

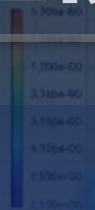
High-Performance Numerical Simulation for Flow in Porous Media

多孔介质渗流高性能数值模拟

偏微分方程与地球系统科学前沿讲习班

Chensong Zhang, AMSS

Version: 2026/04/24 西北天元中心



报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

Neural Network
Methods

08

软件项目

Software
Projects

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

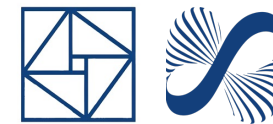
Neural Network
Methods

08

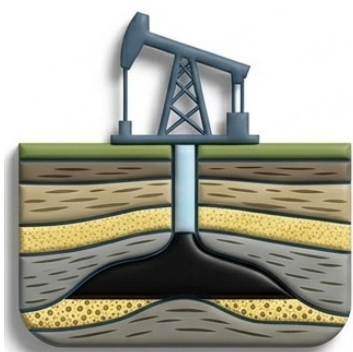
软件项目

Software
Projects

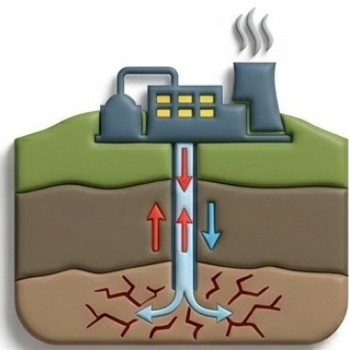
渗流力学问题的典型应用特征



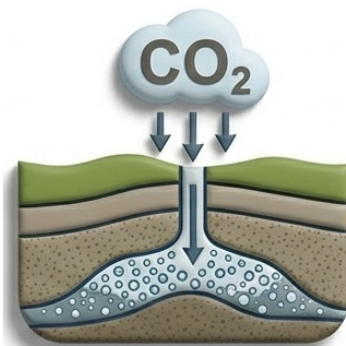
渗流力学应用范围广泛，凡涉及**流体在多孔介质中流动**的场景，都可以看到它的身影：



油气资源开发



地热资源利用



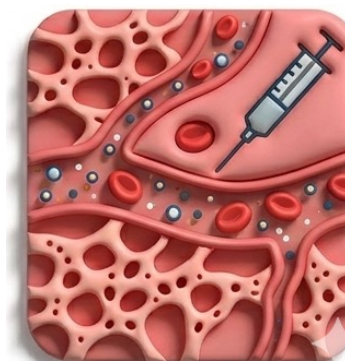
二氧化碳封存



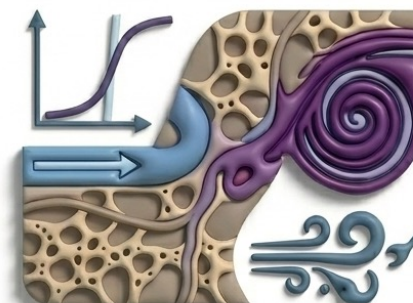
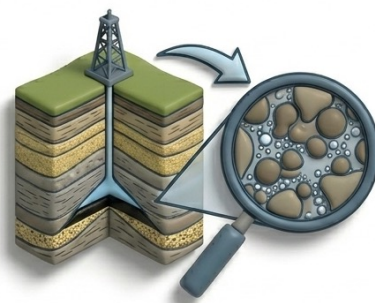
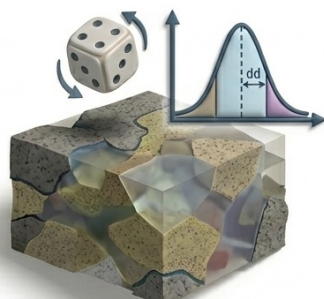
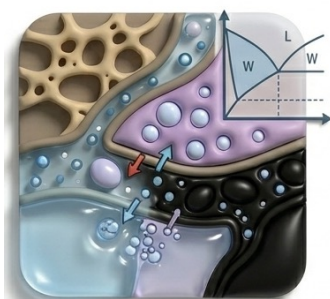
地下水污染治理



核废料封存

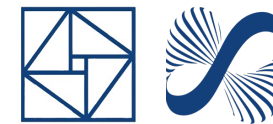


生物医学工程



应用特征：多组分多相渗流、极强的不确定性、跨多个时空尺度、非均质性和强非线性、多物理场耦合特性

一个典型的交叉学科问题



渗流力学

多组分多相流在多孔介质（孔、洞、裂隙）中的流动规律



岩石力学

岩石与流体的物理化学反应，孔隙变形、裂缝扩展等



油藏工程

油气藏开发工程、油井工程、水力压裂、生产机制等



计算科学

应用数学、计算数学、科学计算、并行计算等



软件工程

程序设计、开发、维护、标准，人机接口等



01



02



03

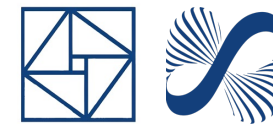


04

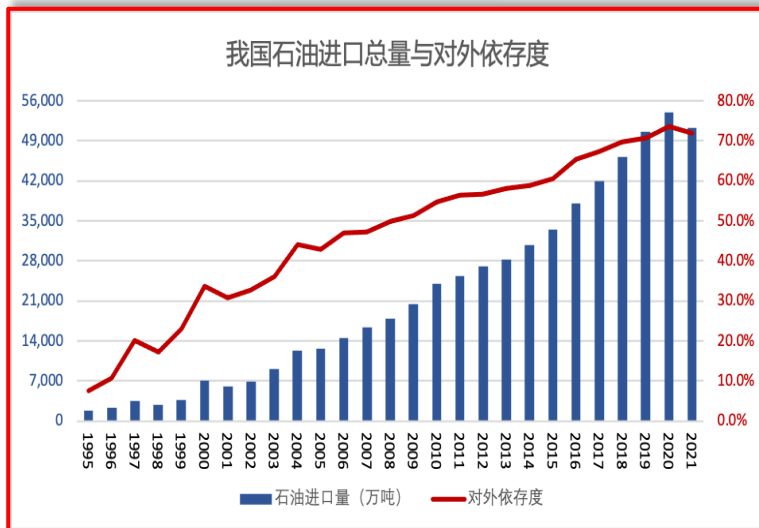


05

在能源与环境工程中的应用



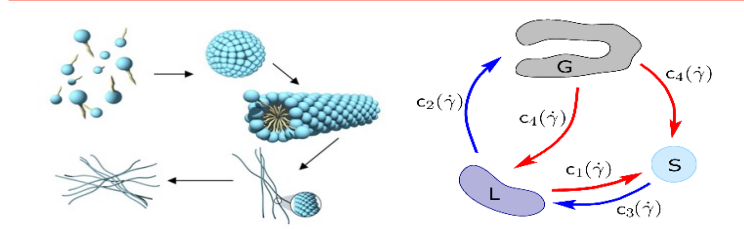
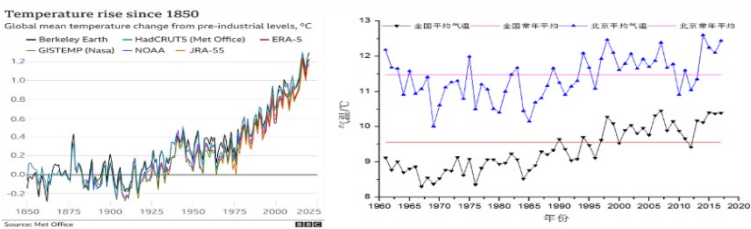
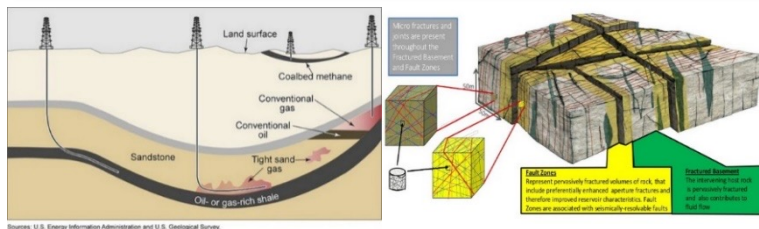
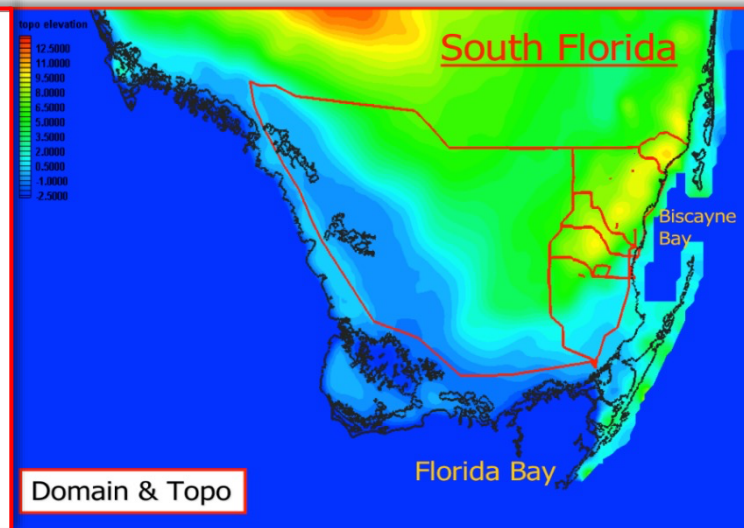
能源困局



双碳目标

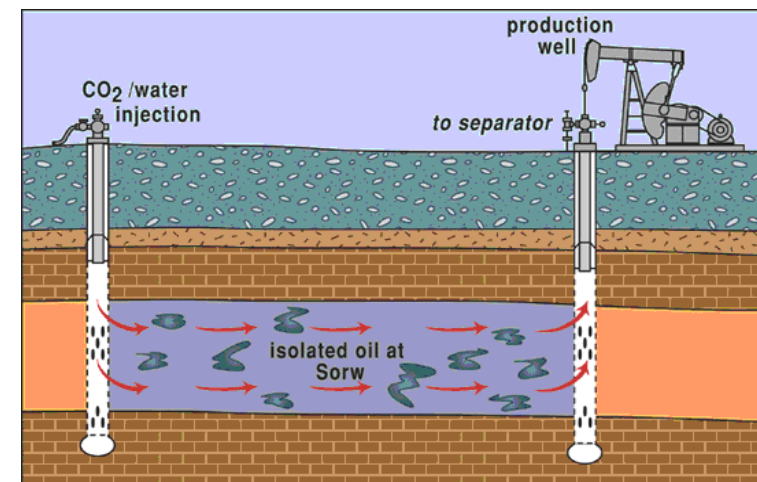
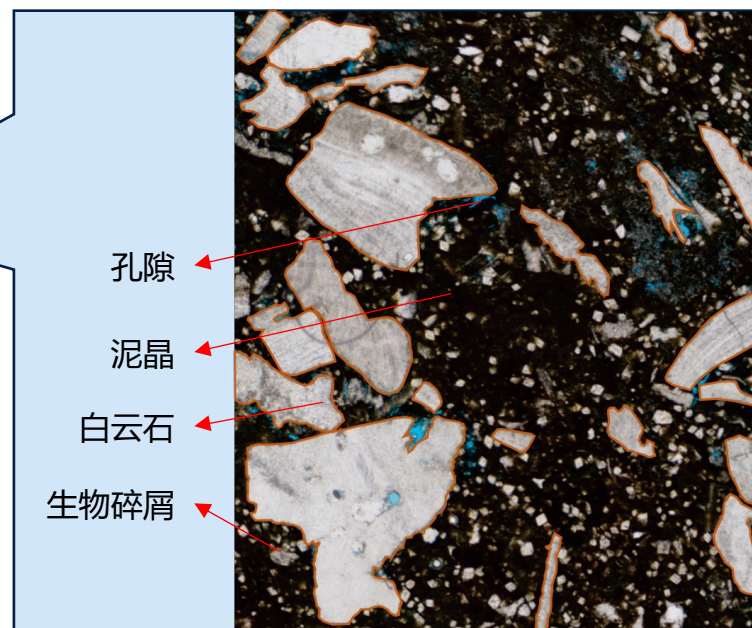
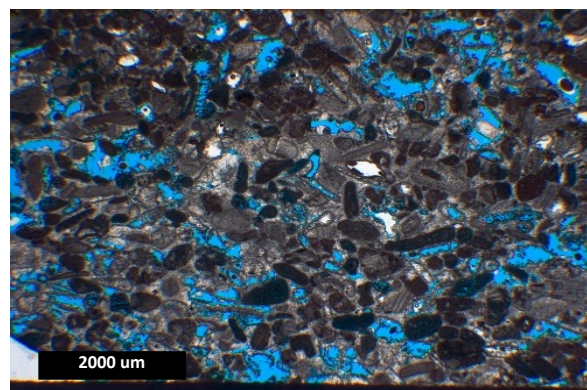
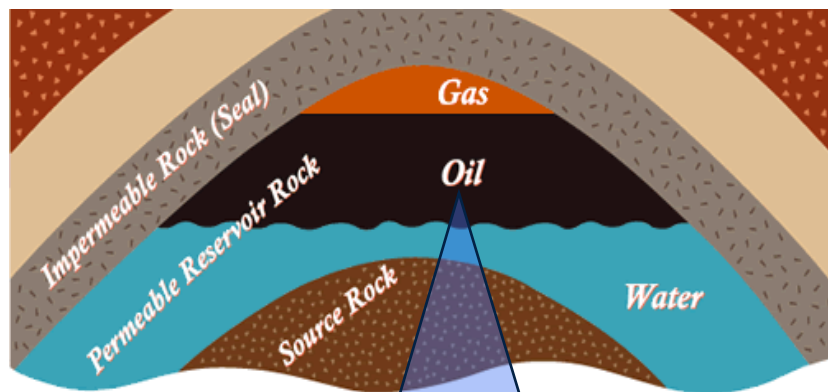
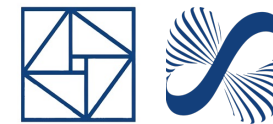


污染治理



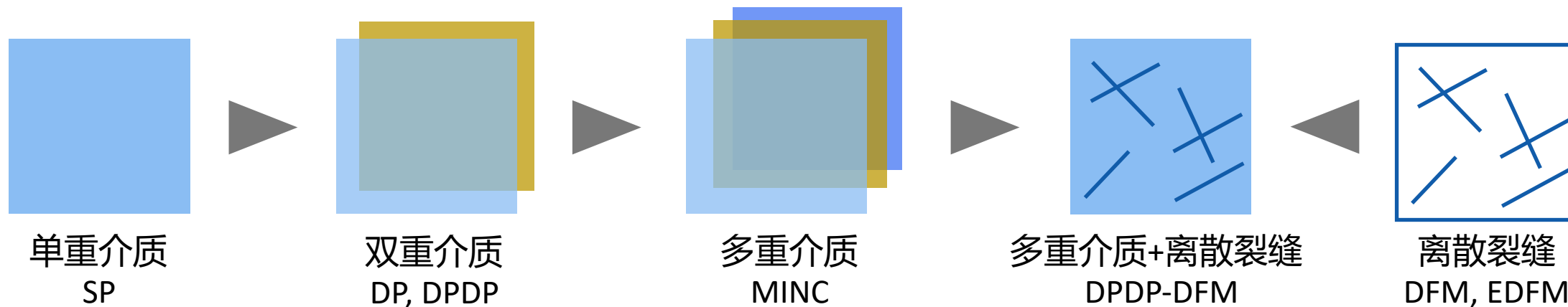
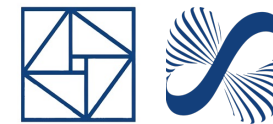
研究目标: 为相关应用场景, 提供一个开源、现代化、高效、并行模拟软件包

多孔介质中的多相多组分渗流



- **Component组分**: 混合物（比如溶液）中的各个物质成分
- **Phase相**: 在热力学系统中的一个区域，在该区域内，材料的所有物理性质基本上是均匀的
- **典型应用**: 油气资源开发、地热资源利用、生物医疗技术等

存在不同特性的介质类型



Easy

General

Limited

Low

Investigation Effort

Practical Applicability

Capability to handle heterogeneity

Computational Cost

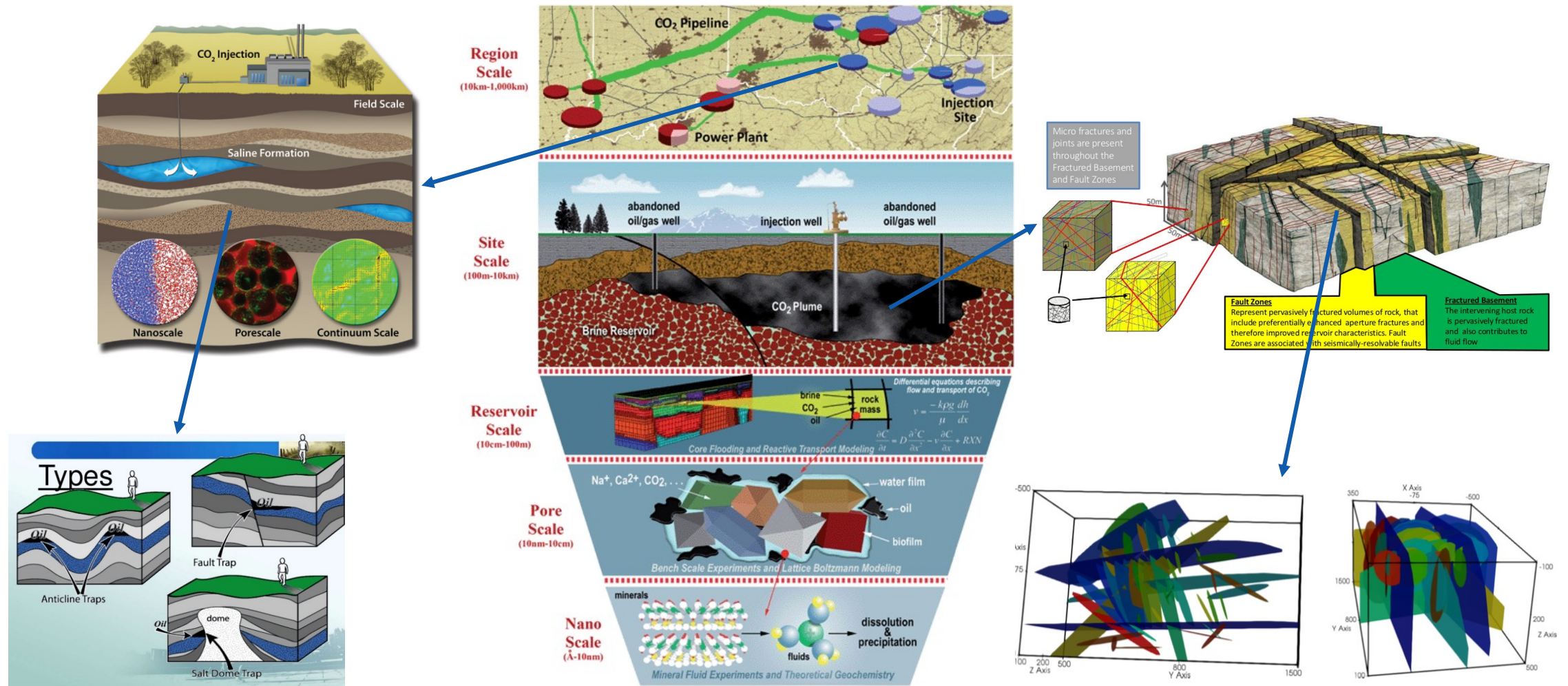
Difficult

Specific

Good

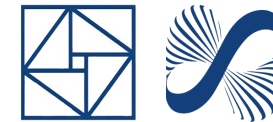
High

存在不同尺度的流动模态

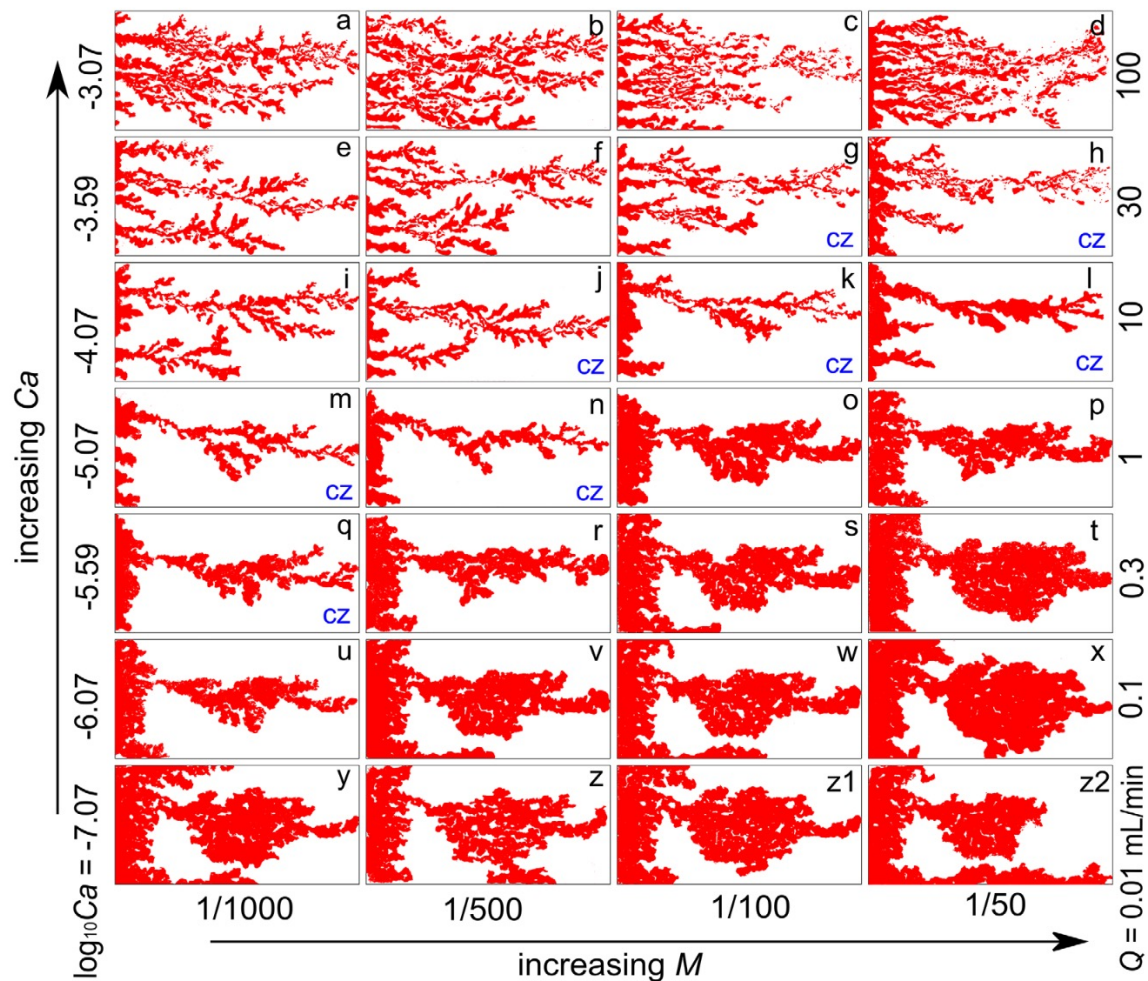
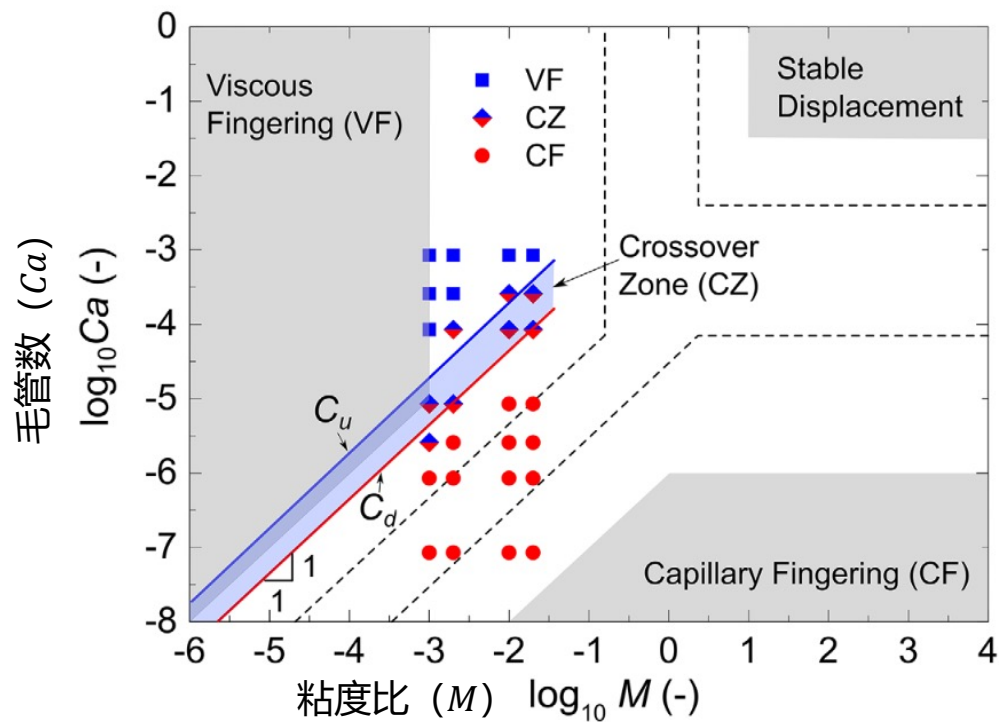


Center for Petroleum & Geosystems Eng, UT Austin

存在不同尺度的相间界面

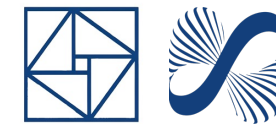


指进 (Fingering) 的形成与演化由毛细力和粘性力的关系主导, 裂隙/介质几何特征、流体物理性质和相界面特征等因素影响毛细力和粘性力大小, 进而调控指进

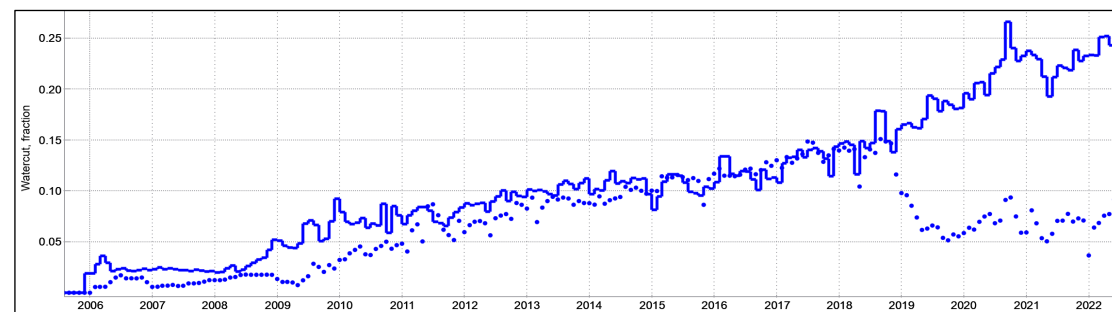
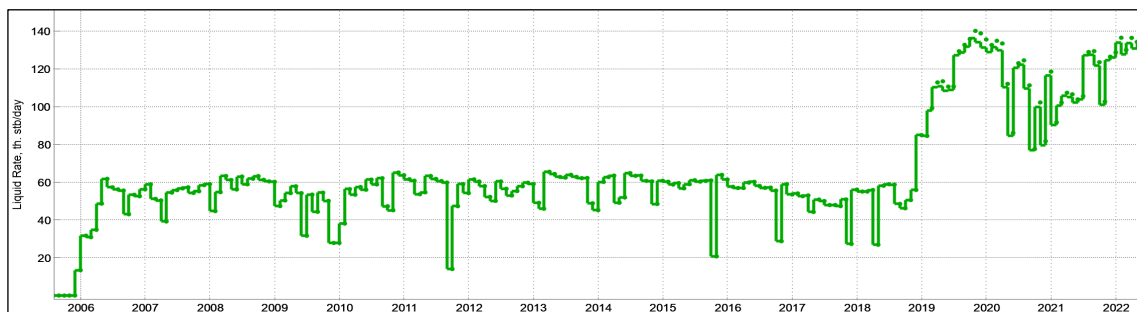
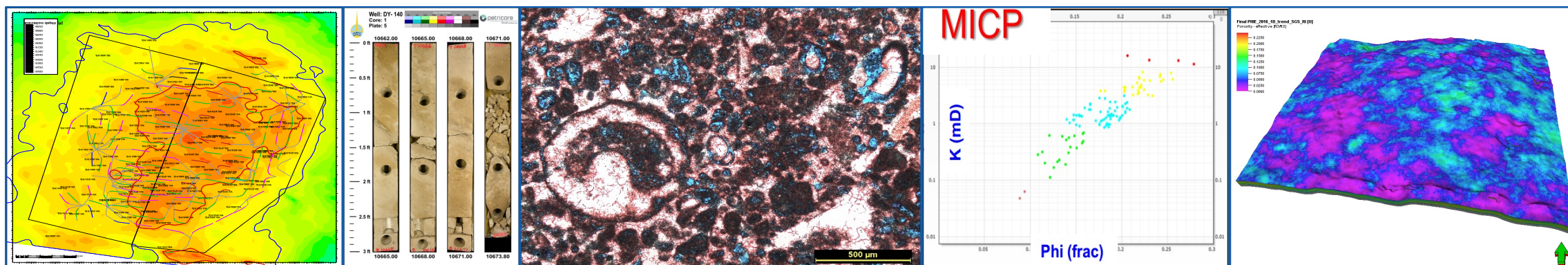


Chen, Y-F et al. Visualizing and quantifying the crossover from capillary fingering to viscous fingering in a rough fracture, *Water Resour Res*, 53, 2017

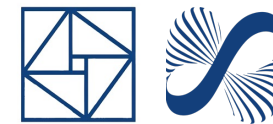
实际应用中的典型困难与挑战



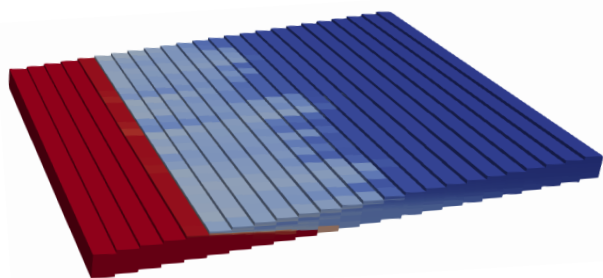
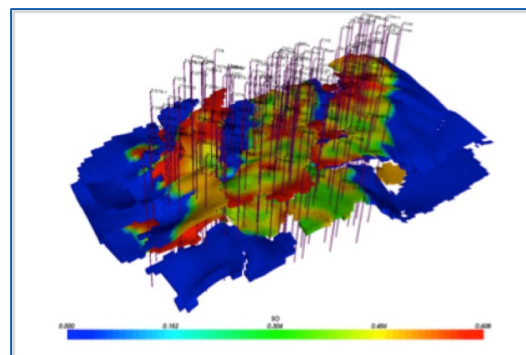
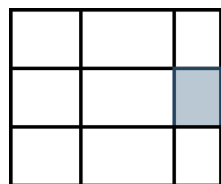
- 中东某碳酸盐岩油藏 (魏晨吉、韩如冰, CNPC & ADNOC) : 短轴背斜, 构造油藏, 饱和油藏 (发育气顶), 原油粘度低, 高孔低渗: 孔隙类型以粒间孔、粒内孔、粒间溶孔、粒内溶孔为主
- 动态模型对非均质性表征程度不理想, 导致常规模型适用性受限: 在定产的情况下, 全油藏气油比、含水率等关键参数拟合效果差, 井底流压数模计算值与实际测试值存在较大偏差



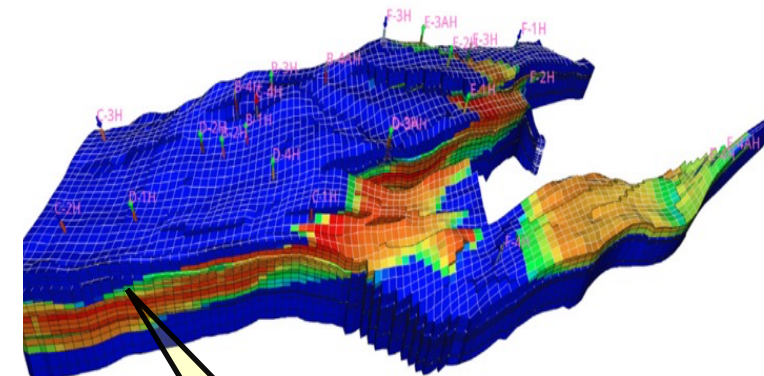
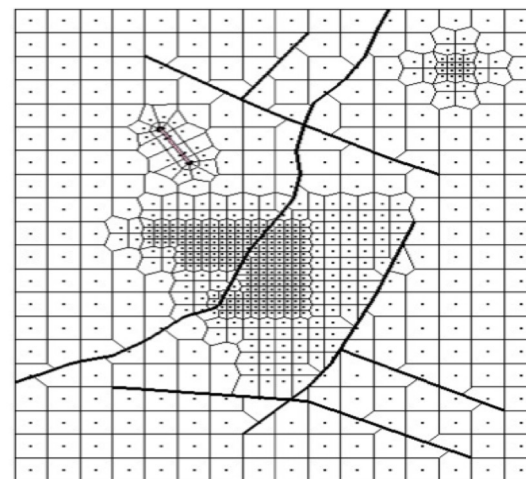
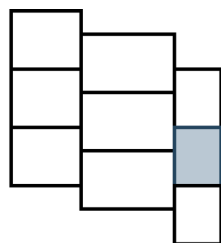
应对复杂几何的计算网格



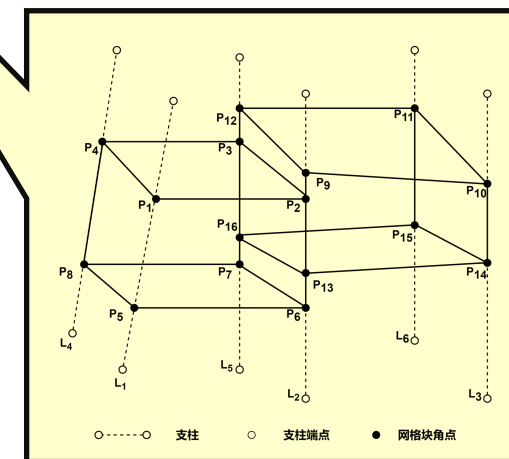
常规均匀正交网格



阶梯均匀正交网格

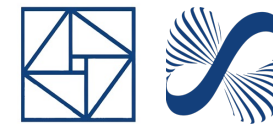


角点网格



- 实际应用中，网格较为复杂：结构网格、PEBI网格、无结构网格、局部加密网格、混合网格等
- 井方程的变量个数与油藏网格块对应的变量个数可能不一致（可能更多，也可能更少）

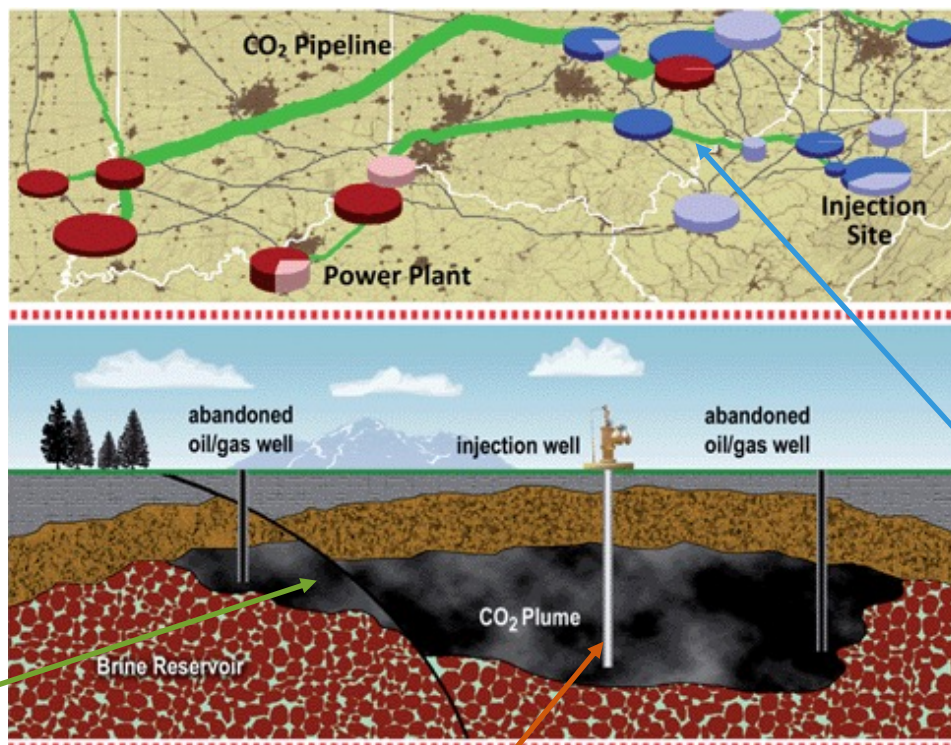
油气藏-井筒-井网全域流体计算



复杂多介质

多相平衡

多物理场



多部件耦合

多尺度现象

多物理场

油藏系统 (Reservoir)

复杂多孔介质中的渗流运动，达西或非达西渗流

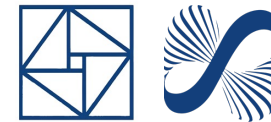
井筒系统 (Wellbore)

管流运动，考虑摩擦、重力、流型转变、热交换等

地面管网 (Surface Network)

集输管网中的流动，包括节流阀、分离器、泵、压缩机等设备

提高仿真工作的效能



百万规模黑油问题，夕发朝至，甚至朝发夕至
——曾经是油藏工程师的梦想

过去需要大规模集群，现在只用笔记本电脑
过去需要几天至几周，现在已在分、秒量级



- 百万量级及以下自由度的问题仍是目前工程中最常用需要求解的规模（中小规模）
- 传统的简化模型和算法已不能满足工程师的需要
- 更精细的模型、更复杂的物理、更快速地求解
- 更复杂的计算硬件环境（CPU、GPU、DCU）

满足高含水、高采出程度的双高油田的开发需求

使用化学驱替剂，如聚合物、表面活性剂等

多种流体物质组分之间，及流体与岩石间的反应

地层中存在裂缝、溶洞、断层、尖灭、高低渗透带

深层及超深层储层，需要考虑地层骨架变形

储层地质情况越发复杂，非均质性强，不确定性强

油气开发越发趋向于强耦合的复杂多物理过程

随着AI4SE的发展，仿真工具的迭代速度迅速提升，曾制约发展的Legacy Code不再是障碍

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

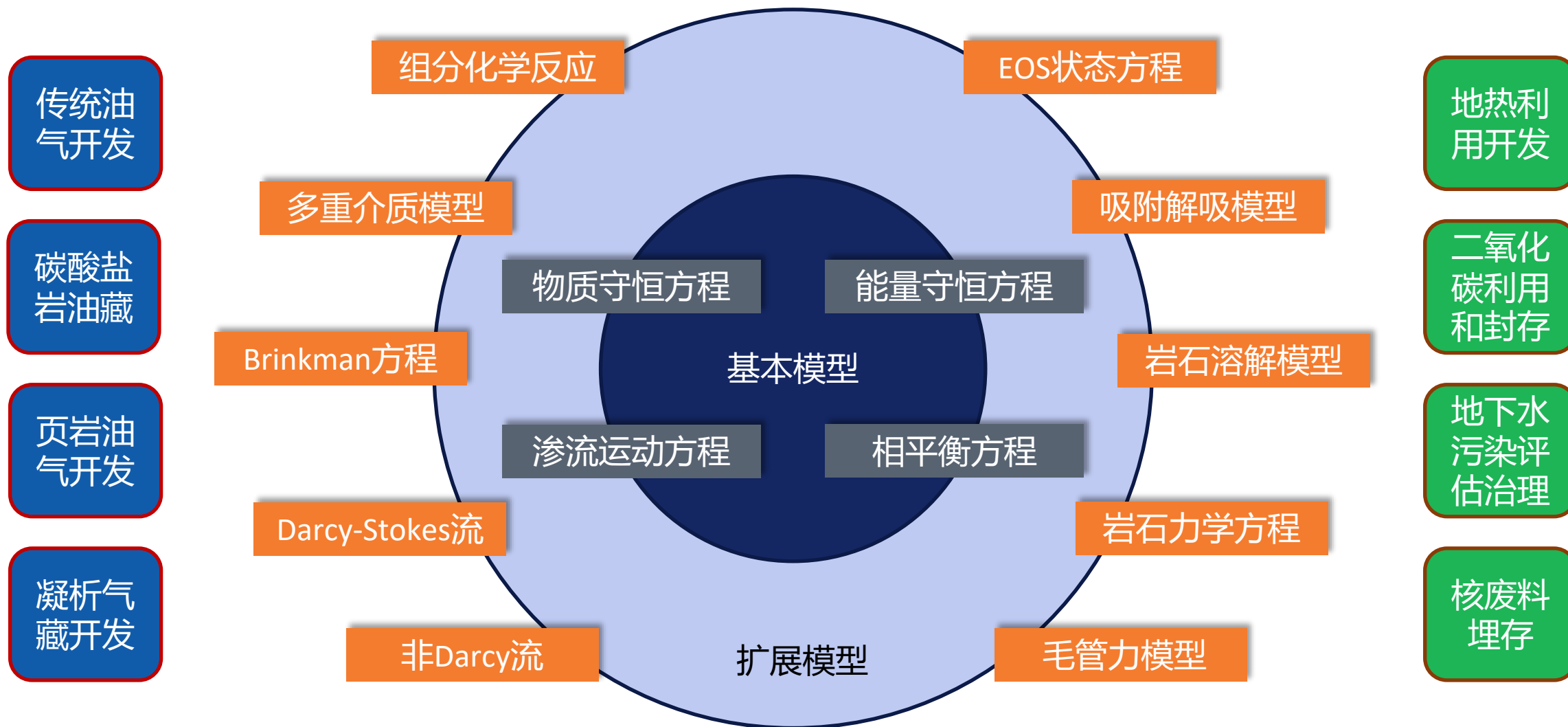
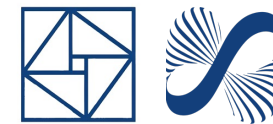
Neural Network
Methods

08

软件项目

Software
Projects

多相多组分渗流力学模型



等温多组分模型框架



$$\frac{\partial}{\partial t} \left(\phi \sum_{j=1}^{n_p} x_{ij} \xi_j S_j \right) + \nabla \cdot \mathbf{F}_i - \sum_{j=1}^{n_p} S_j r_{ij} = Q_i, \quad i = 1 : n_c \quad \text{物质守恒}$$

$$\mathbf{F}_i = \sum_{j=1}^{n_p} \left(x_{ij} \xi_j \mathbf{u}_j - S_j \mathbf{D}_j \nabla (\xi_j x_{ij}) \right), \quad i = 1 : n_c \quad \text{流量方程}$$

$$\mathbf{u}_j = -\frac{\kappa \kappa_{rj}}{\mu_j} (\nabla P_j - \gamma_j \nabla z), \quad j = 1 : n_p \quad \text{Darcy定律}$$

$$\sum_{j=1}^{n_p} S_j = 1, \quad \text{饱和度关系}$$

$$\sum_{i=1}^{n_c} x_{ij} = 1, \quad j = 1 : n_p \quad \text{物料守恒}$$

$$P_1 - P_j = P_{c1j}, \quad j = 2 : n_p \quad \text{表面张力}$$

$$f_{ij} = f_{i1}, \quad i = 1 : n_c, j = 2 : n_p \quad \text{相平衡方程}$$

模型：为工程和科学研究提供更丰富、更精确的物理模型选择

多重介质模型

EOS状态方程

Brinkman方程

化学反应方程

Navier-Stokes方程

非等温模型

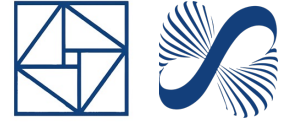
非Darcy模型

吸附解吸模型

岩石溶解模型

岩石力学方程

非等温模型的能量方程



$$\frac{\partial}{\partial t} \left(\phi \sum_{j=1}^{n_p} \xi_j S_j U_j + (1 - \phi) U_r \right) + \nabla \cdot \sum_{j=1}^{n_p} \xi_j \mathbf{u}_j H_j - \nabla \cdot (\kappa_T \nabla T) = \sum_{j=1}^{n_p} q_{j,well} H_j - Q_{loss}, \quad \text{能量守恒}$$

$$U_j = H_j - P/\xi_j, \quad j = 1 : n_p \quad \text{相内能}$$

$$U_r = C_r(T - T_{ref}), \quad \text{岩石内能}$$

$$\kappa_T = \phi \sum_{j=1}^{n_p} S_j \kappa_j + (1 - \phi) \kappa_r, \quad \text{导热率}$$

计算焓的方法

$$H_{g,i} = \int_{T_{ref}}^T (cpg_{1i} + cpg_{2i}t + cpg_{3i}t^2 + cpg_{4i}t^3 + cpg_{5i}t^4) dt,$$

$$H_{v,i} = \begin{cases} hvr_i(T_{crit,i} - T)^{ev_i}, & T < T_{crit,i} \\ 0, & T \geq T_{crit,i} \end{cases}$$

$$H_l = \sum_i^{n_c} x_{il}(H_{g,i} - H_{v,i}), \quad H_g = \sum_i^{n_c} x_{ig}H_{g,i}$$

Gas-based

$$H_{l,i} = \int_{T_{ref}}^T (cpl_{1i} + cpl_{2i}t + cpl_{3i}t^2 + cpl_{4i}t^3 + cpl_{5i}t^4) dt,$$

$$H_l = \sum_{i=1}^{n_c} x_{il}H_{l,i}, \quad H_g = \sum_{i=1}^{n_c} x_{ig}(H_{g,i} + H_{v,i}),$$

Liquid-based

$$H_{l,i} = \int_{T_{ref}}^T (cpl_{1i} + cpl_{2i}t + cpl_{3i}t^2 + cpl_{4i}t^3 + cpl_{5i}t^4) dt,$$

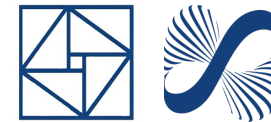
$$H_{g,i} = \int_{T_{ref}}^T (cpg_{1i} + cpg_{2i}t + cpg_{3i}t^2 + cpg_{4i}t^3 + cpg_{5i}t^4) dt,$$

$$H_g = \sum_{i=1}^{n_c} x_{ig}(H_{g,i} + hvapr),$$

$$H_l = \sum_{i=1}^{n_c} x_{il}H_{l,i},$$

Simple Hvap

经典三相黑油 (Black Oil) 模型



可利用下列关系式来给出组分模型框架下的黑油模型，并在统一的程序框架内对黑油和组分模型进行模拟！

$$\begin{array}{c}
 Ww \\
 Oo \\
 Gg \\
 Go
 \end{array}
 \begin{array}{c}
 O \quad G \quad W \\
 \left(\begin{array}{ccc}
 0 & 0 & 1 \\
 1 & 0 & 0 \\
 0 & 1 & 0 \\
 0 & 1 & 0
 \end{array} \right)
 \end{array}
 \begin{array}{c}
 \xi_w \\
 \xi_g \\
 \xi_o
 \end{array}
 =
 \begin{array}{c}
 \frac{1}{B_w} \\
 \frac{1}{B_g} \\
 \frac{1 + R_s}{B_o}
 \end{array}$$

$$\begin{bmatrix}
 x_{Oo} & x_{Go} & x_{Wo} \\
 x_{Og} & x_{Gg} & x_{Wg} \\
 x_{Ow} & x_{Gw} & x_{Ww}
 \end{bmatrix}
 =
 \begin{bmatrix}
 \frac{1}{1+R_s} & \frac{R_s}{1+R_s} & 0 \\
 0 & 1 & 0 \\
 0 & 0 & 1
 \end{bmatrix}$$

数学性质: Trangenstein and Bell, SIAP 1989; SISC 1989

● 物质守恒

$$\frac{\partial}{\partial t} \left(\phi \frac{S_w}{B_w} \right) = -\nabla \cdot \left(\frac{1}{B_w} \mathbf{u}_w \right) + \tilde{q}_w$$

$$\frac{\partial}{\partial t} \left(\phi \frac{S_o}{B_o} \right) = -\nabla \cdot \left(\frac{1}{B_o} \mathbf{u}_o \right) + \tilde{q}_o$$

$$\frac{\partial}{\partial t} \left(\phi \frac{S_g}{B_g} + \phi \frac{R_s S_o}{B_o} \right) = -\nabla \cdot \left(\frac{1}{B_g} \mathbf{u}_g + \frac{R_s}{B_o} \mathbf{u}_o \right) + \tilde{q}_G$$

● Darcy定律

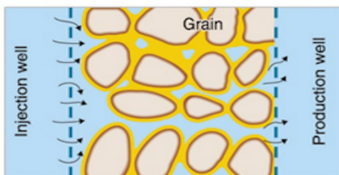
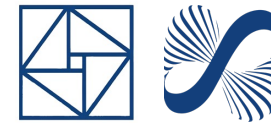
$$\mathbf{u}_j = -\frac{\kappa \kappa_{rj}}{\mu_j} \left(\nabla P_j - \gamma_j \nabla z \right), \quad j = o, w, g$$

● 其它关系式

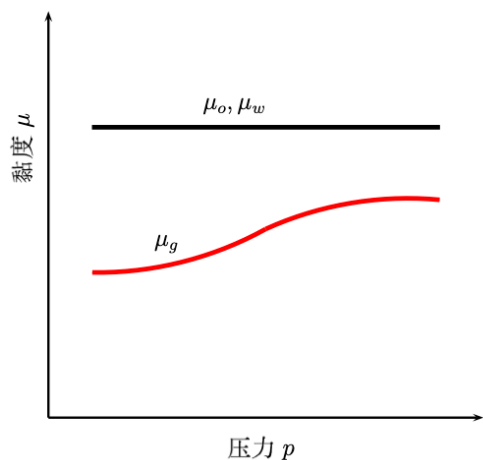
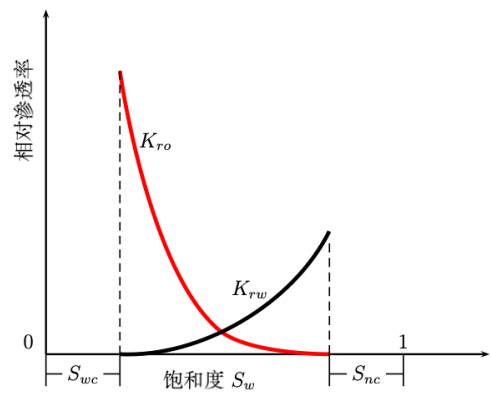
$$S_o + S_g + S_w = 1$$

$$P_{cog} = P_o - P_g \quad P_{cow} = P_o - P_w$$

简化的油水两相模型



基本假设：油水两相，温度固定，流体不可压缩，且黏度与压力无关，多孔介质刚性，忽略毛管力和重力，相对渗透率仅依赖水相饱和度，源汇项已知



● 物质守恒

$$\phi \frac{\partial}{\partial t} S_w + \nabla \cdot \mathbf{u}_w = q_w$$

$$\phi \frac{\partial}{\partial t} S_o + \nabla \cdot \mathbf{u}_o = q_o$$

● Darcy定律

$$\mathbf{u}_j = - \frac{\kappa \kappa_{rj}(S_w)}{\mu_j} \nabla P, \quad j = o, w$$

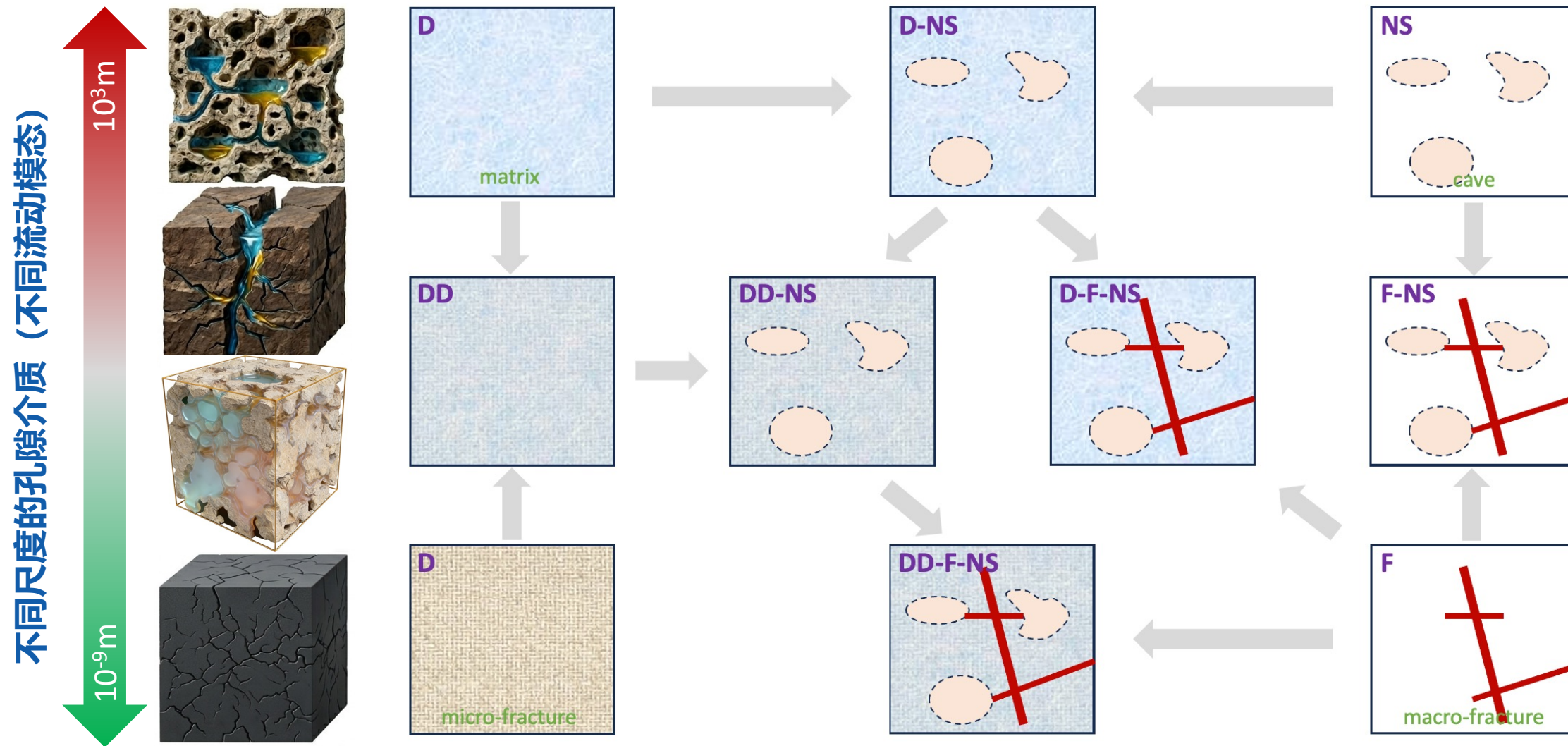
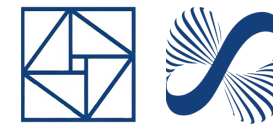
● 其它关系式

$$S_w + S_o = 1$$

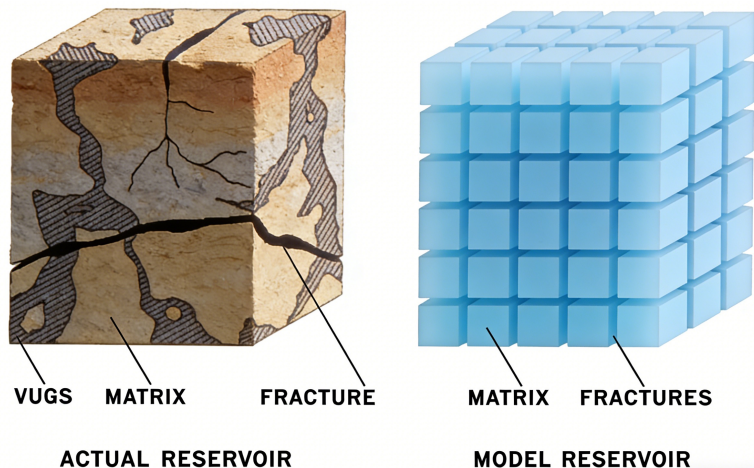
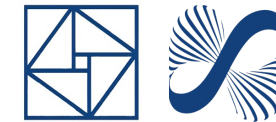


吴淑红、张晨松等，油藏数值模拟中的线性代数求解方法，2020

复杂介质精细模拟面临更多挑战



双重介质的DPDP模型

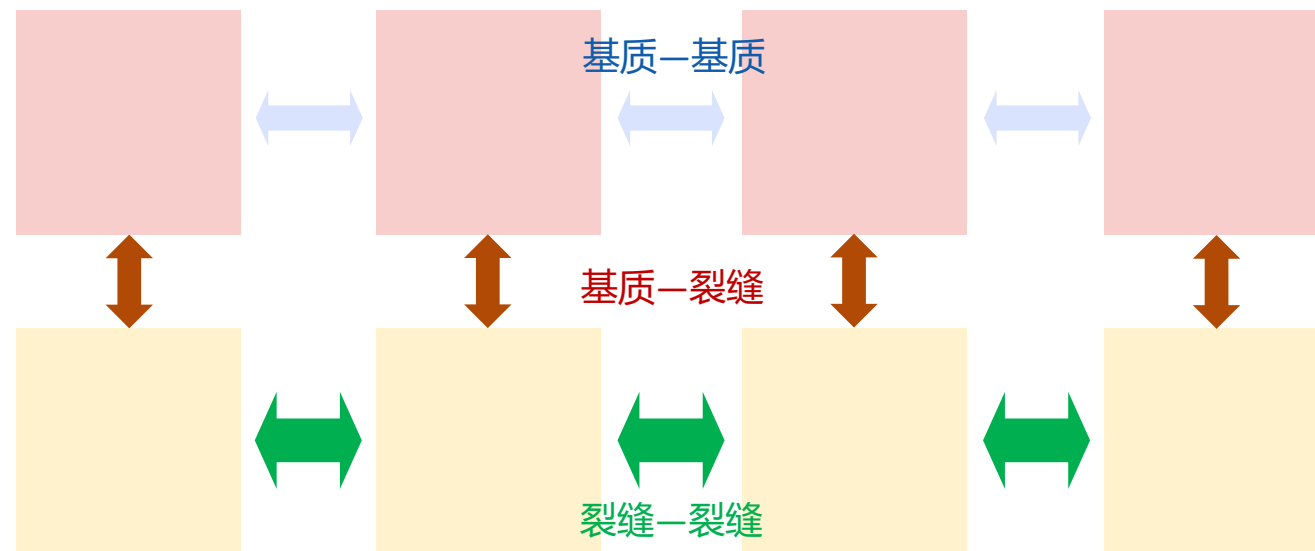
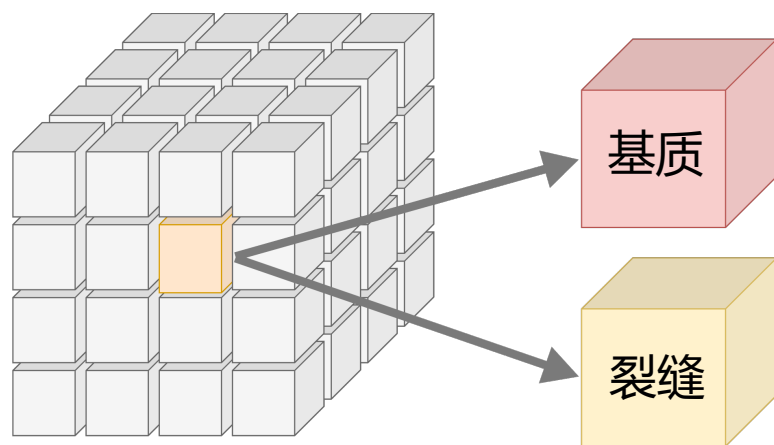


单相DPDP模型:

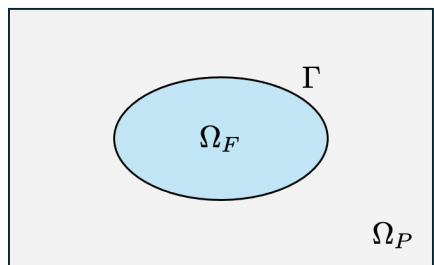
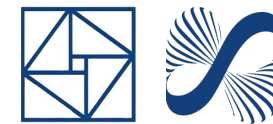
$$\phi_m C_{mt} \frac{\partial p_m}{\partial t} - \nabla \cdot \frac{\kappa_m}{\mu} \nabla p_m = -Q$$

$$\phi_f C_{ft} \frac{\partial p_f}{\partial t} - \nabla \cdot \frac{\kappa_f}{\mu} \nabla p_f = Q + q_p$$

ϕ_m/ϕ_f : porosities
 C_{mt}/C_{ft} : compressibilities
 p_m/p_f : pressures
 κ_m/κ_f : permeabilities
 μ : dynamic viscosities
 q_p : sink/source term
 $Q = \frac{\sigma\kappa}{\mu} (p_m - p_f)$: mass exchange
 σ : shape factor



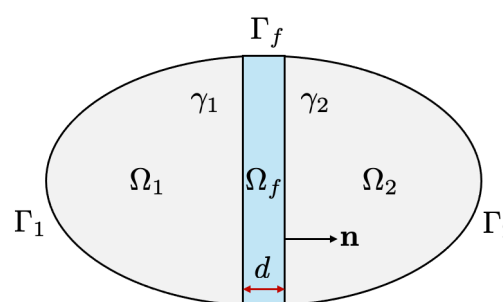
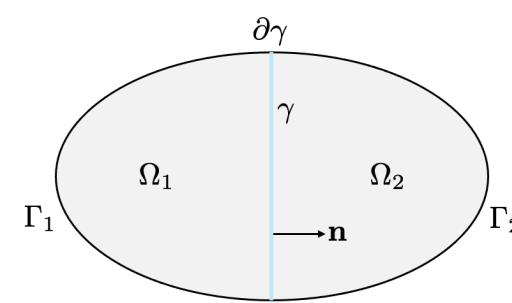
裂缝与溶洞等介质的数学模型



$$\left\{ \begin{array}{l} \mathbf{u} = -\kappa \nabla p_P, \quad \Omega_P, \\ -\mu \Delta \mathbf{u} + \nabla p_F = \mathbf{f}, \quad \Omega_F, \\ \nabla \cdot \mathbf{u} = g, \quad \Omega, \\ \mathbf{u}_F \cdot \mathbf{n} = \mathbf{u}_P \cdot \mathbf{n}, \quad \Gamma, \text{ mass conservation} \\ p_F - 2\mu(\nabla \mathbf{u}_F \mathbf{n}) \cdot \mathbf{n} = p_P, \quad \Gamma, \text{ normal stress} \\ \mathbf{u}_F \cdot \mathbf{t} = -2\beta\mu(\nabla \mathbf{u}_F \mathbf{n}) \cdot \mathbf{t}, \quad \Gamma, \text{ BJS} \\ \mathbf{u} \cdot \mathbf{n} = 0, \quad \partial\Omega. \end{array} \right.$$

$$\left\{ \begin{array}{l} -\nabla \cdot (\mu(x)\nabla \mathbf{u}) + \kappa^{-1} \mathbf{u} + \nabla p = \mathbf{f}, \quad \Omega, \\ \nabla \cdot \mathbf{u} = g, \quad \Omega, \\ \mathbf{u} = \mathbf{0}, \quad \partial\Omega. \end{array} \right.$$

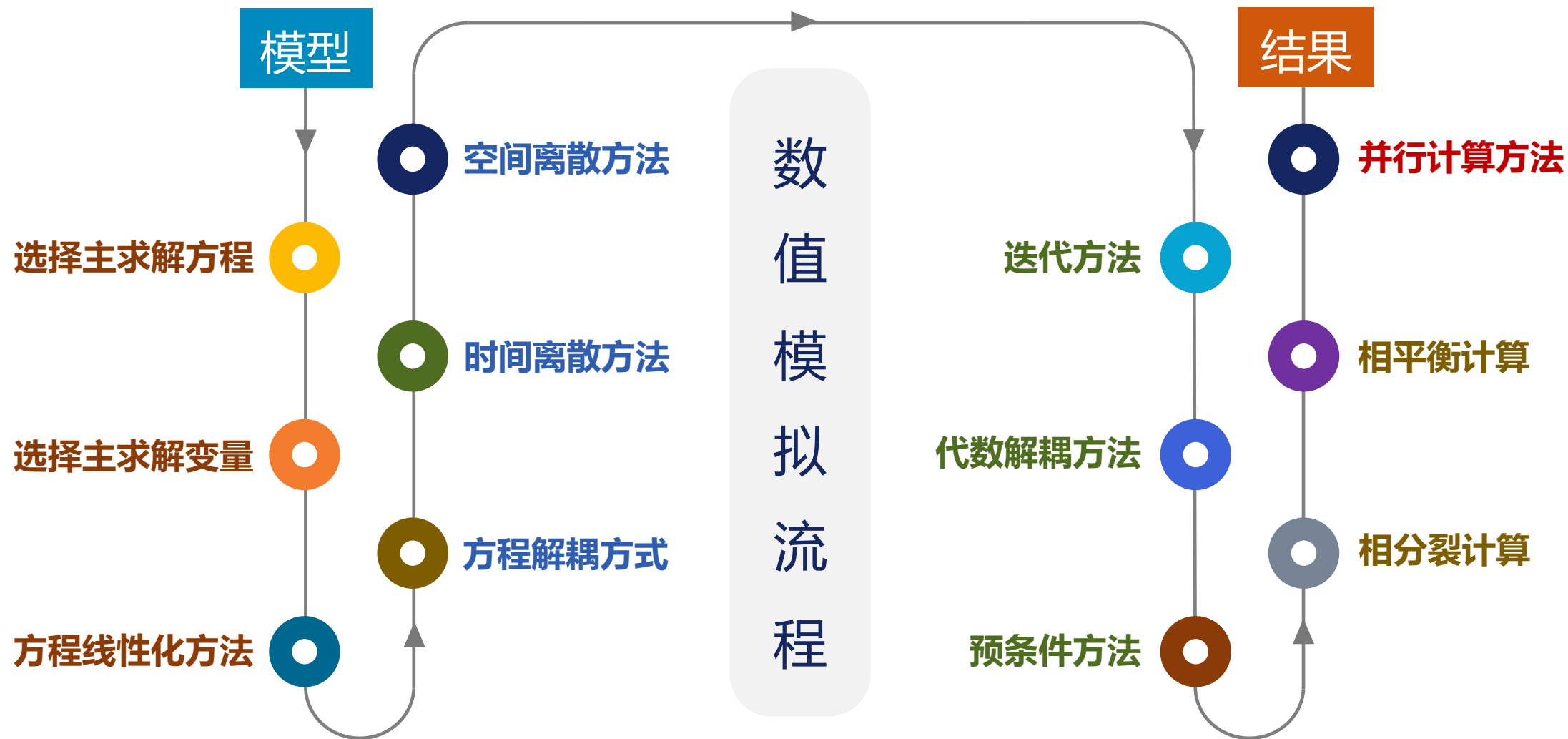
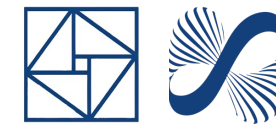
Brinkman



$$\left\{ \begin{array}{l} \mathbf{u}_i = -\kappa_i \nabla p_i, \quad \Omega_i, \quad i = 1, 2, f, \\ \nabla \cdot \mathbf{u}_i = f_i, \quad \Omega_i, \quad i = 1, 2, f, \\ \mathbf{u}_1 \cdot \mathbf{n} = \mathbf{u}_f \cdot \mathbf{n}, \quad \gamma_i, \quad i = 1, 2, \\ p_i = p_f, \quad \gamma_i, \quad i = 1, 2, \\ p_i = p_D, \quad \Gamma_i, \quad i = 1, 2, f. \end{array} \right. \left\{ \begin{array}{l} \mathbf{u}_i = -\kappa_i \nabla p_i, \quad \Omega_i, \quad i = 1, 2 \\ \nabla \cdot \mathbf{u}_i = f_i, \quad \Omega_i, \quad i = 1, 2 \\ \mathbf{u}_f = -\kappa_{f,\tau} \mathbf{d} \nabla_\tau p_f, \quad \gamma, \\ \nabla_\tau \cdot \mathbf{u}_f = f_f + (\mathbf{u}_1 \cdot \mathbf{n}_1 + \mathbf{u}_2 \cdot \mathbf{n}_2)|_\gamma, \quad \gamma, \\ -\xi \mathbf{u}_1 \cdot \mathbf{n}_1 + \alpha_f p_1 = \alpha_f p_f - (1 - \xi) \mathbf{u}_2 \cdot \mathbf{n}_2, \quad \gamma, \\ -\xi \mathbf{u}_2 \cdot \mathbf{n}_2 + \alpha_f p_2 = \alpha_f p_f - (1 - \xi) \mathbf{u}_1 \cdot \mathbf{n}_1, \quad \gamma, \end{array} \right.$$

Discrete Fracture Model

多组分渗流数值模拟方法



报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

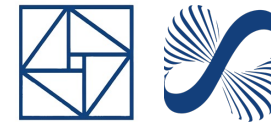
Neural Network
Methods

08

软件项目

Software
Projects

常用离散算法框架简介



- **半隐式方法**: 部分变量 (压力) 隐式求解, 剩余变量显式更新

[Ács et al., 1985; Fernandes et al., 2014; de Araujo Cavalcante Filho et al., 2015]

稳健性较差

时间步长很小

- **顺序隐式方法**: 对变量分批迭代隐式求解

[Watts, 1986; Lu et al., 2007; Moncorgé et al., 2018; Jiang et al., 2021]

稳健性适中

时间步长较小

- **全隐式方法**: 对所有变量隐式求解

[Coats, 1980; Qiao, 2015; Rasmussen et al., 2021; Schlumberger, 2022]

稳健性较强

离散问题难求解

- **自适应隐式方法**: 自适应地对网格单元隐式或显式求解

[Collins et al., 1992; CMG, 2015; Batista Fernandes et al., 2023]

稳健性良好

实现难度更高

目标: 设计统一的自适应求解框架, 在求解过程中可以自适应选择求解方法, 发挥方法各自的优势

油水两相模型的IMPES方法



核心思想： 隐式求解压力，显式推进饱和度，相对渗透率和分流函数等均用上一步的饱和度计算

● 定义总Darcy速度

$$\mathbf{u}_t := \mathbf{u}_w + \mathbf{u}_o$$

$$q_t := q_w + q_o$$

● 定义分流函数

$$\lambda_t := \frac{\kappa_{rw}}{\mu_w} + \frac{\kappa_{ro}}{\mu_o}$$

$$f_w(S_w) := \frac{\lambda_w}{\lambda_t}$$

● Darcy定律相加

$$\mathbf{u}_t = -\kappa \lambda_t(S_w) \nabla p$$

● 质量守恒方程相加

$$\phi \frac{\partial}{\partial t} (S_w + S_o) + \nabla \cdot \mathbf{u}_t = q_t$$

$$-\nabla \cdot (\kappa \lambda_t(S_w) \nabla p) = q_t$$

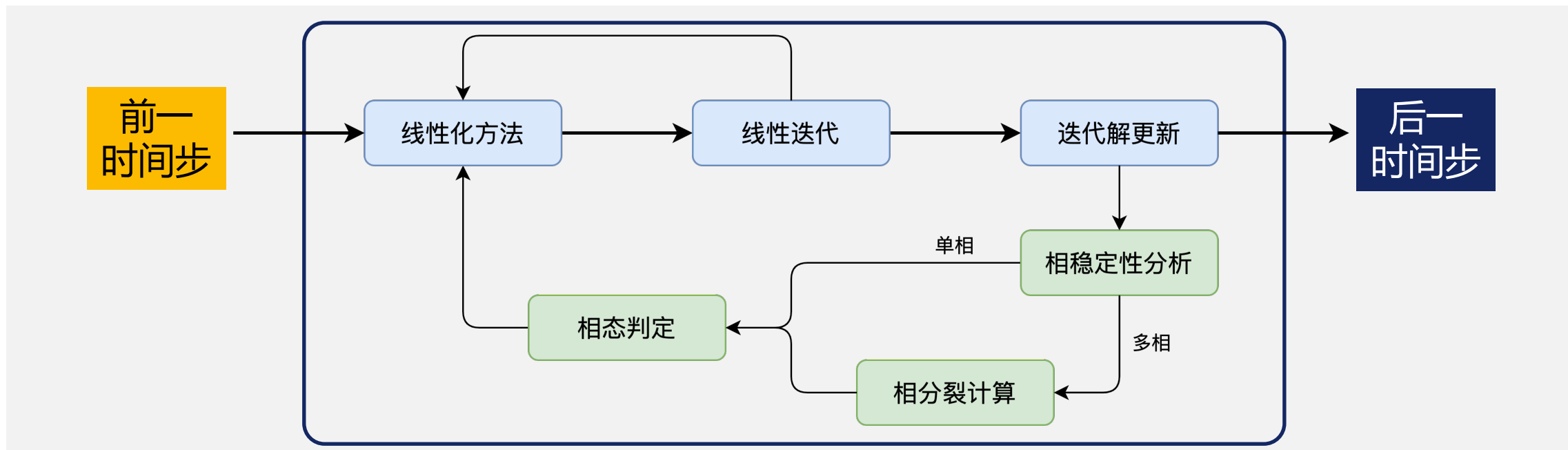
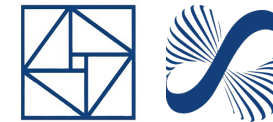
$$-\nabla \cdot (\kappa \lambda_t(S_w^n) \nabla p^{n+1}) = q_t^{n+1}$$

$$\phi \frac{S_w^{n+1} - S_w^n}{\Delta t} + \nabla \cdot (f_w(S_w^n) \mathbf{u}_t^{n+1}) = q_w^{n+1}$$

$$S_w^{n+1} = S_w^n - \frac{\Delta t}{\phi} \left(\nabla \cdot (f_w(S_w^n) \mathbf{u}_t^{n+1}) - q_w^{n+1} \right)$$

Ref: Coats, K. H. (1968). An implicit two-phase reservoir simulator (SPE-2263-MS)

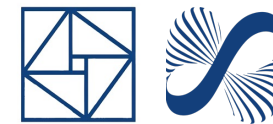
时间步进算法流程简图



求解非线性方程组，使得耦合的方程组成立，该过程对时间步长和初值非常敏感：

- 计算主变量（非线性方程组）
- 检查混合物相态（单相？两相？多相？稳定？）
- 对多相情况进行闪蒸计算（非线性代数方程）
- 更新变量，自动选择时间步长（根据后验误差、非线性迭代次数、线性迭代次数、.....）

时间步长自适应选择方法



主离散变量的选择方法



$$\frac{\partial}{\partial t} N_i + \nabla \cdot F_i - R_i - Q_i = 0, \quad i = 1, \dots, n_c$$

物质守恒方程微分形式

$$\int_{\Omega} \frac{\partial}{\partial t} N_i dV + \int_{\Omega} \nabla \cdot \mathbf{F}_i dV - \int_{\Omega} R_i dV - \int_{\Omega} Q_i dV = 0$$

物质守恒方程积分形式

$$N_i = \phi \sum_{j=1}^{n_p} x_{ij} \xi_j S_j \quad \text{Overall Molar Density}$$

$$F_i = \sum_{j=1}^{n_p} \left[x_{ij} \xi_j \mathbf{u}_j - S_j D_j \nabla (\xi_j x_{ij}) \right]$$

Darcy

$$\mathbf{u}_j = -\frac{k k_{rj}}{\mu_j} (\nabla P + \nabla P_{cj} - \gamma_j \nabla z)$$

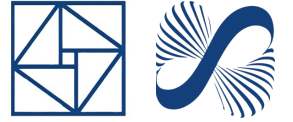
$$R_i = \sum_{j=1}^{n_p} S_j r_{ij}$$

$$Q_i = q_i$$

$n_c + 1$ 个主求解变量 \longleftrightarrow $n_c + 1$ 个主方程

物质守恒 + 流体充满孔隙

有限体积法空间离散



$$\int_{V_k} \frac{\partial}{\partial t} N_i dV + \int_{V_k} \nabla \cdot \mathbf{F}_i dV - \int_{V_k} R_i dV - \int_{V_k} Q_i dV = 0$$

单元积分形式



Green公式 (散度形式)

$$\frac{\partial}{\partial t} \int_{V_k} N_i dV + \int_{\bar{S}_k} \mathbf{F}_i \cdot \mathbf{n} dS - \int_{V_k} R_i dV - \int_{V_k} Q_i dV = 0$$

单元守恒形式



按体积元离散

$$\frac{\partial}{\partial t} N_{i,k} + \sum_{s \in \bar{S}_k} F_{i,s} - R_{i,k} - Q_{i,k} = 0$$

$$\begin{aligned} F_{i,s} &= \left(\frac{Ak}{d} \right)_s \sum_j^{n_p} \left(x_{ij} \xi_j \frac{k_{rj}}{\mu_j} \right)_s \Delta_s (P + P_{cj} - \gamma_j Z) \\ &= T_{c,s} \sum_j^{n_p} \lambda_{j,s} (x_{ij} \xi_j)_s \Delta_s (P + P_{cj} - \gamma_j Z) \end{aligned}$$

传导率系数 (调和平均)

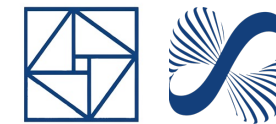
$$T_{c,s} = \frac{2}{\frac{1}{T_{c,k_1}} + \frac{1}{T_{c,k_2}}}$$

流动性、密度 (上游权)

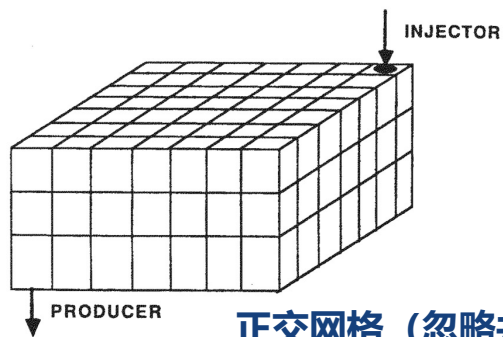
$$\lambda_{j,s} = \lambda_{j,up}$$

$$\{x_{ij} \xi_j\}_s = \{x_{ij} \xi_j\}_{up}$$

全隐式离散系统的稀疏结构

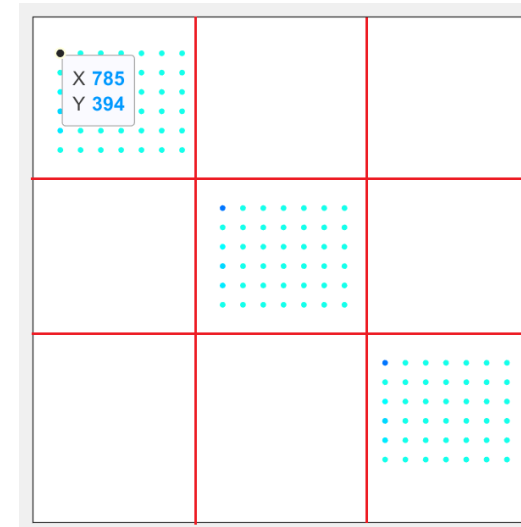
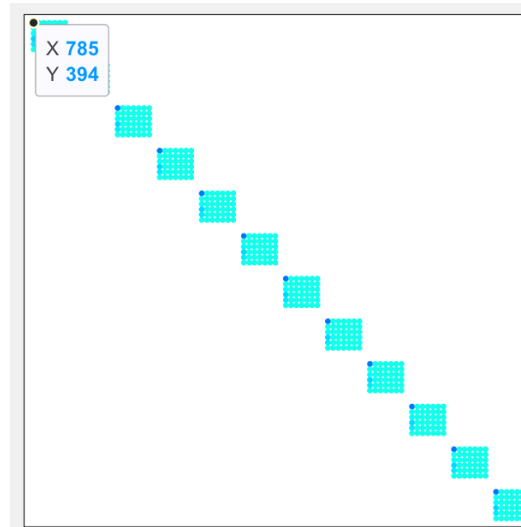
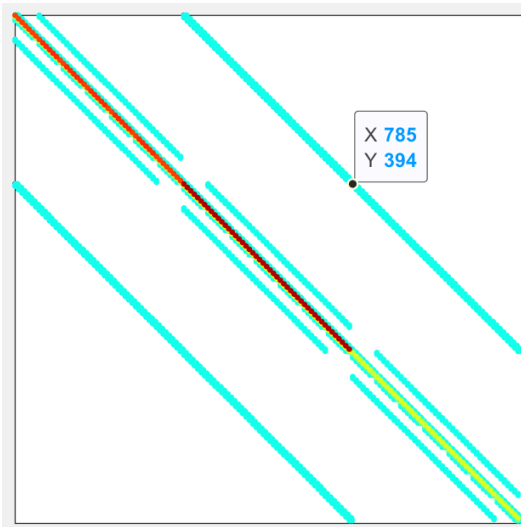
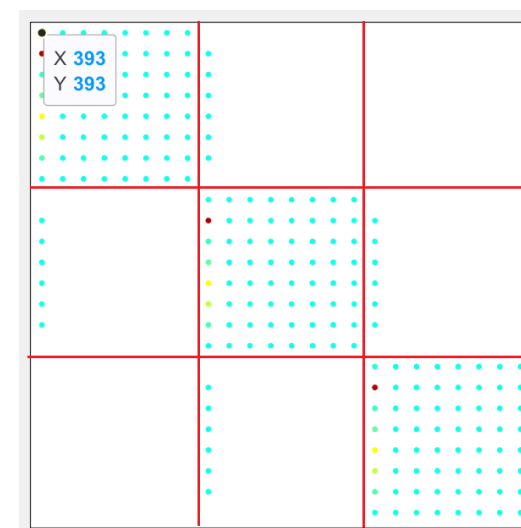
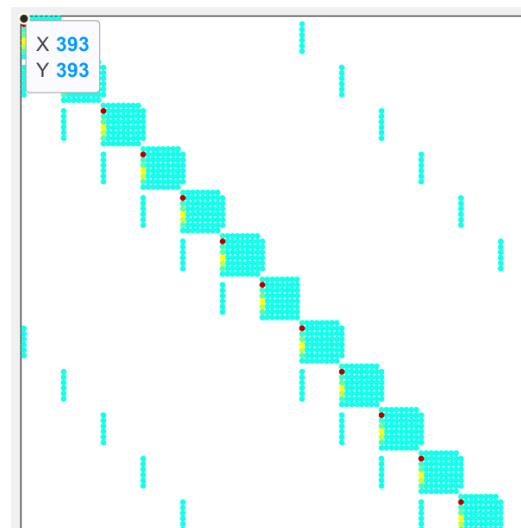
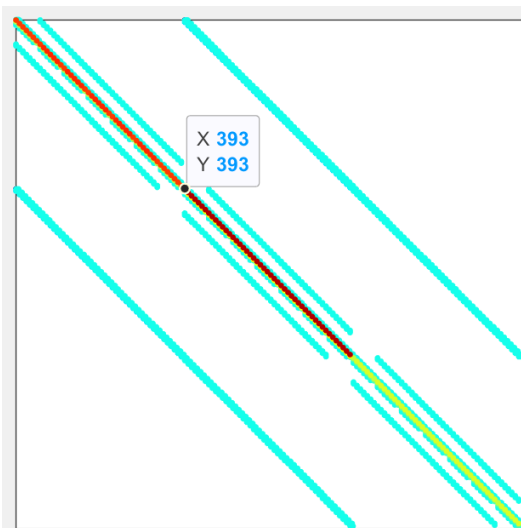
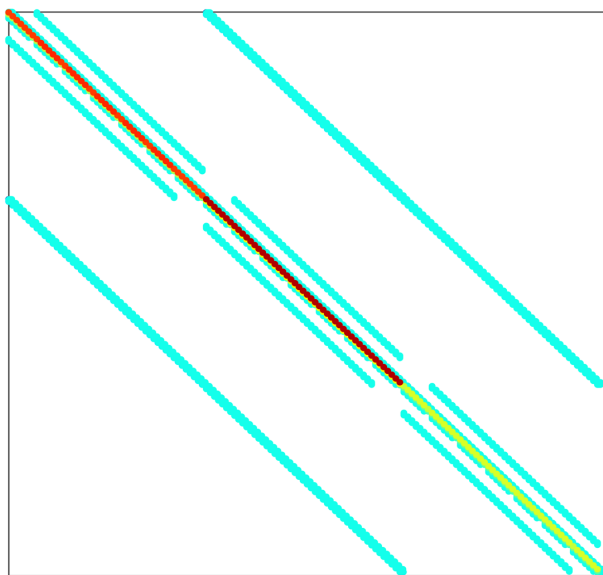


GRID FOR COMPARATIVE SOLUTIONS

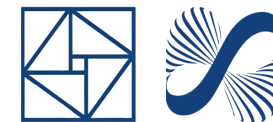


正交网格 (忽略井)

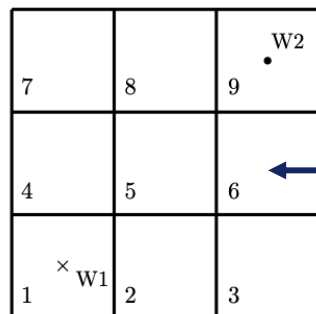
block size = 8



隐式离散的矩阵结构与数据存储

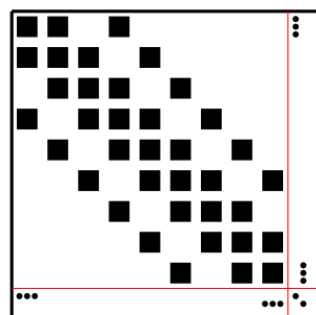


二维
正交
网格

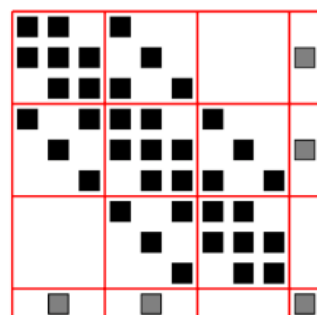


$$A_{ij} = \begin{pmatrix} a_{pp}^{ij} & a_{pN_1}^{ij} & \cdots & a_{pN_{nc}}^{ij} \\ a_{N_1 p}^{ij} & a_{N_1 N_1}^{ij} & \cdots & a_{N_1 N_{nc}}^{ij} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_{nc} p}^{ij} & a_{N_{nc} N_1}^{ij} & \cdots & a_{N_{nc} N_{nc}}^{ij} \end{pmatrix}, \vec{x}_i = \begin{pmatrix} \delta p^i \\ \delta N_1^i \\ \vdots \\ \delta N_{nc}^i \end{pmatrix}, \vec{b}_i = \begin{pmatrix} b_p^i \\ b_{N_1}^i \\ \vdots \\ b_{N_{nc}}^i \end{pmatrix}$$

稀疏
系数
矩阵



统一块
尺寸



统一矩
阵结构

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} = \begin{pmatrix} \vec{b}_1 \\ \vec{b}_2 \\ \vdots \\ \vec{b}_n \end{pmatrix}$$

- **优点:** 适用于各种网格类型, 矩阵存储格式修改后, 算法实现很容易统一
- **缺点:** 消耗更多的内存, 格式转换会产生额外开销 (小)
- **范围:** 适用于并变量很少的情况, 如果并变量很多 (变量个数变化), 则需要考虑其它存储方式

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

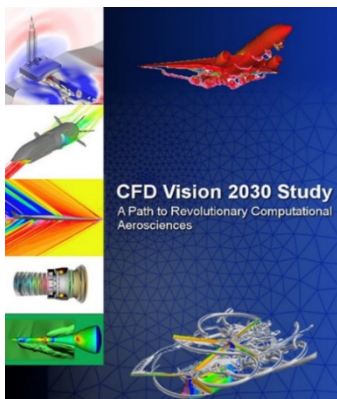
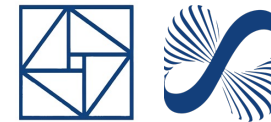
Neural Network
Methods

08

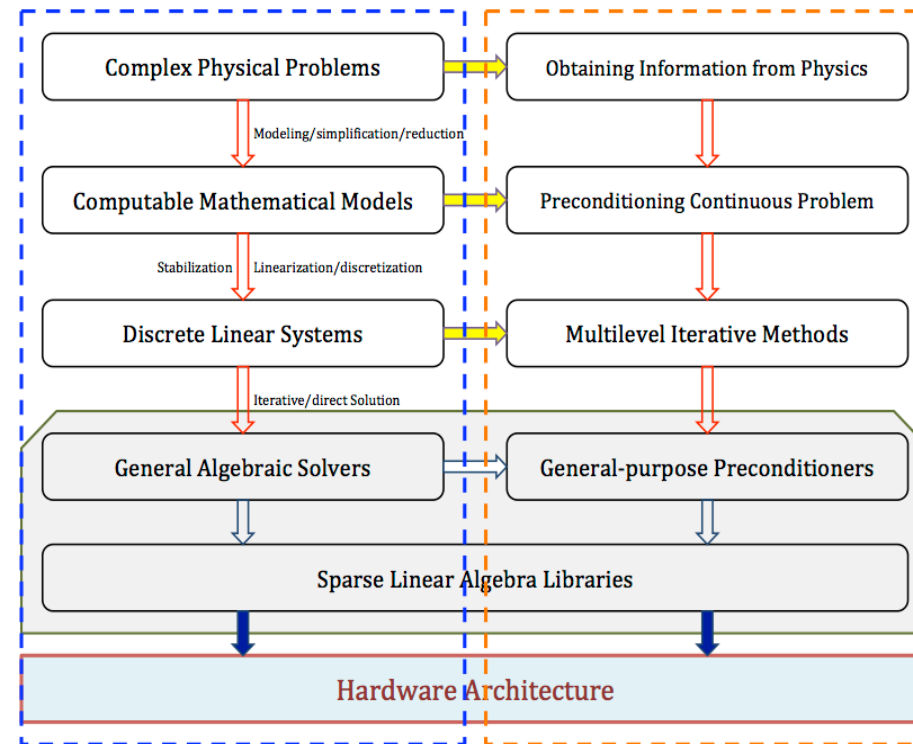
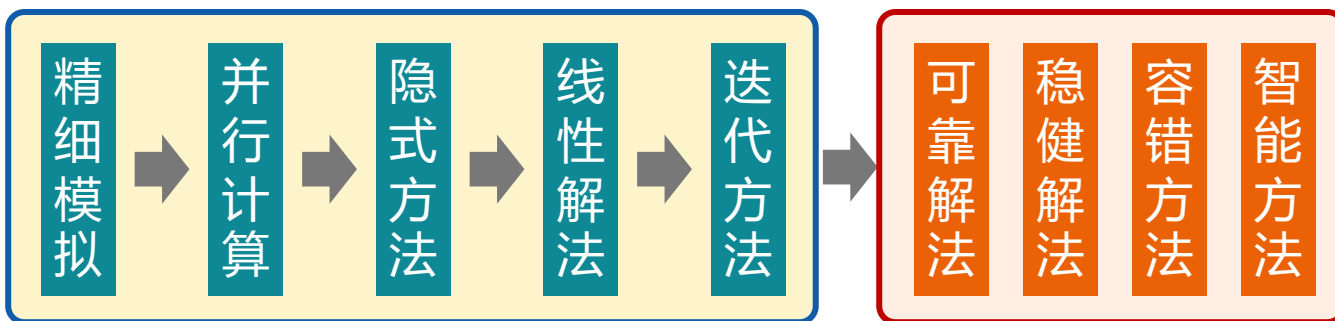
软件项目

Software
Projects

大规模数值模拟中的线性解法器

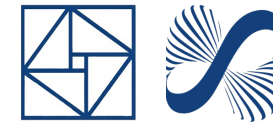


- 大规模高精度的模拟需要变革性的算法支撑
- 大规模网格并行生成和自适应仍是主要技术瓶颈
- **分析和优化过程需要稳健的自动化的求解器技术**
- 精细模拟带来大量的数据需要存储、处理和传输
- HPC硬件发展迅速，其发展趋势难以预测

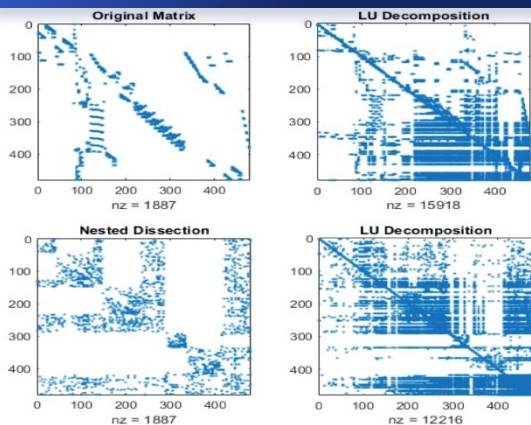


- 线性解法器通常是大规模计算中的主要瓶颈，形成**高效、通用、可扩展**的线性解法器软件是一个难题
- 快速线性解法器是大规模数值模拟的重要组件，在一些应用问题的数值模拟中占用了80%以上的时间
- 对超算硬件的性能排名（如HPC TOP500）常采用线性解法器作为Benchmark（如HPL、HPCG等）

大规模数值模拟中的预条件技术



基于代数的方法

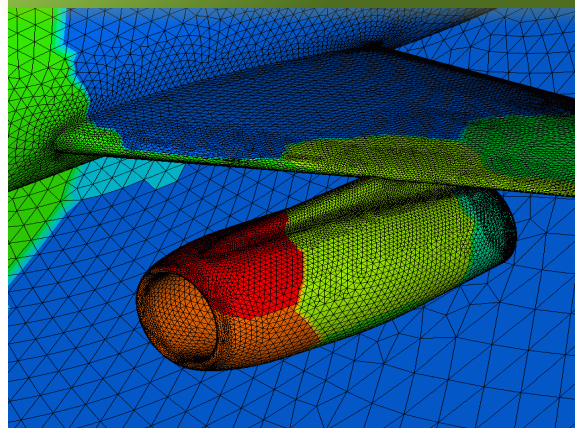


01.

LU, ILU, SAI, ...

纯代数，通用性强，稳健性高，用户友好；效率一般不高，并行可扩展性较差。

基于区域分解的方法

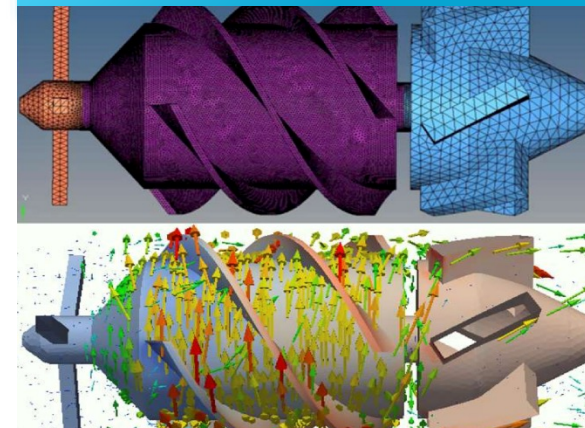


02.

DDM, RAS, FETI-DP, ...

可以基于网格进行，效率高，通用性较强，可扩展性较强；难以兼顾通用性与最优性。

基于物理的方法

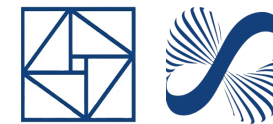


03.

Block Preconditioners

算法灵活，基于成熟算法开发，效率高，可扩展性强；通用性弱，用户友好度差。

全隐式离散方程的解法器



- 全隐式模拟方法更稳健，但会给线性代数方程求解带来挑战，线性方程组求解时间**常常超过80%**!

性质不同的耦合方程

系数的强非均质性

井方程的复杂变化

网格复杂和退化

有效节点数量巨大

$$\frac{\partial}{\partial t} \left(\phi \sum_{j=1}^{n_p} x_{ij} \xi_j S_j \right) + \nabla \cdot \mathbf{F}_i - \sum_{j=1}^{n_p} S_j r_{ij} = Q_i,$$

$$\mathbf{F}_i = \sum_{j=1}^{n_p} \left(x_{ij} \xi_j \mathbf{u}_j - S_j \mathbf{D}_j \nabla (\xi_j x_{ij}) \right),$$

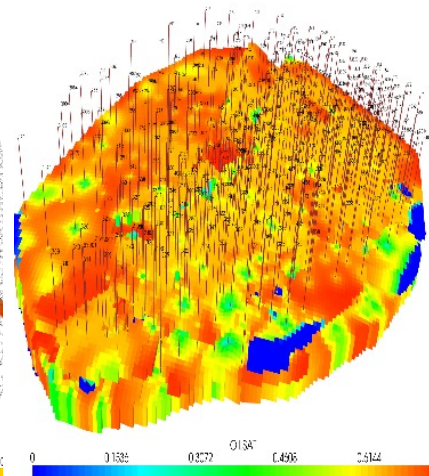
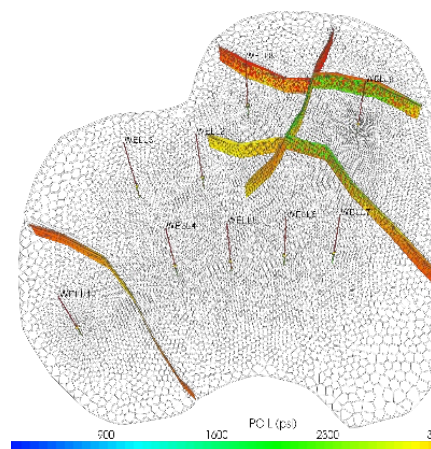
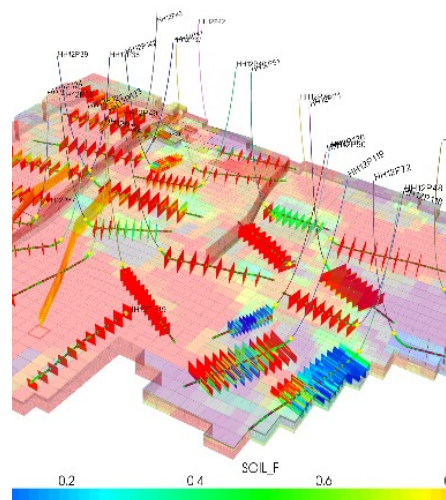
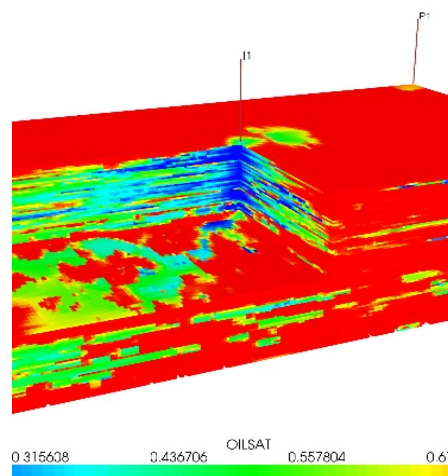
$$\mathbf{u}_j = -\frac{\kappa \kappa_{rj}}{\mu_j} (\nabla P_j - \gamma_j \nabla z),$$

$$f_{ij} = f_{i1}, \quad i = 1 : n_c$$

$$\sum_{j=1}^{n_p} S_j = 1,$$

$$\sum_{i=1}^{n_c} x_{ij} = 1,$$

$$P_1 - P_j = P_{c1j},$$



设计与预条件方法配合的高效解耦方法，在代数层面处理耦合性

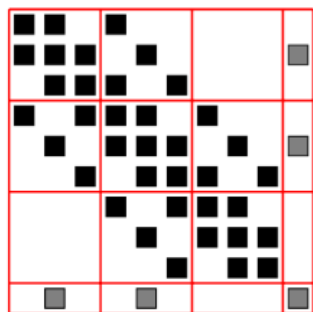
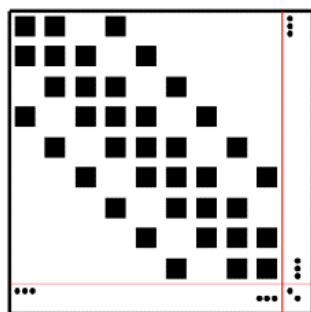
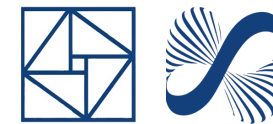
在预条件方法中使用低复杂度代数多重网格法，提升算法稳健性

在预条件方法中使用纯代数方法来处理井方程，提升算法稳健性

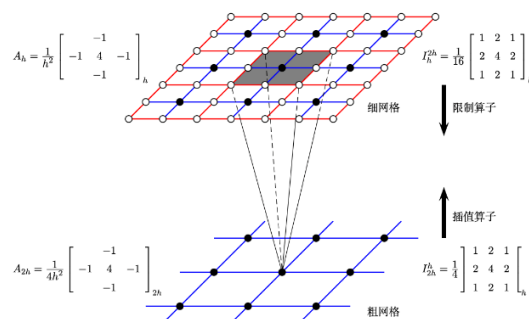
采用无结构的矩阵存储格式BSR，用统一的存储结构处理各种情况

设计最优预条件方法，并结合大规模并行计算技术提高计算效率

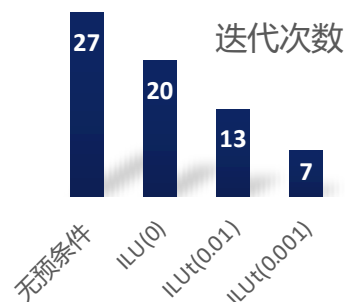
线性解法器：多阶段预条件



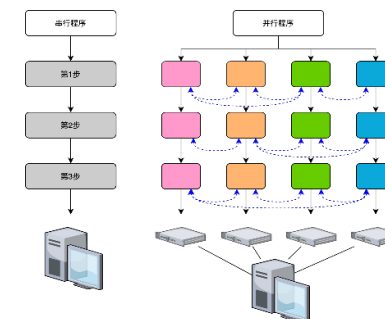
统一矩阵结构



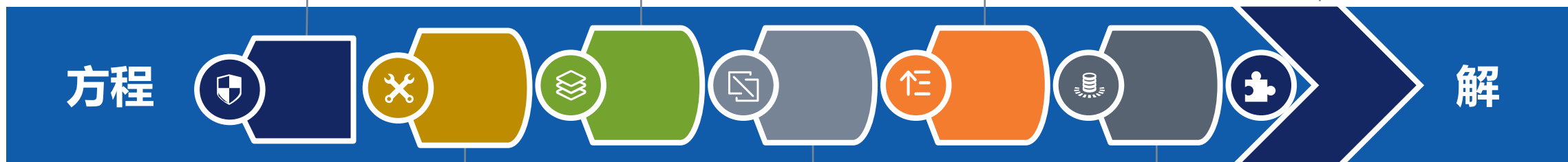
AMG方法



ILU方法



矩阵并行划分

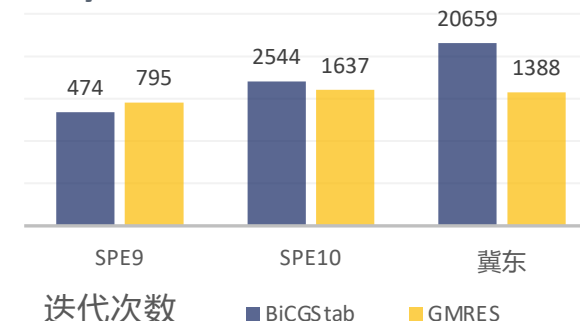
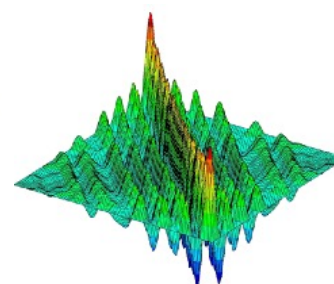
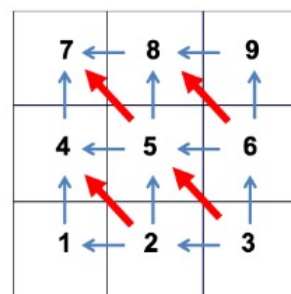
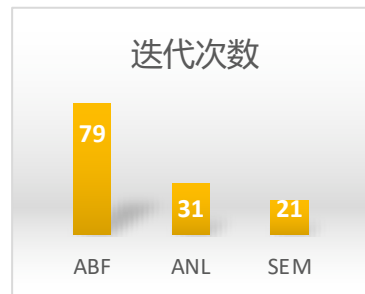


代数解耦方法

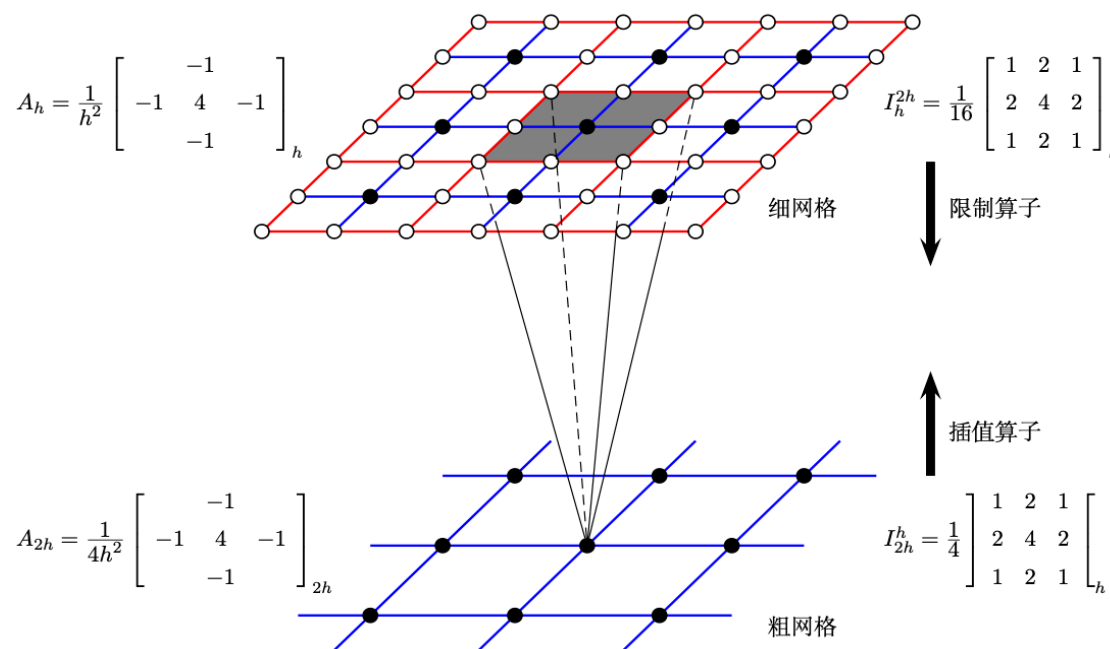
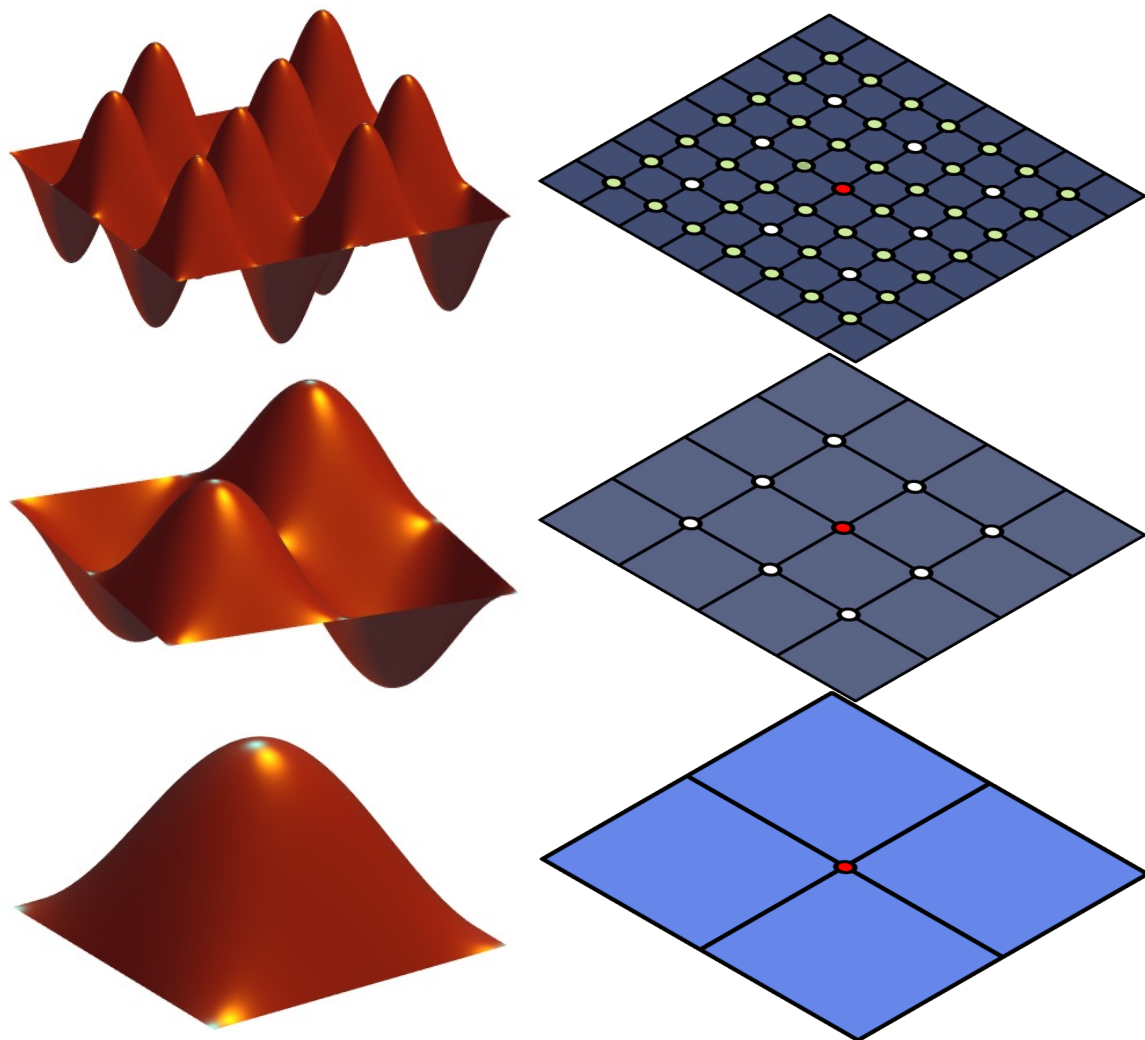
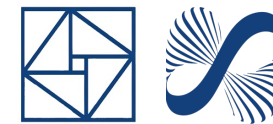
顺风磨光

Krylov子空间方法

张晨松. 油藏数值模拟中的线性解法器. 数值计算与计算机应用, 2022, 43(1): 1-26



多层迭代法：几何多重网格法



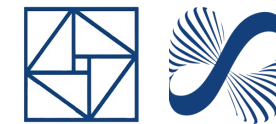
Algorithm (Twogrid method in operator form)

Given an initial guess $u^{(0)} \in V$.

- i Pre-smoothing: $u^{(1)} = u^{(0)} + \mathcal{S}(f - Au^{(0)})$;
- ii Coarse-grid Correction: $u^{(2)} = u^{(1)} + (\mathcal{I}_c \mathcal{B}_c \mathcal{I}_c^T)(f - Au^{(1)})$;
- iii Post-smoothing: $u^{(3)} = u^{(2)} + \mathcal{S}^T(f - Au^{(2)})$.

嵌套调用TG方法 → MG方法

线性解法器最优算法的重要性

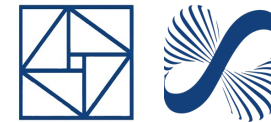


问题规模	网格剖分	64 ³			128 ³	256 ³	512 ³	1024 ³
	变量个数	0.25M			2M	16M	135M	1024M
稀疏直接法 @北京超算	计算核数	8x1	16x1	32x1	16x8	16x64	16x512	16x1024
	求解时间	5.38s	3.86s	3.26s	59.78s	999.46s	/	/
GMG@CPU	求解时间	0.030s			0.303s	2.815s	23.54s	/
GMG@GPU	求解时间	0.013s			0.026s	0.045s	0.097s	0.715s

三维Poisson方程（均匀网格七点差分格式）的线性解法器对比，稀疏直接法软件Intel MKL Pardiso（北京超级云超算）和几何多重网格法FASP（笔记本电脑，nVidia A100，80GB显存）：线程数x进程数。2020年，北京超级云计算中心A分区以Linpack测试性能3.74PFlops，单节点两颗32核AMD EPYC7452，256GB内存。2024年，张林杰测试GPU版本，Intel Xeon8358+A100。

Ref: C. Feng et al. Numerical study of geometric multigrid on CPU–GPU heterogenous computers, *Advances in Applied Math and Mech*, 2014.

多层迭代法：代数多重网格法

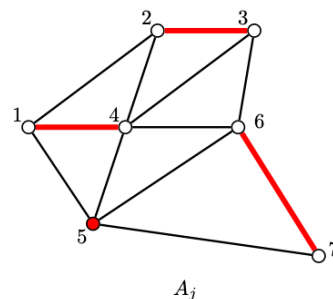


- 代数多重网格法通过系数矩阵构造限制和提升算子，从而得到不同层之间的信息传递方式和粗网格问题
- 粗网格自由度：C/F分裂、聚集法
- 插值和限制算子：
 - 确定插值局部区域 – 稀疏模式
 - 确定插值权系数 – 获得 P (and R)
- 粗空间解法器：
 - 确定粗水平上的求解问题
 - 多层迭代: V, W, AMLI, K, ...
- 磨光算子：降低误差的光滑部分
 - 简单通用的光滑子
 - 依赖问题的光滑子

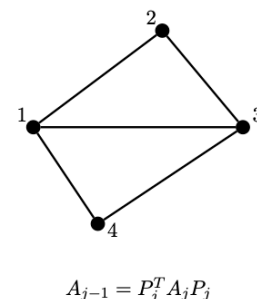
聚集法：

$$P_{ij} = \begin{cases} 1, & \text{if } i \in A_j \\ 0, & \text{if } i \notin A_j \end{cases}$$

$$P_j : \mathbb{R}^{n_c} \mapsto \mathbb{R}^{n_f}$$

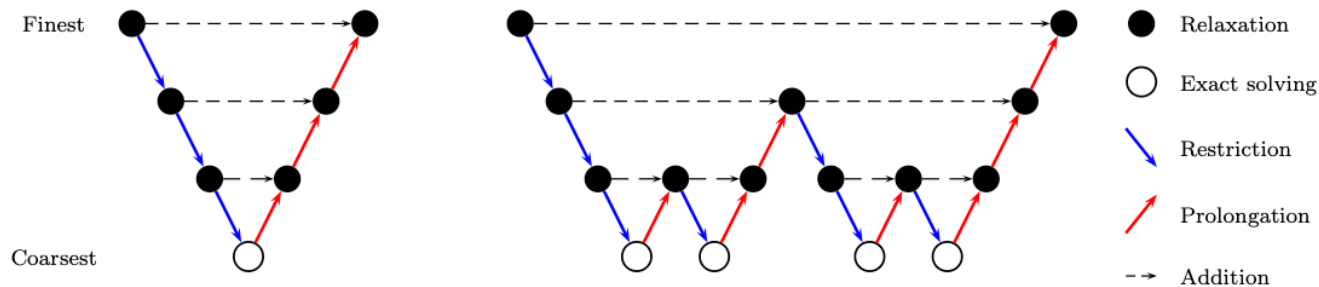


$$P_j = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

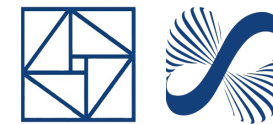


$$A_{j-1} = P_j^T A_j P_j$$

- 非光滑聚集法的插值算子构造简单，基于聚集法的AMG方法过程更简单，对并行更友好！
- 光滑聚集法AMG的插值算子是在非光滑聚集法AMG的插值算子的基础上左乘一个磨光算子



解耦方法与预条件方法



压力	耦合项
耦合项	饱和度等

$$\text{Original} \begin{bmatrix} * & * & \dots & * \\ 0 & 1 & \dots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & \dots & 0 \\ * & * & \dots & * \\ * & \vdots & \ddots & \vdots \\ * & * & \dots & * \end{bmatrix}$$

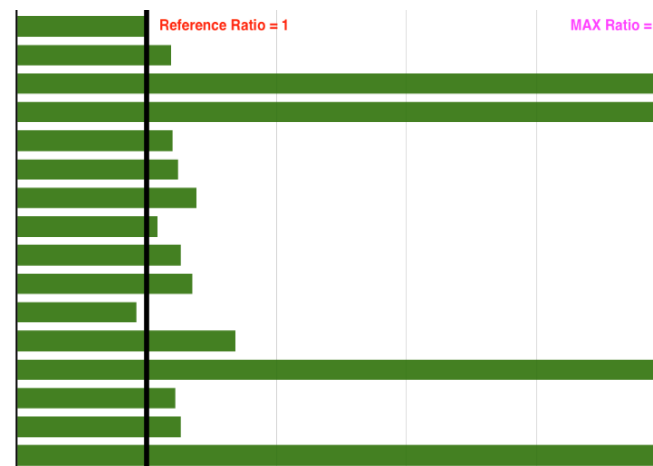
- 解耦方法简单，复杂度低，本身不需要收敛
- 解耦后的问题，需要适用于整体多阶段预条件方法
- 解耦后的问题，每个子问题都“易于求解”

部分解耦方法

压力	0
耦合项	饱和度等

$$\text{SEM} \begin{cases} \text{ANL} \begin{bmatrix} * & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} + \begin{bmatrix} * & * & \dots & * \\ * & * & \dots & * \\ * & \vdots & \ddots & \vdots \\ * & * & \dots & * \end{bmatrix} \\ \text{ABF} \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \end{cases}$$

Ref: Qiao, Wu, Xu, Zhang. JCP 2017

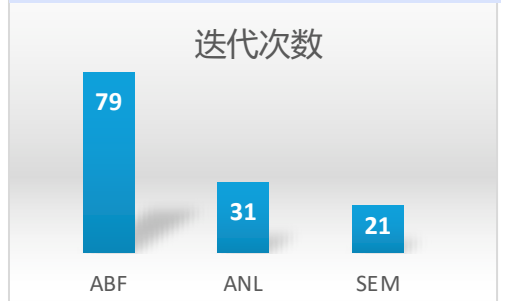


ABF方法的解耦效果强，但是会使得压力方程更难求解，需要对其进行特殊处理才能获得较高的整体性能

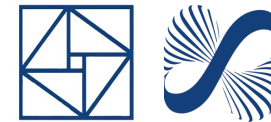
完全解耦方法

压力	0
0	饱和度等

Method	Time Steps	Nonlinear Iterations	Linear Iterations	AMG Iterations	Linear Solver Time (s)
ABF	60	352	2505	37235	7756
Analytical	57	332	2209	16212	3149
Semi-analytical	56	320	1338	13813	2464



线性解法器：解耦方法



中国石油大学（华东）油藏工程系：SimFast模拟器的解法器优化

- 继承了石油部引进的黑油模拟器代码，进行二次开发
- 增加前后处理、各种现场应用功能和实用模型等，.....，**已经开发了近30年时间**

中国石油勘探开发研究院：个性化井网设计软件求解性能测试与优化

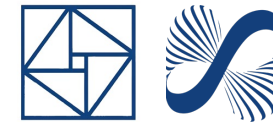
- 通过斯坦福大学的Industrial Consortium获得油藏模拟器代码
- **18万行代码，消化这些代码比较困难，更重要的是：没有人愿意去读**
- 一些算例不能算，一些算法不能算；或者跑一段时间死机，或者跑得很慢.....



AMG求解时间	算例A			算例B		
	耗时 (秒)	调用次数	加速比	耗时 (秒)	调用次数	加速比
修改前	1705.188	192	---	2034.765	107	---
修改后	7.815	72	219倍	94.475	37	22倍

解法器代码改进（解耦方法，修改一行代码）：前后计算速度对比

求解算法性能大幅提升



SPE10



利用CPU-GPU异构并行,
将模拟时间缩短至6m

使用NVIDIA P100, 将多阶段预条件方法移植到GPU; 同时移植到国产DCU (网络中心419)

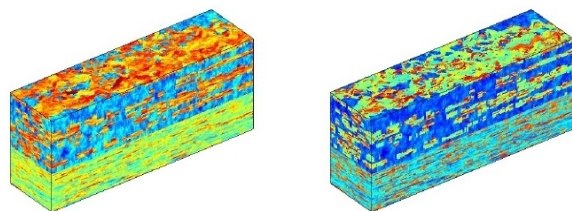


OpenCAEPoro模拟器将
模拟时间缩短至47s

计算节点配备两颗CPU (32核, 运行频率为3.1GHz, 256GB内存), 使用128进程分布式并行

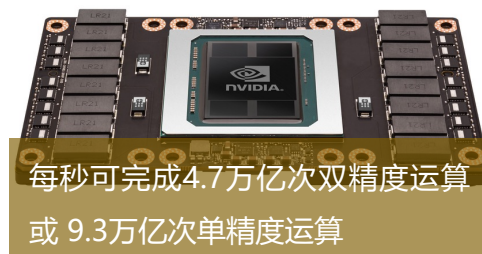
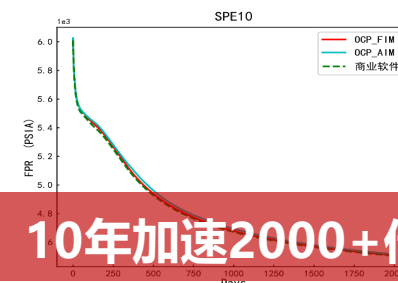
FASP中构造MSP预条件,
40m完成模拟 (串行)

中石油自研MURS 29h, 商业软件: Ecl100 100h, Ecl300 8-node 5h, tNavigator 18h

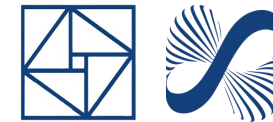


获得首届CSIAM数学应用落地认证

为不同应用场景构造了代数解耦方法、多阶段预条件、组合型预条件等一整套解决方案



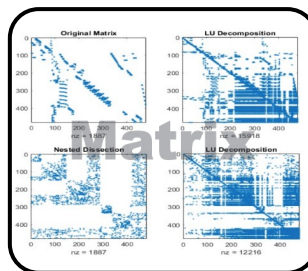
线性解法器不可能三角



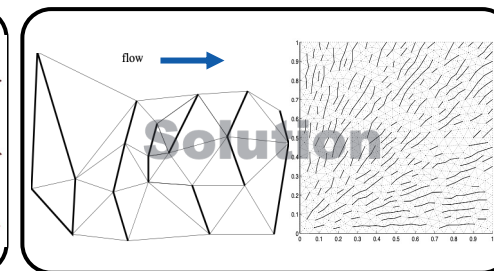
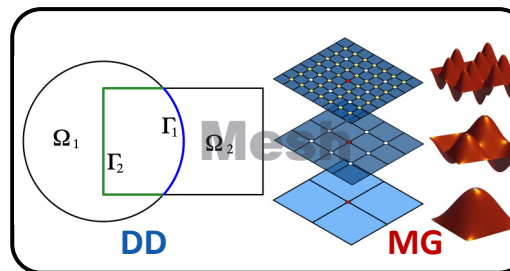
$Ax = b$

↓ IR

$x' = x + B(b - Ax)$



$$\begin{cases} -\Delta u + \nabla p = \mathbf{f}, & \Omega; \\ \nabla \cdot \mathbf{u} = 0, & \Omega; \\ \mathbf{u} = 0, & \partial\Omega. \end{cases}$$



Efficiency

Efficient, optimal, and scalable

One size does not fit all!

Robustness

Robust, resilient, and reliable



Q: How to choose or develop a solution algorithm in practice?

Usability

Applicable, user-friendly, and portable

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

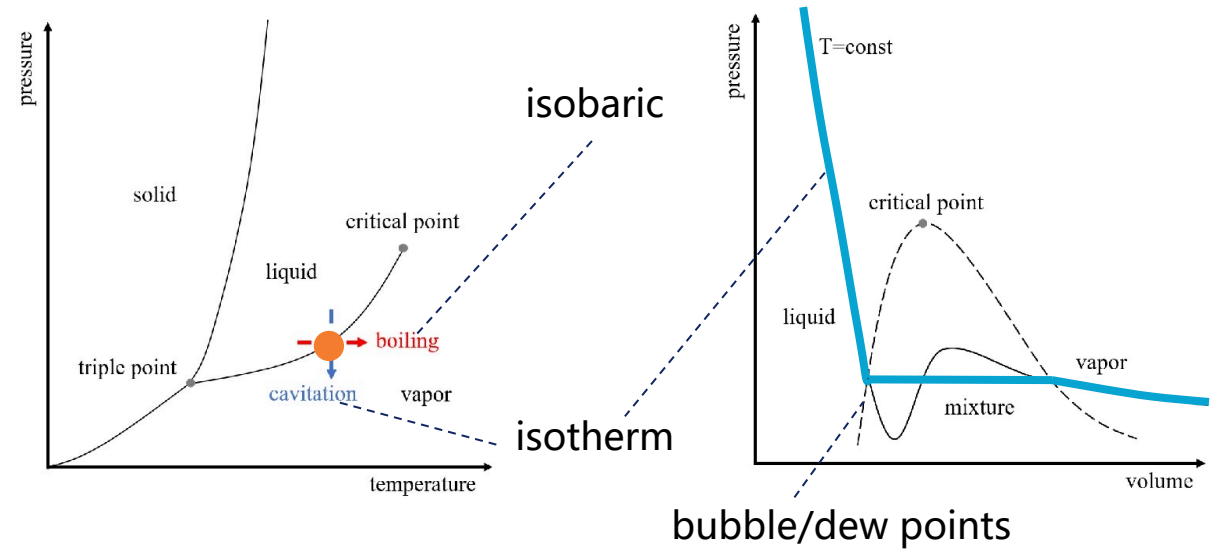
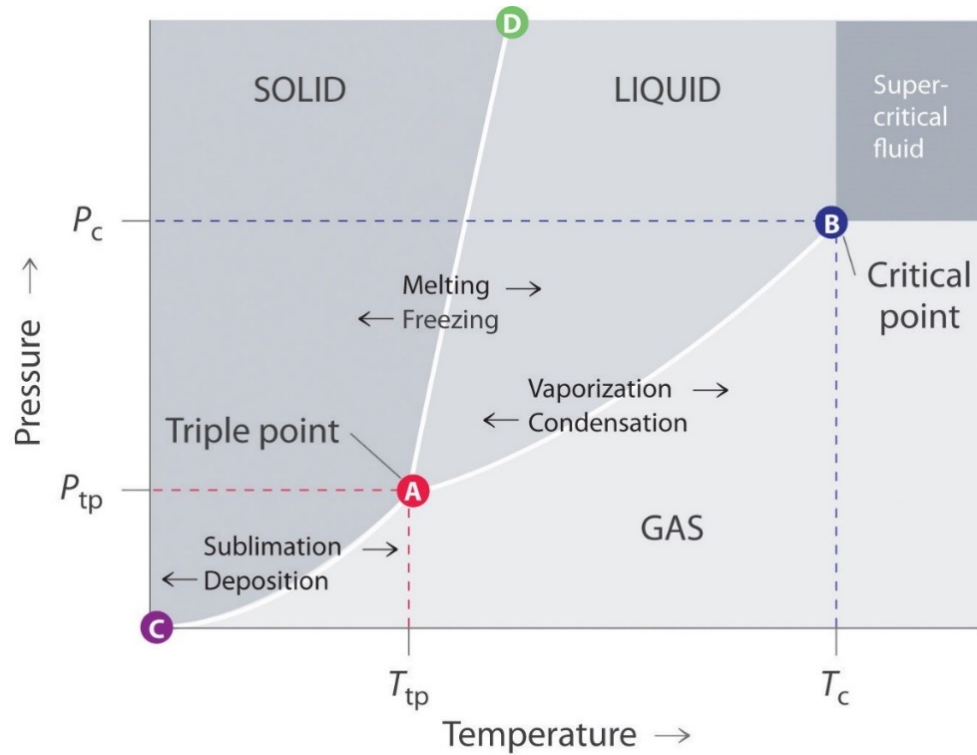
Neural Network
Methods

08

软件项目

Software
Projects

单组分子物质的相态变化



Reference: General Chemistry -- Principles, Patterns, and Applications

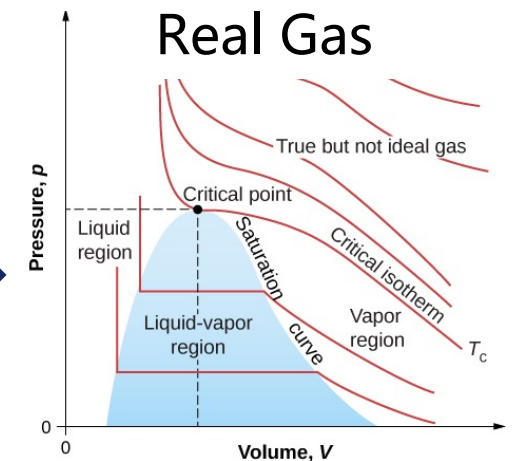
http://saylordotorg.github.io/text_general-chemistry-principles-patterns-and-applications-v1.0/index.html

Ideal Gas EOS

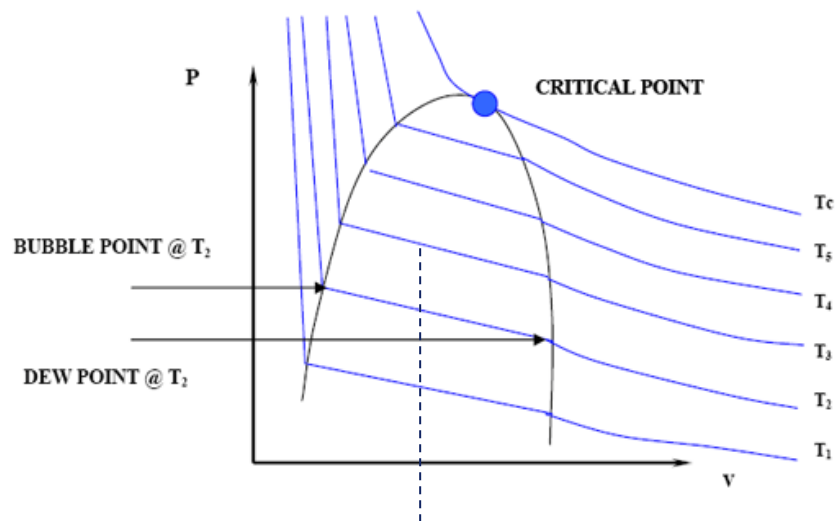
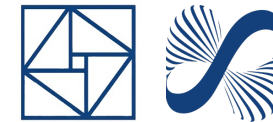
$$pV = nRT$$

Pressure in Pascal (Pa) Number of moles Temperature in Kelvin (K)
 Volume in m³ Gas constant = 8.314 JK⁻¹mol⁻¹

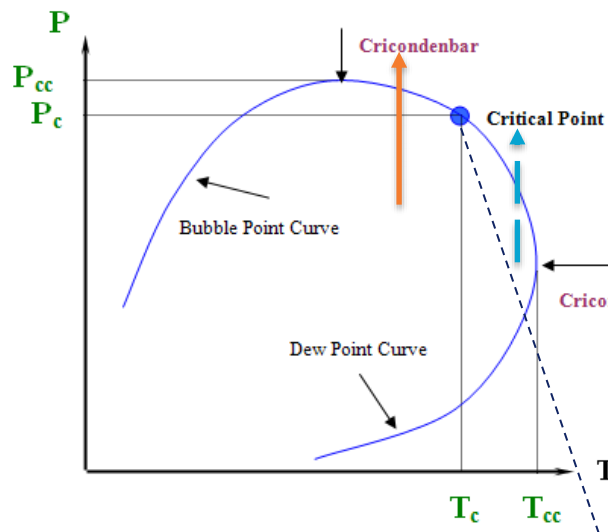
Real Gas



多组分物质的相态变化



PV曲线的形态发生明显变化



临界点可能不对应泡点线上的最大压力

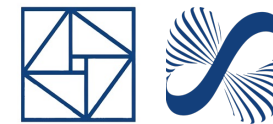
正常现象（和单组分类似）：
达到饱和条件后，越压缩体系，析出的液体就越多

反常现象：即使压缩系统，液体也可能会蒸发而不是凝结

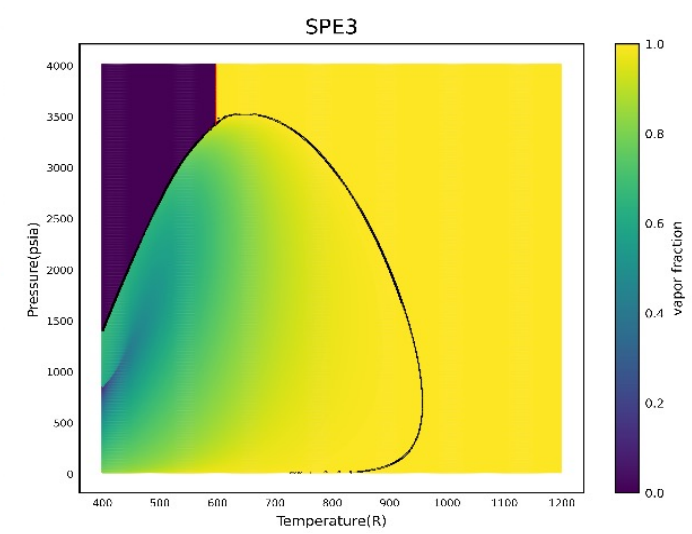
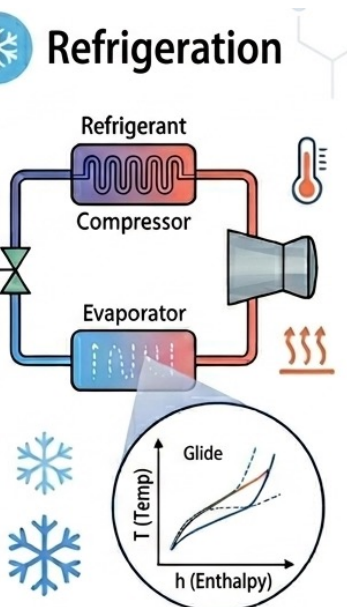
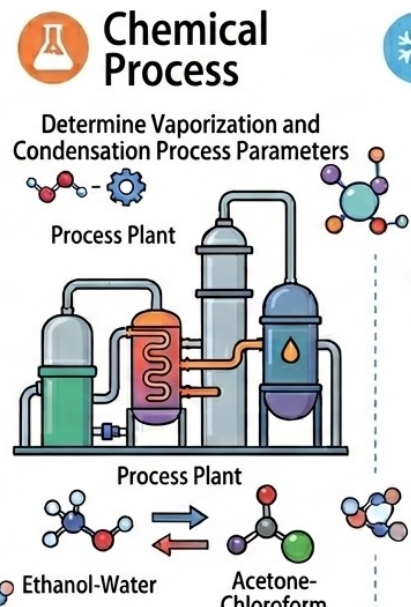
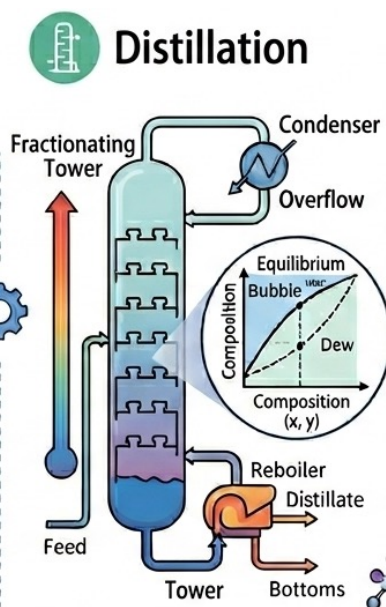
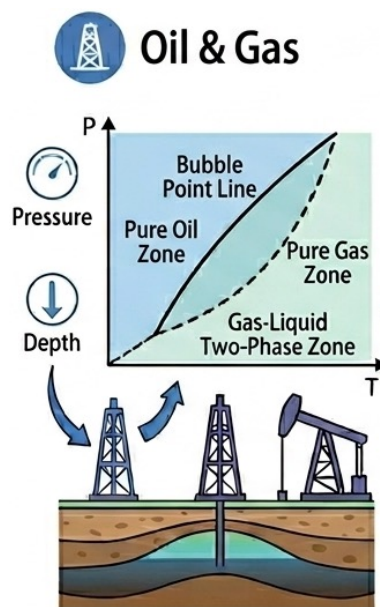
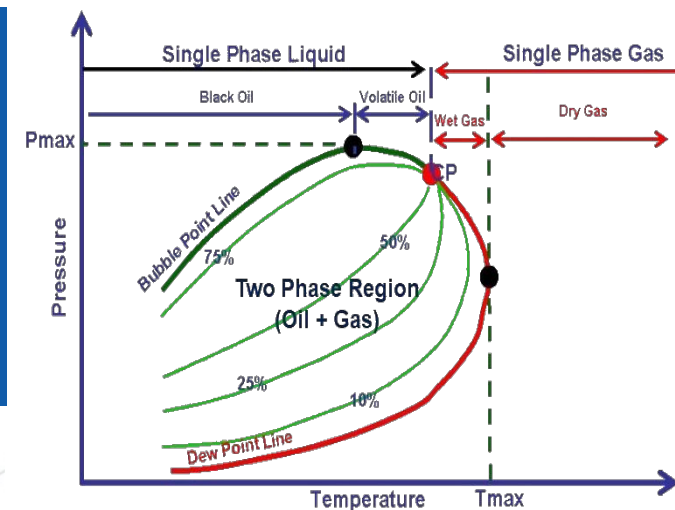
多组分情形：分子间距离与作用力改变，和单组分情形有较大变化

- 当组分数增加时，系统变得更加复杂，不仅相平衡计算的难度加大了，而且建模本身也更困难
- 多种组分的混合物处于两相状态时，不同种类分子同时存在，它们可以是液态或气态（两相状态）
- 有些分子“更愿意”处于气态，而有些分子“更愿意”处于液态；这种“偏好”就是挥发性

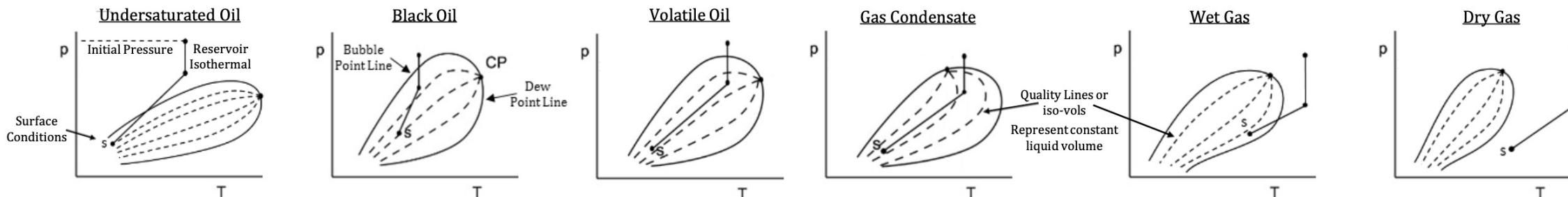
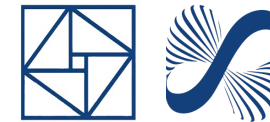
泡点线和露点线的含义与应用



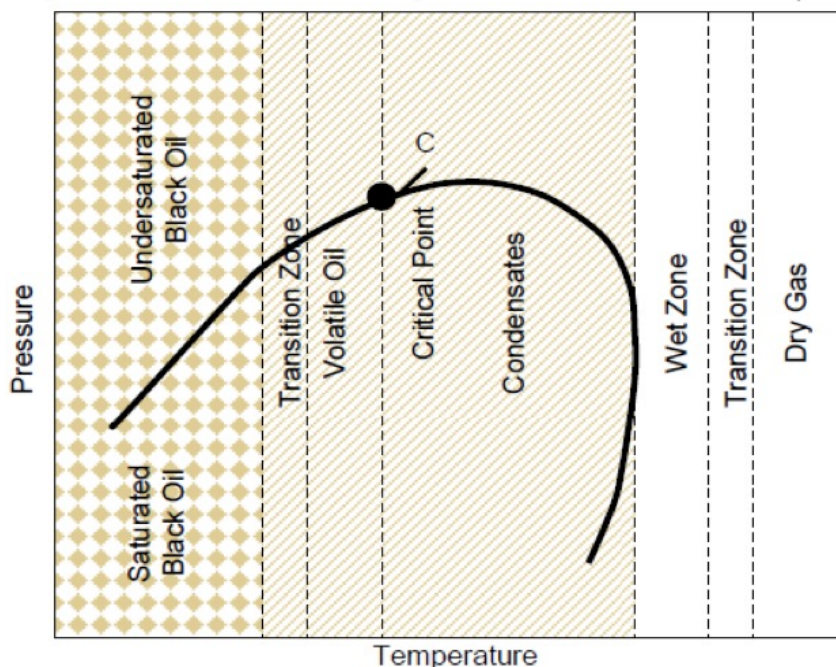
- **泡点线 (Bubble Point Line)** 是描述液体何时开始蒸发成气体的热力学边界, **露点线 (Dew Point Line)** 是描述气体何时开始凝结成液滴的热力学边界
- 泡点线和露点线依赖于混合物的压力和组成; 在不同的压力下, 同一混合物的泡点露点温度可能会不同; 同理, 含有不同组成成分的混合物在相同压力下的泡点露点温度也会不同。它们在工程问题中有广泛应用:



油气藏的典型相态变化特征



油藏模拟常用模型
适用范围与P-T相图



Ref: Thermodynamics and Phase Behavior, J. Riddle, Univ of Texas at Austin

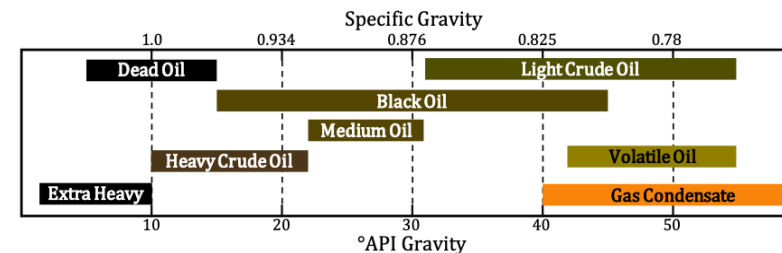
Laboratory and Field Characteristics

Fluid Type	T^{RC}	Initial GOR scf/STB	Oil Gravity °API	Initial B_o RB/STB	C_{7+} Fraction Mole %	Phase Behavior
Dry Gas	$> T^{crc}$	--	--	--	--	Single phase at all pressures and temperatures
Wet Gas	$> T^{crc}$	$> 50,000$	--	--	--	Gas phase in reservoir at all P & T. Separator within two-phase region.
Gas Condensate	$T^c - T^{crc}$	3300 – 50000	40 – 60	--	< 12.5	All gas initially; As pressure falls below dew point a liquid phase forms
Volatile Oil	$< T^c$	2000 – 3300	42 – 55	1.7 – 3.0	12.5 – 20	T^{RC} close to T^c . Key difference between volatile and black oils is that the gas phase can produce liquid at surface.
Black Oil	$< T^c$	200 – 2000	15 – 45	1.1 – 1.7	25 – 50	As reservoir pressure falls below bubble point, gas phase forms.
Undersaturated Oil	$< T^c$	200 – 2000	15 – 45	1.1 – 1.7	25 – 50	Oil phase in reservoir at all P & T. Separator within two-phase region.
Dead Oil	$< T^c$	--	5 – 15	1.0 – 1.1	> 50	Single phase at all pressures and temperatures.

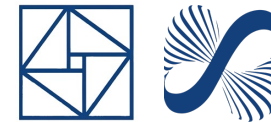
T^{crc} = cricidentherm

API Classification

Fluid Type	°API Gravity
Light Crude Oil	> 31.1
Medium Oil	22.3 – 31.1
Heavy Crude Oil	10.0 – 22.3
Extra Heavy Oil	< 10.0



单组分物质的PVT性质刻画



对于一种混合物来说，物质处于什么相态？存在多少相？是否处于稳定的平衡状态？某一具体的相中包含了多少种物质？以及包含了多少该物质？

van der Waals / Redlich–Kwong / Redlich–Kwong–Soave EOS:

$$P := \frac{RT}{V-b} - \frac{a}{V(V+b)}$$

Peng–Robinson EOS [Peng, Robinson 1976]:

$$P := \frac{RT}{V-b} - \frac{a}{V(V+b) + b(V-b)}$$

$$a := 0.45724 \frac{R^2 T_c^2}{P_c} \alpha(T), \quad b := 0.07780 \frac{RT_c}{P_c}, \quad \alpha(T) := \left(1 + m(\omega)(1 - \sqrt{T/T_c})\right)^2$$

PR EOS: 40 years

through cubics (2017)

Change of variables:

$$A := \frac{aP}{R^2 T^2}, \quad B := \frac{bP}{RT}, \quad Z := \frac{PV}{RT}$$

$$Z^3 - (1 - B)Z^2 + (A - 2B - 3B^2)Z - (AB - B^2 - B^3) = 0 \quad \text{三次模型}$$

三次状态方程的求解方法

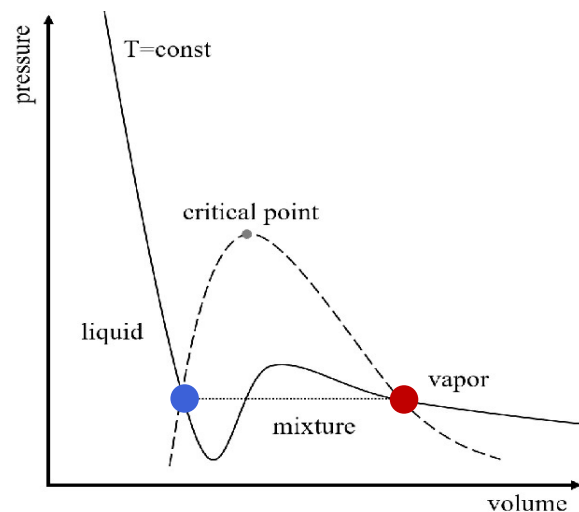


- Cubic EOS models (like the PR EOS):

$$A := \frac{aP}{R^2 T^2}, \quad B := \frac{bP}{RT}, \quad Z := \frac{PV}{RT} = \frac{V_m}{V_m^{\text{ideal}}}$$

$$Z^3 - (1 - B)Z^2 + (A - 2B - 3B^2)Z - (AB - B^2 - B^3) = 0$$

- Cubic models in terms of Z (or the volume V , equivalently, in fact)
- The **compression factor** (Z-factor) $Z=1 \rightarrow$ **ideal gas**



Example:

$$Z^3 - Z^2 + 0.089Z - 0.0013 = 0$$

$$Z_1 = 0.018301 \quad \text{liquid phase}$$

$$Z_2 = 0.078661 \quad \text{discard}$$

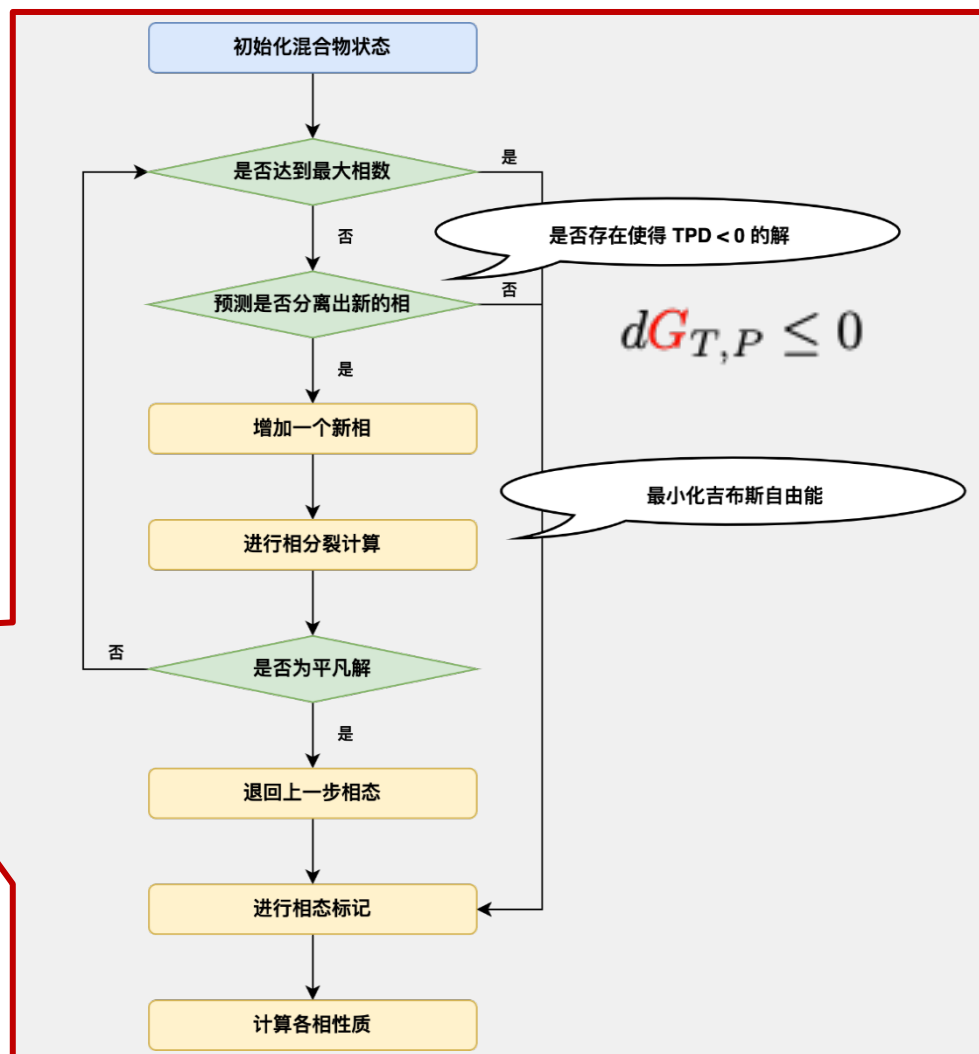
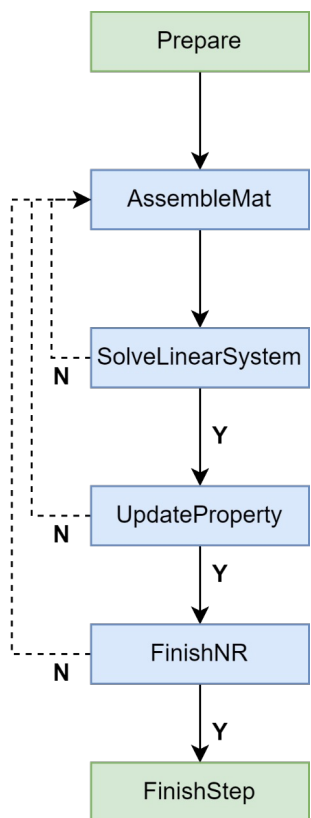
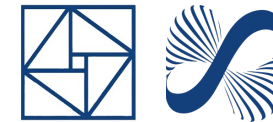
$$Z_3 = 0.903038 \quad \text{vapor phase}$$

最小化Gibbs自由能



Vapor-Liquid Equilibrium (VLE)

多相平衡计算方法的标准过程



气液两相平衡典型计算流程:

计算混合物成分并初始化

相稳定性分析

计算初始气液平衡系数

计算气液相平衡比

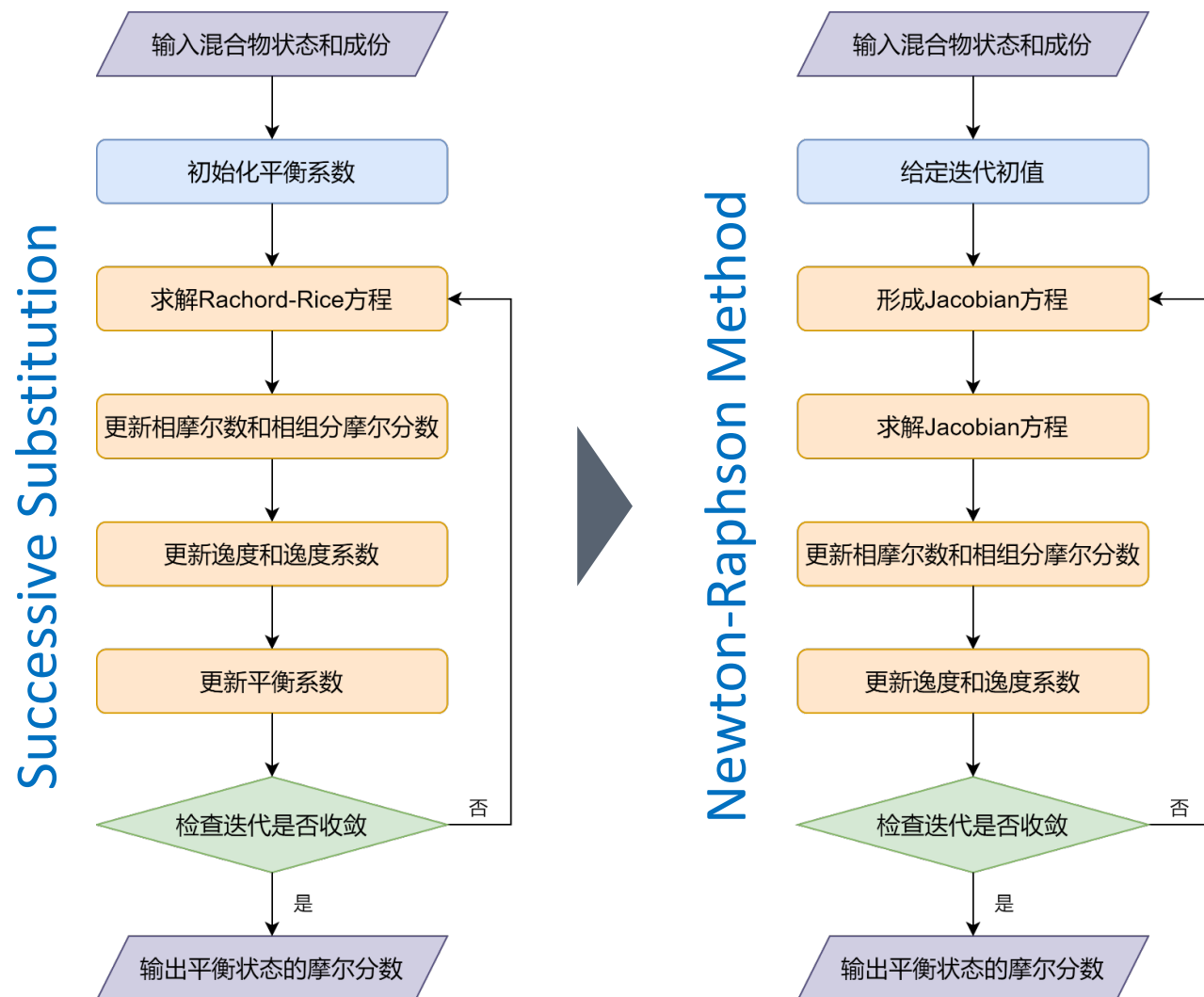
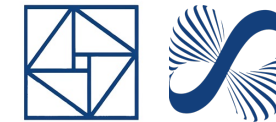
计算相的组分摩尔数占比

计算EOS模型对应的逸度

检查是否达到相平衡

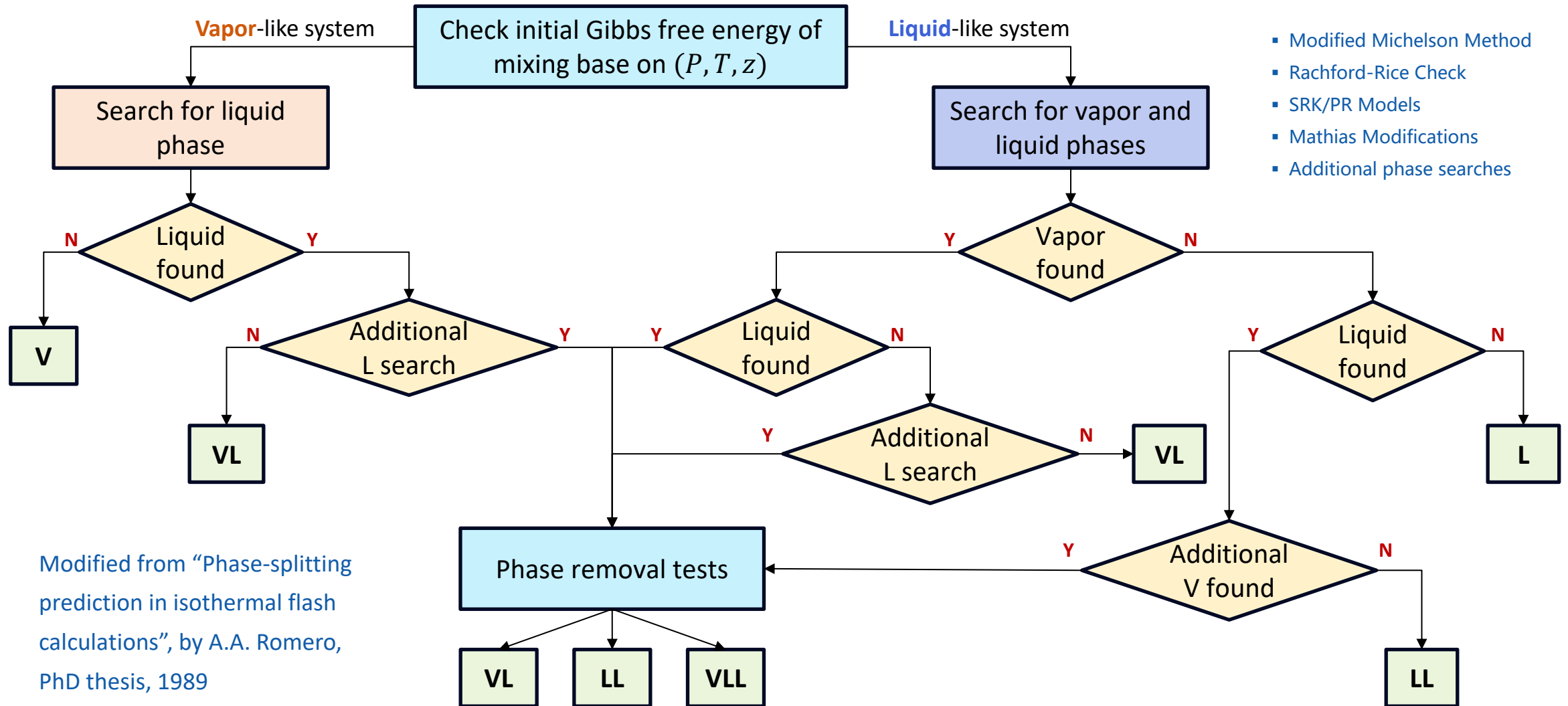
更新气液平衡系数

相平衡计算非线性方程求解



- 两种方法（SSM与Newton迭代）可以进行结合：首先使用SSM方法，收敛到一定程度之后，切换到Newton法，增强迭代法的稳健性
- 在临界点附近，一般会使用ASSM方法加快迭代收敛速度
- 可以设计自适应策略，根据情况选择迭代方法

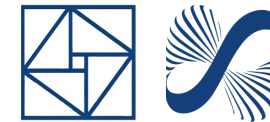
三相或多相平衡的计算流程



- Modified Michelson Method
- Rachford-Rice Check
- SRK/PR Models
- Mathias Modifications
- Additional phase searches

Modified from "Phase-splitting prediction in isothermal flash calculations", by A.A. Romero, PhD thesis, 1989

多相平衡计算中的挑战



相态“可知性”问题

计算前无法确定系统是两相还是三相，算法必须自行判断相态数量，这是最根本的难点。

$$\text{稳定性分析: } \min \text{TPD}(w) \geq 0$$



模型准确性要求

需要模型能准确地预测液相分裂行为，普通EOS往往失效，必须使用高级混合模型。

$$a_m = \sum_i \sum_j y_i y_j (a_{ij})^{1/2} (1 - k_{ij})$$

三相及更多相平衡计算的复杂性和计算挑战远超两相平衡计算，主要体现在相态判断、变量激增、初始化难度和模型精度等方面



变量维数提高

变量数量随相数呈指数级增长，需要为两个液相提供不同且物理上合理的初始组成估计。

$$\text{物料平衡: } F = V + L_1 + L_2$$

$$z_i F = y_i V + x_i^I L_1 + x_i^{II} L_2$$



极致精度要求

在相边界处，算法需要在接近机器精度的情况下区分微小的逸度差异，对数值稳定性要求苛刻。

$$\max \left| \frac{f_i^k}{f_i^m} - 1 \right| < \epsilon \quad (10^{-10} \text{或更小})$$



收敛到亚稳态风险

算法易收敛到亚稳态解（如误将三相系统识别为稳定的两相系统），而非全局最稳定的热力学状态。

$$G^{\text{system}} = \sum_k \sum_i n_i^k \mu_i^k$$



相态动态管理

算法必须动态处理相的出现与消失，在迭代过程中可能需要增加或减少相数。

$$\text{相分率约束: } \psi_V + \psi_{L1} + \psi_{L2} = 1$$

鲁棒算法 = 稳定性分析 + 闪蒸计算 + 动态相数管理

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

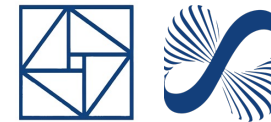
Neural Network
Methods

08

软件项目

Software
Projects

并行效率与最优算法



Amdahl's Law 1967



如果串行部分占总时间的10%，那并行加速比不可能超过10倍

对于大规模系统来说，很多应用程序的强可扩展性是很难实现的！

对于很多应用来说，更需要的是计算越来越大规模的问题！需要是弱可扩展性，而不是强可扩展性

Gustafson-Barsis's Law 1988



当问题规模与计算资源同比例增大 s 倍时，加速比最高就可达 $0.9s+0.1$

Gabriel Wittum: HPC Paradox



当购买了大10倍的硬件系统，希望能更快地求解大10倍的问题；但现实很残酷，必须有最优算法才可以！

最优算法对于充分发挥HPC效率及实现弱可扩展性至关重要！

硬件投资



并行效率

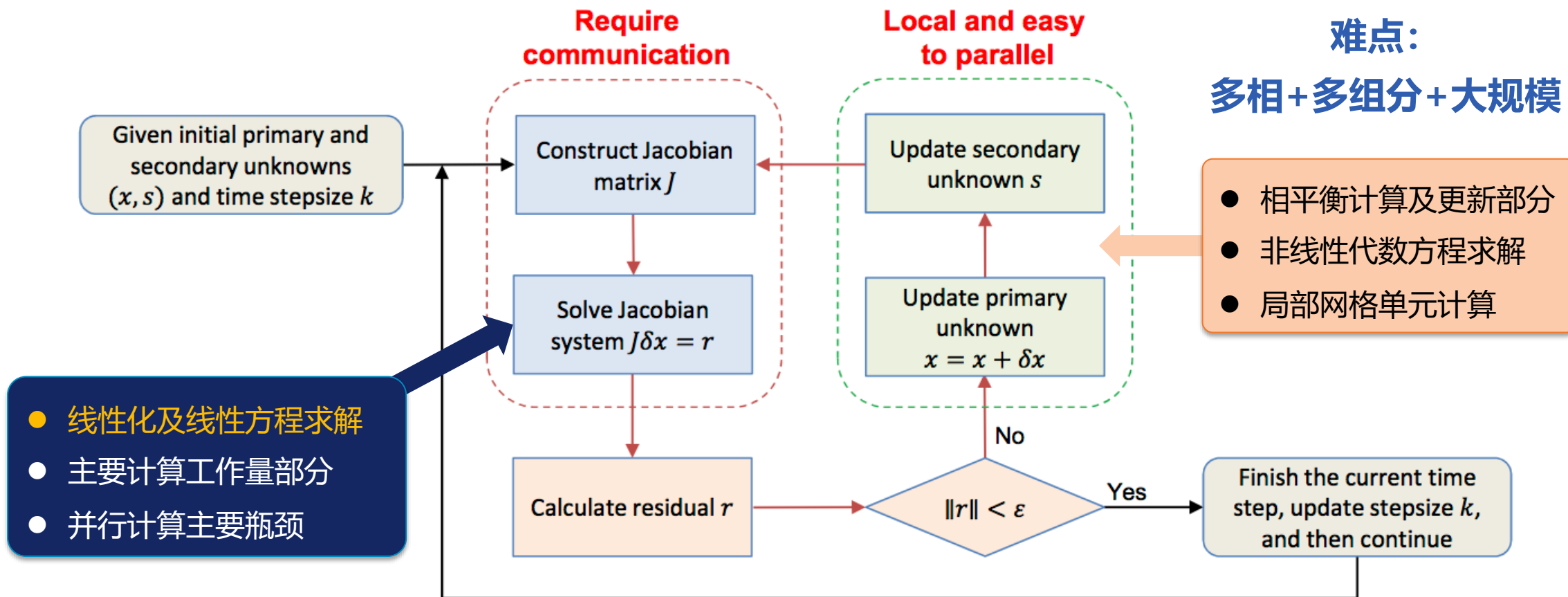
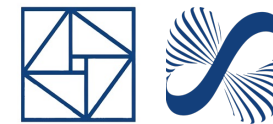


最优算法



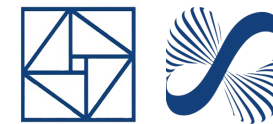
软件投资

并行油藏模拟的性能瓶颈



瓶颈：代数求解、相平衡计算、Jacobian矩阵形成等；当规模扩大，时间步长更小，非线性迭代次数更多

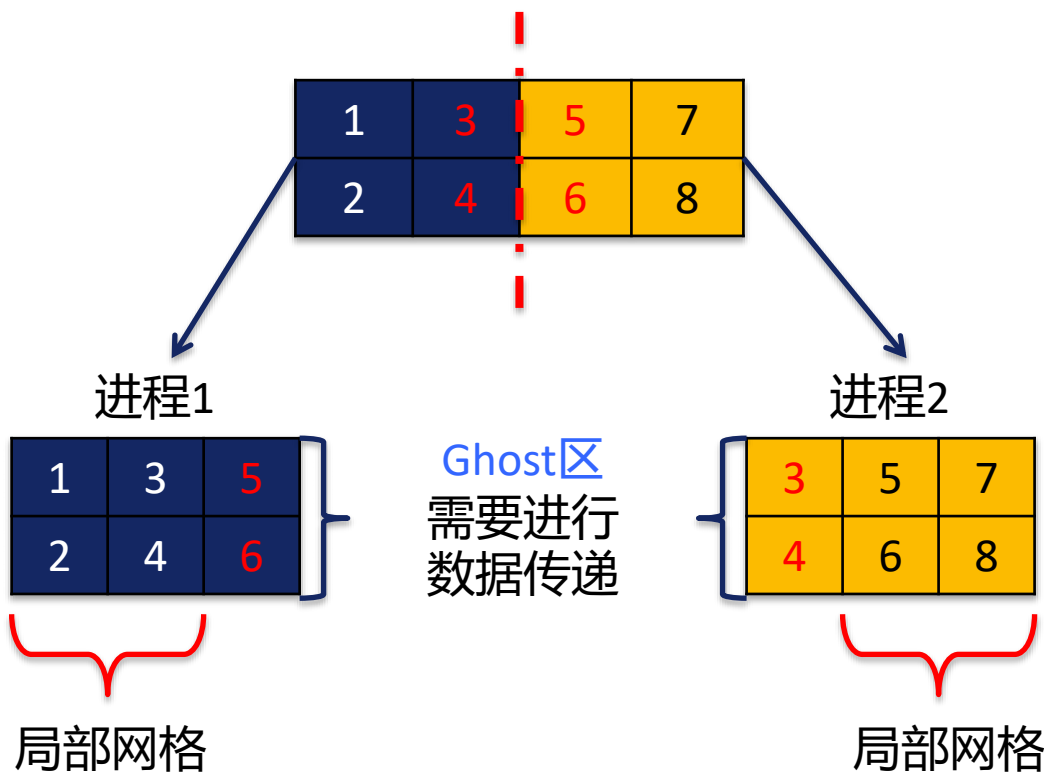
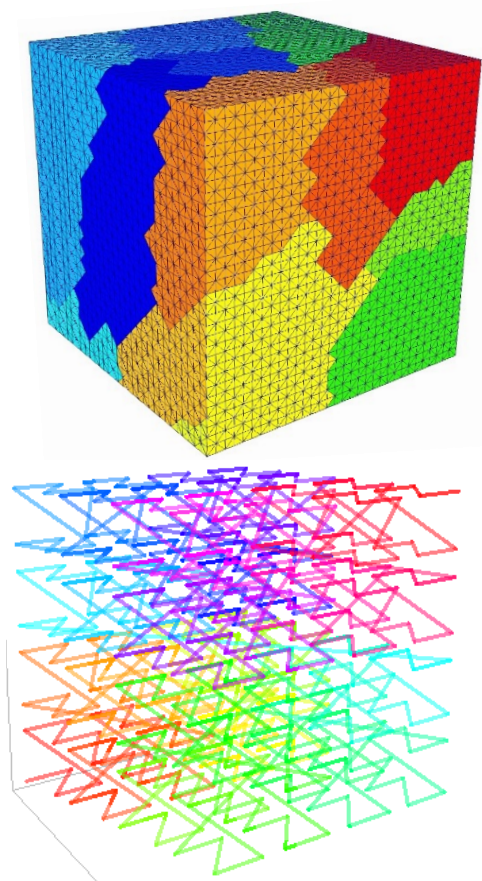
并行网格划分与数据分发



初始网格处理

空间填充曲线

负载均衡剖分

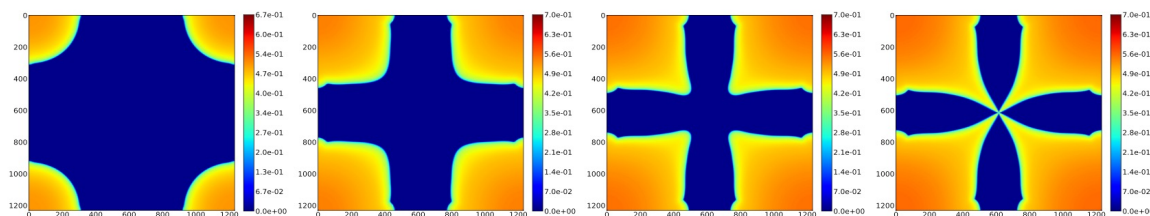
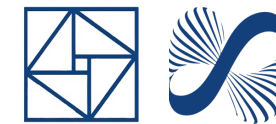


网格划分

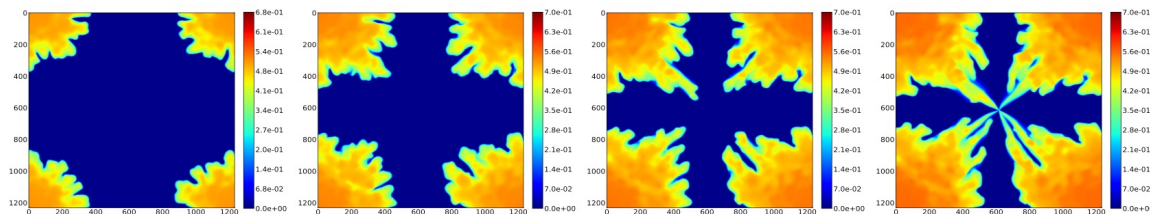
数据分发

多目标优化问题：每个部分的工作量的变化极小，界面面积极小，裂缝和井等特征不跨区域.....

非线性求解器的初值选取

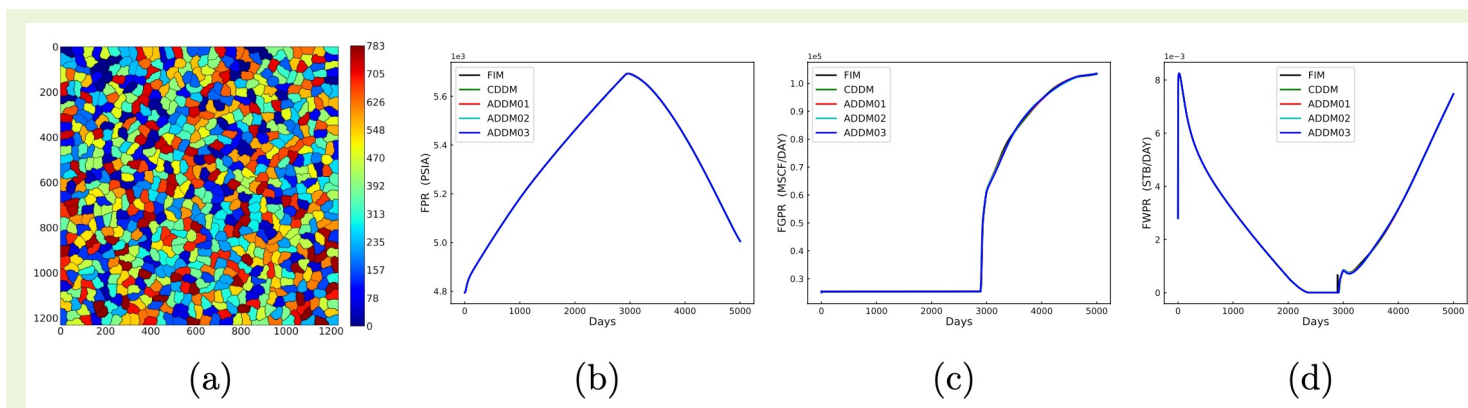


(a) 750 day (b) 2000 day (c) 2600 day (d) 3000 day



(e) 900 day (f) 1800 day (g) 2400 day (h) 3000 day

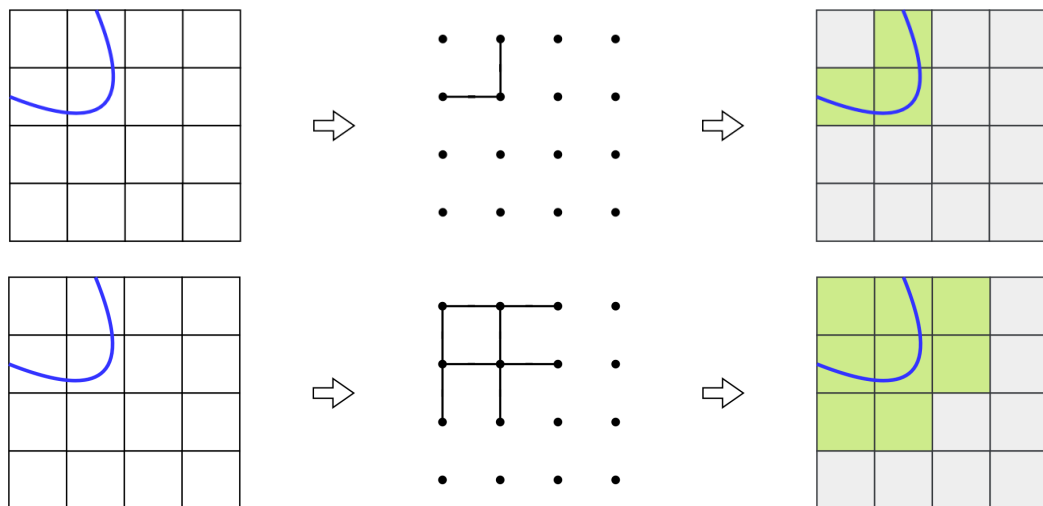
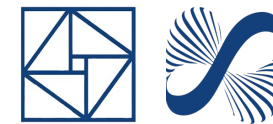
- 多组分多相流动非常复杂（特别是在各向异性和非均质性的介质中的流动），会产生复杂的流动前缘
- 为了解决强非线性导致的收敛困难，提供好的初始条件是最简单、最直接、最有效的手段
- 如何获得更好的初值呢？
 1. 减小时间步长（可能导致时间步长明显增多）
 2. 外推（一般是线性外插，不总能提供好初值）
 3. 求解近似问题（粗网格问题、子区域问题、.....）



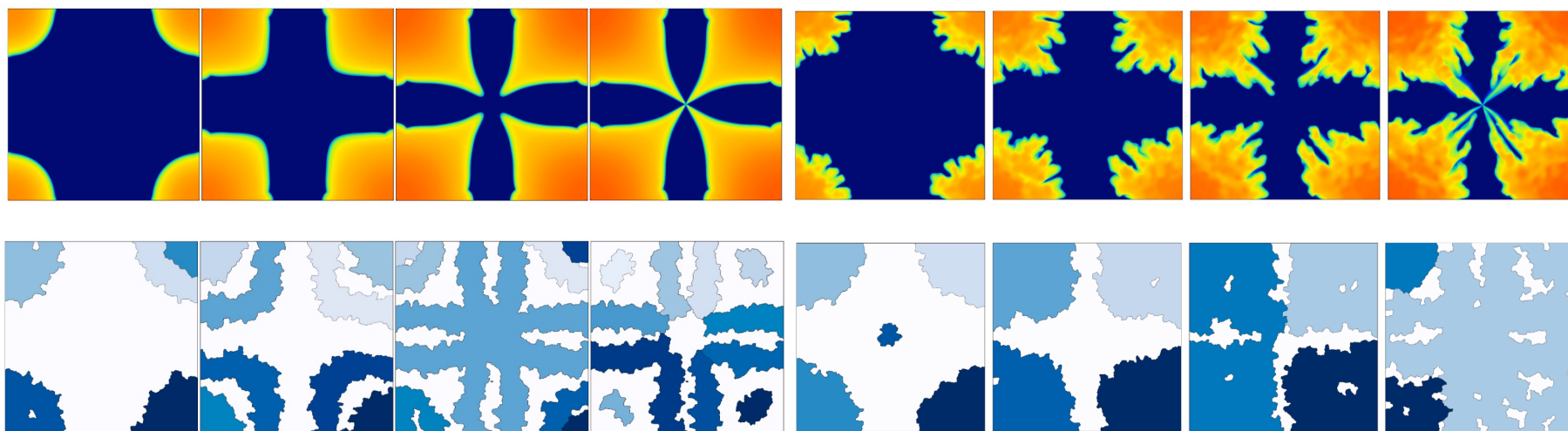
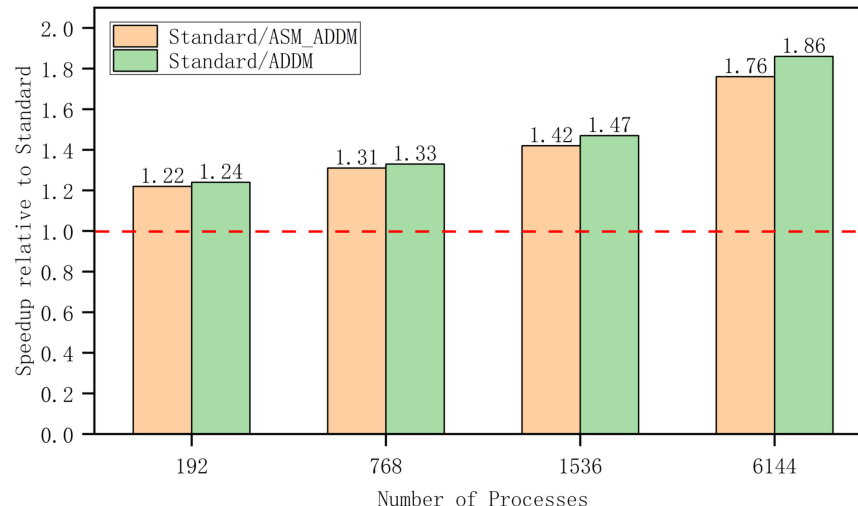
(a) (b) (c) (d)

- (a) domain partitioning by 784 processes using ParMetis
- (b), (c), and (d): field average pressure (FPR), field gas production rate (FGPR) and field water production rate (FWPR)
- Increasing number of domains could deteriorate convergence

自适应区域耦合ADDM方法

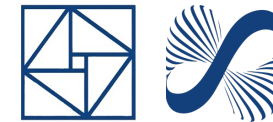


结果一致
效率提升



Ref: S. Li, L. Zhao, C.-S. Zhang, An adaptive subdomain coupling approach in domain decomposition for multiphase porous media flow, *Journal of Computational and Applied Mathematics*, 117644 (04/2026).

ADDM方法并行可扩展性测试



弱可扩展性

三相黑油模型问题

最大网格数量为1,001,089,600

最大自由度总数为4,004,358,400

表 6-20 测试问题 1 加密算例的节点间弱可扩展测试

Table 6-20 Inter-node parallel weak scalability test for the refined Case 1

网格规模	进程数	时间步	牛顿迭代数	线性迭代数	油藏更新 (s)	矩阵组装 (s)	解法器 (s)	总时间 (s)
15,642,025	192	41	137(+0)	534(+0)	7.76	20.3	146 (75.3%)	194
62,568,100	768	42	154(+0)	643(+0)	8.44	23.3	207 (79.3%)	261
125,136,200	1536	41	192(+0)	851(+0)	11.0	28.8	364 (83.7%)	435
500,544,800	6144	47	247(+16)	1215(+89)	16.8	43.8	927 (90.4%)	1025
1,001,089,600	12288	50	290(+36)	1518(+218)	20.8	50.0	1824 (94.4%)	1933

强可扩展性

多组分模型问题

网格数量为21,934,464

自由度总数为175,475,712

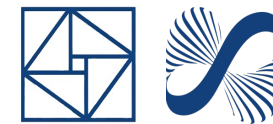
表 6-16 测试问题 4 扩展算例 2 的并行强可扩展性测试

Table 6-16 Parallel strong scalability test for the second extended case of Case 4

进程数	时间步	牛顿迭代数	线性迭代数	油藏更新 (s)	矩阵组装 (s)	解法器 (s)	总时间 (s)
512	20	68(+0)	225(+0)	24.1(16.7%)	11.6(8.06%)	96.1(66.7%)	144
1024	20	68(+0)	226(+0)	12.2(14.5%)	5.75(6.84%)	58.4(69.4%)	84.1
2048	20	67(+0)	229(+0)	6.19(10.0%)	2.82(4.56%)	47.2(76.3%)	61.9
4096	20	67(+0)	232(+0)	3.19(5.66%)	1.40(2.48%)	46.7(82.8%)	56.4
8192	20	67(+0)	243(+0)	1.68(1.69%)	0.71(0.71%)	91.3(91.9%)	99.4

- 李仕哲, 博士论文《多相多组分渗流力学并行数值方法与软件研究》, 中国科学院大学, 2024
- 高效性 (特别是可扩展性) 与稳健性是否可以兼得? 线性解法器 “不可能三角”
- 在CPR类预条件方法中, AMG和ILU是并行模拟的最主要瓶颈, 需要进一步改进或优化

整体求解的BAMG方法



核心问题：如何对向量型方程形成粗空间？

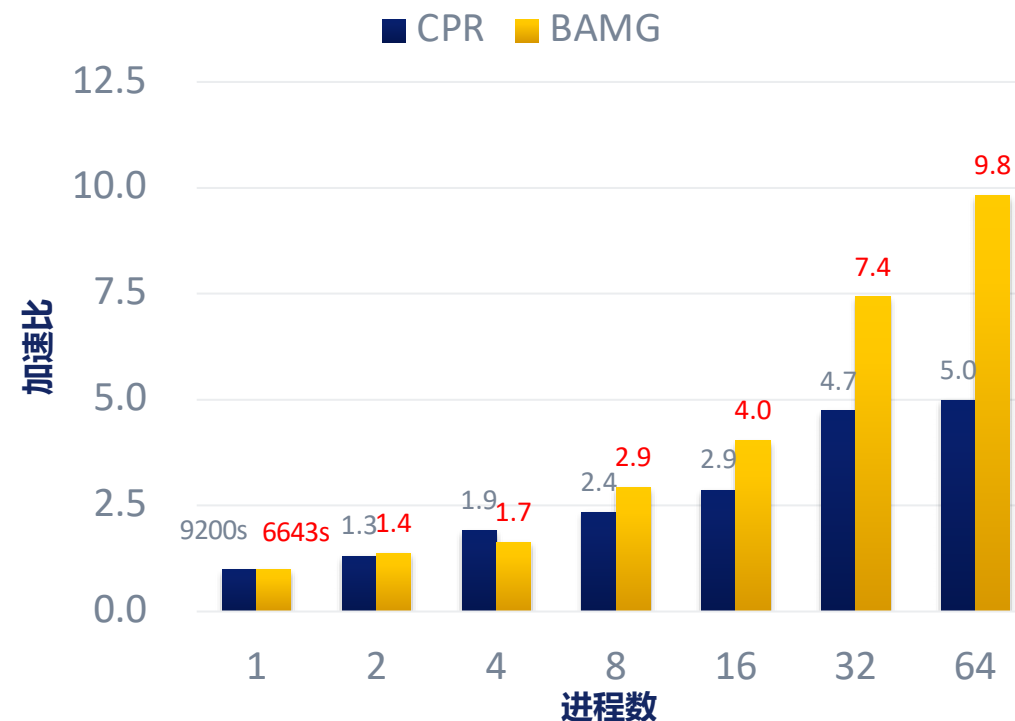
$$N_i(\varepsilon) = \left\{ j : \|W \odot A_{ij}\|_F > \varepsilon \sqrt{\|W \odot A_{ii}\|_F \|W \odot A_{jj}\|_F} \right\},$$

$$W_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad W_{22} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad W_{\text{all}} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

权矩阵	迭代次数	层数	算子复杂度
W_1	9	4	1.416
W_2	8	4	1.478
W_3	11	×	12.348
W_4	13	×	11.841
W_{all}	8	4	1.481
W_{22}	7	4	1.480

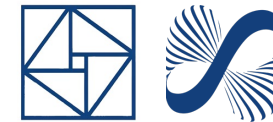
解耦方法	None	QI	ANL	SEM
W_2	3715	3757	3619	3906
W_{22}	3567	3551	3475	3637
迭代次数减少率	4.15%	5.80%	4.14%	7.40%

北京超算上的单节点强可扩展性测试
(非等温, 两相, 基于SPE10, 1.1百万网格)

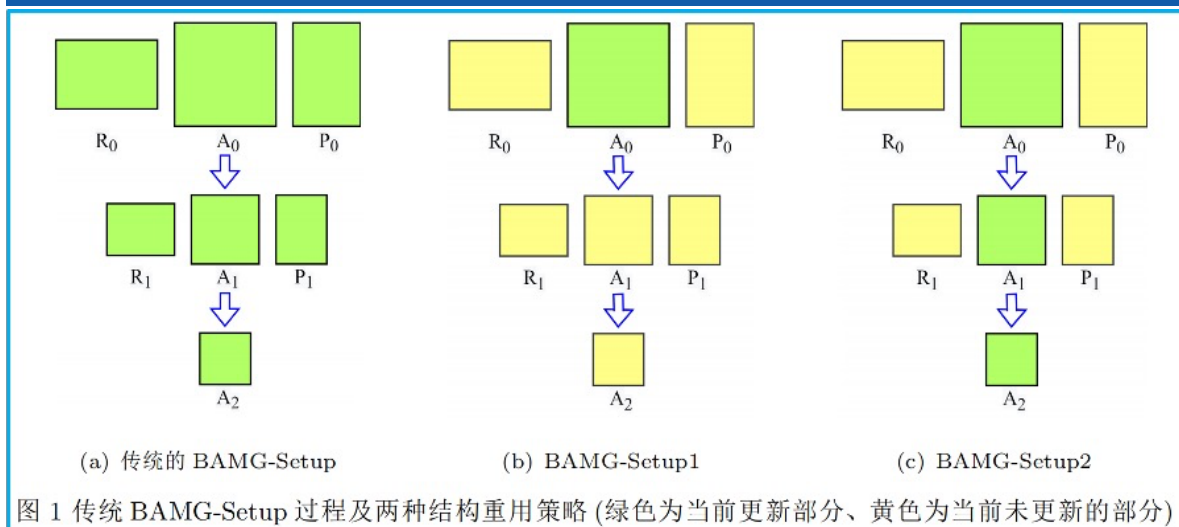


Ref: 赵梨, 博士论文《两类油藏模型的多水平预条件方法研究》, 湘潭大学, 2023

BAMG预条件的自适应初始化



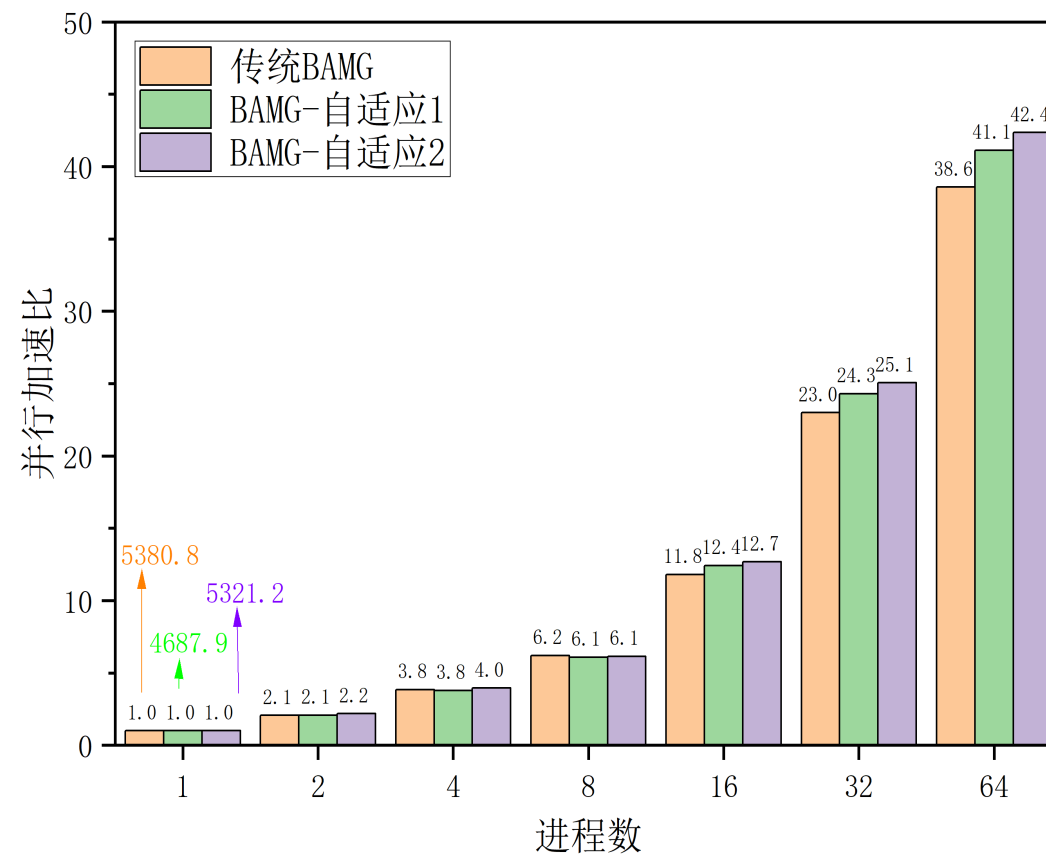
- BAMG的并行瓶颈是初始化阶段 (Setup)
- 利用矩阵的结构相似性, 复用插值、限制、粗网格矩阵等
- 设计自适应方法提升初始化效率和并行可扩展性



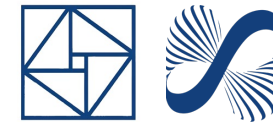
自适应准则:

$$B^k = \begin{cases} \text{AMG-Setup1 或 AMG-Setup2,} & \text{若 } IT \leq IT^{\text{ref}} + \alpha, \\ \text{AMG-Setup,} & \text{否则.} \end{cases} \quad \alpha = \frac{T_{\text{setup}}^{\text{ref}} - T_{\text{setup}}^{\text{share}}}{T_{\text{solve_single}}^{\text{ref}}}$$

盐城超算上的单节点强可扩展性测试 (多组分SPE5, 14.7万网格, 8个组分)



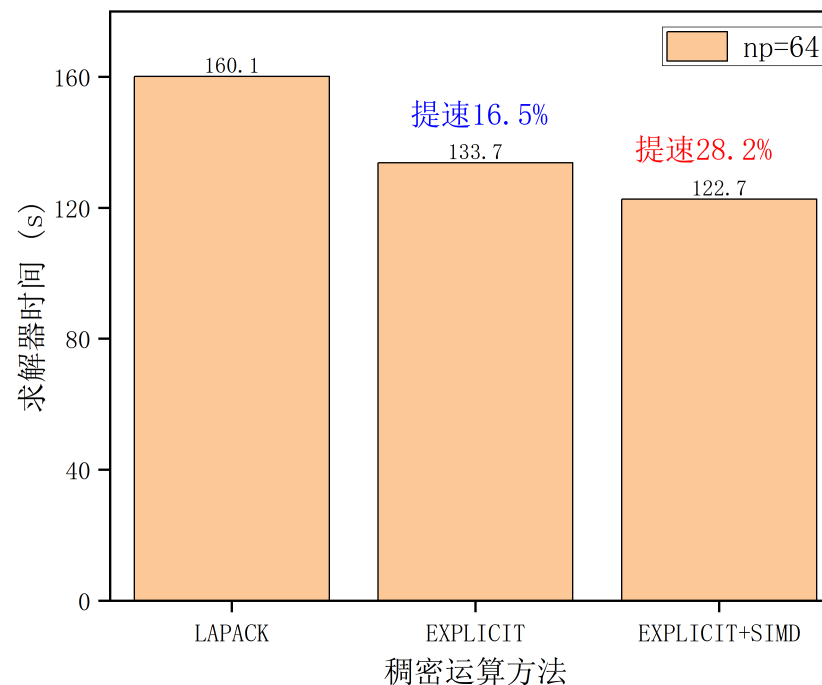
BAMG中稠密子块的向量化优化



- 稠密子块运算：矩阵向量乘、两矩阵乘、矩阵求逆
- EXPLICIT：显式展开 (roll-out)
- SIMD：单指令多数据流向量化方法

BAMG中稠密子块运算优化测试

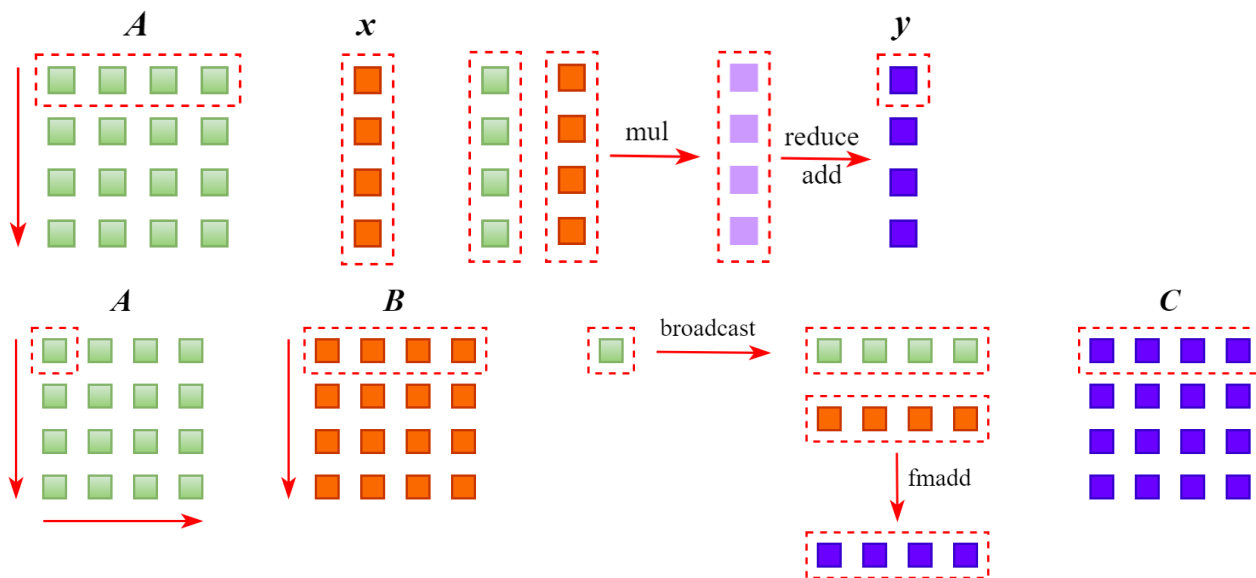
(多组分SPE5, 14.7万网格, 8个组分)



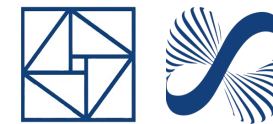
与稠密运算库LAPACK相比，**显式展开法提速16.5%**，**显式展开法+向量化并行提速28.2%**

$$A^{-1} = \frac{1}{|A|} A^*$$

```
const REAL det = a0*a3 - a1*a2;
REAL det_inv = 1.0 / det;
a[0] = a3 * det_inv; a[1] = -a1 * det_inv;
a[2] = -a2 * det_inv; a[3] = a0 * det_inv;
```



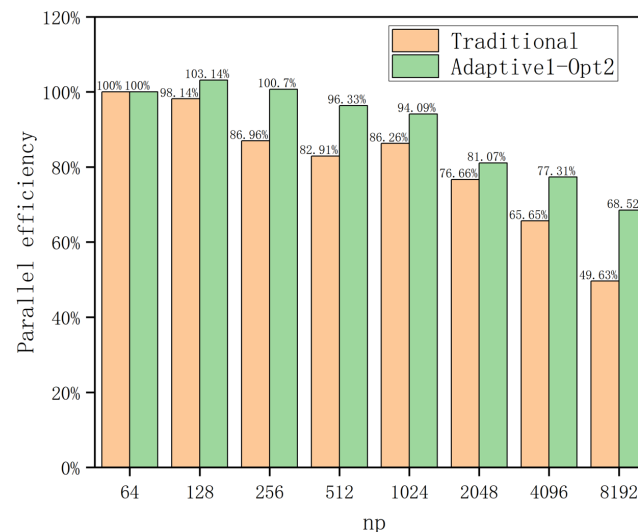
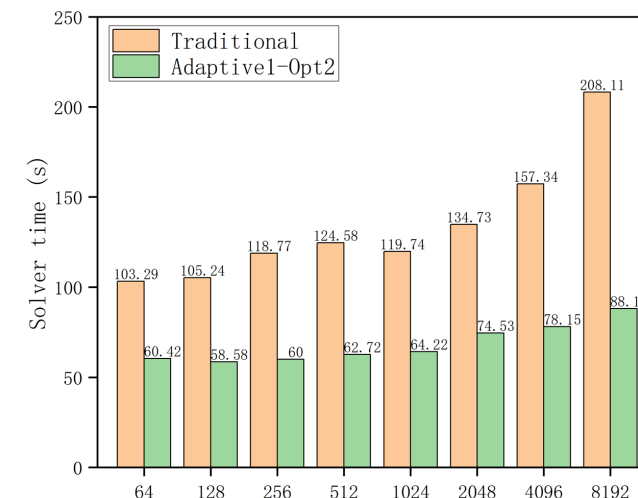
亿级网格测试：标准算例



盐城超算上的弱可扩展性测试

(多组分SPE5模拟100天, 每个进程1.4万网格, 8个组分)

np	Mesh	Traditional			Adaptive1-Opt2		
		IT	SolverTime	PE	IT	SolverTime (PI)	PE
64	390×390×6	345	103.29	100.00%	396	60.42 (41.50%)	100.00%
128	552×552×6	315	105.24	98.14%	373	58.58 (44.34%)	103.14%
256	781×781×6	316	118.77	86.96%	359	60.00 (49.48%)	100.70%
512	1104×1104×6	295	124.58	82.91%	353	62.72 (49.66%)	96.33%
1024	1561×1561×6	281	119.74	86.26%	340	64.22 (46.37%)	94.09%
2048	2208×2208×6	288	134.73	76.66%	361	74.53 (44.68%)	81.07%
4096	3123×3123×6	291	157.34	65.65%	359	78.15 (50.33%)	77.31%
8192	4416×4416×6	298	208.11	49.63%	336	88.18 (57.63%)	68.52%

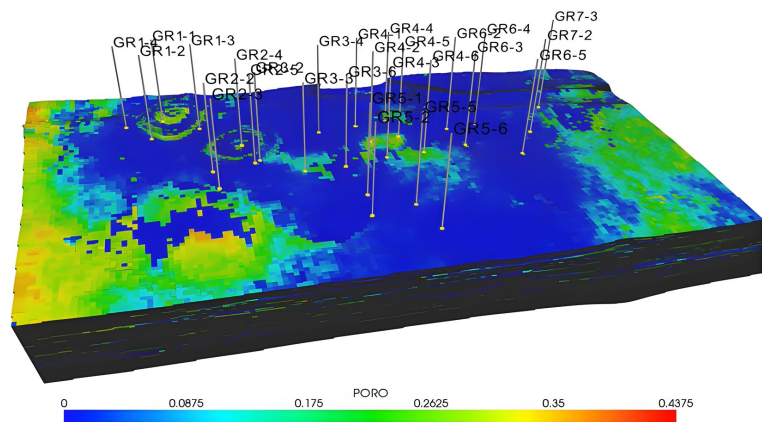
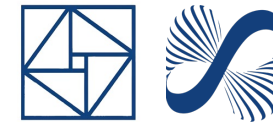


1.2 亿网格

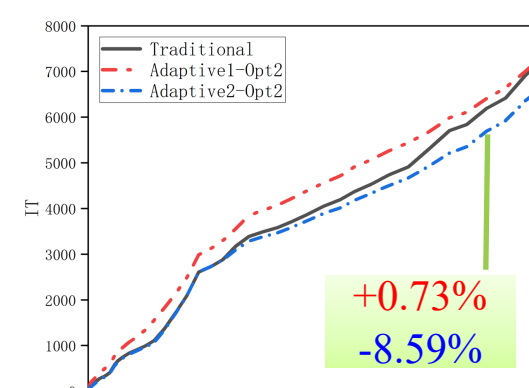
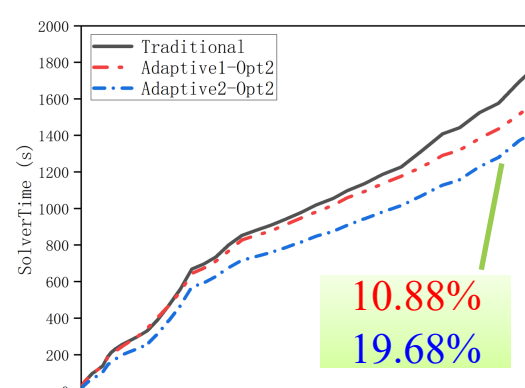
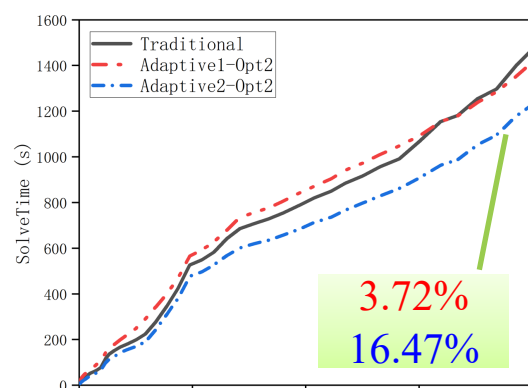
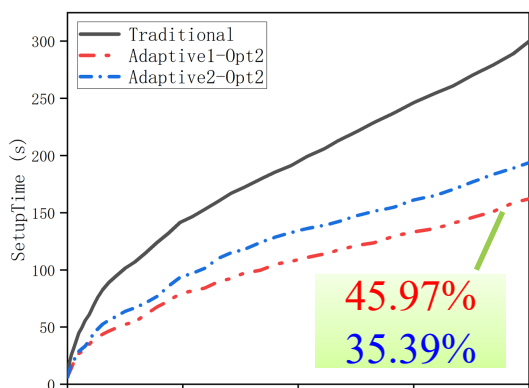
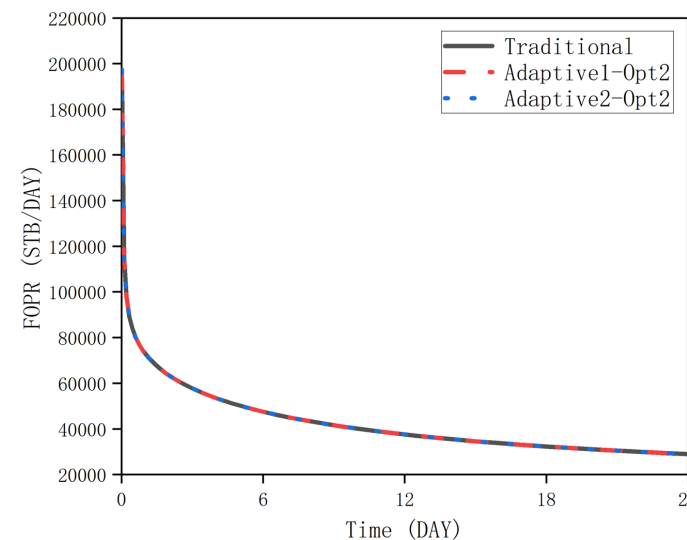
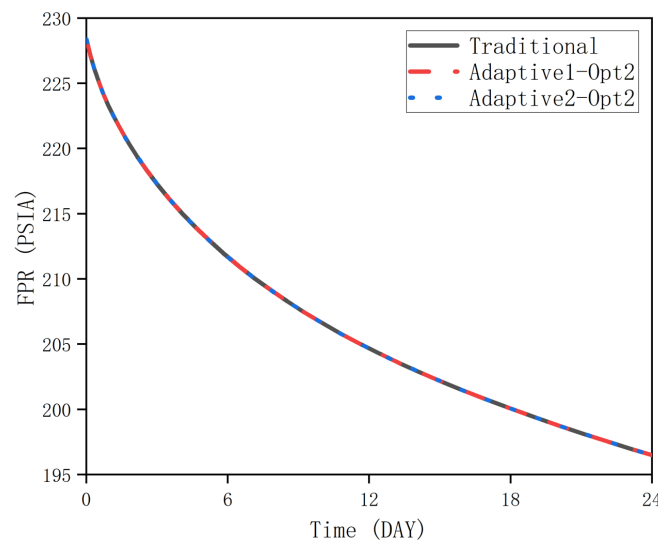
- 计算时间提速：57.63%
- 并行效率提升：38.06%

PE: 并行效率, PI: 性能提升比

亿级网格测试：实际算例



- 网格588 X 328 X 560，总数超1亿，黑油模型
- 使用3072个进程数，模拟油藏启动阶段24天



相较于Hisim中的多阶段预条件方法，自适应启动BAMG方法可获得近20%的性能加速，模拟时间越长可以获得的性能提升越高

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

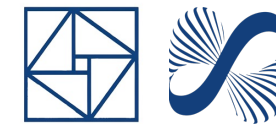
Neural Network
Methods

08

软件项目

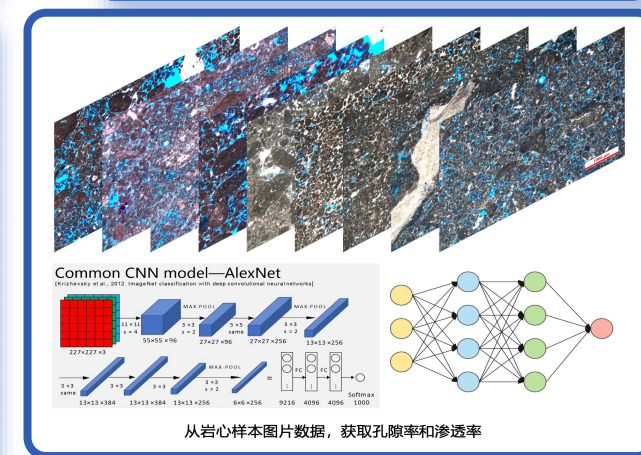
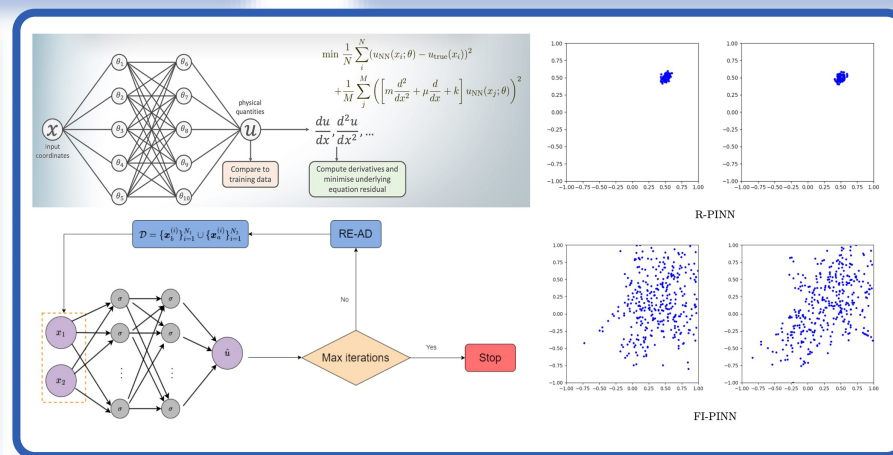
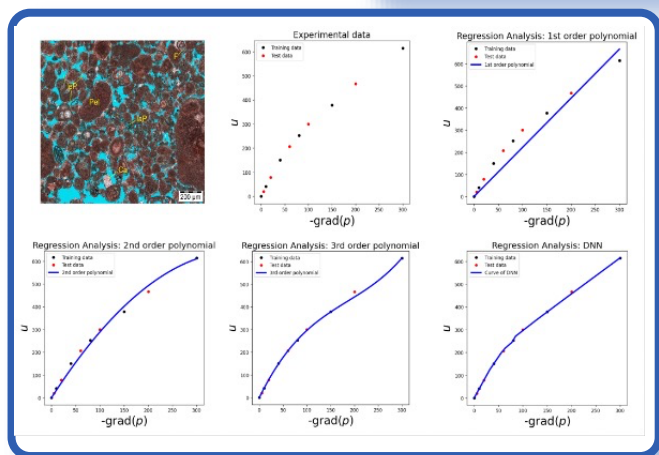
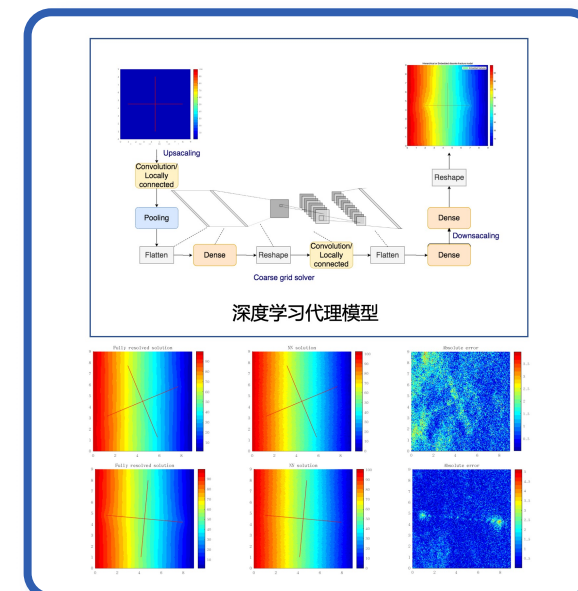
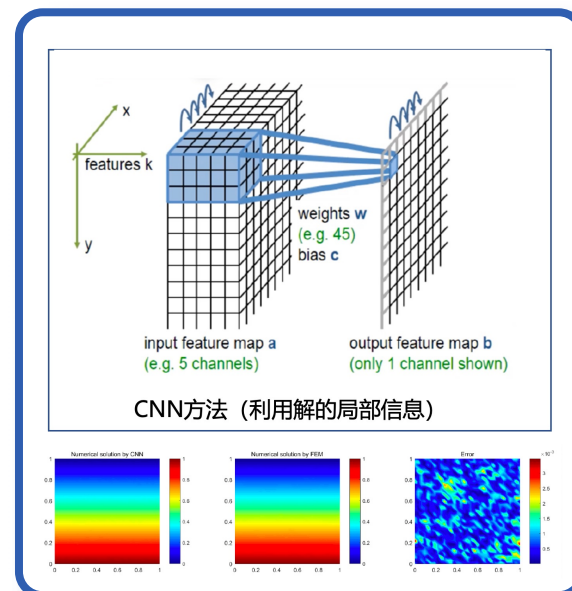
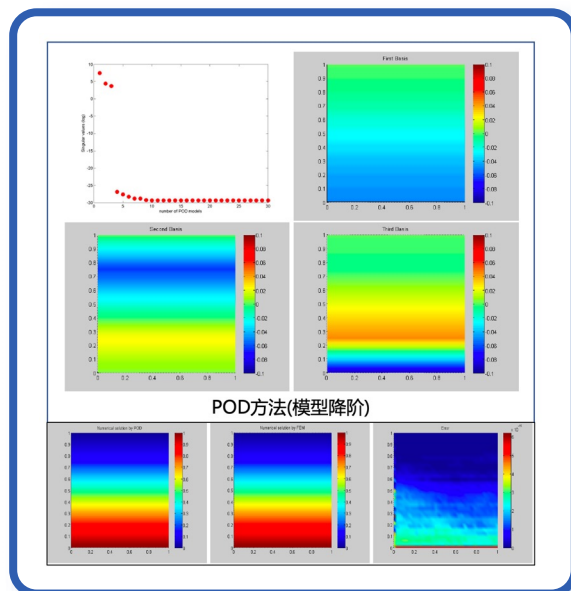
Software
Projects

智能渗流模拟技术路线探索



2019

2021



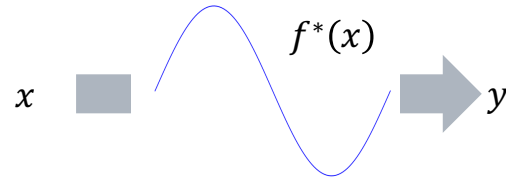
Neural Operator Learning



Data $\{(x_j, y_j)\}_{j=1}^N$

NN function space

$$\Xi_n := \left\{ \sum_{i=1}^n a_i \sigma(w_i x + b_i) \mid a_i, b_i, w_i \in \mathbb{R}, i = 1, \dots, n \right\}$$



NN function approx

$$f^{NN} := \operatorname{argmin}_{f \in \Xi_n} \frac{1}{N} \sum_{i=1}^N [\|f(x_i) - y_i\|^2]$$

- Classical model-based numerical methods

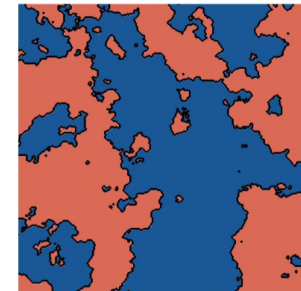
- Require a-prior knowledge of the underline PDE
- Model might not be accurate enough
- Model might be too complicate for computing

- Data-based NOL methods

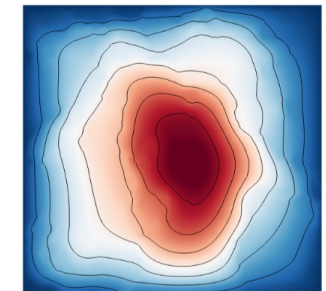
$$\left\{ (\kappa_j, u_j) : \kappa_j \stackrel{\text{iid}}{\sim} \mu, u_j = \mathcal{O}^*(\kappa_j) \right\}_{j=1}^N$$

- Infinite dimensional regression
- Treat input and output as elements in function spaces

κ



u



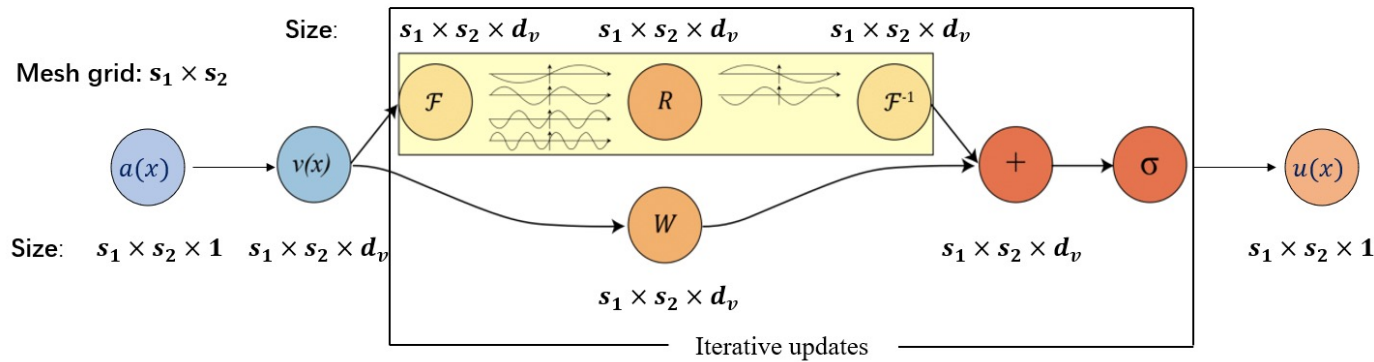
$$\begin{cases} -\nabla \cdot (\kappa(x) \nabla u(x)) = f(x), & x \in D \\ u(x) = 0, & x \in \partial D \end{cases}$$

Resolution Invariance: Trained at low resolution and perform zero-shot super-resolution without retraining

FNO Neural Operator Architecture



$$\mathcal{K}_\theta(\mathbf{v})(x) := (\mathbf{k}_{\theta_\ell} \star \mathbf{v})(x) = \int_D \mathbf{k}_{\theta_\ell}(x - y) \mathbf{v}(y) dy, \quad \forall x \in D.$$



$$v_{t+1}(x) := \sigma\left(Wv_t(x) + (\mathcal{K}(a; \phi)v_t)(x)\right), \quad \forall x \in D$$

FNO

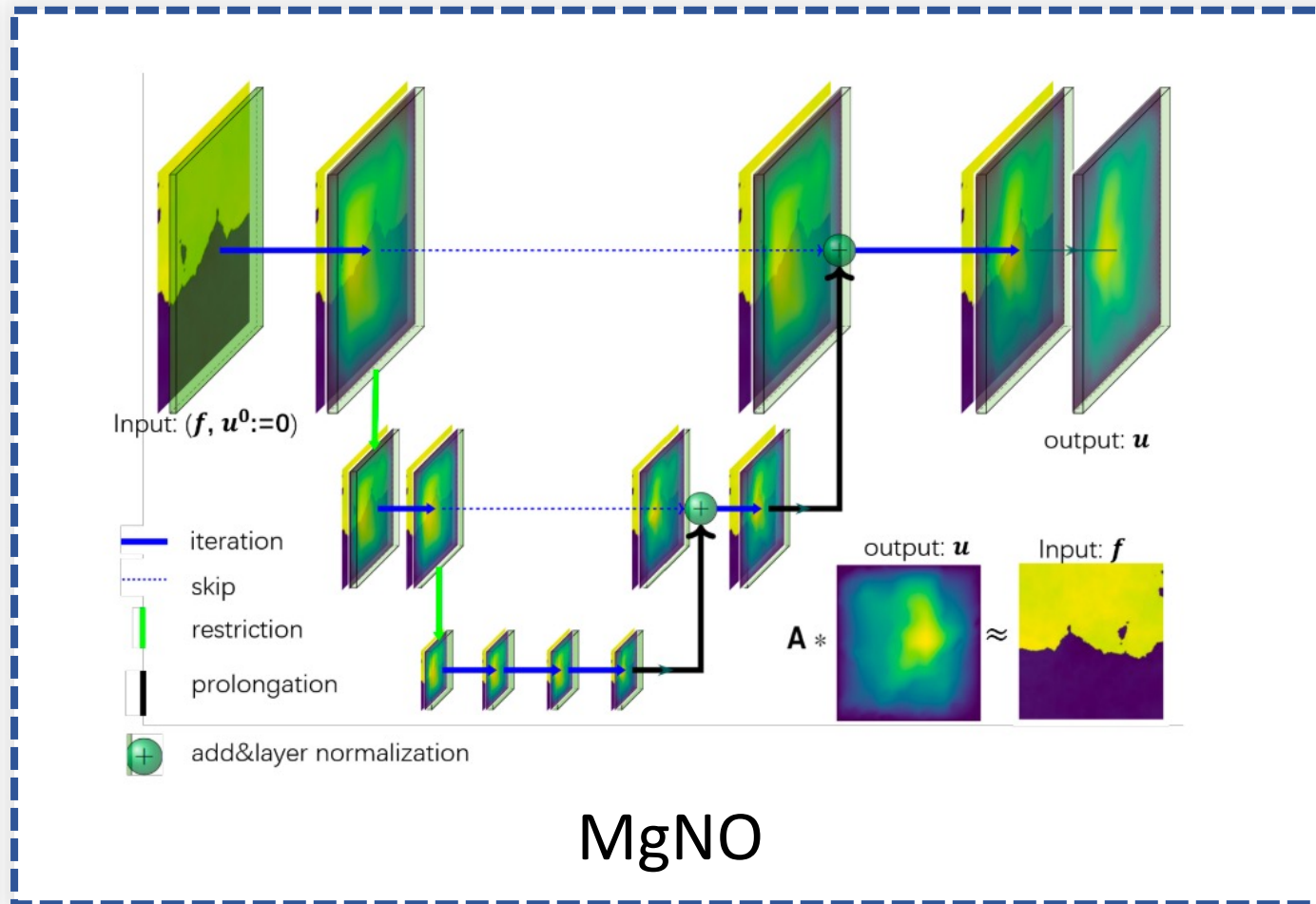
Key

Fourier Neural Operator

- ✓ Resolution invariant due to (latent) Fourier transform in the convolution layers
- ✓ Fast implementation using FFT operations
- ✓ Fourier transform can be replaced by other transforms such as wavelet (MWT)

Ref: Zongyi Li et al. Fourier Neural Operator for Parametric Partial Differential Equations, ICLR 2022

MgNO Neural Operator Architecture



Key

Multigrid Neural Operator

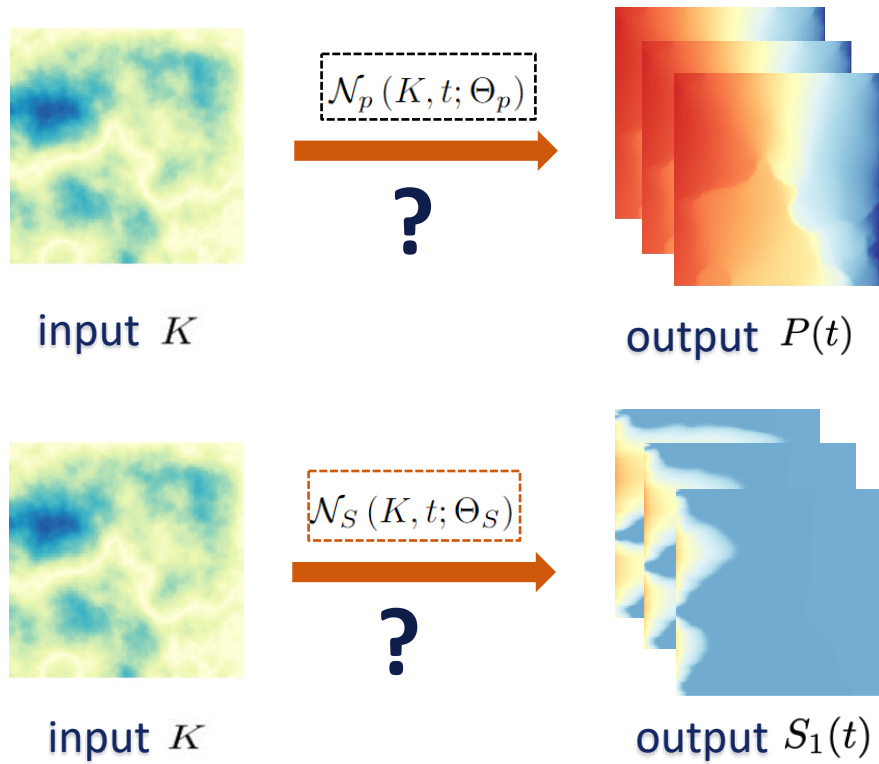
- ✓ Implement a neural network version of the multigrid method by defining Mglte
- ✓ Make use of multi-channel V-cycles in Mglte instead of one
- ✓ Incorporate parameterization of linear operators and the ability to handle given boundary conditions, ensuring accurate simulation

Ref: He, Juncai, Xinliang Liu, and Jinchao Xu. MgNO: Efficient parameterization of linear operators via multigrid, ICLR 2024

Extension to Multiphase Flow Problems



Neural Operators \mathcal{N}_p and \mathcal{N}_S



Two-phase oil-water model on a uniform grid $128 \times 1 \times 128$ with step length of 10 inches. The initial porosity is 0.3, and the initial water saturation is 0.2. The simulation runs for 60 days. The distribution of absolute permeability K follows Equation $A * |N(0, (-\Delta + 9I) - 2)|$, where $A = 10$ represents the jump magnitude.

- K is independent of time; If expressed as a time series, it can be written as $K(x, t_0) = K(x, t_1) = \dots = K(x)$
- In each cell, K_i is a diagonal matrix (assuming the absolute permeabilities in the x , y , and z directions are K_{x_i} , K_{y_i} , and K_{z_i} , respectively)

Ref: Liu, X., Yang, X., Zhang, CS., et al. (2025). Proceedings of the 16th Annual Meeting Conference on Porous Media

Training Parameters



Dataset size: 2000 * 25
Timesteps: 0-24
Resolution: 128*1*128

ntrain = 1600 * 25
loss_type: 'L2' or 'H1'
epoch: 500
weight_decay: 1e-5

ntest = 400 * 60
batch_size: 50
learning_rate: 1e-4
scheduler: Adam

Same
settings for
FNO &
MgNO

Loss Functions

$$L2(y, \hat{y}) = \frac{\|\hat{y} - y\|_2}{\|y\|_2}$$

$$H1(y, \hat{y}) = \|y - \hat{y}\|_{L_2}^2 + \|\nabla y - \nabla \hat{y}\|_{L_2}^2$$

K-S Model: input K and T_i , output water saturation at timesteps 0-24

K-P Model: input K and T_i , output pressure field at timesteps 0-24

Numerical Results: FNO vs MgNO

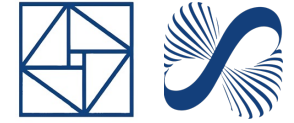


Prediction	Model	Parameter Count	Relative L2 Error	Time (s) per Sample
K-P	MgNO-L2	18,500,124	7.88E-03	1.34E-03
	MgNO-H1	18,500,124	7.92E-03	1.48E-03
	FNO-L2	11,989,761	9.51E-03	7.99E-04
	FNO-H1	11,989,761	9.34E-03	7.87E-04
K-S	MgNO-L2	18,500,124	2.28E-02	1.45E-03
	MgNO-H1	18,500,124	2.40E-02	1.48E-03
	FNO-L2	11,989,761	4.60E-02	9.07E-04
	FNO-H1	11,989,761	4.08E-02	7.98E-04

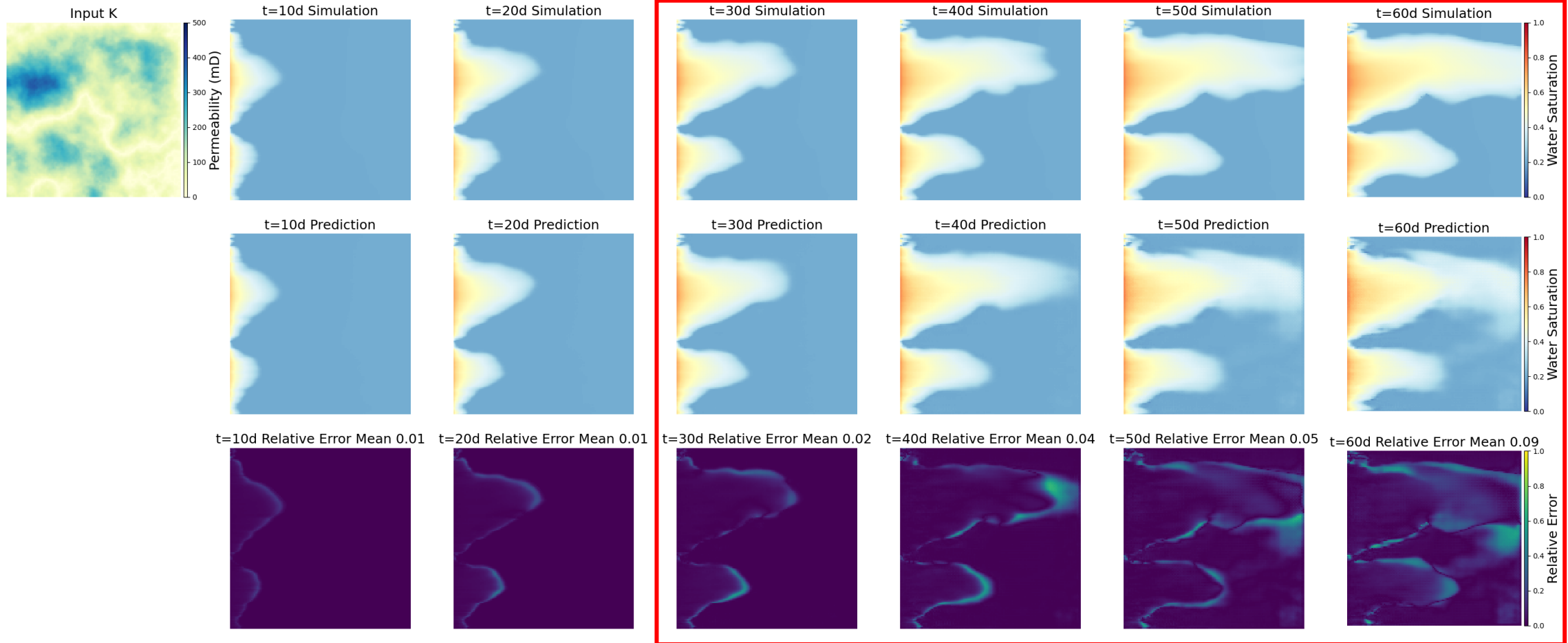
Comparison: Classical simulation requires 0.164s for each sample step (125 times speedup)

Hardware: 2 * Huawei Kungpeng-920 @ 3.0GHz, 4 * Nvidia A100 PCIe 40GB

MgNO: K-S Model Results



MgNO-L2-KS: Simulation Results, Predictions, and Errors at Different Time Steps

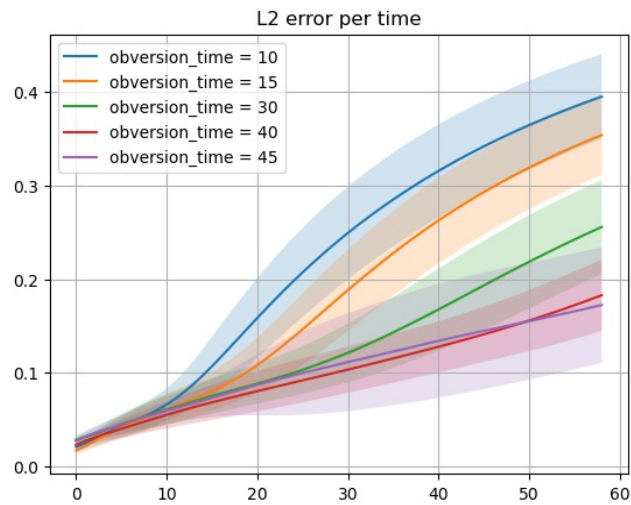


Recurrent Neural Operator

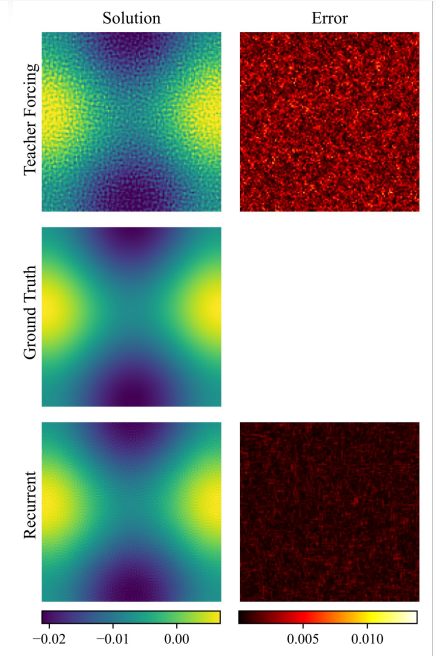
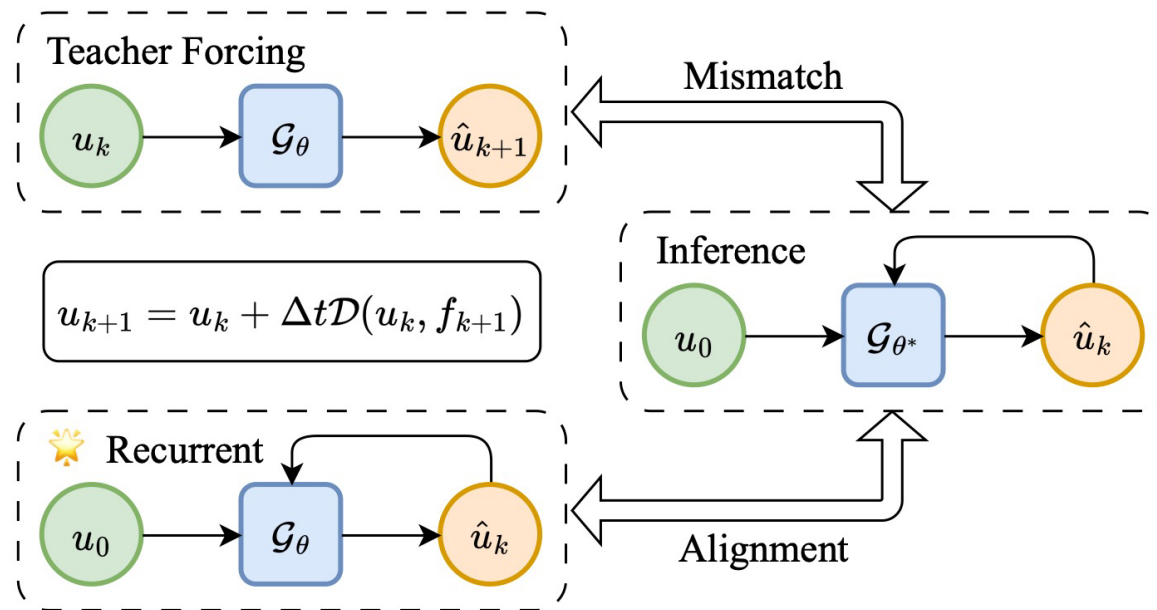


Key

- How to enhance the predictive capability of neural operators for evolution equations under the constraint of limited observational data?
- Align the training and inference processes to reduce distribution shift in the NO predictions



Long-term prediction requires sufficient data



Error Estimates on RNO



Key

Error Bound Comparison: Teacher Forcing vs. Recurrent Training

- Recurrent training is likely to reduce the worst-case exponential error growth (typical in teacher forcing) to linear growth.

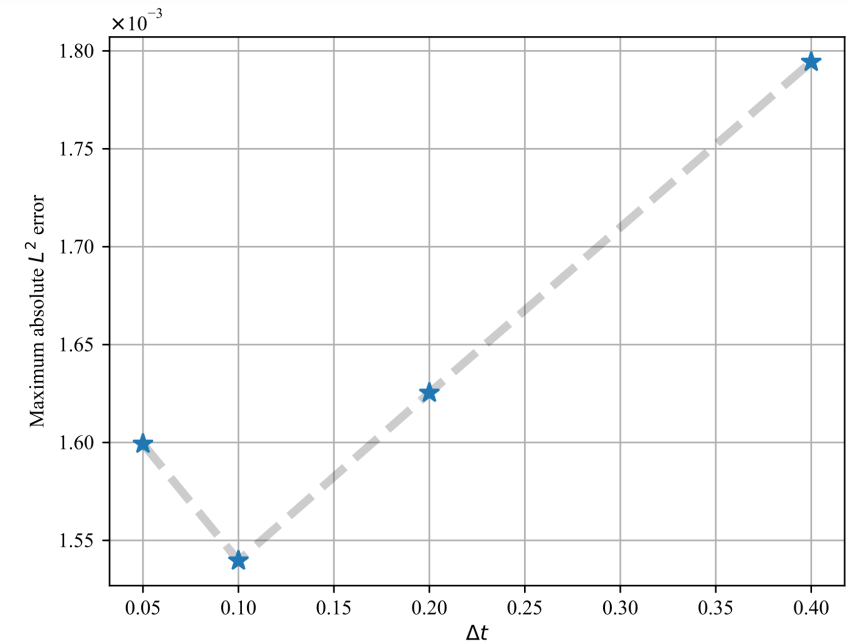
Let \hat{u}_n^{TF} and \hat{u}_n^{RNO} be predictions from models trained using Teacher Forcing and Recurrent Training methods, respectively. Assume that approximation error of NO is bounded by ϵ .

Teacher forcing error grows exponentially with time:

$$\max_{0 \leq n \leq N} \|\hat{u}_n^{TF} - u_n\| \leq e^{CT} \|\hat{u}_0 - u_0\| + (e^{CT} - 1) C^{-1} (\epsilon + O(\Delta t))$$

Recurrent error grows linearly with time:

$$\max_{0 \leq n \leq N} \|\hat{u}_n^{RNO} - u_n\| \leq \|\hat{u}_0 - u_0\| + T (\epsilon + O(\Delta t))$$



Ref: Ye Z, Zhang C S, Wang W. Recurrent Neural Operators: Stable Long-Term PDE Prediction. arXiv:2505.20721, 2025.

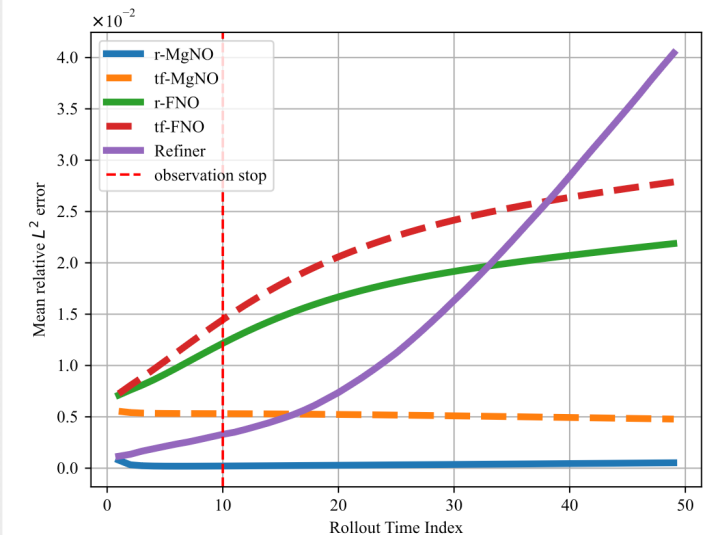
Numerical Results on RNO



Comparison of mean relative L2 error for rollout prediction at $n = 5$ (interpolation) and $n = 50$ (extrapolation) steps. Models were trained up to $n = 10$. Lower is better.

Model (Param)	Heat		Allen-Cahn		Cahn-Hilliard		Navier-Stokes	
	$n = 5$	$n = 50$	$n = 5$	$n = 50$	$n = 5$	$n = 50$	$n = 5$	$n = 50$
Refiner (9.1M)	3.7e-03 (±3.2e-04)	5.1e-02 (±3.8e-02)	1.7e-03 (±2.2e-04)	2.9e-02 (±6.7e-03)	1.5e-03 (±2.4e-04)	6.1e-03 (±4.5e-04)	3.8e-02 (±4.8e-02)	1.4e-01 (±1.8e-01)
tf-FNO (1.1M)	3.9e-02 (±8.9e-04)	1.8e-01 (±1.3e-01)	1.0e-02 (±8.6e-04)	3.3e-02 (±5.4e-03)	3.7e-02 (±5.3e-03)	1.6e-01 (±7.0e-03)	5.7e-02 (±2.6e-03)	2.4e+02 (±3.0e+02)
tf-MgNO (0.8M)	7.4e-03 (±1.9e-04)	2.9e-02 (±2.2e-03)	6.0e-03 (±8.5e-04)	5.8e-03 (±1.1e-03)	1.3e-03 (±2.4e-04)	3.4e-03 (±8.4e-04)	2.2e-02 (±1.7e-02)	1.4e-01 (±9.5e-02)
r-FNO (1.1M)	3.2e-02 (±3.1e-03)	1.2e-01 (±6.1e-02)	9.7e-03 (±1.3e-03)	3.0e-02 (±7.4e-03)	3.2e-02 (±5.6e-03)	2.1e-01 (±3.2e-02)	5.5e-02 (±2.9e-03)	3.7e-01 (±3.5e-01)
r-MgNO (0.8M)	3.6e-03 (±3.3e-04)	1.1e-02 (±1.4e-03)	3.1e-04 (±1.2e-04)	6.0e-04 (±3.8e-05)	1.2e-03 (±2.4e-04)	2.9e-03 (±6.2e-04)	3.4e-03 (±6.6e-04)	8.9e-02 (±5.4e-02)

Error accumulation during the long-term rollout process based on the Allen-Cahn problem.



Ref: Lippe P, Veeling B, Perdikaris P, et al. PDE-Refiner: Achieving accurate long rollouts with neural PDE solvers. *Advances in Neural Information Processing Systems*, 2023, 36: 67398-67433. PDE-Refiner serves as a strong baseline method.

报告内容

01

背景介绍

Application
Background

02

数学模型

Mathematical
Modeling

03

离散方法

Discretization
Methods

04

代数求解

Algebraic
Solution

05

平衡计算

Flash
Calculation

06

并行计算

Parallel
Computing

07

神经网络

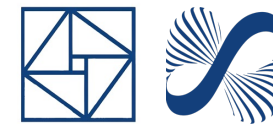
Neural Network
Methods

08

软件项目

Software
Projects

OpenCAEPoro软件开发团队



当前版本代码贡献者

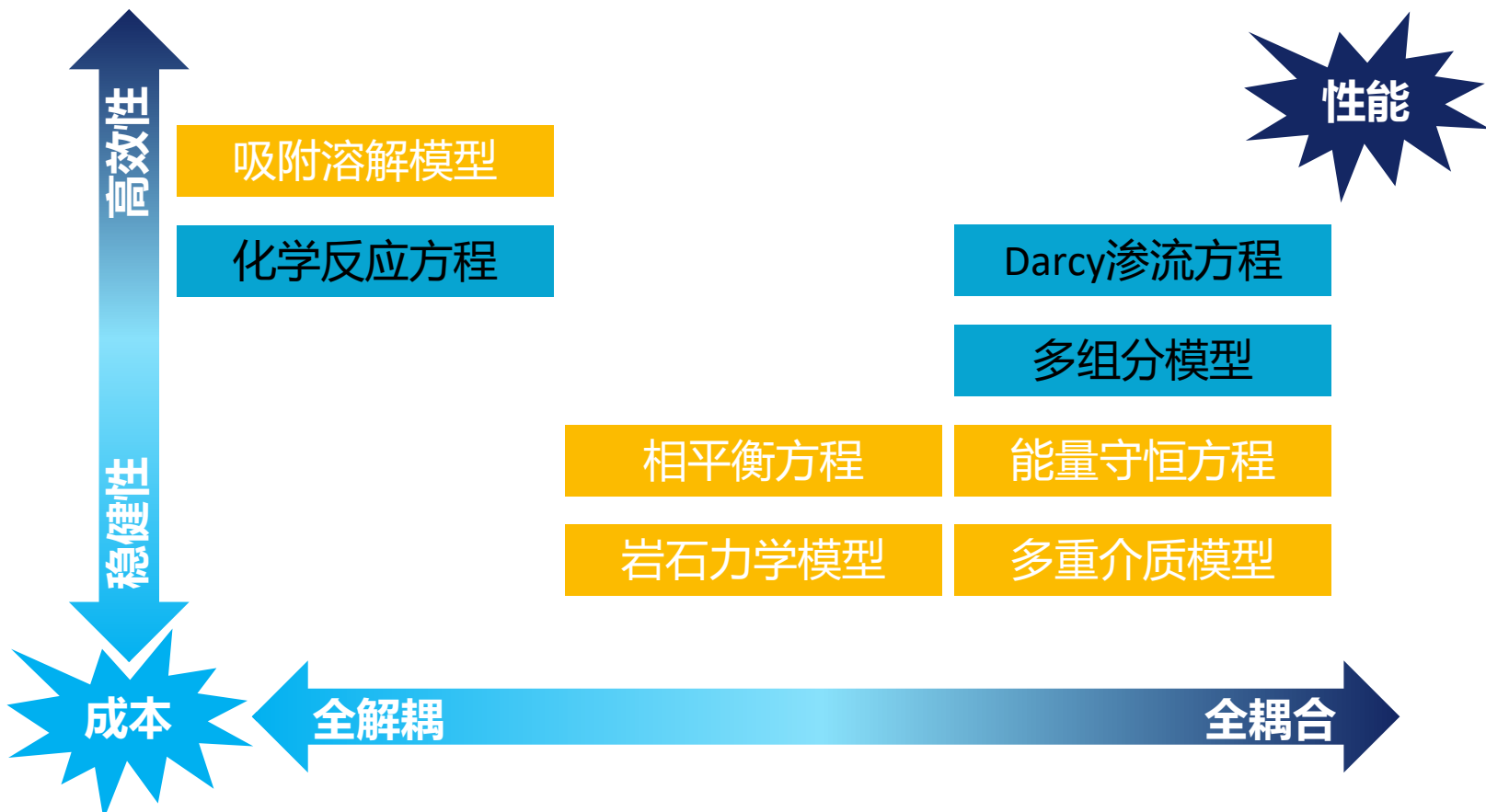
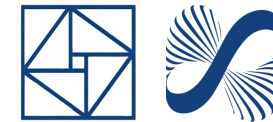
- 张晨松, 中科院数学院
- 李仕哲, 中科院数学院
- 赵 梨, 中科院数学院
- 冯春生, 湘潭大学
- 范荣红, 深圳大数据研究院
- 李 瑞, 陕西师范大学

早期版本代码贡献者

- 乔长河, 宾州州立大学
- 张鸿轩, 宾州州立大学
- 关文超, 中山大学
- 李 政, 湘潭大学
- 王锦东, 北京大学
- 王 慧, 湘潭大学
- 赖 婷, 湘潭大学
- 刘生豪, 湘潭大学
- 张林杰, 湘潭大学
- 齐浩然, 宾州州立大学
- 沈博芊, 中科院数学院
- 程晓行, 深圳大数据研究院

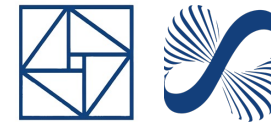
<https://github.com/opencaeplus>

OpenCAEPoro求解方法的选择



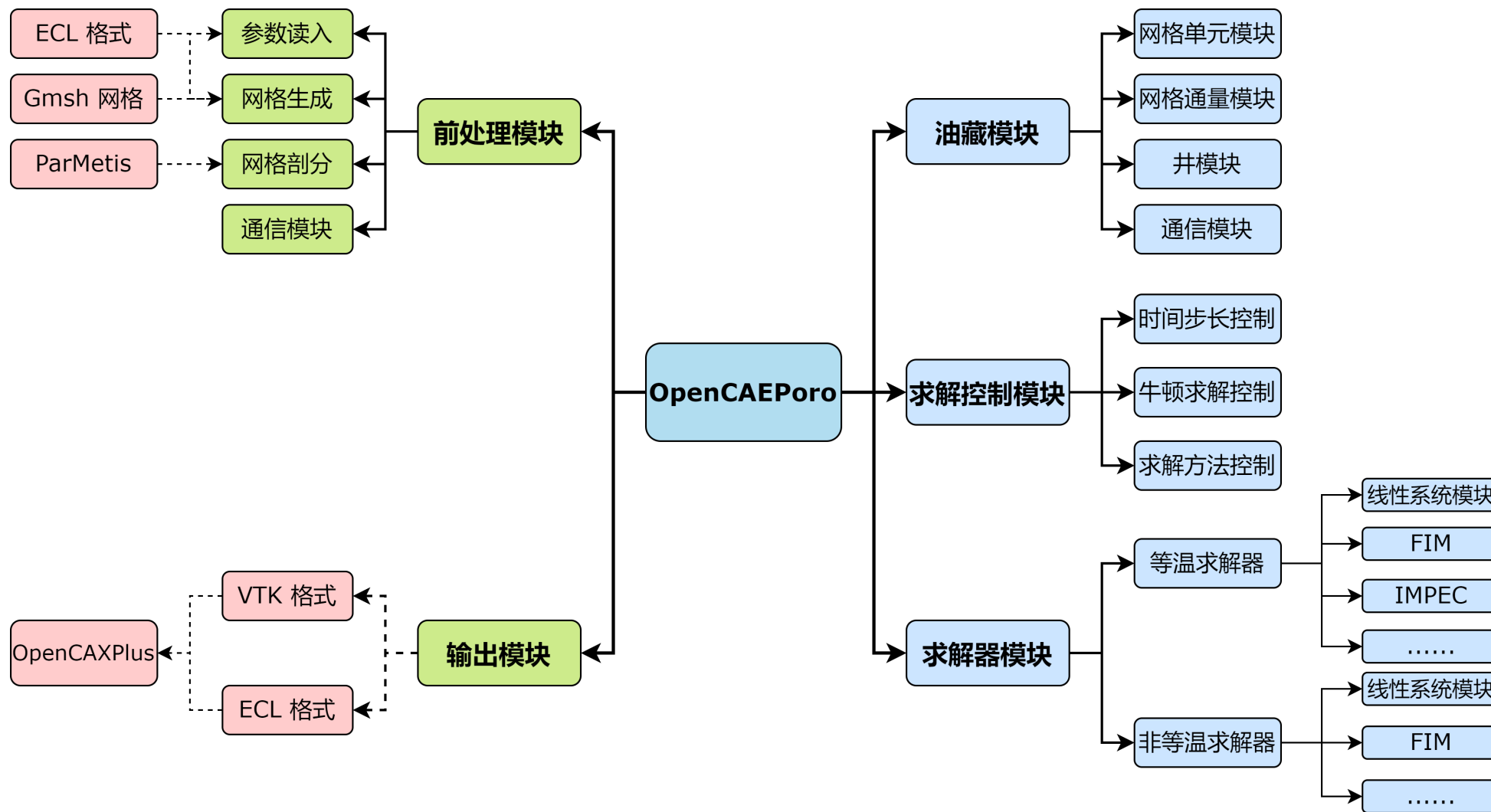
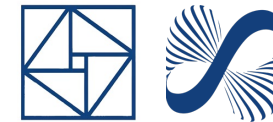
全隐式方法 + 向后Euler时间离散 + 自适应时间步长 + Newton线性化 + 无结构网格
 + TPF有限体积法 + 上游加权 + 拟Newton法 + 半解析解耦方法 + 多阶段预条件 + FGMRES

OpenCAEPoro主程序的计算流程

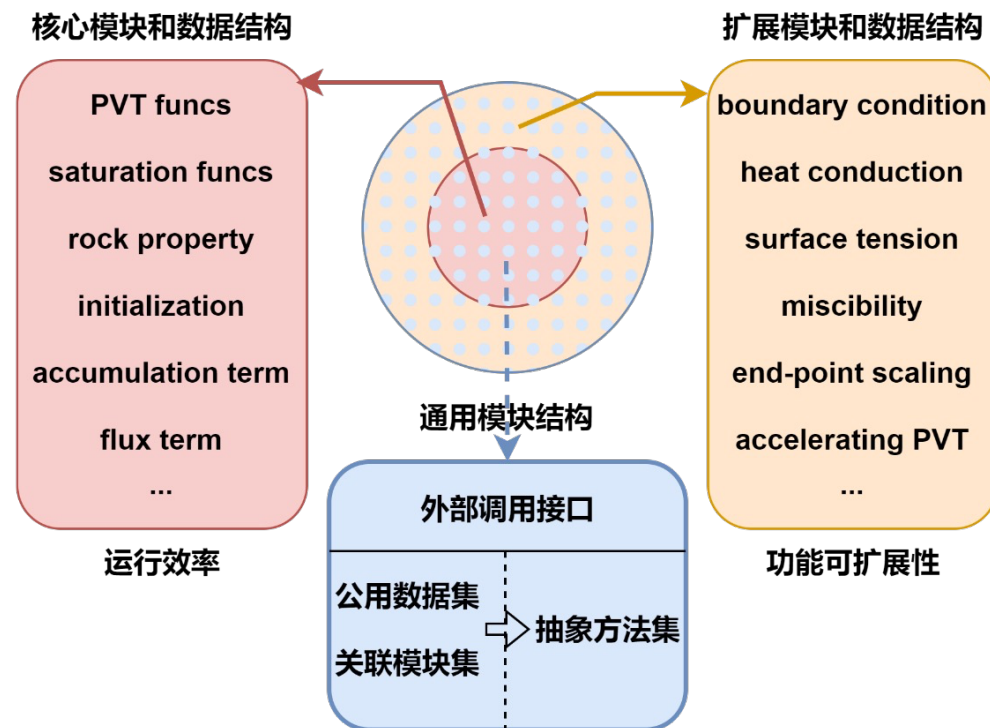
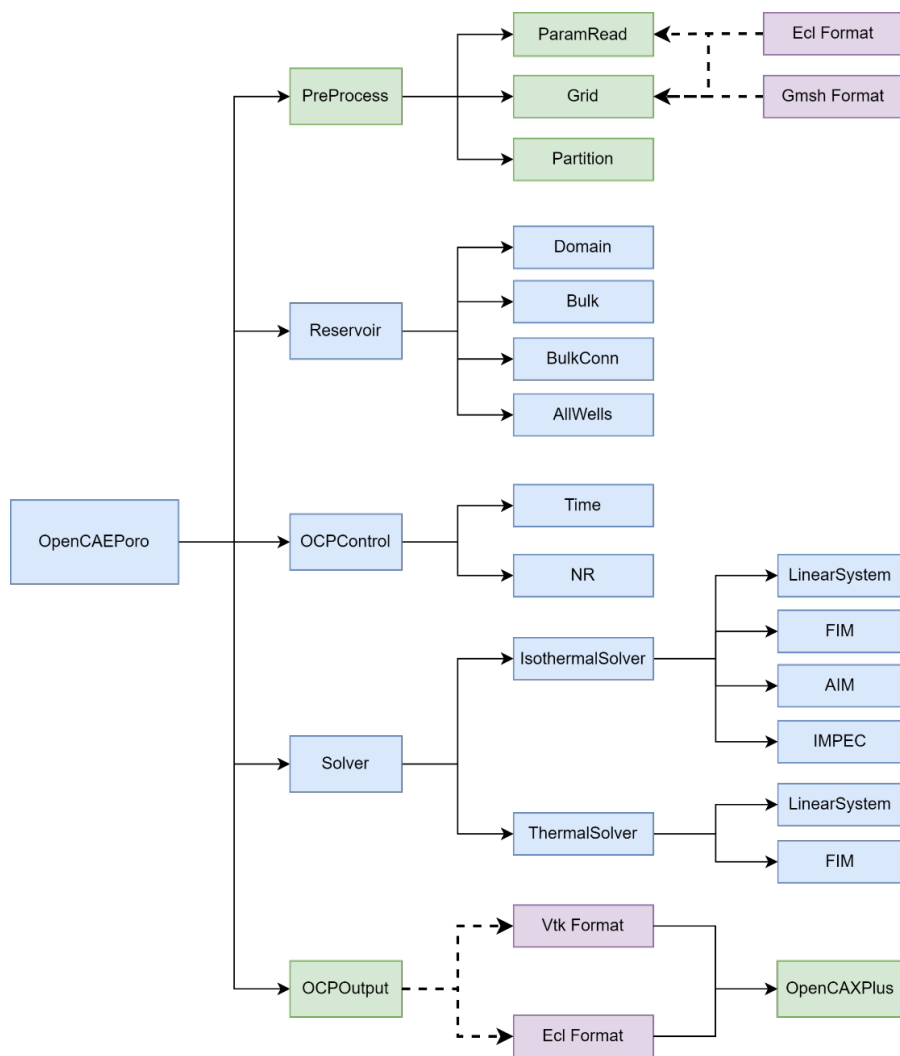
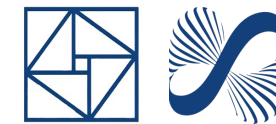


网格数据读入 与预处理	<pre>PreParamGridWell::InputFile() // 主进程读入网格相关的参数 PreParamGridWell::Setup() // 设置活跃网格, 构建拓扑连接关系 Partition::SetPartition() // 调用 Metis/ParMetis 对网格进行并行划分 Partition::SetDistribution() // 重新分配网格单元至各个进程 Domain::Setup() // 构建通信系统</pre>
参数配置	<pre>ParamRead::InputFile() // 所有进程读入网格无关的参数 Reservoir::Setup() // 设置网格属性, 创建物理模型 Control::Setup() // 装配求解控制模块 Output::Setup() // 装配结果输出模块</pre>
求解设置	<pre>Solver::Setup() // 设置求解器: 分配内存, 设置线性求解器模块</pre>
初始化	<pre>Solver::InitReservoir() // 初始化储层状态及相关辅助变量</pre>
动态模拟	<pre>Begin each time segment: // 根据输入参数, 将模拟时长分段 Control::ApplyControl() // 应用针对当前分段的求解控制策略 Reservoir::ApplyControl() // 应用针对当前分段的井控制策略 Time stepping within segment: // 在当前分段内使用时间步进求解 Solver::GoOneStep() // 求解当前时间步的储层状态 Output::SetValAtTimeStep() // 保存当前时间步的计算结果 Output::PrintAtPeriod() // 在每个时间段结束时将结果打印到临时文件中</pre>
结果输出	<pre>Output::PostProcess() // 主进程对打印结果后处理 OpenCAEPoro::PrintTimeInfo() // 打印与运行时间相关的信息</pre>

OpenCAEPoro软件整体结构

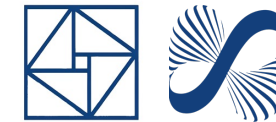


OpenCAEPoro软件模块设计

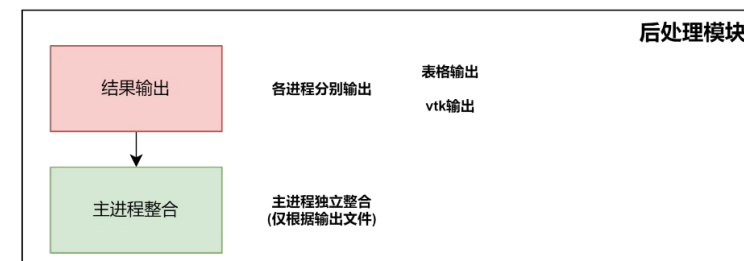
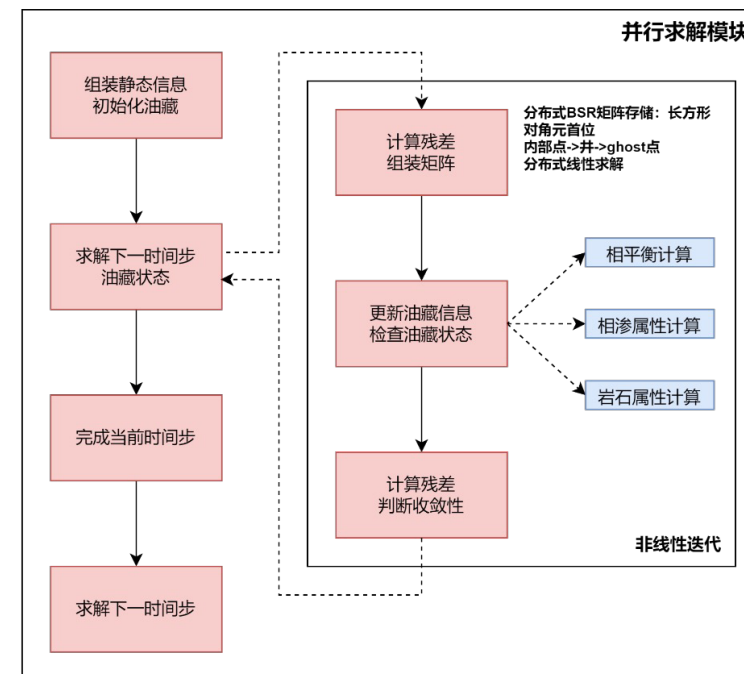
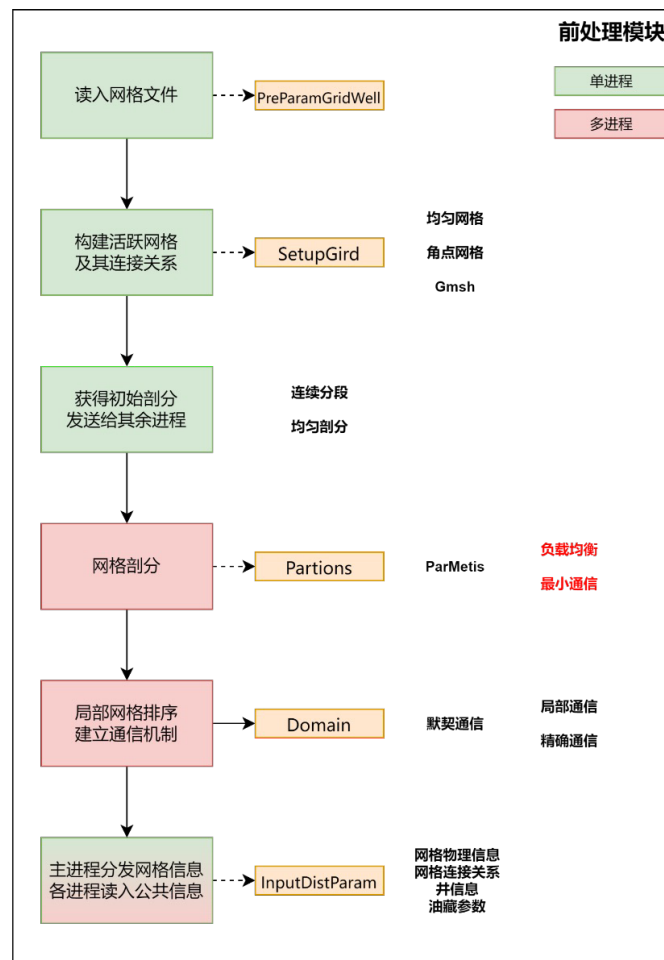


- (1) 模块化设计: 可复用性
- (2) 抽象化设计: 可替换性
- (3) 关联性设计: 可协作性
- (4) 统一化设计: 可理解性

OpenCAEPoro分布式并行设计

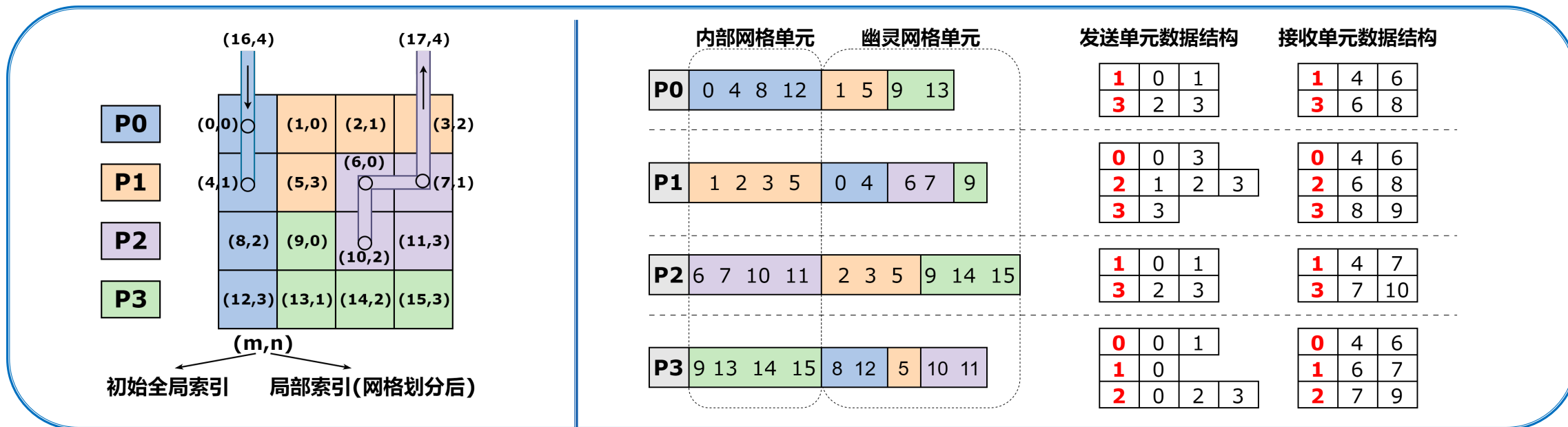
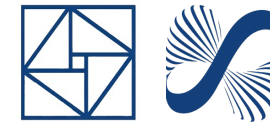


组分模型并行数值模拟



1. 串行地读取输入文件是潜在的瓶颈所在, 这一块也是绝大部分软件所缺失的, 但更大的问题在于串行地对一般网格的前处理——得到网格信息和连接关系, 需要进一步分析和研究。
2. 在模拟过程中, 油藏的状态不断发生变化, 每个网格上的计算量以及网格间的联通强弱也在不断发生变化, 难以在整个模拟过程中始终保持各进程的负载均衡或是达到具有某种优势的网格剖分。
3. 求解器的并行效率是整个模拟过程能否高效并行的关键。

OpenCAEPoro并行划分与通信



- 同一口井的射孔所在网格单元分配至同一进程
- 网格数量在各进程间分布均匀
- 区域边界最小化

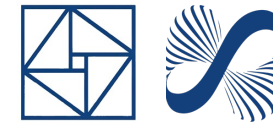
```

map<int, set<int>>    send_element_loc; // 发送单元数据结构
map<int, vector<int>> recv_element_loc; // 接收单元数据结构

MPI_Comm global_comm; // 全局通信器
int      glocal_rank; // 当前进程的全局编号

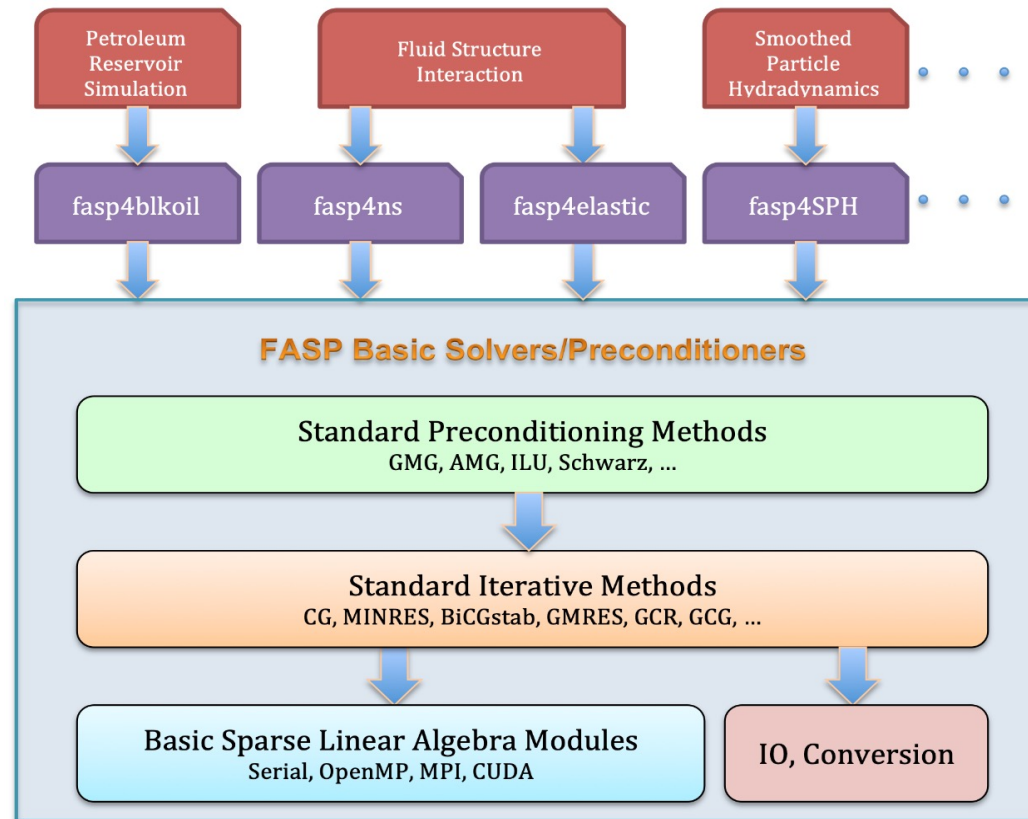
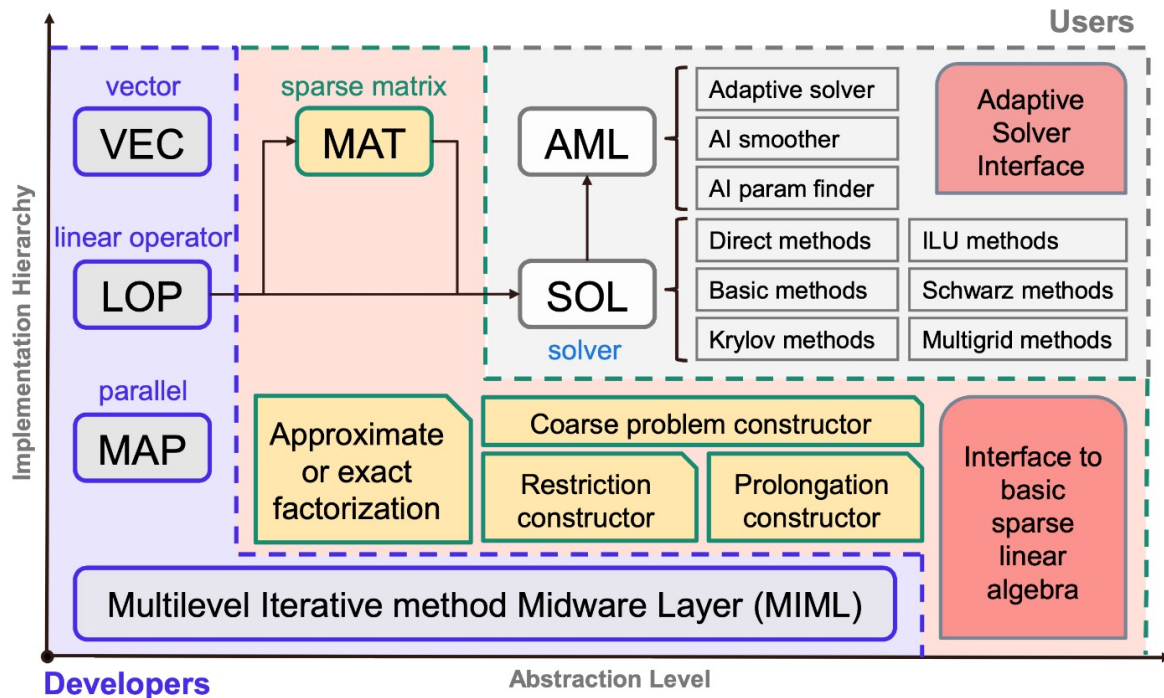
MPI_Comm cs_comm; // 耦合求解专用通信器
int      cs_rank; // 当前进程在 cs_comm 的局部编号
set<int> cs_group_local_rank; // cs_comm 中进程的局部编号
set<int> cs_group_global_rank; // cs_comm 中进程的全局编号
    
```

FASP线性解法器软件介绍



```

FaspRetCode LOP :: Solve(VEC& b, VEC& x)
FaspRetCode LOP :: Solve(VEC& b, VEC& x, SOL& pc)
FaspRetCode MAT :: Solve(VEC& b, VEC& x, SOL& pc, Param& par)
    
```



FASP 开源项目网址

- <http://www.multigrid.org/fasp>
- <https://github.com/FaspDevTeam/faspsolver>
- <https://github.com/FaspDevTeam/fasppx>

国际油藏工程师协会标准算例



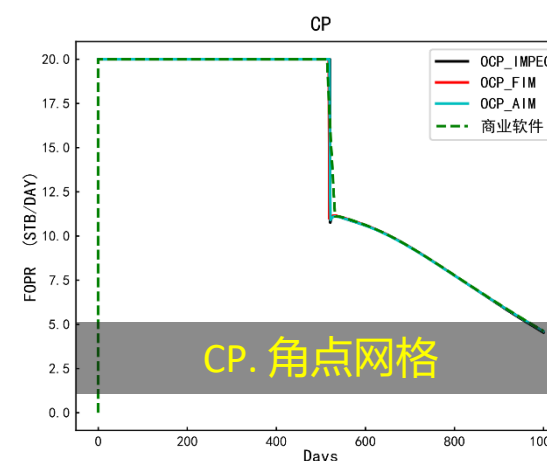
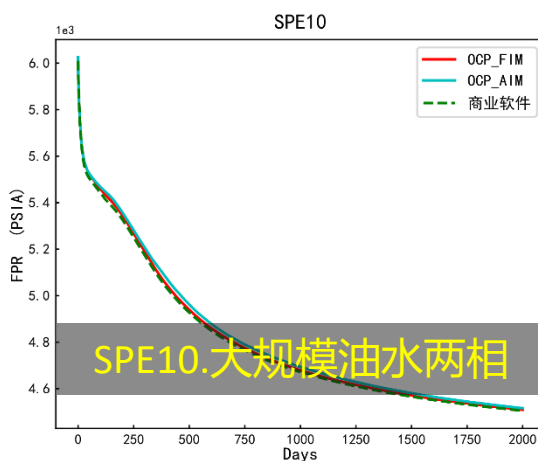
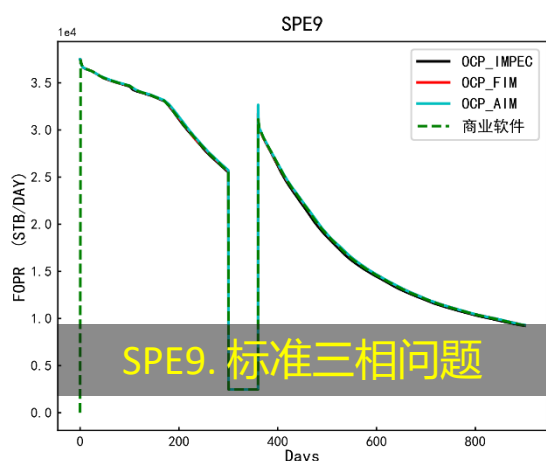
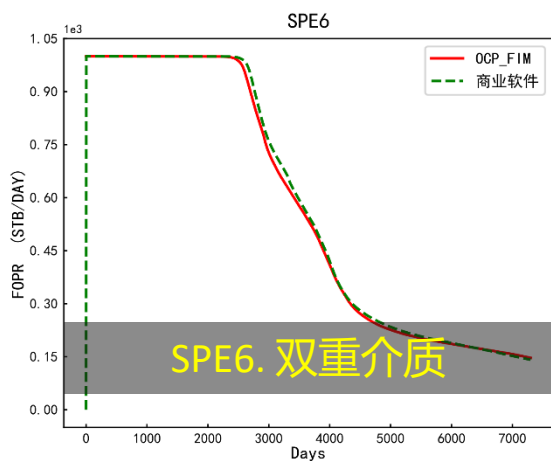
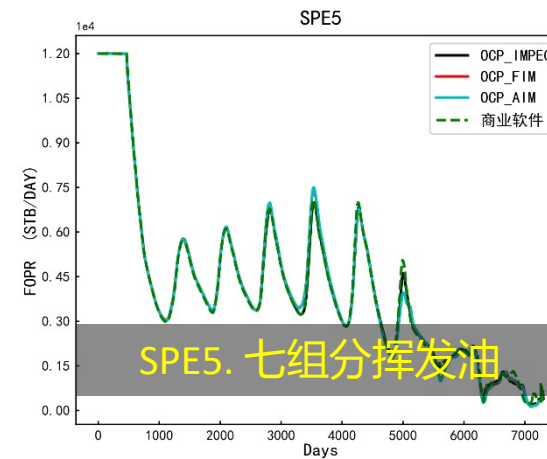
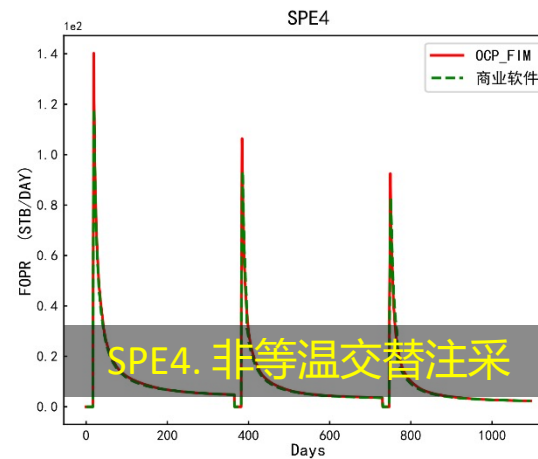
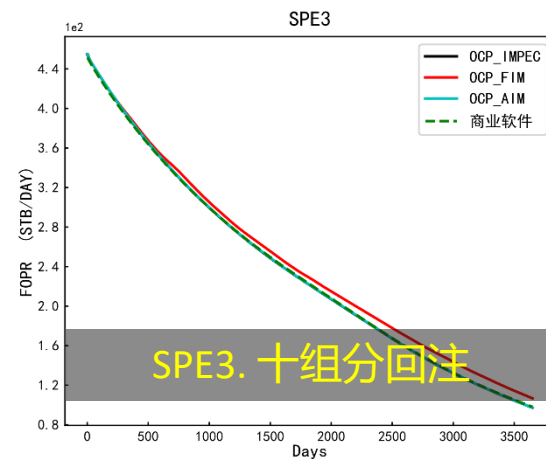
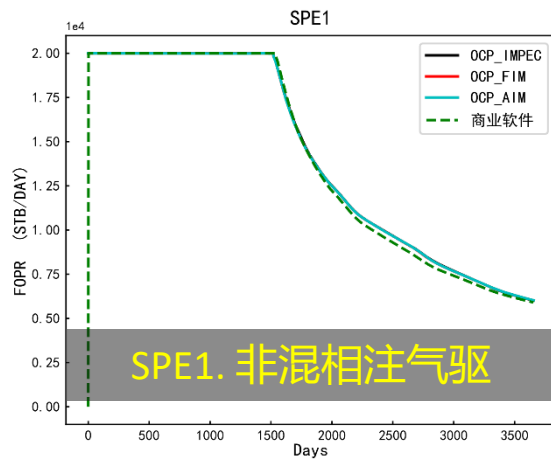
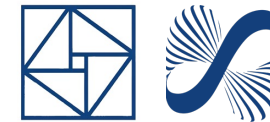
TABLE 17.1—SPE COMPARATIVE SOLUTION PROJECT PROBLEMS

SPE1	Three-phase black oil 10×10×3 300-block grid 3,650-day depletion with gas injection
SPE2	Three-phase black oil 10×1×15 150-block <i>r-z</i> grid 900-day single-well coning depletion
SPE3	Nine-component retrograde gas 9×9×4 324-block grid 5,480-day cycling and blowdown
SPE4	Cyclic steam injection and steam displacement of heavy oils
SPE5	Six-component volatile oil 7×7×3 147-block grid 20-year WAG injection
SPE6	Three-phase black oil Single-block and cross-sectional dual porosity with drainage and gas and water injection cases
SPE7	Three-phase black oil 9×9×6 486-block grid with horizontal wells Eight 1,500-day injection-production cases
SPE8	Two-phase gas-oil black oil 10×10×4 400-block grid Comparison of 2,500-day 400-block grid results with 20-block unstructured and locally refined grid results
SPE9	Three-phase black oil 24×25×15 9,000-block 25-well grid with geostatistical description 900-day depletion
SPE10	Model 1: Two-phase gas-oil case with a 2,000-block 100×1×20 grid and gas injection to 2000 days Model 2: Two-phase water-oil case with a 1.12-million block 60×220×85 grid and water injection to 2,000 days Both models have geostatistical descriptions

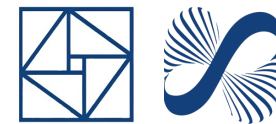
- 1) Odeh, A.S. 1981. Comparison of Solutions to a Three-Dimensional Black-Oil Reservoir Simulation Problem. *J Pet Technol* 33 (1): 13–25. SPE-9723-PA.
- ~~2) Weinstein, H.G., Chappellear, J.E., and Nolen, J.S. 1986. Second Comparative Solution Project: A Three-Phase Coning Study. *J Pet Technol* 38 (3): 345–353. SPE-10489-PA.~~
- 3) Kenyon, D. 1987. Third SPE Comparative Solution Project: Gas Cycling of Retrograde Condensate Reservoirs. *J Pet Technol* 39 (8): 981-997. SPE-12278-PA.
- 4) Aziz, K., Ramesh, A.B., and Woo, P.T. 1987. Fourth SPE Comparative Solution Project: Comparison of Steam Injection Simulators. *J Pet Technol* 39 (12): 1576–1584. SPE-13510-PA.
- 5) Killough, J.E. and Kossack, C.A. 1987. Fifth Comparative Solution Project: Evaluation of Miscible Flood Simulators. Presented at the SPE Symposium on Reservoir Simulation, San Antonio, Texas, SPE-16000-MS.
- 6) Firoozabadi, A. and Thomas, L.K. 1990. Sixth SPE Comparative Solution Project: Dual-Porosity Simulators. *J Pet Technol* 42 (6): 710-715, 762-763. SPE-18741-PA.
- 7) Nghiem, L., Collins, D.A., and Sharma, R. 1991. Seventh SPE Comparative Solution Project: Modelling of Horizontal Wells in Reservoir Simulation. Presented at the SPE Symposium on Reservoir Simulation, Anaheim, California. SPE-21221-MS.
- ~~8) Quandalle, P. 1993. Eighth SPE Comparative Solution Project: Gridding Techniques in Reservoir Simulation. Presented at the SPE Symposium on Reservoir Simulation, New Orleans, Louisiana, 1993. SPE-25263-MS.~~
- 9) Killough, J.E. 1995. Ninth SPE Comparative Solution Project: A Reexamination of Black-Oil Simulation. Presented at the SPE Reservoir Simulation Symposium, San Antonio, Texas, USA. SPE 29110.
- 10) Christie, M.A. and Blunt, M.J. 2001. Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. *SPE Res Eval & Eng* 4 (4): 308–317. SPE-72469-PA.

Source: https://petrowiki.spe.org/Reservoir_simulation

SPE标准算例结果对比



一体化全生命周期数字孪生



两驱一体化数值模拟

水驱+化学驱，微乳液相，复杂相态变化，四相平衡计算带来巨大挑战

多介质多场渗流模拟

多尺度裂缝、溶洞等形成的非均匀介质，热-流-固-化多场耦合效应明显

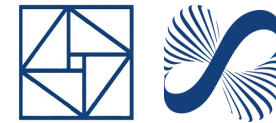
油藏管网一体化模拟

地下油气储层和井组、地面管网的耦合作用，油气资产整体效益最大化

全生命周期数字孪生

地震勘探、实验室模拟、油藏尺度模拟、自动历史拟合等全周期数字孪生

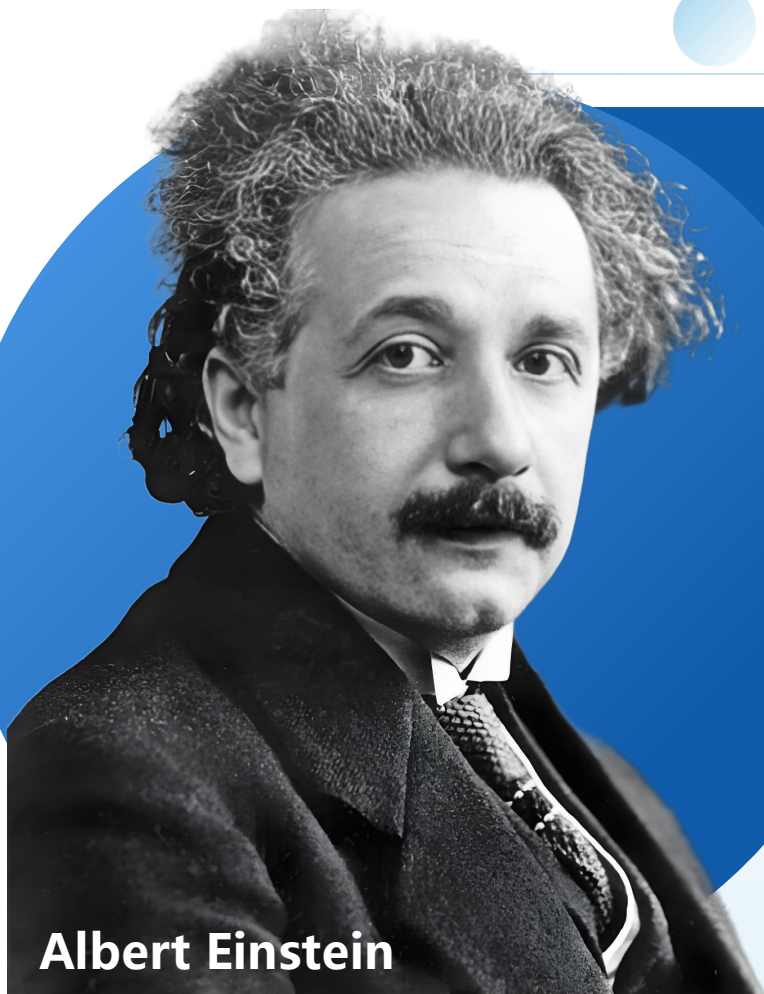
数值仿真的结果“可信”吗



理论信其真，唯创建之人；实验疑其假，唯操作之人。

A theory is something nobody believes, except the person who made it. An experiment is something everybody believes, except the person who made it.

- ✓ 仿真信其真，唯软件作者？…… 很可惜，仿真连软件作者都不信！
- ✓ 仿真的终极问题并不在于你信不信，而在于**误差有多大**。当实验不可行（过于昂贵、过于耗时、过于危险、没有实验条件）时，仿真仍能提供理性的分析手段——进行仿真恰恰是为了探索那些“仅凭相信”还无法触及的领域！
- ✓ 没有**精度**，则仿真毫无意义；没有**速度**，则仿真无法实用；而不**好用**，则仿真没有市场。



Albert Einstein

Thank You All Very Much

感谢各位专家!

 zhangcs@lsec.cc.ac.cn



主要参考文献



1. Linear solvers for petroleum reservoir simulation, C.-S. Zhang, **Journal on Numerical Methods and Computer Applications**, 43, 1–26 (03/2022), in Chinese.
2. Convergence analysis of inexact two-grid methods: A theoretical framework, X. Xu and C.-S. Zhang, **SIAM Journal on Numerical Analysis**, 60, 133–156 (01/2022)
3. Multiscale hybrid-mixed finite element method for flow simulation in fractured porous media, P. Devloo, W. Teng, and C.-S. Zhang, **Computer Modeling in Engineering and Sciences**, 119, 145–163 (04/2019)
4. On the ideal interpolation operator in algebraic multigrid methods, X. Xu and C.-S. Zhang, **SIAM Journal on Numerical Analysis**, 56, 1693–1710 (06/2018).
5. Analytical decoupling techniques for fully implicit reservoir simulation, C. Qiao, S. Wu, J. Xu, and C.-S. Zhang, **Journal of Computational Physics**, 336, 664–681 (05/2017)
6. On Robust and Efficient Parallel Reservoir Simulation on Tianhe-2, W. Guan, C. Qiao, H. Zhang, C.-S. Zhang, et al., **SPE Reservoir Characterization and Simulation** Conference and Exhibition, SPE-175602 (08/2015)
7. A multilevel preconditioner and its shared memory implementation for a new generation reservoir simulator, S. Wu, J. Xu, C. Feng, C.-S. Zhang, et al., **Petroleum Science**, 11, 540–549 (10/2014)
8. Numerical study of geometric multigrid on CPU–GPU heterogenous computers, C. Feng, S. Shu, J. Xu, and C.-S. Zhang, **Advances in Applied Mathematics and Mechanics**, 6, 1–23 (01/2014)
9. Application of auxiliary space preconditioning in field-scale reservoir simulations, X. Hu, J. Xu, and C.-S. Zhang, **Science China Mathematics**, 56, 2737–2751 (12/2013)
10. Combined preconditioning with applications in reservoir simulation, X. Hu, S. Wu, X.-H. Wu, J. Xu, C.-S. Zhang, S. Zhang, and L. Zikatanov, **Multiscale Modeling and Simulation**, 11, 507– 521 (08/2013)