澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

1981 2021
不惑新航
揚帆追夢
SET SAIL ANEW ON THE RUBY JUBILEE

# Introduction to Applied Mathemtics --- Numerical Methods for Differential Equations

MATH 3014

Monday & Thursday 14:30-15:45

Instructor: **Dr. Luo Li**

https://www.fst.um.edu.mo/personal/math3014/

Department of Mathematics

Faculty of Science and Technology

**Purpose**

To provide an introduction to finite difference methods and finite element methods for solving ordinary and partial differential equations of initial and boundary value problems.

**Emphasis**

- Classical finite difference schemes
- Classical finite element schemes
- Implementation using Matlab codes
- Mathematical theory: consistency, stability, convergence

**Prerequisite**

- Calculus
- Linear algebra
- Numerical analysis
- Ordinary differential equations
- Partial differential equations

# Outline

## Part I — Finite difference methods for ordinary differential equations

- Explicit and implicit methods
- Local/global truncation error
- Stability, convergence

## Part II — Finite difference methods for partial differential equations

- One-dimensional boundary value problems
- Two-dimensional elliptic partial differential equations
- Parabolic  partial differential equations
- Hyperbolic partial differential equations

## Part III — Finite element methods for partial differential equations

- One-dimensional boundary value problems
- Two-dimensional elliptic partial differential equations

# Some Abbreviations

ODE: Ordinary differential equation

PDE: Partial differential equation

IVP: Initial value problem

BVP: Boundary value problem

1D: One-dimensional

2D: Two-dimensional

# Initial Value Problems (IVP)

Examples of ODE

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0 \,; \tag{1.1}$$

A single higher-order differential equation may be rewritten as a first-order system.

**Example** Convert the following IVP to a first-order system of ODEs of an IVP.

$$\begin{aligned} &u''(t) + a(t)u'(t) + b(t)u(t) = f(t), \\ &u(0) = u_0, \quad \boxed{u'(0) = v_0}. \end{aligned} \tag{1.2}$$

An ODE IVP can often be solved using Runge–Kutta methods, with adaptive time steps. In Matlab, there is the ODE-Suite which includes ode45, ode23, ode23s, ode15s.

# Boundary Value Problems (BVP)

ODE BVP:

$$u''(x) + a(x)u'(x) + b(x)u(x) = f(x), \quad 0 < x < 1,$$
$$u(0) = u_0, \quad \boxed{u(1) = u_1}; \tag{1.3}$$

PDE BVP:

Linear elliptic type

$$u_{xx} + u_{yy} = f(x, y), \quad (x, y) \in \Omega,$$
$$u(x, y) = u_0(x, y), \quad (x, y) \in \partial\Omega, \tag{1.4}$$

Eigenvalue problems:

$$u''(x) = \lambda u(x),$$
$$u(0) = 0, \quad u(1) = 0. \tag{1.6}$$

Mixed BVP and IVP

$$\begin{aligned} &u_t = au_{xx} + f(x,t), \\ &u(0,t) = g_1(t), \quad u(1,t) = g_2(t), \qquad \text{BC} \qquad (1.5) \\ &u(x,0) = u_0(x), \qquad \text{IC}, \end{aligned}$$

Diffusion and reaction equations

Diffusion term

Advection term

$$\frac{\partial u}{\partial t} = \nabla \cdot (\beta \nabla u) + \mathbf{a} \cdot \nabla u + f(u)$$

Reaction term

The incompressible Navier–Stokes equations

$$\begin{aligned} &\rho\,(\mathbf{u}_t + (\mathbf{u} \cdot \nabla)\mathbf{u}) = \nabla p + \mu \Delta \mathbf{u} + \mathbf{F}, \\ &\nabla \cdot \mathbf{u} = 0. \end{aligned}$$

A linear second-order PDE has the following general form:

$$a(x,y)u_{xx} + 2b(x,y)u_{xy} + c(x,y)u_{yy}$$

$$+ d(x,y)u_x + e(x,y)u_y + g(x,y)u(x,y) = f(x,y)$$

Linear: the coefficients are independent of u(x, y)

The classification of the PDE

- Elliptic if $b^2 - ac < 0$ for all $(x,y) \in \Omega$,
- Parabolic if $b^2 - ac = 0$ for all $(x,y) \in \Omega$, and
- Hyperbolic if $b^2 - ac > 0$ for all $(x,y) \in \Omega$.

## Analytic methods

- Too few differential equations can be solved exactly in terms of elementary functions such as polynomials, $\log x$ , $ex$, trigonometric functions ($\sin x, \cos x, \ldots$), etc.
- Considerable effort and sound mathematical theory are often needed.
- The closed form of the solution may even turn out to be too messy to be useful.

## Numerical methods

- Discrete numerical values may represent the solution to a certain accuracy.
- Obtained using computers.
- Provide effective solutions of many problems that were impossible to obtain before.

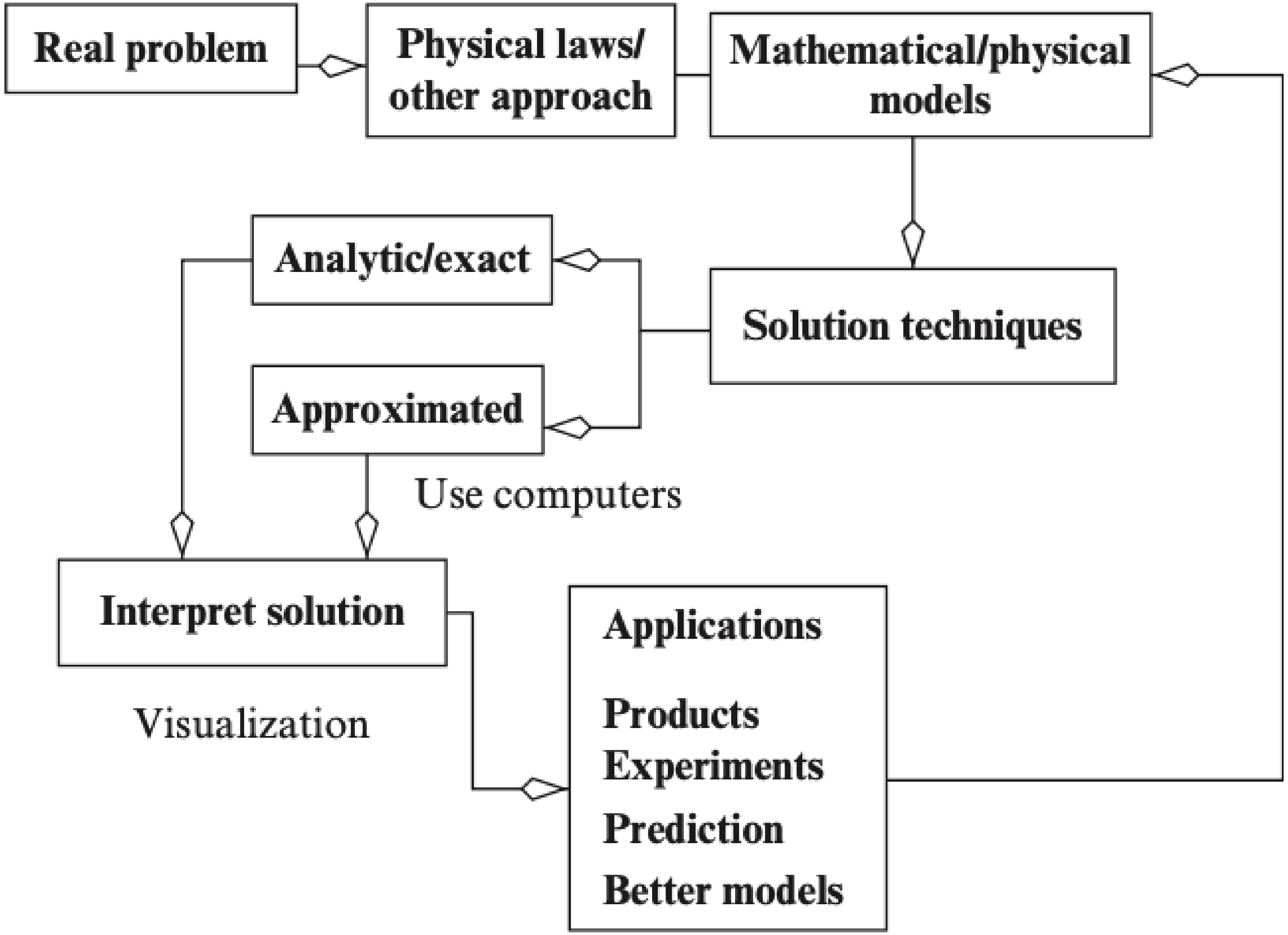



Figure 1.1. A flowchart of a problem-solving process.

# Finite difference methods

• Easy to implement for regular domains, e.g., rectangular domains in Cartesian coordinates, and circular or annular domains in polar coordinates.

• Their discretization and approximate solutions are pointwise, and the fundamental mathematical tool is the Taylor expansion.

• There are many fast solvers and packages for regular domains.

• Difficult to implement for complicated geometries.

• Requires the existence of high-order derivatives

# Finite element methods

- Very successful for elliptic type problems

- Suitable approach for problems with complicated boundaries

- Sound theoretical foundation especially for elliptic PDE

- Less regularity requirements.

- Many commercial packages, e.g., Ansys, Matlab PDE Tool-Box, Triangle.

# Reference

*Numerical Solution of Differential Equations: Introduction to Finite Difference and Finite Element Methods*, Zhilin Li, Zhonghua Qiao, and Tao Tang.
https://zhilin.math.ncsu.edu/

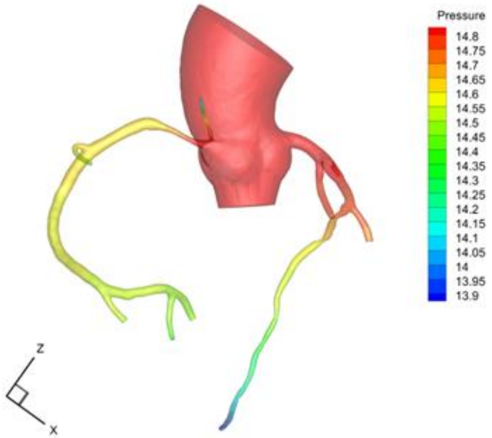
*Numerical Solution of Ordinary Differential Equations*, Kendall Atkinson, Weimin Han, David Stewart.
https://onlinelibrary.wiley.com/doi/book/10.1002/9781118164495

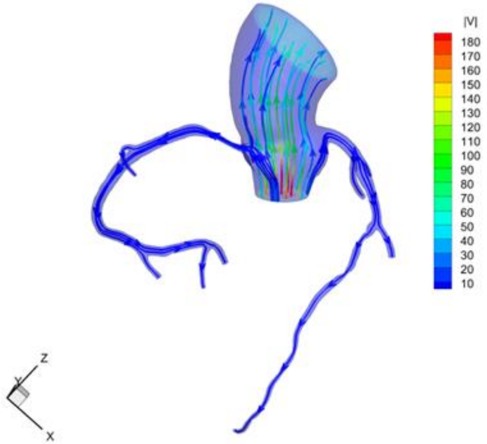
*Finite Difference Methods for Ordinary and Partial Differential Equations: Steady State and Time Dependent Problems*, Randy LeVeque.
http://faculty.washington.edu/rjl/fdmbook/

*Numerical Solution of Partial Differential Equations*, Louise Olsen-Kettle
https://espace.library.uq.edu.au/view/UQ:239427

# Biofluid Dynamics

- ❑ Prediction of stroke/myocardial infarction: non-invasive, personalized, flexible




The incompressible Navier-Stokes equation with initial and boundary conditions

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) - \nabla\cdot\boldsymbol{\sigma} = \rho\mathbf{f}, \quad \text{in} \quad \Omega,$$

$$\nabla\cdot\mathbf{u} = 0, \quad \text{in} \quad \Omega.$$

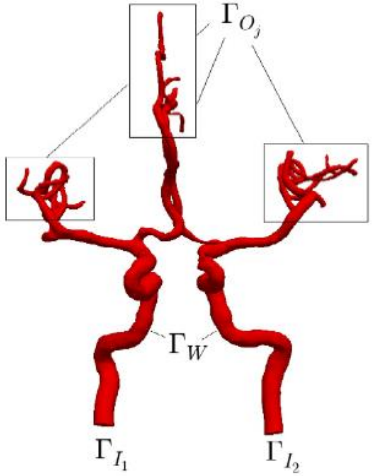
$$\mathbf{u} = \mathbf{0}, \quad \text{on} \quad \Gamma_W,$$

$$\mathbf{u} = \mathbf{v}_{I_i}, \quad \text{on} \quad \Gamma_{I_i}, \quad i = 1, 2.$$

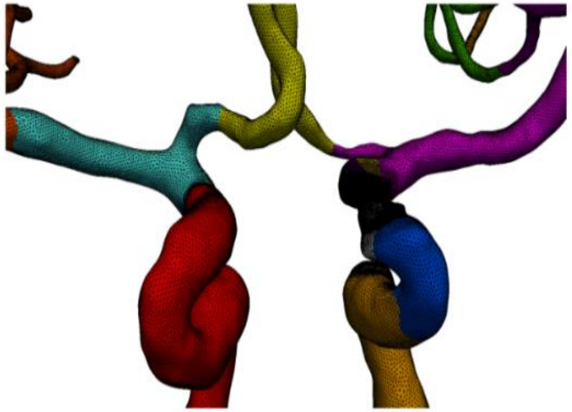
$$p = R_j Q_j, \quad \text{on} \quad \Gamma_{O_j}, \quad j = 1, \ldots, m.$$

$$Q_j = \int_{\Gamma_{O_j}} \mathbf{u}\cdot\mathbf{n}_j \mathrm{d}\Gamma, \quad j = 1, \ldots, m$$

The physical problem




Extraction of geometry



Mesh generation

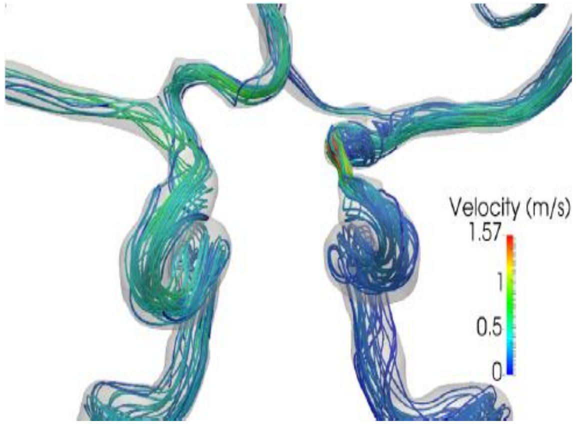
$$\int_\Omega \rho \frac{\partial \mathbf{u}}{\partial t}\cdot \mathbf{w}\, d\Omega + \int_\Omega \rho\, (\mathbf{u}\cdot\nabla)\, \mathbf{u}\cdot\mathbf{w}\, d\Omega - \int_\Omega p\nabla\cdot\mathbf{w}\, d\Omega + \int_\Omega \mu\left(\nabla\mathbf{u}+\nabla\mathbf{u}^T\right):\nabla\mathbf{w}\, d\Omega$$

$$+ \int_\Omega (\nabla\cdot\mathbf{u})\, q\, d\Omega - \int_\Omega \rho\, \mathbf{f}\cdot\mathbf{w}\, d\Omega + \sum_{j=1}^{m} \int_{\Gamma_{O_j}} \left(R_j Q_j \mathbf{n}_j\cdot\mathbf{w} - \mu\, \mathbf{n}_j\cdot\left(\nabla\mathbf{u}+\nabla\mathbf{u}^T\right)\cdot\mathbf{w}\right) d\Gamma = 0$$

$$A\mathbf{x} = b$$

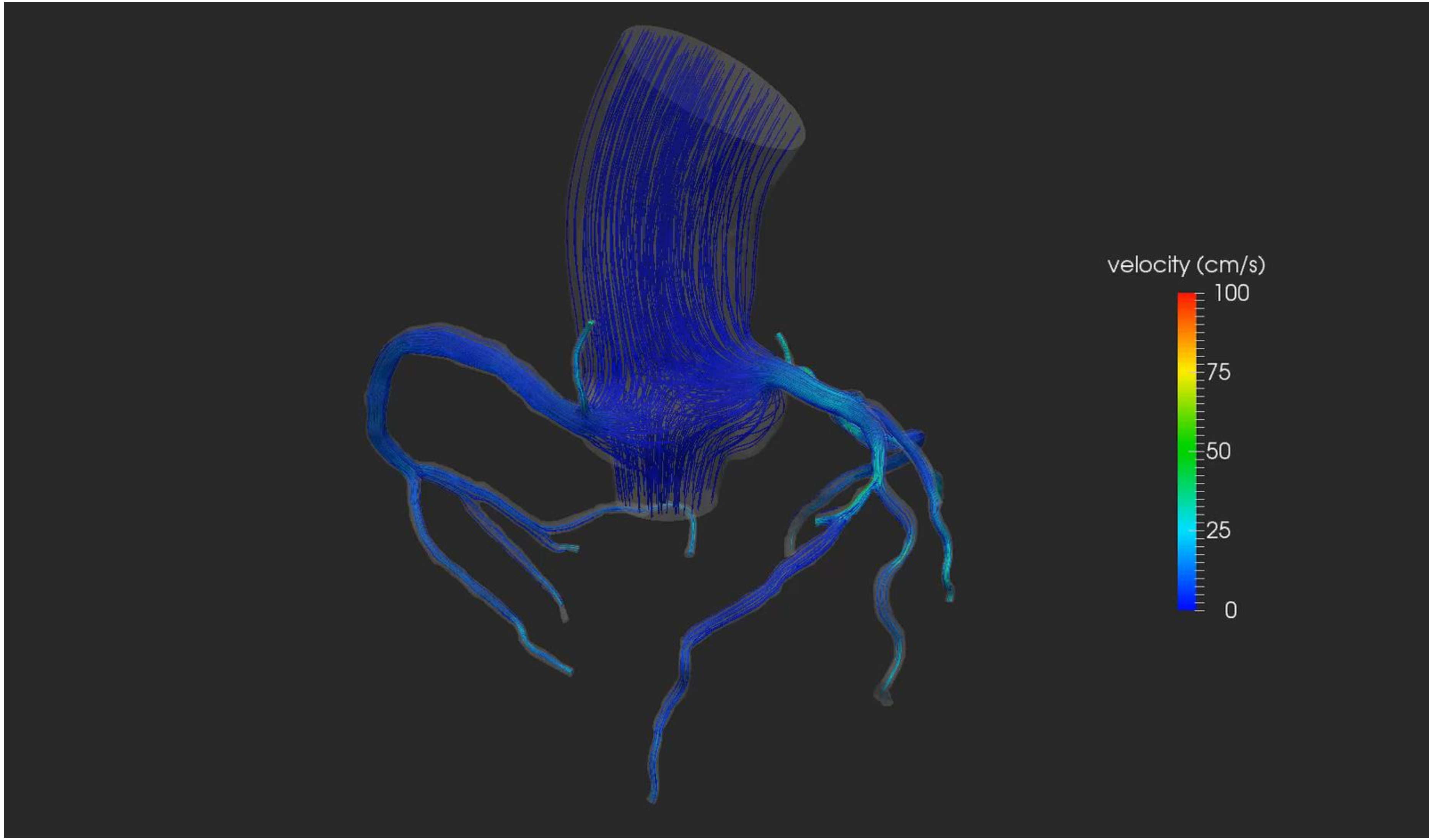
Numerical method

High-performance computing




Visualization of results

A **non-invasive** assessment for heart artery disease: personalized, accurate, low cost, and easy-to-implement

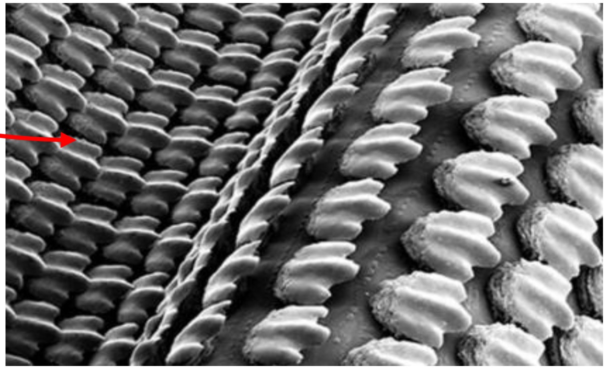


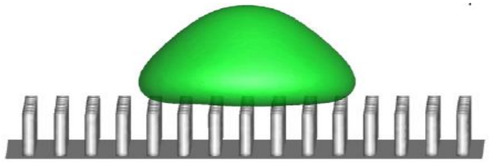
# Microfluidics

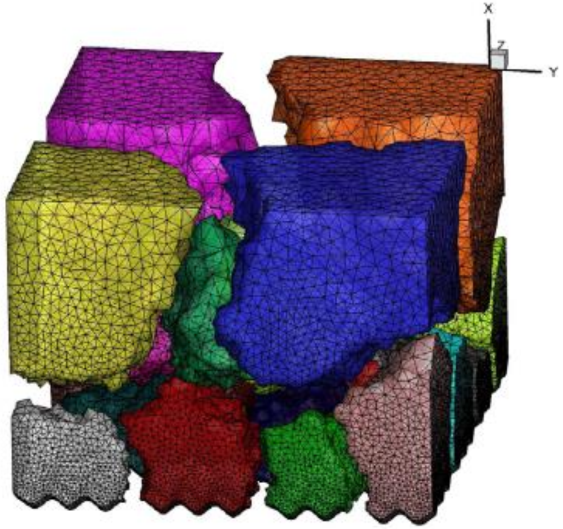
- Applications: waterproof materials, inkjet printheads, coatings

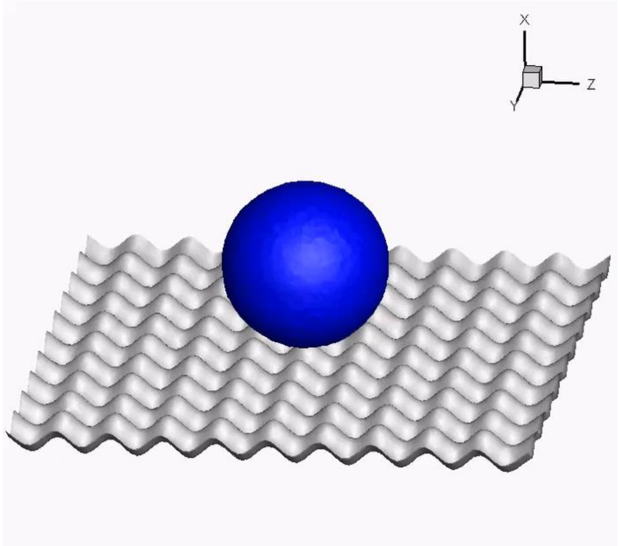
A coupled Cahn-Hilliard and Navier-Stokes equations

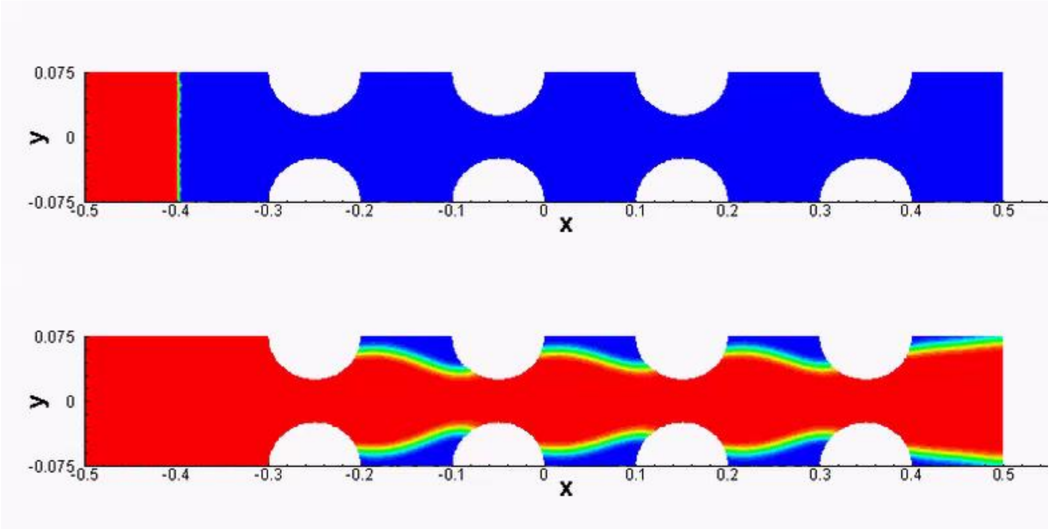
$$
\begin{cases}
\dfrac{\partial \phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = \mathcal{L}_d \Delta\mu, & \text{in} \quad \Omega,\\
\mu = -\epsilon\Delta\phi - \dfrac{\phi}{\epsilon} + \dfrac{\phi^3}{\epsilon}, & \text{in} \quad \Omega,\\
Re\rho\left(\dfrac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = -\nabla p + \nabla\cdot(\eta D(\mathbf{u})) + \mathcal{B}\mu\nabla\phi + \mathbf{g}_{ext}, & \text{in} \quad \Omega,\\
\nabla\cdot\mathbf{u} = 0, & \text{in} \quad \Omega,\\
\dfrac{\partial \phi}{\partial t} + u_\tau \partial_\tau \phi = -\mathcal{V}_s L(\phi), \qquad \partial_n \mu = 0, & \text{on} \quad \Gamma_w,\\
u_\tau := \mathbf{u}\cdot\tau = \mathcal{L}_s \mathcal{B} L(\phi)\partial_\tau\phi/\eta - \mathcal{L}_s \mathbf{n}\cdot D(\mathbf{u})\cdot\tau, & \text{on} \quad \Gamma_w,\\
u_n := \mathbf{u}\cdot\mathbf{n} = 0, & \text{on} \quad \Gamma_w.
\end{cases}
$$



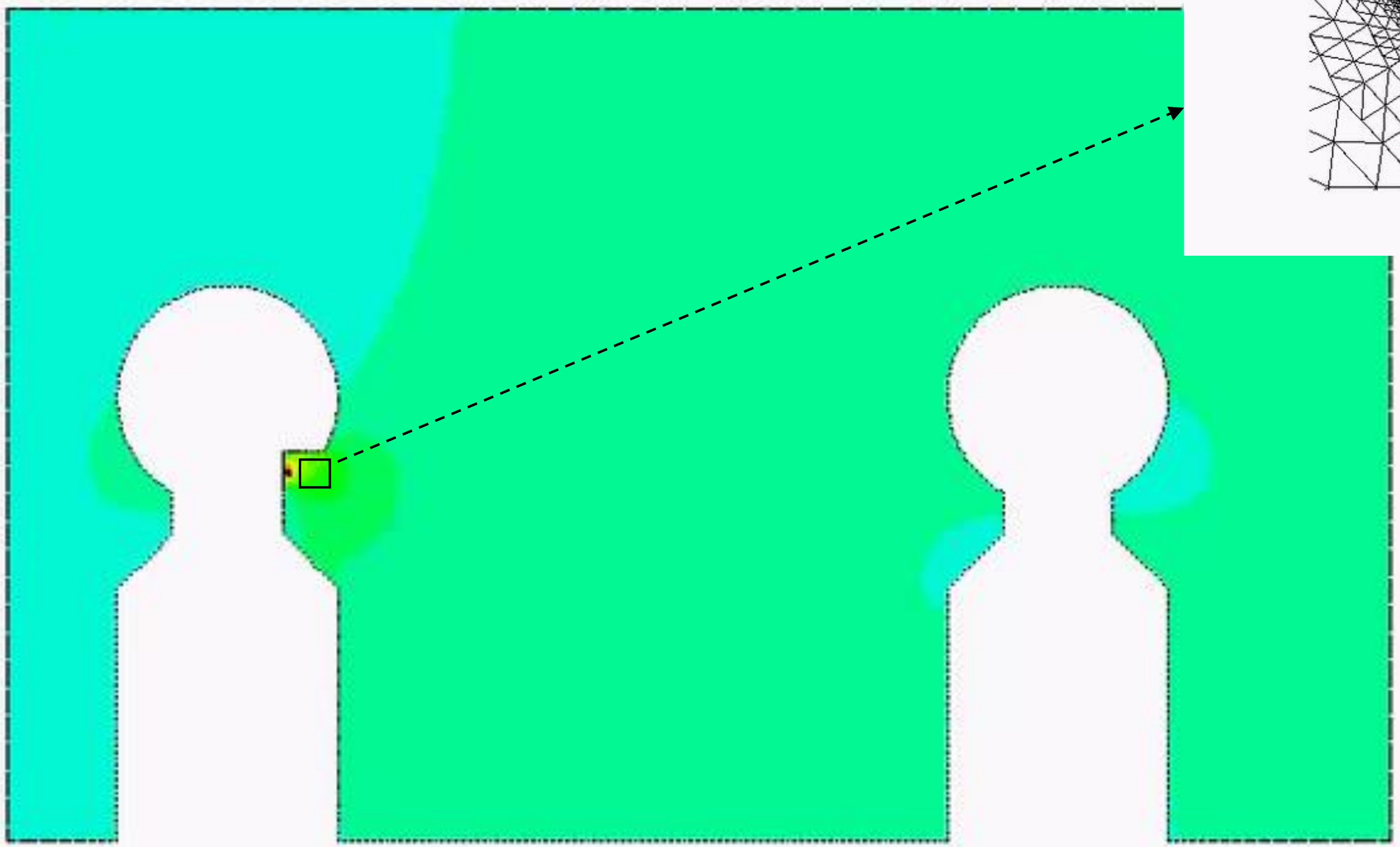


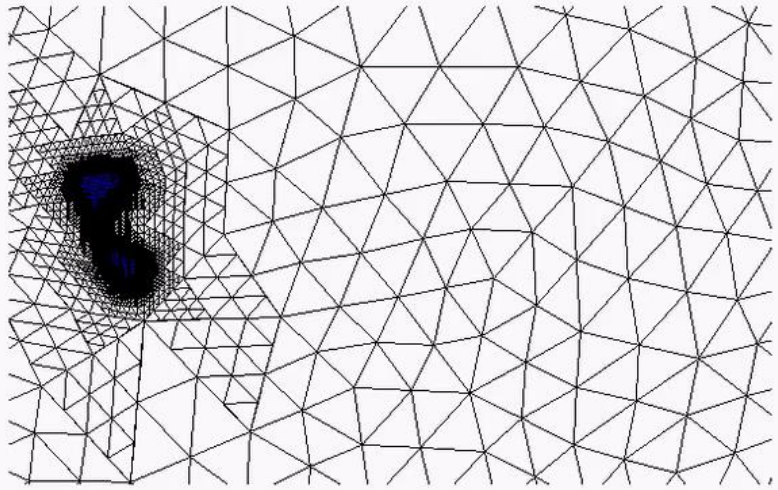

X
Z
Y


X
Z
Y

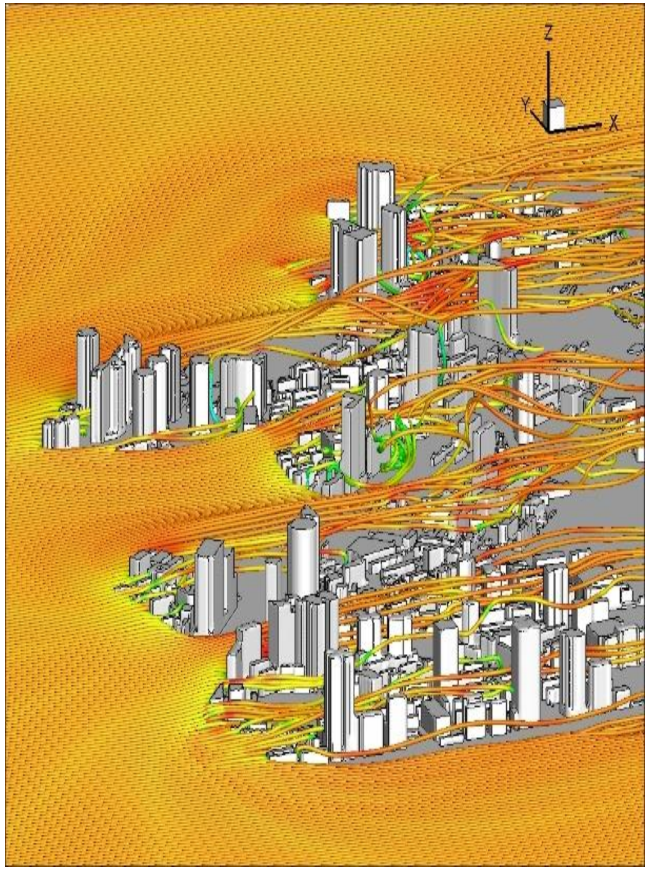

0.075
y
0
-0.075
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5
x
0.075
y
0
-0.075
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5
x

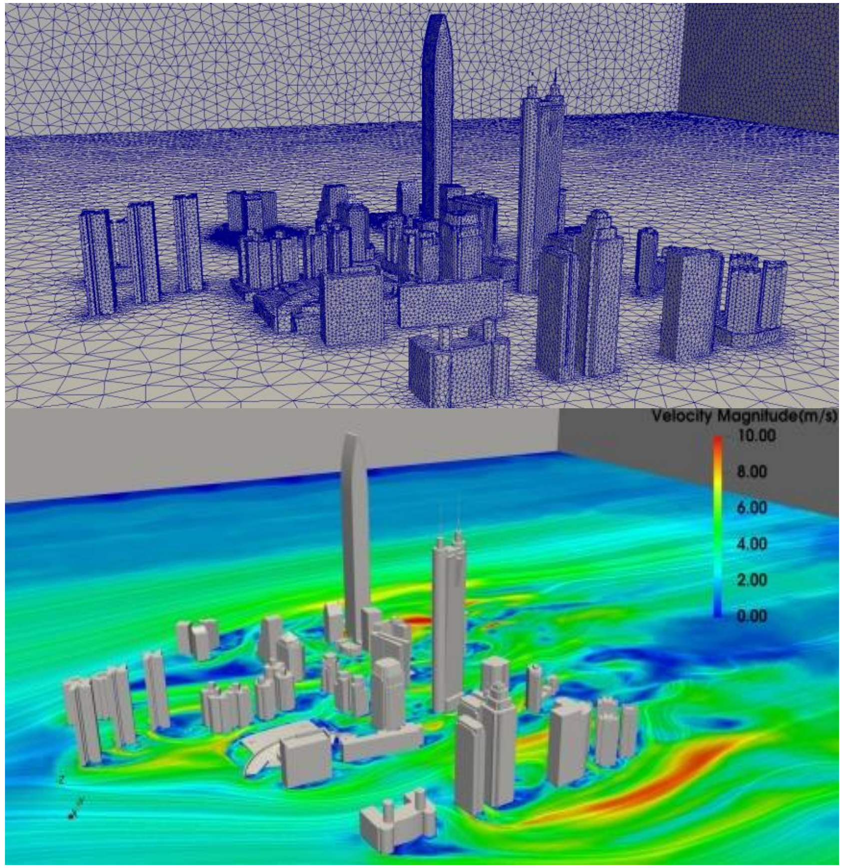
# Droplet Travelling in Air





The wind flow around the city affects the safety of buildings. The simulation of urban wind flow plays an important role in typhoon warning and pollution forecasting.

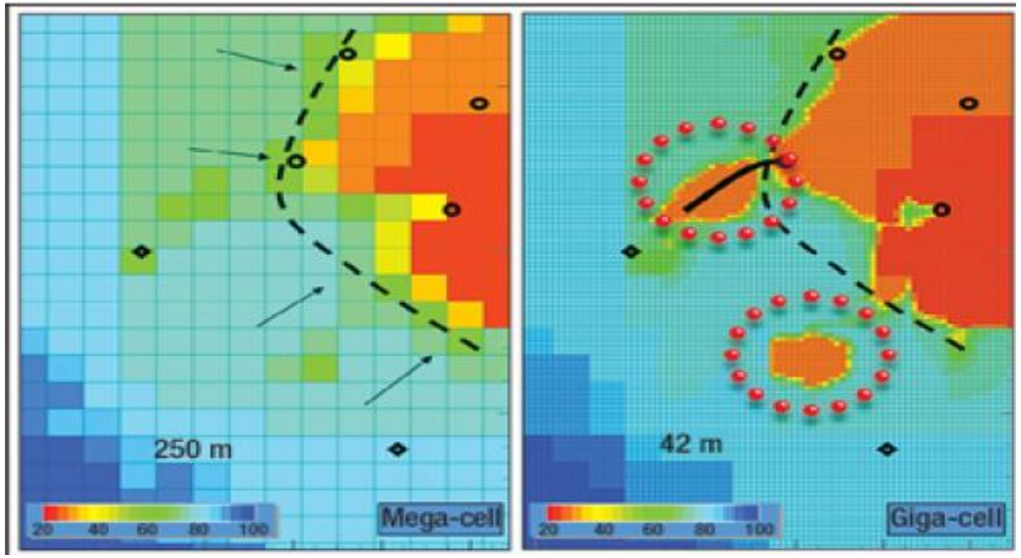


# Geophysical Flows

- Applications: prediction of groundwater contamination, protection of submarine tunnels/optical fiber cables

$$
\begin{aligned}
-\phi\frac{\partial s_w}{\partial t} - \nabla\cdot(\lambda_n \mathbf{K}(\nabla p_w + \nabla p_c - \rho_n g \nabla D)) &= q_n, && \text{in} \quad \Omega,\\
\phi\frac{\partial s_w}{\partial t} - \nabla\cdot(\lambda_w \mathbf{K}(\nabla p_w - \rho_w g \nabla D)) &= q_w, && \text{in} \quad \Omega.\\
\mathbf{u}_w\cdot\mathbf{n} = f_w^{in},\ \mathbf{u}_n\cdot\mathbf{n} &= f_n^{in}, && \text{on} \quad \Gamma_{in},\\
p_w = p_w^{out},\ \lambda_n\mathbf{K}\nabla p_c\cdot\mathbf{n} &= 0, && \text{on} \quad \Gamma_{out},\\
\mathbf{u}_w\cdot\mathbf{n} = 0,\ \mathbf{u}_n\cdot\mathbf{n} &= 0, && \text{on} \quad \Gamma_0,
\end{aligned}
$$

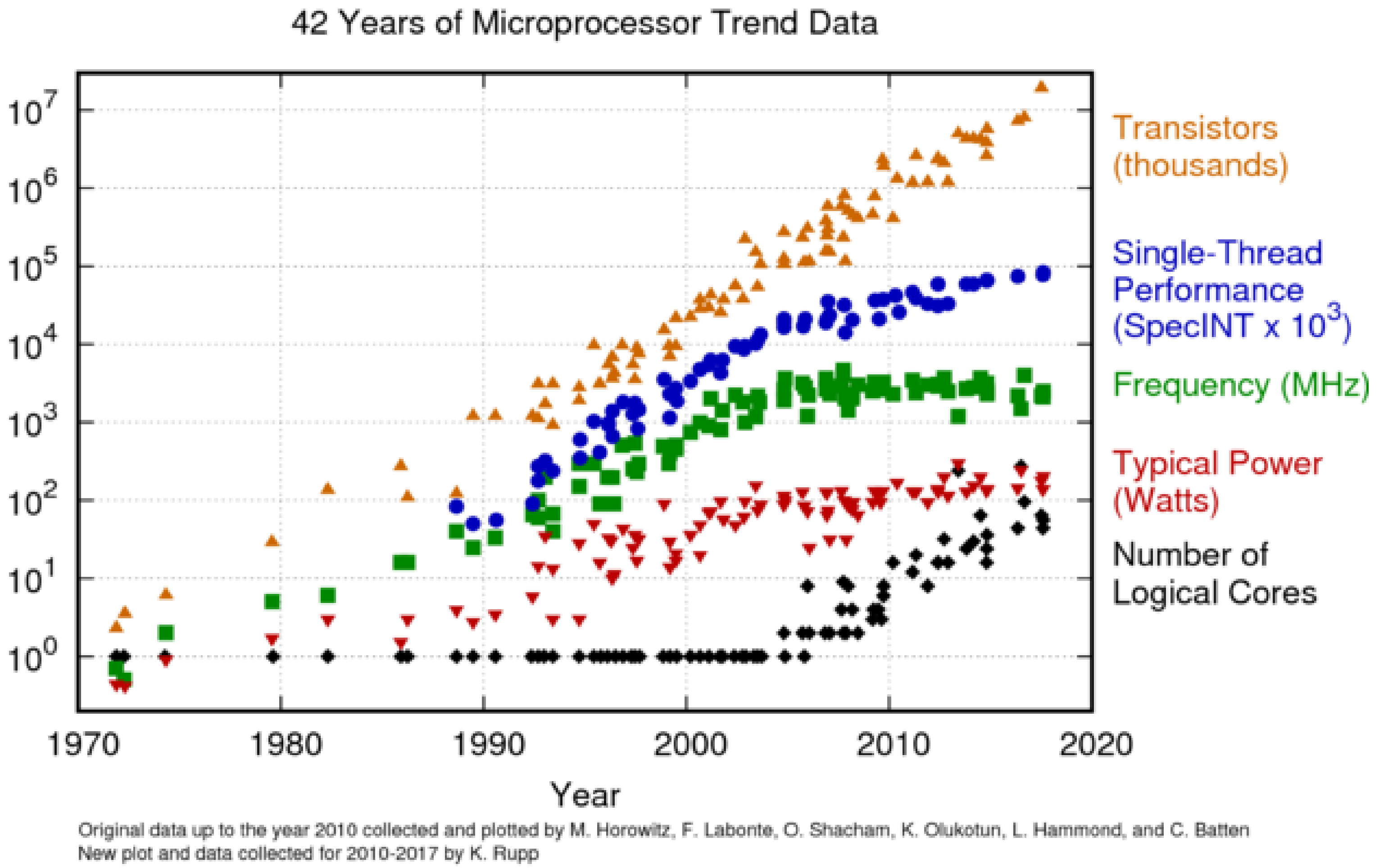



Ali H. Dogru, WorldOil 2010

Saudi Aramco أرامكو السعودية

**Submarine tunnel environment**

# Supercomputing is getting Moore and Moore hot here




https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/


## TIANHE-2A

Top 1st 06/2013

Cores: 4,981,760

Theor: 100,679 TF/s

Linpack: 61,444.5 TF/s


## SUNWAY TAIHULIGHT

Top 1st 06/2016

Cores: 10,649,600

Theor: 125,436 TF/s

Linpack: 93,014.6 TF/s

HPCG: 480.848 TF/s


## SUMMIT

Top 1st 06/2018

Cores: 2,414,592

Theor: 200,795 TF/s

Linpack: 148,600 TF/s

HPCG: 2,925.75 TF/s

## FUGAKU

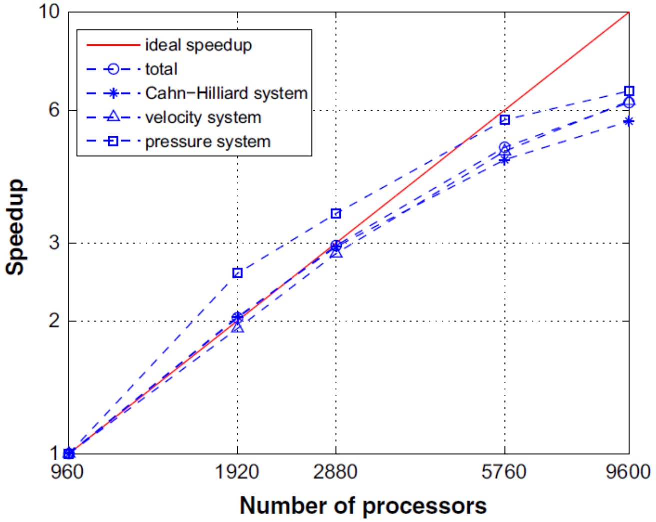
Top 1st 06/2020

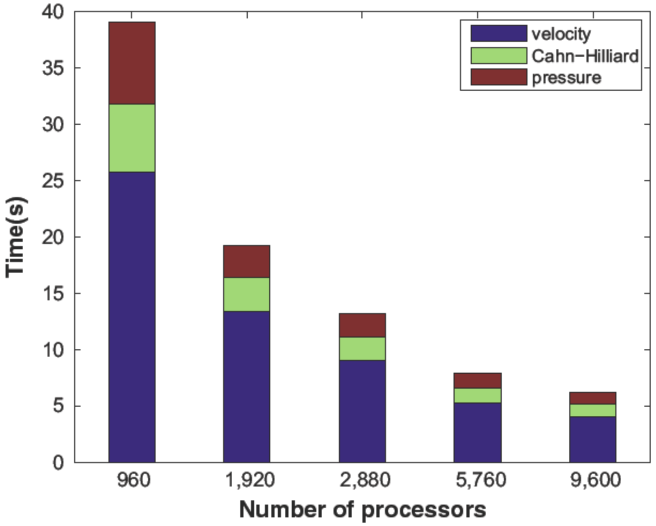
Cores: 7,630,848

Theor: 537,212 TF/s

Linpack: 442,010 TF/s

HPCG: 16,004.5 TF/s




澳大的第三代HPCC集合了超過2,000個計算核心、85臺服務器、34張圖形處理器，配備超大內存3TB的計算服務器

NEW!!  An upcoming new cluster provides 4096 cores...