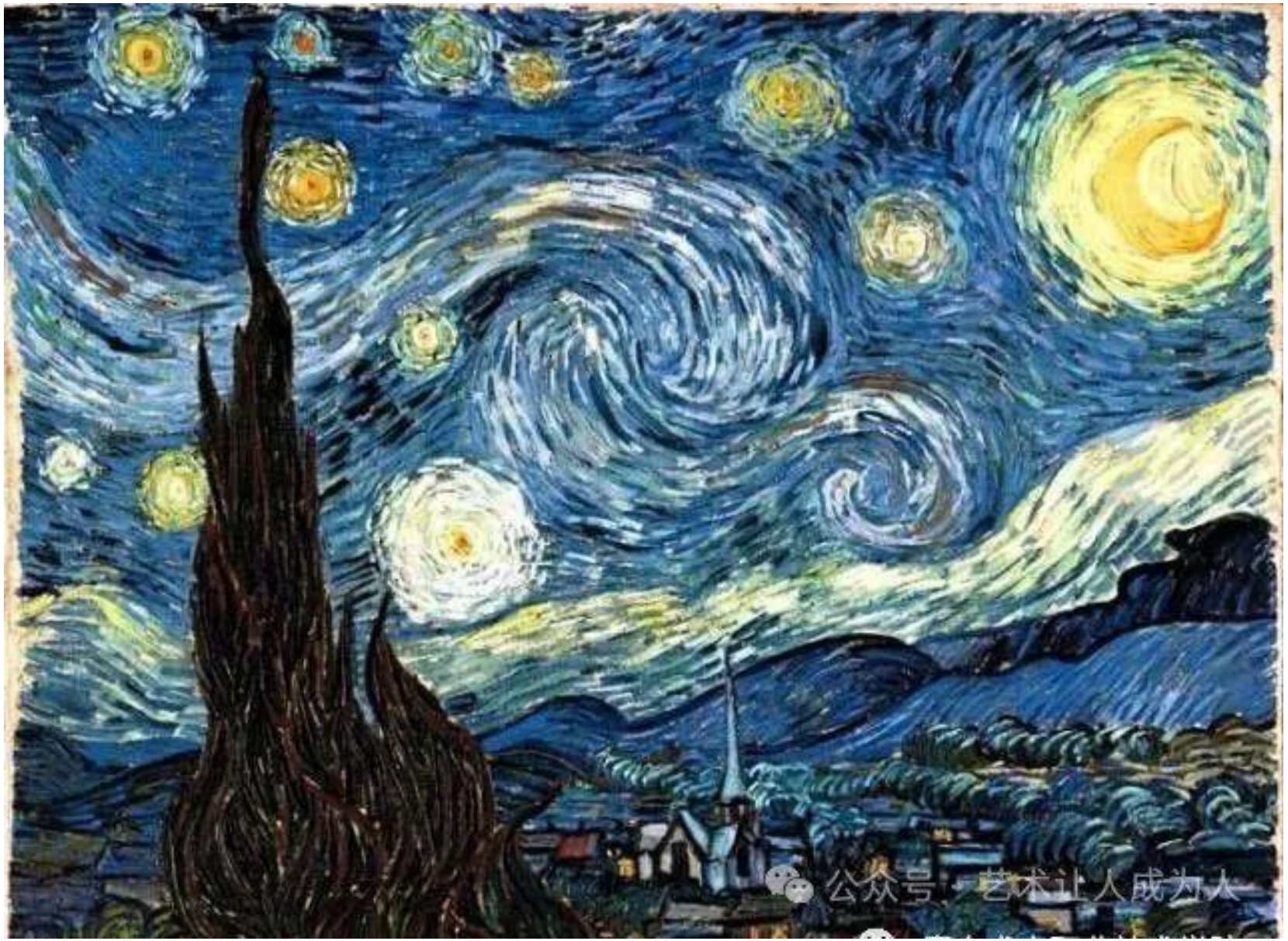


Physical problem:cin a rectangular enclosure under the Boussinesq approximation. The aspect ratios are Lx/L=3.5, Lz/L=2.1, Pr=2.5, Ra=20000. Boundary conditions are as follows:

- **Bottom wall (y=0): no-slip, fixed high temperature;**
- **Top wall (y=L): no-slip, fixed low temperature;**
- **Four vertical side walls: no-slip and adiabatic (绝热）.**



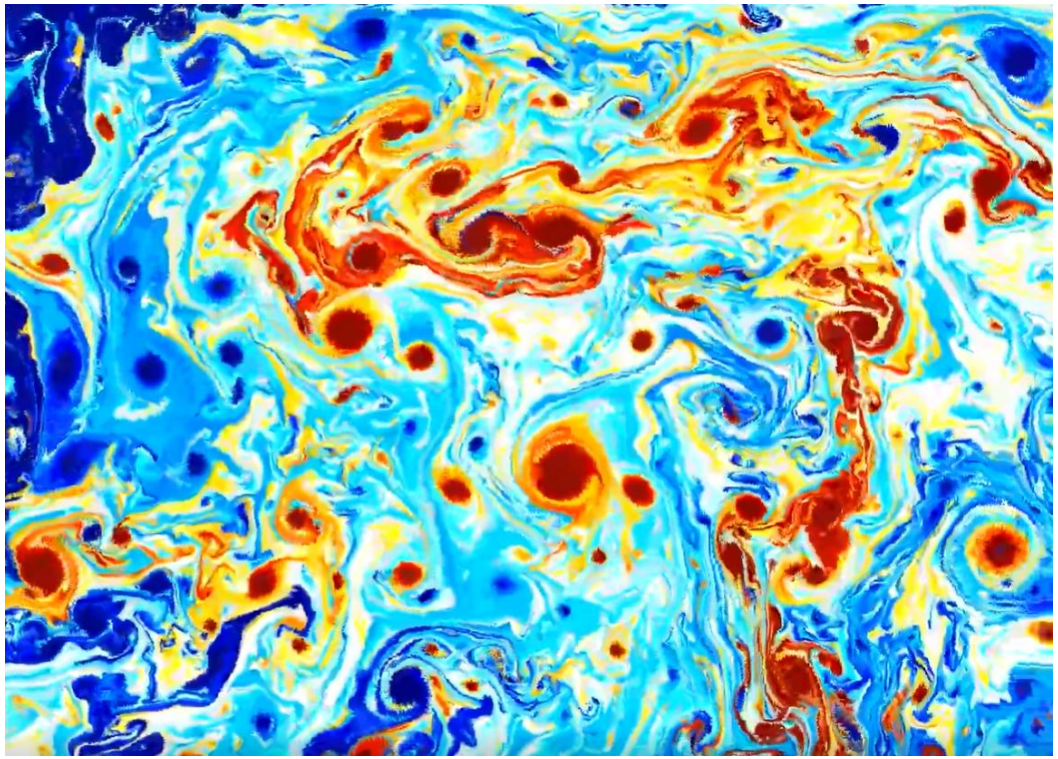
2D Rayleigh–Bénard natural convection with high Ra

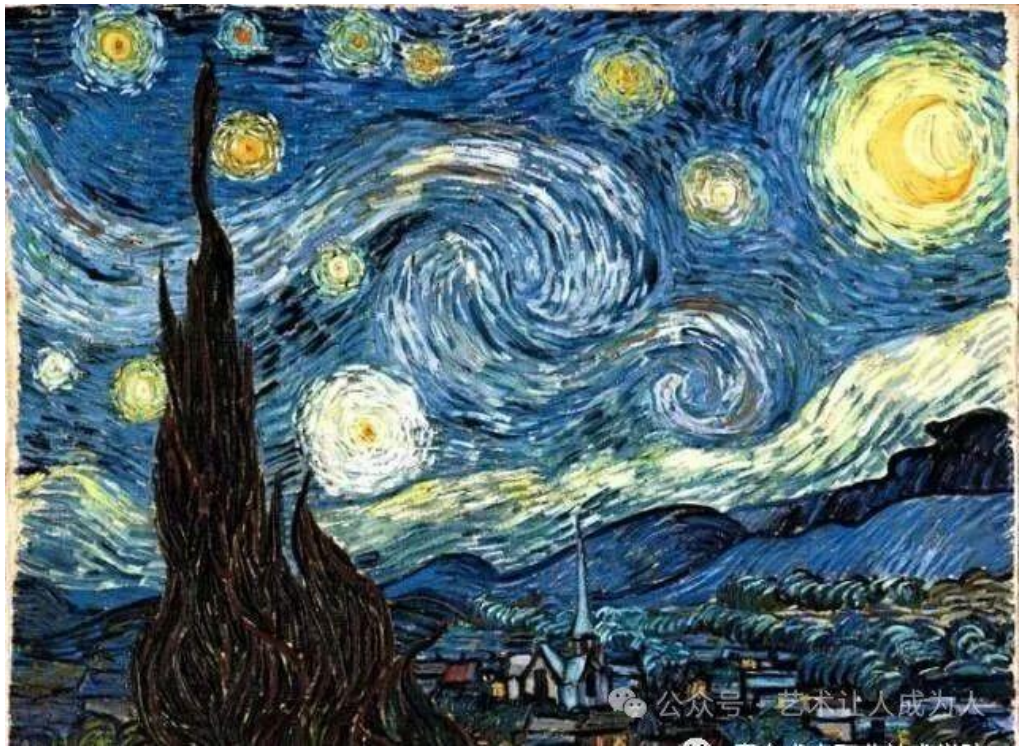




Vincent van Gogh
梵高

The Starry Night (星空）





Science and art

4h　　2h　　h

网格	20×20×20	40×40×40	80×80×80
Nu	2.646	2.586	2.571
U_{max}	42.75	42.97	43.01
$X_{max}(U_{max})$	0.70	0.61	0.66
$Y_{max}(U_{max})$	0.82	0.81	0.81
$Z_{max}(U_{max})$	0.47	0.50	0.51
V_{max}	46.67	51.05	51.95
$X_{max}(V_{max})$	0.09	0.04	0.02
$Y_{max}(V_{max})$	0.55	0.32	0.52
$Z_{max}(V_{max})$	1.0	1.02	1.04
W_{max}	7.81	8.13	8.18
$X_{max}(W_{max})$	0.09	0.04	0.02
$Y_{max}(W_{max})$	0.22	0.21	0.21
$Z_{max}(W_{max})$	0.315	0.315	0.315

$$n=\frac{\ln[(\phi_{2h}-\phi_{4h})/(\phi_h-\phi_{2h})]}{\ln 2}$$

$$\phi=\frac{2^n\phi_h-\phi_{2h}}{2^n-1}$$

$$C=\frac{\phi-\phi_h}{h^2}$$

$$C=\frac{\phi-\phi_h}{h^2}=\frac{2.566-2.571}{h^2}=-\frac{0.005}{h^2}$$

网格	20×20×20	40×40×40	80×80×80	Richardson 外推法	相对偏差/%
Nu	2.646	2.586	2.571	2.566	3.12
U_{max}	42.75	42.97	43.01	43.02	0.63
$X_{max}(U_{max})$	0.70	0.61	0.66	0.677	
$Y_{max}(U_{max})$	0.82	0.81	0.81	0.18	—
$Z_{max}(U_{max})$	0.47	0.50	0.51	0.513	
V_{max}	46.67	51.05	51.95	52.25	10.38
$X_{max}(V_{max})$	0.09	0.04	0.02	0.0133	
$Y_{max}(V_{max})$	0.55	0.32	0.52	0.52	—
$Z_{max}(V_{max})$	1.0	1.02	1.04	1.048	
W_{max}	7.81	8.13	8.18	8.200	4.76
$X_{max}(W_{max})$	0.09	0.04	0.02	0.0133	
$Y_{max}(W_{max})$	0.22	0.21	0.21	0.21	—
$Z_{max}(W_{max})$	0.315	0.315	0.315	0.315	

Contents

7.1 Consistency, Convergence and Stability of Discretized Equations

7.2 von Neumann Method for Analysing Stability of Initial Problems

7.3 Conservation of Discretized Equations

7.4 Classification and V&V of Numerical Solution Errors

7.5 Analysis of Truncation Error of Discretization Schemes

7.6 Analysis of Grid Convergence Index and Incomplete Iteration Error

7.6 Analysis of Grid Convergence Index (网格收敛指标) and Incomplete Iteration Error （不完全迭代误差）

7.6.1. The grid convergence index (GCI)

7.6.2 The incomplete iteration error

7.6.3 Methods for Verifying Numerical Results and Analyzing Errors

Problem: There are two sets of simulations, in each set there are two simulations: A-1, A-2, B-1, B-2. The grid size of A-1 and B-1 is the same.

(1) In the first set (A-1 and A-2), the truncation error is 1-order. The grid of A-2 is refined （细化） by 50% compared with A-1, the relative error between results from A-1 and A-2 is **4%**.

(2) In the second set (B-1 and B-2), the truncation error is 2-order. The grid of B-2 is refined by 100% compared with B-1, the relative error between result from B-1 and B-2 is **6%**.

Question: A-2 is more accurate or B-2 is more accurate? It is hard to answer because the truncation error is different and the grid size is also different.

Therefore, a reasonable, unified method is required to evaluate and compare results from different grid sizes and different truncation errors.

Assume ϕ_{exact} is the exact solution, the solution of the case with p-order of truncation error and mesh size h is assumed as follows:

$$\phi_1 = \phi_{exact} + g_1 h^p + g_2 h^{p+1} + g_3 h^{p+2} + \cdots \qquad \textbf{(a)}$$

With the mesh size increases to αh

$$\phi_2 = \phi_{exact} + g_1 (\alpha h)^p + g_2 (\alpha h)^{p+1} + g_3 (\alpha h)^{p+2} + \cdots \qquad \textbf{(b)}$$

α^p*(a)-(b) leads to the exact solution ϕ_{exact} without T.E.

$$\phi_{exact} = \phi_1 + \frac{\phi_1 - \phi_2}{\alpha^p - 1} + O(h^{p+1}) = \frac{\alpha^p \phi_1 - \phi_2}{\alpha^p - 1} + O(h^{p+1})$$

The difference between ϕ_{exact} and ϕ_1 is as follows $\quad \phi_1 - \phi_{exact} = \dfrac{\phi_2 - \phi_1}{\alpha^p - 1}$

The difference between ϕ_{exact} and ϕ_1 is as follows $\phi_1 - \phi_{exact} = \dfrac{\phi_2 - \phi_1}{\alpha^p - 1}$

The relative error or the discrepancy is defined as follows:



$$E_1 = \frac{\phi_1 - \phi_{exact}}{\phi_1} = \left(\frac{\phi_2 - \phi_1}{\phi_1}\right)\frac{1}{\alpha^p - 1} = \frac{\varepsilon}{\alpha^p - 1}$$

$$\varepsilon = \frac{\phi_2 - \phi_1}{\phi_1}$$

It can be seen from the above equation that the relative error is not simplify calculated by

$$\varepsilon = \frac{\phi_2 - \phi_1}{\phi_1}$$



E_1 not only contains the *p*-order, but also the mesh ratio α, which is more suitable for evaluating the relative error.

The grid convergence index (GCI) is defined as

$$GCI_{fine} = \frac{3|\varepsilon|}{\alpha^{p}-1} \qquad \varepsilon = \frac{\phi_2 - \phi_1}{\phi_1}$$

If α=2 and p=2, α^p-1=3; Then, GCI=|ε|.

Therefore, for arbitrary p and α, the relative error ε, is the same as the value of GCI with p=2 and α=2.

This again demonstrates that it is not suitable to use ε to evaluate the accuracy of simulations with different p and α.

The original problem:

A-1, A-2: T.E. p=1, grid refine ratio α=1.5, ε=4%; GCI=3*4%/(1.5-1)=24%

B-1, B-2: T.E. p=2, grid refine ratio α=2.0, ε= 6%; GCI=3*6%/(4-1)=6%

Therefore, the second set is more accurate.

7.6.2 The incomplete iterative error

Definition: The iteration convergence error refers to the difference between the current solution obtained from numerical computation and the exact solution of the discrete equations on the same mesh.

In other words, the discrepancy of the solution of the discrete equations from its exact solution when iteration is terminated with insufficient iteration times. (**由于不完全迭代次数导致的解与离散方程精确解之间的差别).**

From Section 2, it is known that $\phi^{n+1} = A\phi^n + g$

In fact, the incomplete iterative error also propagates as follows: $\varepsilon^{n+1} = A\varepsilon^n$

The incomplete iterative error can be calculated as （|| ||, norm operator,**范数**）

$$\varepsilon^n = \frac{\phi^{n+1} - \phi^n}{\lambda - 1} \qquad \lambda = \frac{\| \phi^{n+1} - \phi^n \|_2}{\| \phi^n - \phi^{n-1} \|_2} \qquad \| \phi^{n+1} - \phi^n \|_2 = \sqrt{\sum_i^N (\phi_i^{n+1} - \phi_i^n)^2}$$

7.5.3 Methods for Verifying Numerical Results and Analyzing Errors

The method proposed by Celik in 2008

Step 1: choose three sets of mesh system to conduct numerical simulation.

Step 2: calculate the typical mesh size

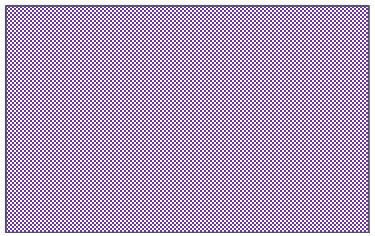
3D $$h = \left[\frac{1}{N} \sum_{i=1}^{N} (\Delta V_i) \right]^{1/3}$$ **2D** $$h = \left[\frac{1}{N} \sum_{i=1}^{N} (\Delta A_i) \right]^{1/2}$$

Step 3: calculate the apparent order p (视在或表观截断误差阶数)

- h_1, h_2, and h_3 for the three sets of meshes
- $h_1 < h_2 < h_2$, meaning h_1 corresponding to the finest mesh
- $\alpha_{21} = h_2 / h_1$, $\alpha_{32} = h_3 / h_2$.

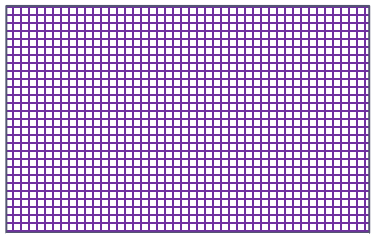
Step 4: calculate the exact value using the extrapolation method

Mesh



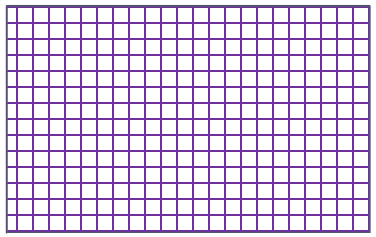
h1

$$\alpha_{21} = \frac{h_2}{h_1}$$



h2

$$\alpha_{32} = \frac{h_3}{h_2}$$



h3

Numerical Solution

ϕ1 **ϕ2** **ϕ3**

$$e_{21} = \phi_2 - \phi_1 \qquad e_{32} = \phi_3 - \phi_2$$

Because

$$\phi_{\text{exact}} = \phi_1 + \frac{\phi_1 - \phi_2}{\alpha^p - 1} + O(h^{p+1}) = \frac{\alpha^p \phi_1 - \phi_2}{\alpha^p - 1} + O(h^{p+1})$$

Then

$$\phi_{\text{exact}}^{21} = (\alpha_{21}^p \phi_1 - \phi_2)/(\alpha_{21}^p - 1)$$

$$\phi_{\text{exact}}^{32} = (\alpha_{32}^p \phi_1 - \phi_2)/(\alpha_{32}^p - 1)$$

Step 4: calculate the exact value using the extrapolation method

Mesh **h1** $\alpha_{21}=\frac{h_2}{h_1}$ **h2** $\alpha_{32}=\frac{h_3}{h_2}$ **h3**

Solution **ϕ1** **ϕ2** **ϕ3**

$$e_{21}=\phi_2-\phi_1 \qquad e_{32}=\phi_3-\phi_2$$

Then

$$\phi_{\text{exact}}^{21}=(\alpha_{21}^{p}\phi_1-\phi_2)/(\alpha_{21}^{p}-1)$$

$$\phi_{\text{exact}}^{32}=(\alpha_{32}^{p}\phi_1-\phi_2)/(\alpha_{32}^{p}-1)$$

The apparent T.E.

$$p=\frac{1}{\ln(\alpha_{21})}[\ln|e_{32}/e_{21}|+q(p)]$$

$$q(p)=\ln\left(\frac{\alpha_{21}^{p}-s}{\alpha_{32}^{p}-s}\right)$$

$$s=1\cdot\operatorname{sgn}(e_{32}/e_{21})$$

$$\operatorname{sgn}(x)=\begin{cases}+1, & x>0\\ 0, & x=0\\ -1, & x<0\end{cases}$$

Step 5: calculate the error and the apparent order of the T.E.

$$\phi_{\text{exact}}^{21} = (\alpha_{21}^{p}\phi_1 - \phi_2)/(\alpha_{21}^{p} - 1)$$

$$\phi_{\text{exact}}^{32} = (\alpha_{32}^{p}\phi_1 - \phi_2)/(\alpha_{32}^{p} - 1)$$

$$p = \frac{1}{\ln(\alpha_{21})}[\ln|e_{32}/e_{21}| + q(p)]$$

$$q(p) = \ln\left(\frac{\alpha_{21}^{p} - s}{\alpha_{32}^{p} - s}\right)$$

$$s = 1 \cdot \text{sgn}(e_{32}/e_{21})$$

The conventional error $\varepsilon_{\text{a}}^{21} = \left|\dfrac{\phi_1 - \phi_2}{\phi_1}\right|$

The conventional error using the exact solution $\varepsilon_{\text{exact}}^{21} = \left|\dfrac{\phi_{\text{exact}}^{21} - \phi_1}{\phi_{\text{exact}}^{21}}\right|$

The GCI $\text{GCI}_{\text{fine}}^{21} = \dfrac{F_{\text{GCI}}\varepsilon_{\text{a}}^{21}}{\alpha_{21}^{p} - 1}$

Example: the exact solution and the error estimation Table 7-7

参数符号	相应数值(单调收敛解)
N_1，N_2，N_3	18000，8000，4500
α_{21}	1.5
α_{32}	1.333
ϕ_1	6.063
ϕ_2	5.972
ϕ_3	5.863
p	1.53
ϕ_{exact}^{21}	6.1685
ε_a^{21}	1.5%
ε_{exact}^{21}	1.7%
GCI_{fine}^{21}	2.2%

Thanks for your attention