Numerical methods for conservation law equation

有限差分法求解泊松方程

Outline

I. 拉普拉斯算子

II. 问题描述

III. 离散代数方程组以及基本的迭代方法(JacobiG-S, SOR)

IV. 计算结果

V. Fortran codes

Poisson equation

(I) 二维拉普拉斯算子 $L = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$

常见表现形式：

(1) 热传导方程
(2) N-S方程中的粘性项
(3) 泊松方程(压力泊松方程、电势泊松方程)

热传导方程：$\frac{\partial \varphi}{\partial t} = \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2}$

泊松方程：$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = f(x, y)$

拉普拉斯方程：$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = 0$

Time-consuming

(1) 多重网格加速收敛技术
(2) CPU并行(MPI, OpenMP等)
(3) GPU并行(CUDA)

Poisson equation

(II) 问题描述

泊松方程是科学计算中常见的模型方程，本讲介绍几个基本的求解策略，
并进行误差分析

(1) 雅可比迭代
(2) 高斯-赛德尔迭代
(3) 超松驰迭代(SOR)

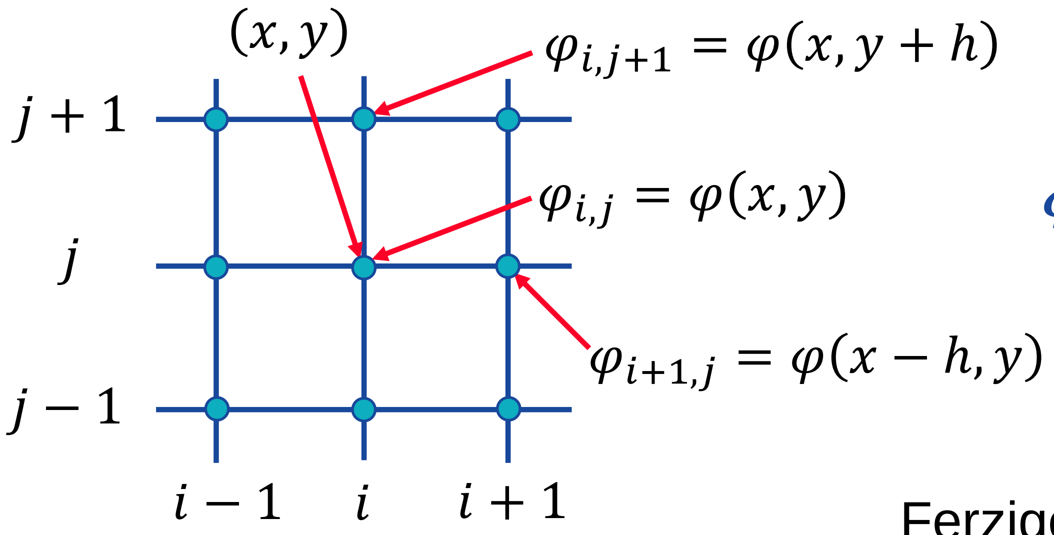
对于泊松方程：$$\frac{\partial^2 \varphi}{\partial x^2}+\frac{\partial^2 \varphi}{\partial y^2}=f(x,y)=-4\cdot sin(x-y)\cdot e^{(x-y)}, x\in[-1,1], y\in[-1,1]$$

其解析解为：$$\varphi(x,y)=cos(x-y)\cdot e^{(x-y)}$$

Poisson equation

(III) 离散代数方程组以及基本的迭代方法

$$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = f(x, y) \qquad \frac{\varphi_{i+1,j} - 2\varphi_{i,j} + \varphi_{i-1,j}}{\Delta x^2} + \frac{\varphi_{i,j+1} - 2\varphi_{i,j} + \varphi_{i,j-1}}{\Delta y^2} = f_{i,j}$$



离散示意图

简单起见，令： $\Delta x = \Delta y = h$

$$\varphi_{i,j} = \frac{1}{4}\left(\varphi_{i+1,j} + \varphi_{i-1,j} + \varphi_{i,j+1} + \varphi_{i,j-1} - h^2 f_{i,j}\right)$$

Ferziger《 Computational Methods for Fluid Dynamics 》

Poisson equation

(III) 离散代数方程组以及基本的迭代方法

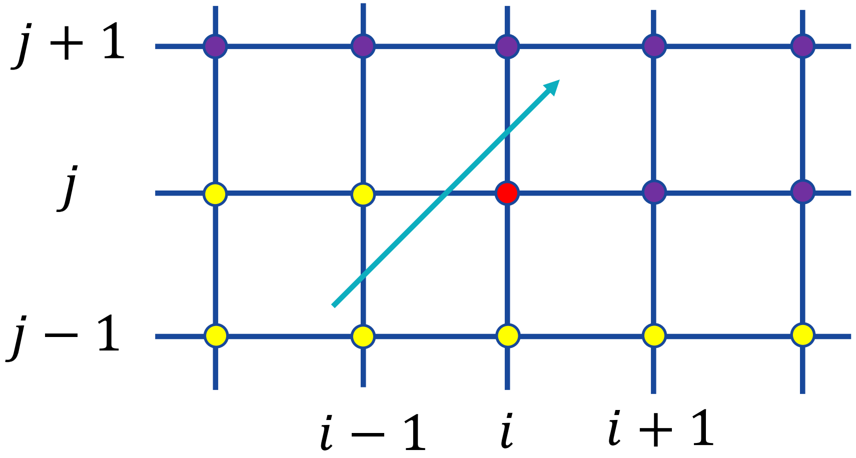
$$\varphi_{i,j}=\frac{1}{4}\left(\varphi_{i+1,j}+\varphi_{i-1,j}+\varphi_{i,j+1}+\varphi_{i,j-1}-h^2 f_{i,j}\right)$$

$\varphi_{i,j}^{n}$ 表示本次迭代的值

$\varphi_{i,j}^{n+1}$ 表示下一次迭代的值

雅可比迭代：

$$\varphi_{i,j}^{n+1}=\frac{1}{4}\left(\varphi_{i+1,j}^{n}+\varphi_{i-1,j}^{n}+\varphi_{i,j+1}^{n}+\varphi_{i,j-1}^{n}-h^2 f_{i,j}\right)$$



高斯-赛德尔迭代：

$$\varphi_{i,j}^{n+1}=\frac{1}{4}\left(\varphi_{i+1,j}^{n}+\varphi_{i-1,j}^{n+1}+\varphi_{i,j+1}^{n}+\varphi_{i,j-1}^{n+1}-h^2 f_{i,j}\right)$$

Poisson equation

(III) 离散代数方程组以及基本的迭代方法

超松驰(SOR)迭代： $\boldsymbol{\varphi_{i,j}^{n+1} = \frac{\beta}{4}\left(\varphi_{i+1,j}^{n} + \varphi_{i-1,j}^{n+1} + \varphi_{i,j+1}^{n} + \varphi_{i,j-1}^{n+1} - h^2 f_{i,j}\right) + (1-\beta)\varphi_{i,j}^{n}}$

收敛判据：
(平均相对误差)

$$Res = \left(\sum_{i=1}^{nCells} \frac{\left|\varphi_i^{n+1} - \varphi_i^{n}\right|}{\left|\varphi_i^{n+1}\right|}\right) / nCells \le 10^{-5}$$

$N_x = N_y = 256$	迭代次数
Jacobi	37715
Gauss-Seidler	23985
SOR(β = 1.50)	10242
SOR(β = 1.75)	4970
SOR(β = 1.90)	2606
SOR(β = 1.95)	1270

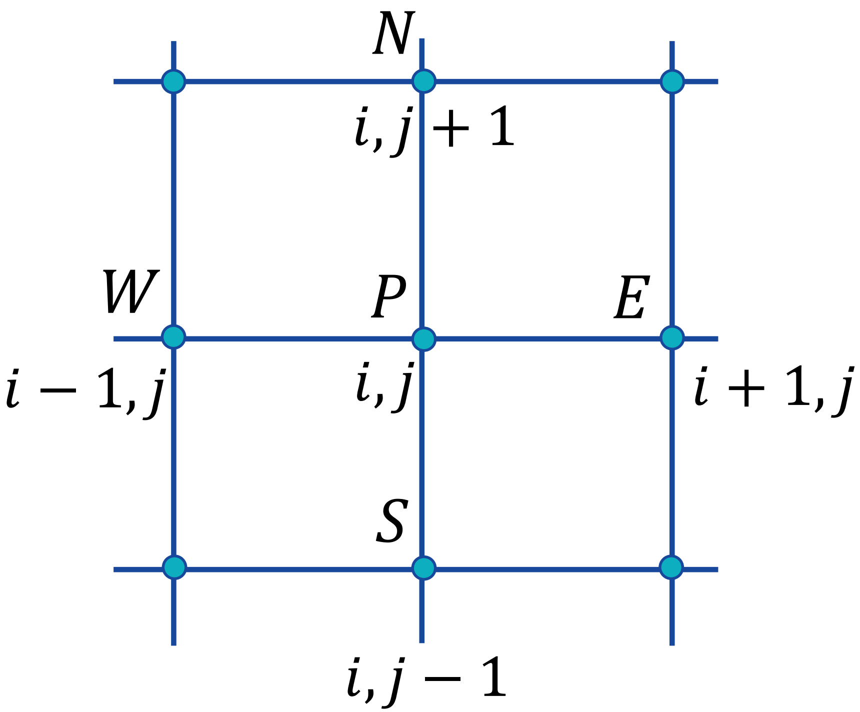
Poisson equation

(4) 计算结果

(5) Fortran codes

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) 网格记号



Grid Notation

Grid location	Compass notation	Storage location
i,j	P	$k=j+(i-1)*N_y$
$i-1,j$	W	$k-N_y$
$i,j-1$	S	$k-1$
$i,j+1$	N	$k+1$
$i+1,j$	E	$k+N_y$

不同表示方式之间的转换关系

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) 二阶导数项的一般离散形式

$$\left(\frac{\partial^2\varphi}{\partial x^2}\right)_{i,j}=\frac{\partial}{\partial x}\left(\frac{\partial\varphi}{\partial x}\right)_{i,j}=\frac{\left(\frac{\partial\varphi}{\partial x}\right)_{i+\frac{1}{2},j}-\left(\frac{\partial\varphi}{\partial x}\right)_{i-\frac{1}{2},j}}{\frac{1}{2}\left(x_{i+1}-x_{i-1}\right)}=\frac{\left(\frac{\varphi_{i+1,j}-\varphi_{i,j}}{x_{i+1}-x_i}\right)-\left(\frac{\varphi_{i,j}-\varphi_{i-1,j}}{x_i-x_{i-1}}\right)}{\frac{1}{2}\left(x_{i+1}-x_{i-1}\right)}$$

$$\left(\frac{\partial^2\varphi}{\partial y^2}\right)_{i,j}=\frac{\partial}{\partial y}\left(\frac{\partial\varphi}{\partial y}\right)_{i,j}=\frac{\left(\frac{\partial\varphi}{\partial y}\right)_{i,j+\frac{1}{2}}-\left(\frac{\partial\varphi}{\partial y}\right)_{i,j-\frac{1}{2}}}{\frac{1}{2}\left(y_{j+1}-y_{j-1}\right)}=\frac{\left(\frac{\varphi_{i,j+1}-\varphi_{i,j}}{y_{j+1}-y_j}\right)-\left(\frac{\varphi_{i,j}-\varphi_{i,j-1}}{y_j-y_{j-1}}\right)}{\frac{1}{2}\left(y_{j+1}-y_{j-1}\right)}$$

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) 问题描述

对于泊松方程：

$$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = f(x,y) = -4 \cdot sin(x-y) \cdot e^{(x-y)}, x \in [0,3], y \in [-1,2]$$

其解析解为：

$$\varphi(x,y) = cos(x-y) \cdot e^{(x-y)}$$

一般离散形式(适用于非均匀网格)：

$$\frac{\left(\frac{\varphi_{i+1,j} - \varphi_{i,j}}{x_{i+1} - x_i}\right) - \left(\frac{\varphi_{i,j} - \varphi_{i-1,j}}{x_i - x_{i-1}}\right)}{\frac{1}{2}(x_{i+1} - x_{i-1})} + \frac{\left(\frac{\varphi_{i,j+1} - \varphi_{i,j}}{y_{j+1} - y_j}\right) - \left(\frac{\varphi_{i,j} - \varphi_{i,j-1}}{y_j - y_{j-1}}\right)}{\frac{1}{2}(y_{j+1} - y_{j-1})} = f(x,y)$$

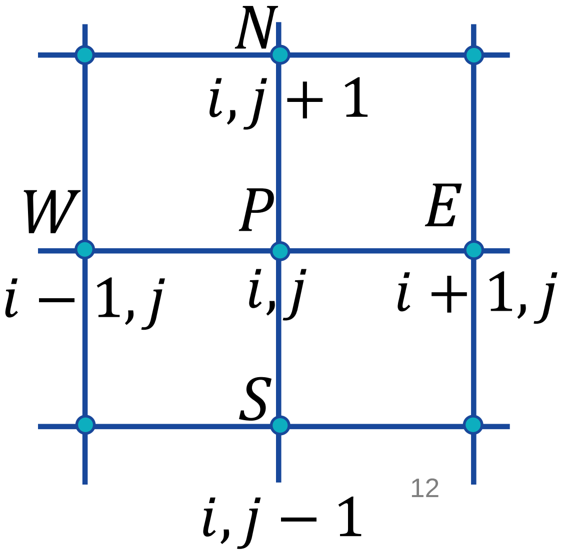
离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) 离散代数方程组

一般离散形式(适用于非均匀网格)：

Grid location	Compass notation	Storage location
i,j	P	$k=j+(i-1)*N_y$
$i-1,j$	W	$k-N_y$
$i,j-1$	S	$k-1$
$i,j+1$	N	$k+1$
$i+1,j$	E	$k+N_y$

$$\frac{\left(\frac{\varphi_{i+1,j}-\varphi_{i,j}}{x_{i+1}-x_i}\right)-\left(\frac{\varphi_{i,j}-\varphi_{i-1,j}}{x_i-x_{i-1}}\right)}{\frac{1}{2}(x_{i+1}-x_{i-1})}+\frac{\left(\frac{\varphi_{i,j+1}-\varphi_{i,j}}{y_{j+1}-y_j}\right)-\left(\frac{\varphi_{i,j}-\varphi_{i,j-1}}{y_j-y_{j-1}}\right)}{\frac{1}{2}(y_{j+1}-y_{j-1})}=f(x,y)$$

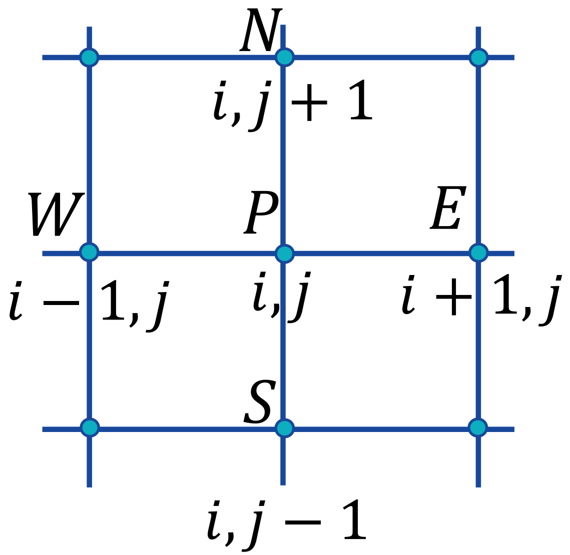


$$AP*\varphi_P+AE*\varphi_E+AW*\varphi_W+AN*\varphi_N+AS*\varphi_S=Q$$

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) Stone’s strong implicit method (SIP)

$$\boldsymbol{A\Phi = Q}$$



二阶中心差分得到了五对角矩阵，没有快速的通用解法

Stone的思路是：是否可以将五对角矩阵进行LU分解？
如果可以，求解过程将大为简化。

$\boldsymbol{LY = Q}$　Forward substitution

$\boldsymbol{U\Phi = Y}$　Backward substitution

$\boldsymbol{A = LU}$是否严格成立？

参考资料：

1. Stone - Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations
2. Ferziger《 Computational Methods for Fluid Dynamics 》
3. Sandip Mazumder《 Numerical methods for Partial Differential Equations》

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) Stone's strong implicit method (SIP)

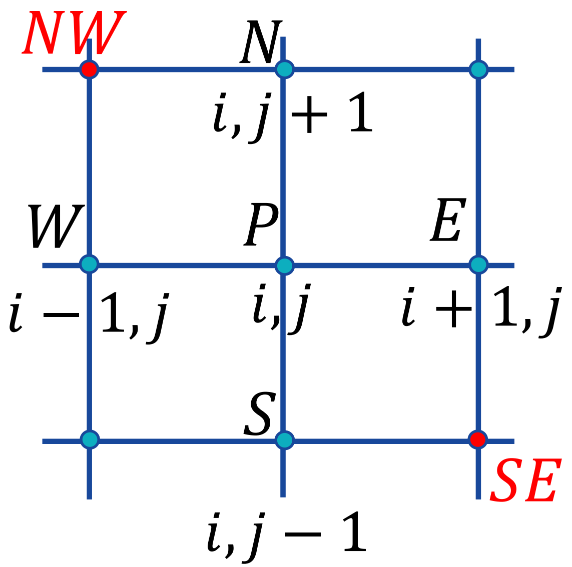
$$\boldsymbol{A\Phi} = \boldsymbol{Q}$$

将矩阵进行分解之后，矩阵的乘积是七对角矩阵，
不是原来的五对角矩阵

$\boldsymbol{A} = \boldsymbol{LU}$不是严格成立

如果分解成功，只需对$\boldsymbol{L}, \boldsymbol{U}$进行回代即可得到方程组的解

现在分解不成功，能否利用这个近似$\boldsymbol{LU}$分解来构造迭代关系？

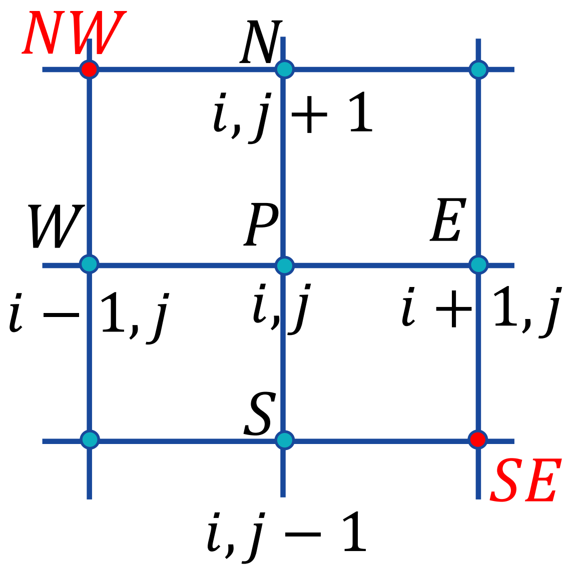


离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) Stone’s strong implicit method (SIP)

$$\boldsymbol{A\Phi} = \boldsymbol{Q}$$

七对角矩阵中多余的两项分别对应NW和SE这两个节点上的值，能否将这两个节点的值近似表示出来？



$$\delta_{NW} \approx \delta_P - \left(\frac{\partial \delta}{\partial x}\right)_P \Delta x + \left(\frac{\partial \delta}{\partial y}\right)_P \Delta y$$

$$\left(\frac{\partial \delta}{\partial x}\right)_P \approx \frac{\delta_P - \delta_W}{\Delta x}$$

$$\left(\frac{\partial \delta}{\partial y}\right)_P \approx \frac{\delta_N - \delta_P}{\Delta y}$$

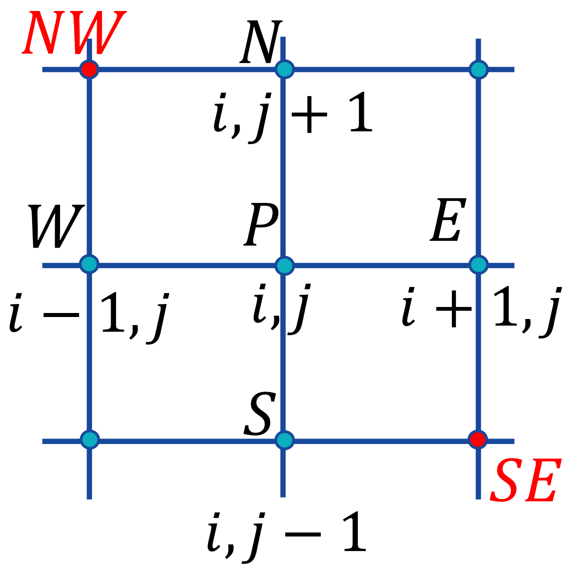
SIP方法中两个重要的近似关系：

$$\delta_{NW} \approx -\delta_P + \delta_N + \delta_W$$

$$\delta_{SE} \approx -\delta_P + \delta_E + \delta_S$$

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) Stone’s strong implicit method (SIP)



SIP算法的执行过程

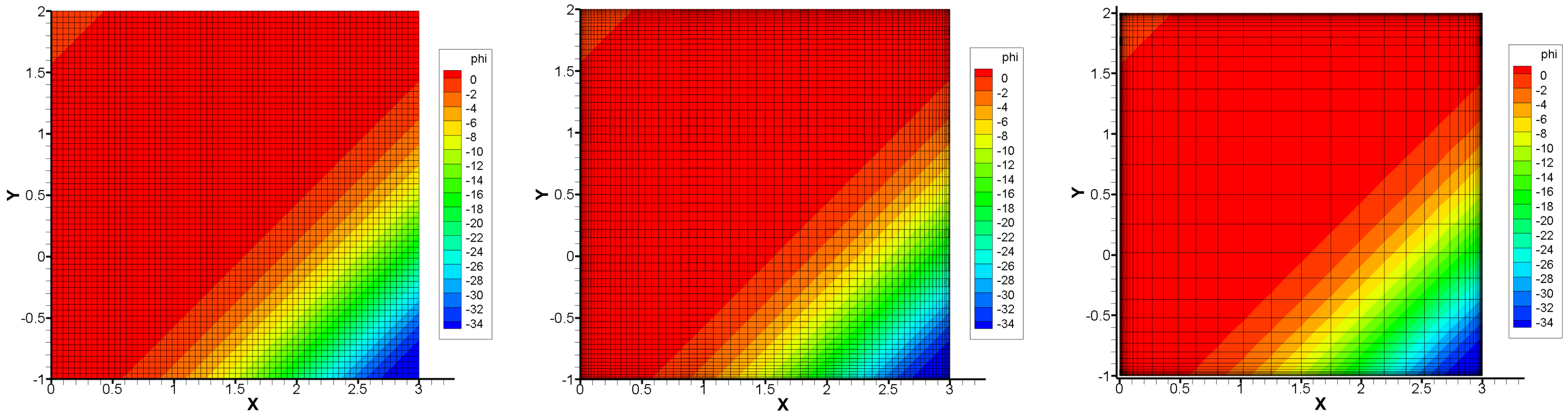
1. 计算$\boldsymbol{L}, \boldsymbol{U}$矩阵中的元素
2. 求解$\boldsymbol{LY} = \boldsymbol{Q}$
3. 求解$\boldsymbol{U\Phi} = \boldsymbol{Y}$
4. 更新代数方程组的解
5. 判断是否收敛

参考资料：

1. Stone - Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations
2. Ferziger《 Computational Methods for Fluid Dynamics 》
3. Sandip Mazumder《 Numerical methods for Partial Differential Equations》

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) SIP的计算结果



均匀网格(64×64)

非均匀网格-1(64×64)

$$z = d(Asin(\zeta\pi/2) + (1 - A)\zeta)$$

非均匀网格-2(64×64)

$$z = d\frac{tanh(A\zeta)}{tanh(A)}$$

Here d is the channel half-width, $-1 < \zeta < 1$ is the uniformly distributed coordinate, z is the wall-normal coordinate and A is the parameter that accounts for the strength of clustering.

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) SIP的计算结果

网格(64×64)	迭代次数	与真实值的误差
均匀网格	350	6.4154×10^{-2}
非均匀网格1	216	9.8634×10^{-2}
非均匀网格2	103	1.1959×10^{-1}

网格(128×128)	迭代次数	与真实值的误差
均匀网格	1259	7.7993×10^{-2}
非均匀网格1	745	6.1139×10^{-2}
非均匀网格2	243	5.5799×10^{-2}

离散代数方程组的高效迭代求解，以求解 Poisson equation 为例

(III) SIP的计算结果

网格(64×64)	迭代次数	与真实值的误差
均匀网格	350	6.4154×10^{-2}
非均匀网格1	216	9.8634×10^{-2}
非均匀网格2	103	1.1959×10^{-1}

网格(128×128)	迭代次数	与真实值的误差
均匀网格	1259	7.7993×10^{-2}
非均匀网格1	745	6.1139×10^{-2}
非均匀网格2	243	5.5799×10^{-2}

Numerical methods for conservation law equation

求解离散代数方程组的共轭梯度法，以 Poisson equation 为例

Outline

I. 问题描述

II. 最速降线法(MSD)介绍

III. 共轭梯度法(CG)介绍

IV. 数值结果比较

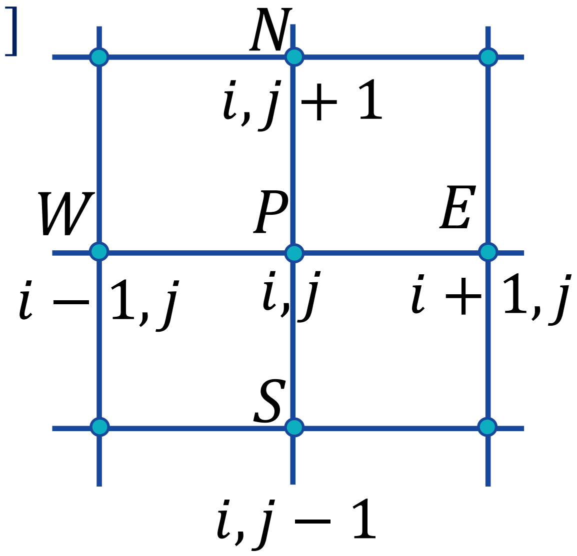
V. Fortran codes

求解离散代数方程组的共轭梯度法，以 Poisson equation 为例

(I) 问题描述

对于泊松方程：

$$\frac{\partial^2\varphi}{\partial x^2}+\frac{\partial^2\varphi}{\partial y^2}=f(x,y)=-4\cdot sin(x-y)\cdot e^{(x-y)}, x\in[0,3], y\in[-1,2]$$



其解析解为：

$$\varphi(x,y)=cos(x-y)\cdot e^{(x-y)}$$

一般离散形式(适用于非均匀结构网格)：

$$\frac{\left(\frac{\varphi_{i+1,j}-\varphi_{i,j}}{x_{i+1}-x_i}\right)-\left(\frac{\varphi_{i,j}-\varphi_{i-1,j}}{x_i-x_{i-1}}\right)}{\frac{1}{2}(x_{i+1}-x_{i-1})}+\frac{\left(\frac{\varphi_{i,j+1}-\varphi_{i,j}}{y_{j+1}-y_j}\right)-\left(\frac{\varphi_{i,j}-\varphi_{i,j-1}}{y_j-y_{j-1}}\right)}{\frac{1}{2}(y_{j+1}-y_{j-1})}=f(x,y)$$

求解离散代数方程组的共轭梯度法，以 Poisson equation 为例

(II) 最速降线法(MSD)

将代数方程组的求解转化为求二次型相关的优化(极值)问题

$$\boldsymbol{Ax} = \boldsymbol{b}$$

矩阵A是正定(负定)对称矩阵

⇓

$$f(x_1, x_2, \cdots, x_n) = \boldsymbol{x}^T\boldsymbol{Ax} - \boldsymbol{x}^T\boldsymbol{b} - \boldsymbol{c}$$

⇓

$\boldsymbol{x}$取何值时，函数$f(x_1, x_2, \cdots, x_n)$取极小(大)值？

⇓

求导 $\nabla_x f(\boldsymbol{x})$，找极值点

Ref: Richard Shewchuk—An Introduction to the Conjugate Gradient Method Without the Agonizing Pain

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(II) 最速降线法(MSD)

$$\boldsymbol{Ax} = \boldsymbol{b}$$

$$f(x_1, x_2, \cdots, x_n) = \boldsymbol{x}^T\boldsymbol{Ax} - \boldsymbol{x}^T\boldsymbol{b} - \boldsymbol{c}$$

$$\nabla_x f(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} = \boldsymbol{Ax} - \boldsymbol{b} = 0$$

沿着负梯度的方向进行搜索，下降最快

$$[\boldsymbol{x}]^{n+1} = [\boldsymbol{x}]^n - \boldsymbol{\alpha}^n[\nabla_x f(\boldsymbol{x})]^n$$

假设对于第n次迭代过程，残差向量满足：

$$[\boldsymbol{R}]^n = \boldsymbol{b} - \boldsymbol{A}[\boldsymbol{x}]^n$$

$$[\boldsymbol{R}]^n = -[\nabla_x f(\boldsymbol{x})]^n$$

残差向量沿着梯度的反方向！

$$[\boldsymbol{x}]^{n+1} = [\boldsymbol{x}]^n - \boldsymbol{\alpha}^n[\nabla_x f(\boldsymbol{x})]^n = [\boldsymbol{x}]^n + \boldsymbol{\alpha}^n[R]^n$$

α^n怎么取，走一步迈多远？

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(II) 最速降线法(MSD)

α^n怎么取，走一步迈多远？

$$\left.\frac{\partial f}{\partial \alpha}\right|^{n+1} = 0$$

关于α的极值点

链式求导，建立新的关系

$$\left[\frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial \alpha} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial \alpha} + \cdots + \frac{\partial f}{\partial x_n}\frac{\partial x_n}{\partial \alpha}\right]^{n+1} = \mathbf{0}$$

$$[\nabla_x f(\boldsymbol{x})]^{n+1} = -[\boldsymbol{R}]^{n+1}$$

$$[\boldsymbol{x}]^{n+1} = [\boldsymbol{x}]^n + \boldsymbol{\alpha}^n[\boldsymbol{R}]^n$$

$$\frac{\partial \boldsymbol{x}_i}{\partial \boldsymbol{\alpha}} = \boldsymbol{R}_i^n$$

$$[\boldsymbol{R}^{n+1}]^T[\boldsymbol{R}^n] = 0$$

相邻两次搜索方向相互垂直

$$\boldsymbol{\alpha}^n = \frac{[\boldsymbol{R}^n]^T[\boldsymbol{R}^n]}{[\boldsymbol{R}^n]^T[\boldsymbol{A}][\boldsymbol{R}^n]}$$

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(II) 最速降线法(MSD)

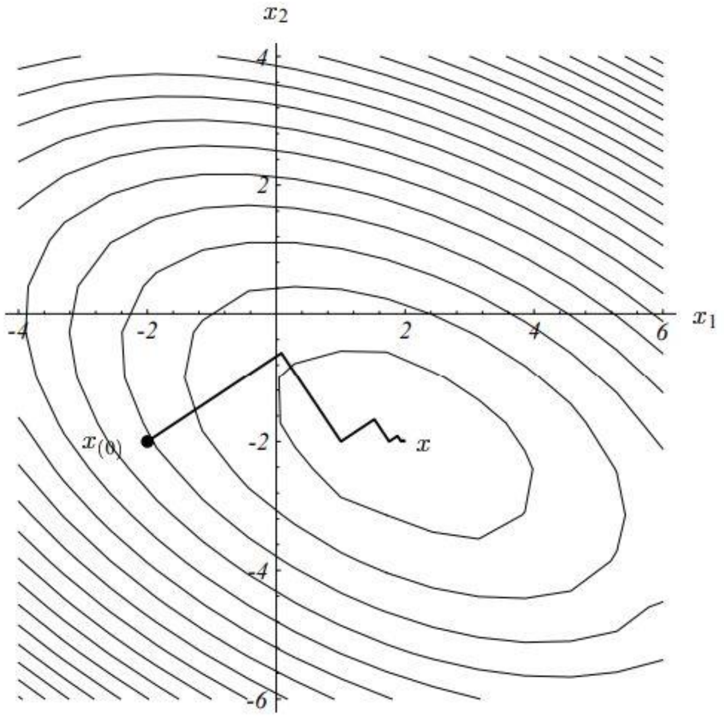
算法流程图：

Step-1: $[R]^n = b - A[x]^n$

Step-2: $\alpha^n = \dfrac{[R^n]^T[R^n]}{[R^n]^T[A][R^n]}$

Step-3: $[x]^{n+1} = [x]^n + \alpha^n[R]^n$

判断是否收敛 $R2^{n+1} < \varepsilon_{tol}$



$[\boldsymbol{R}^{n+1}]^T[\boldsymbol{R}^n]$ =0

相邻两次搜索方向相互垂直极大地限制了搜索范围，
降低了MSD算法的收敛速度

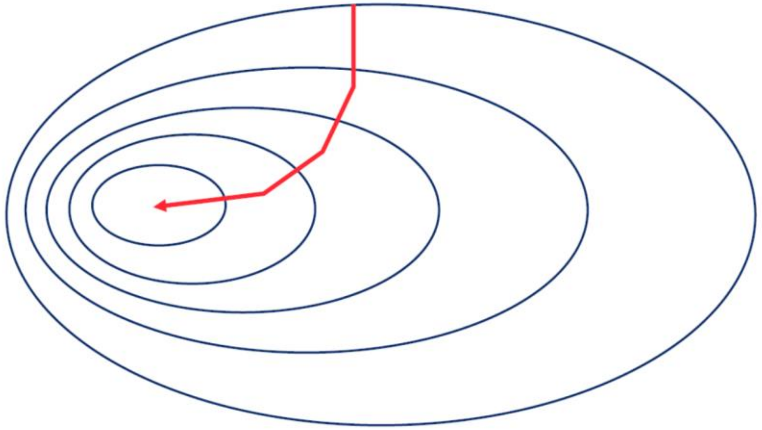
Ref: Richard Shewchuk—An Introduction to the Conjugate Gradient Method Without the Agonizing Pain

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(III) 共轭梯度法(CG)

相邻两次搜索方向相互垂直极大地限制了MSD算法的收敛速度，如何在最速降线法的基础上进行改进？

将相互垂直的两次搜索方向进行线性组合，得到新的搜索方向！



算法流程图：

Step-1: $\alpha^{n+1} = \frac{[R^n]^T[R^n]}{[D^n]^T[A][D^n]}$

Step-2: $[x]^{n+1} = [x]^n + \alpha^{n+1}[D]^n$

Step-3: $[R]^{n+1} = b - A[x]^{n+1}$

Step-4: $\beta^{n+1} = \frac{[R^{n+1}]^T[R^{n+1}]}{[R^n]^T[R^n]}$

Step-5: $[D]^{n+1} = [R]^{n+1} + \beta^{n+1}[D]^n$

判断是否收敛 $R2^{n+1} < \varepsilon_{tol}$

使用共轭梯度法的前提：

系数矩阵A必须是对称矩阵($A^T = A$)

Krylov子空间方法：将前n次相互垂直的搜索方向进行线性组合，进行搜索！

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(III) 其他改进的共轭梯度方法

CG方法无法处理非对称矩阵，限制了CG的使用范围，
如何在CG的基础上进行改进？

↓

Conjugate Gradient Square method (CGS)
Bi-Conjugate Gradient method (BCG)
……

↓ 预处理

CGS-Stab
BiCG-Stab

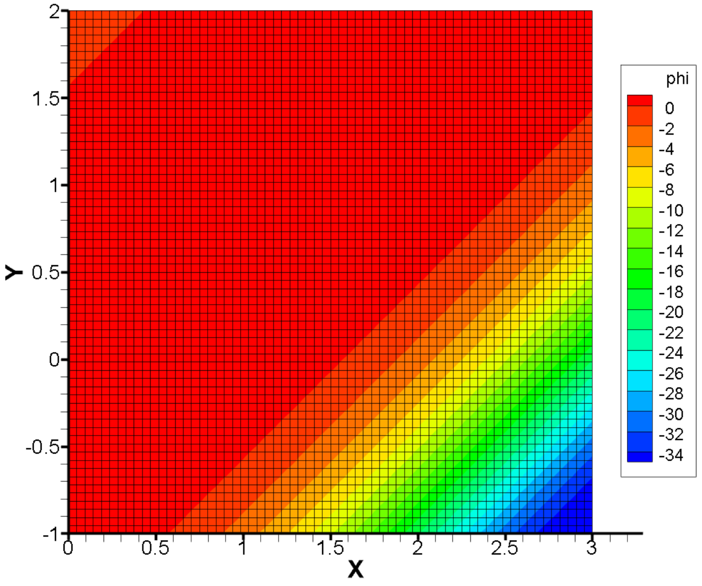
Ref: 1. Richard Shewchuk—An Introduction to the Conjugate Gradient Method Without the Agonizing Pain

2. Ferziger《 Computational Methods for Fluid Dynamics 》

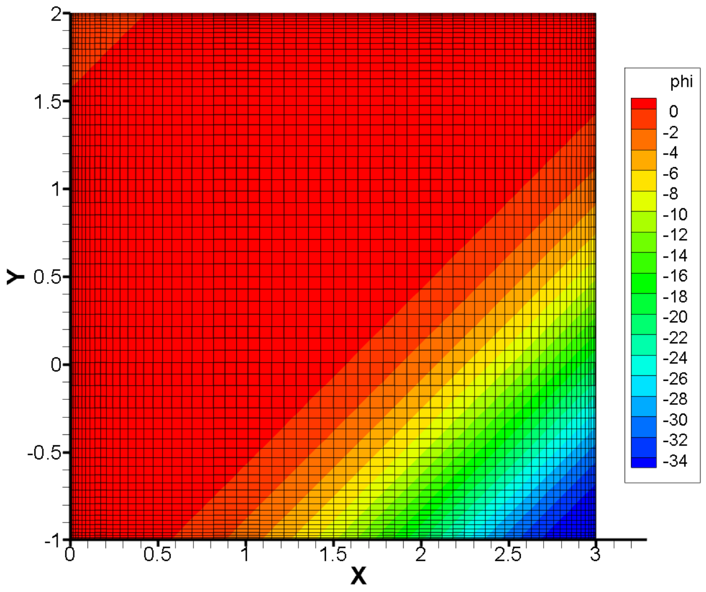
3. Sandip Mazumder《 Numerical methods for Partial Differential Equations》

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(IV) 计算结果

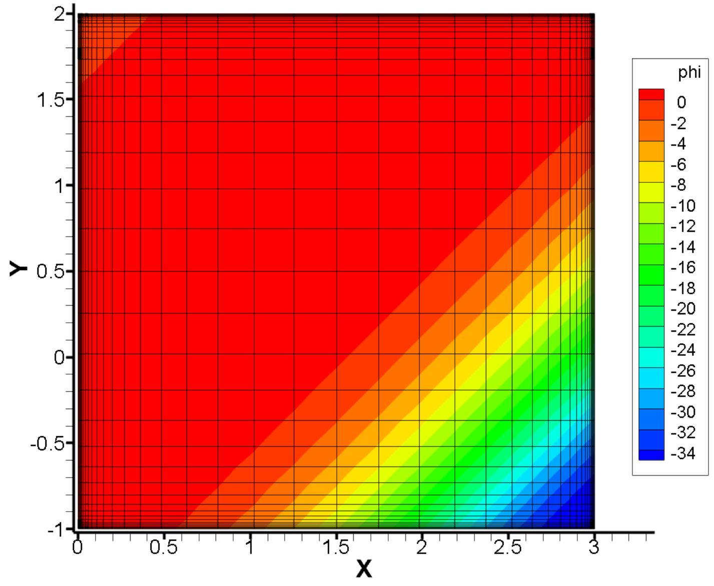


均匀网格(64×64)



非均匀网格-1(64×64)

$$z = d(A\sin(\zeta\pi/2) + (1 - A)\zeta)$$



非均匀网格-2(64×64)

$$z = d\frac{\tanh(A\zeta)}{\tanh(A)}$$

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(IV) 均匀网格的计算结果

迭代方法	迭代次数			CPU计算时间(秒)		
	32×32	64×64	128×128	32×32	64×64	128×128
Jacobi	1489	5984	23976	4.17	22.00	91.38
G-S	746	2993	11989	2.74	13.4	65.35
SOR	143	580	2190	0.49	2.56	14.58
SIP	123	452	1675	0.50	2.34	15.22
MSD	2265	8523	31899	7.09	30.17	82.72
CG	96	185	367	0.40	0.86	1.63
CGS	73	150	306	0.34	0.70	1.79

SIP：线性收敛 $\dfrac{452}{123}\approx 3.6$ $\dfrac{1675}{452}\approx 3.7$

CG、CGS：超线性收敛 $\dfrac{185}{96}\approx 1.9$ $\dfrac{367}{185}\approx 2.0$

求解离散代数方程组的共轭梯度法，以求解 Poisson equation 为例

(IV) 均匀网格的计算结果

1.经典迭代法的收敛速度很慢，效率很低

2.MSD方法的收敛速度比Jacobi更慢

3.SIP、CG、CGS等方法效率很高，收敛快，CPU耗时少

4.非均匀网格下：
可以提高经典迭代法的收敛速度，任意非均匀网格迭代都保持稳定；

对于CG方法，由于失去正定对称条件，迭代发散；

对于CGS方法，由于矩阵性质变差，收敛速度下降，无法对任意非均匀网格保持稳定！

需要进行预处理！