From IB to IIM, from Solution to Gradient Computations

0011 0010 1010 1101 0001 0100 1011

Zhilin Li

CRSC & Department of Mathematics

North Carolina State University

Supported by NSF-560168

Happy Birthday to Charlie!!!

Outline

- From IB (Peskin) to IIM (LeVeque/Li)
- Motivations of this talk: Accurate gradient computation ***at*** the interface/boundary for ***Cartesian*** grid methods
- ***A new augmented IIM***
  - FD Poisson equations, regular problem →piecewise constant coef. →Variable coef.
  - Optimal complexity ***O(N log(N))***, 2nd accurate solution & ***gradient*** and proof (Claim to be the '***best***')
- Numerical results
- Convergence analysis
- Conclusions

From IB to IIM

- Peskin's IB method
  - Mathematical modeling
  - Numerical method: discrete delta function
  - Simple, robust, many applications
  - First order, elliptic (Li, elliptic with Dirichlet BC), Stokes with periodic BC (Mori)
- IIM (LeVeque/Li)
  - Second order or higher
  - Use jump conditions (from PDE or physics) instead of `delta functions
  - Best discrete delta function?
  - Finite difference (IIM, AIIM) and element (IFEM)
- ***How to compute the solution & gradient accurately?***

Motivations for Accurate Gradient

- Many free boundary/moving interface problems depend on the first order ***derivatives*** of the solution
- For finite difference (***FD***) methods based on ***Cartesian meshes***, there are a number of 2^{nd} or higher order methods, but the derivatives are less accurate especially near the boundary/interface
- FEM: L^2: $O(h^2)$, H^1: $O(h)$, ***at interface?***

Some Examples

- The 1D Stefan problem modeling the ice-water interface, let ***s(t)*** be the free boundary, ***u(x,t)*** be the temperature

$$\frac{\partial u}{\partial t} = \beta^2 \frac{\partial^2 u}{\partial x^2}, \quad 0 < x < s(t)$$

$$-\frac{\partial u}{\partial t}(0,t) = f(t), \ \text{inlet heat flux at left end}$$

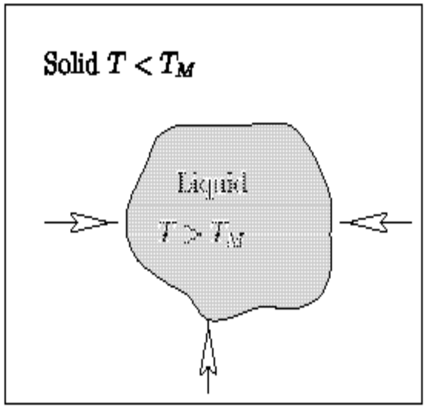
$$u(s(t),t) = 0, \quad \text{the right end is the freezing temperature}$$

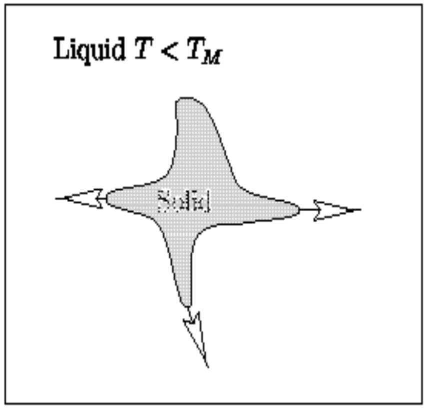
$$\frac{ds}{dt} = -\frac{\partial u}{\partial x}(s(t),t), \ \text{the Stefan condition}$$

$$u(x,0) = 0, \quad s(t) = 0, \ \text{Initial conditions}$$

Stefan problem in 2D & Crystal Growth

- Left: 2D Stefan problem. Right: Formulations of Snowflakes. Heat equation with ***non-linear BC***



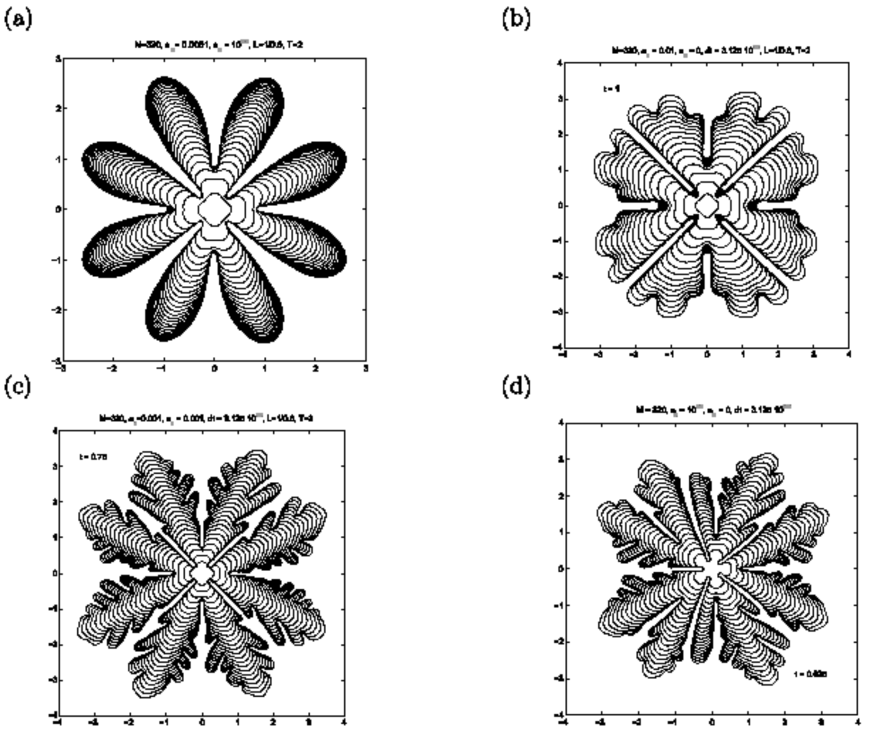


$$\rho c \frac{\partial T}{\partial t} = \nabla \bullet (\beta \nabla T), \quad \rho L V = -\left[\beta \frac{\partial T}{\partial n} \right]$$

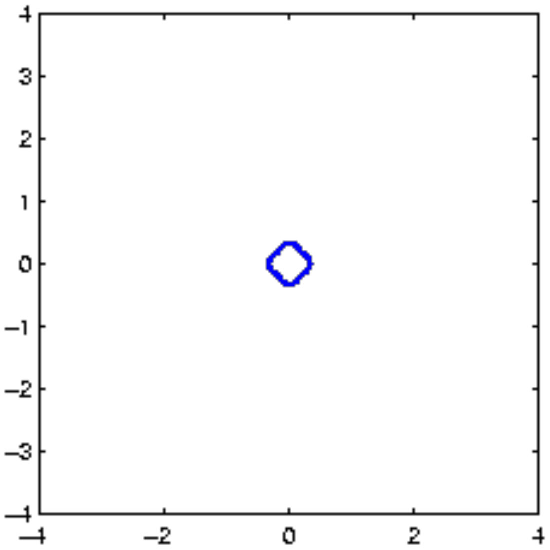
$$T(x,t) = -\varepsilon_c \kappa - \varepsilon_V V, \quad \frac{dX}{dt} \bullet n = V$$

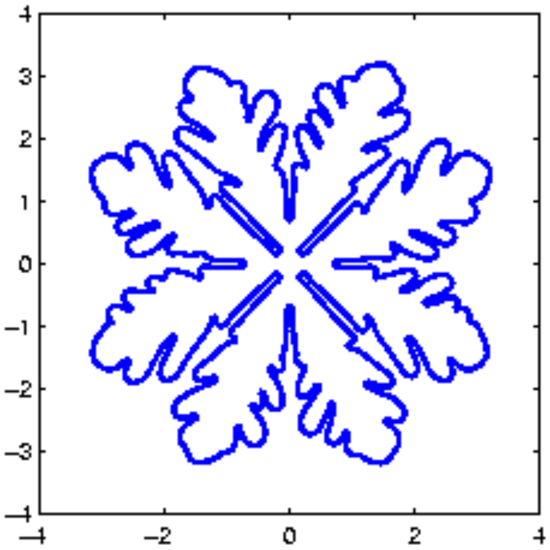
Stefan Problem and Crystal Growth

- ***1st derivatives are involved***
- ***Stability analysis:*** dynamically unstable for some medium modes (*exp(-k l t)*)



Simulation: Crystal Growth





A moving interface example

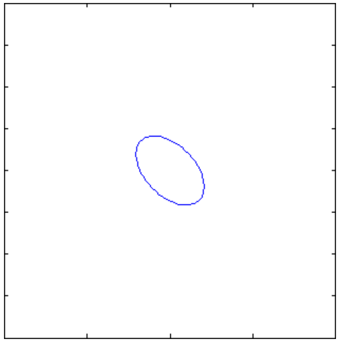
- ***NSE*** equations with ***unknown surface tension***, an inverse problem
- Both the area/length should be preserved.

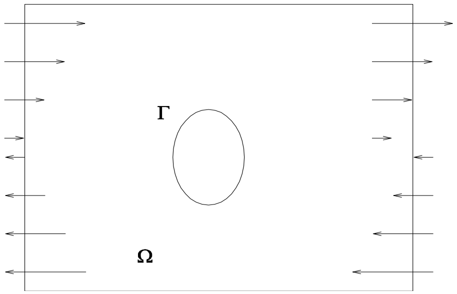
$$\rho\left(\frac{\partial u}{\partial t}+u\bullet\nabla u\right)+\nabla p=\mu\Delta u+\int_{\Gamma} f(s,t)\delta(x-X(s,t))\,ds+g$$

$$f(s,t)=\frac{\partial}{\partial s}\big(\sigma(t,s)\tau\big)+f_b$$

$$=\sigma(s,t)\kappa n+\frac{\partial\sigma(s,t)}{\partial s}\tau+f_b$$

$$\nabla\bullet u=0,\quad \left(\partial_s\bullet u\right)_{\Gamma}=\frac{\partial u}{\partial\tau}\bullet\tau=0$$



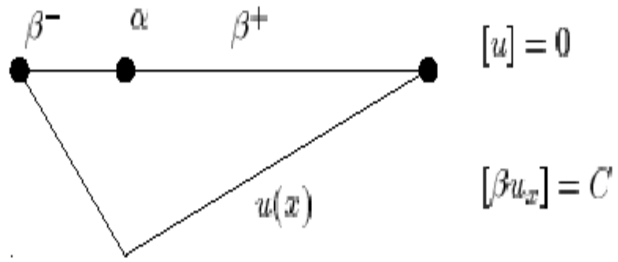


Zhilin LI

Model Problems

- 1D:

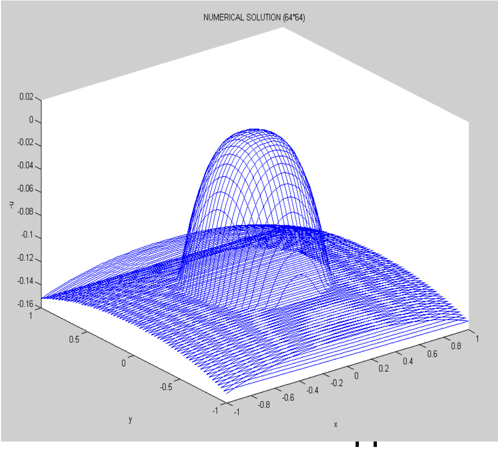
$$(\beta u')' - \sigma u = C\delta(x-\alpha) \qquad 0 < x < 1$$
$$u(0) = 0, \qquad u(1) = 0$$



- 2D

$$\nabla \cdot \left(\beta \nabla u\right) - \sigma u = f + \int_{\Gamma} C(s)\delta(x - X(s))\,ds$$

or $\nabla \cdot \left(\beta \nabla u\right) - \sigma u = f, \;\; [u] = w, \; [\beta u_n] = C(s)$



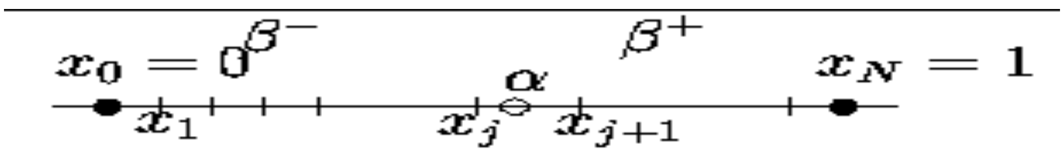
Methods for *Gradients* Review

- FD with Cartesian mesh and central FD scheme: For ***regular*** **problem &** ***regular*** **domain**, the derivatives have the ***same*** order as the solution.
- The difficulty is for *general* boundaries and interfaces.
  - In FEM, ***posterior error analysis*** to get more accurate derivatives, depends on mesh quality
  - In FEM, ***mixed*** FEM or least squares FEM. It will lead to saddle problem and computationally expensive
  - DG for conservation laws
  - FD for elliptic and parabolic problems: ???

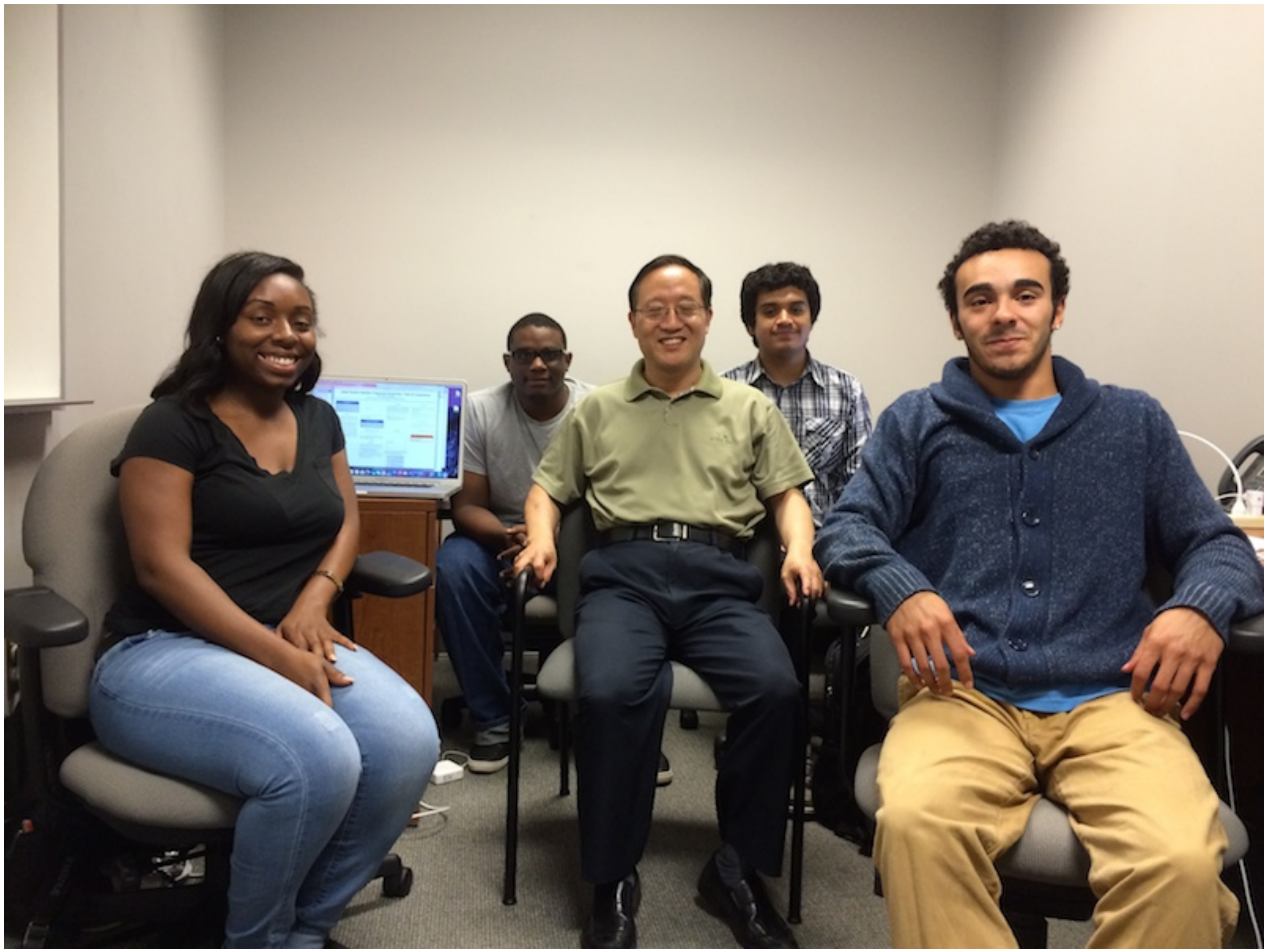
Results (old & *new*) in 1D

Accuracy of u_x at the boundary/interface

- At the boundary, 3-point *one-sided*, provide 2nd u_x
- At the interface (singular source or discontinuous coef
  - 3-point one-sided FD scheme is 1st order
  - ***IIM (compact FD, two-sided) is 2nd order in Cartesian, polar, and spherical***. NCSU-2015 REU project.
  - 1D



My NCSU 2015 REU Group



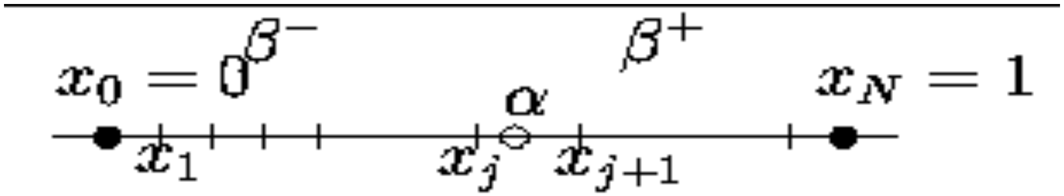
IIM in 1D, simple case

- FD scheme for $(\beta u')' - \sigma u = C\delta(x-\alpha)$

$$\gamma_{j-1}U_{j-1} + \gamma_j U_j + \gamma_{j+1}U_{j+1} = f_j + C_j$$

- Determine the coefficients and the correction term $\gamma_{j-1}, \gamma_j, \gamma_{j+1}, C_j$
- Interface relations:

$$u^+ = u^-, \quad u_x^+ = \frac{\beta^-}{\beta^+}u_x^- + C, \quad u_{xx}^+ = \frac{\beta^-}{\beta^+}u_{xx}^-$$



IIM in 1D: Set-up equation

- The linear system for the coefficients

$$\gamma_{j-1} + \gamma_{j-1} + \gamma_{j+1} = 0$$

$$\gamma_{j-1}(x_{j-1} - \alpha) + \gamma_j (x_j - \alpha) + \gamma_{j+1} \frac{\beta^-}{\beta^+}(x_{j+1} - \alpha) = 0$$

$$\gamma_{j-1} \frac{(x_{j-1} - \alpha)^2}{2} + \gamma_j \frac{(x_j - \alpha)^2}{2} + \gamma_{j+1} \frac{\beta^-}{\beta^+} \frac{(x_{j+1} - \alpha)^2}{2} = \beta^-$$

- The correction term is

$$C_j = C\gamma_{j+1} \frac{\beta^-}{\beta^+}(x_{j+1} - \alpha)$$

Interpolation scheme for u_x

- Three points from both sides plus correction term

$$u_x(\alpha-) = \tilde{\gamma}_{j-1} U_{j-1} + \tilde{\gamma}_j U_j + \tilde{\gamma}_{j+1} U_{j+1} + \tilde{C}_j$$

$$\tilde{\gamma}_{j-1} + \tilde{\gamma}_{j-1} + \tilde{\gamma}_{j+1} = 0$$

$$\tilde{\gamma}_{j-1}(x_{j-1} - \alpha) + \tilde{\gamma}_j (x_j - \alpha) + \tilde{\gamma}_{j+1} \frac{\beta^-}{\beta^+}(x_{j+1} - \alpha) = 1$$

$$\tilde{\gamma}_{j-1} \frac{(x_{j-1} - \alpha)^2}{2} + \tilde{\gamma}_j \frac{(x_j - \alpha)^2}{2} + \tilde{\gamma}_{j+1} \frac{\beta^-}{\beta^+} \frac{(x_{j+1} - \alpha)^2}{2} = 0$$

2D Results for Gradients (old & *new*)

Accuracy of u_x & u_y at the interface

- Singular source only (i.e. $\beta=1$, $[u]\neq 0$, $[u_n]\neq 0$)
  - One-sided FD scheme is ***1st order.***
  - IIM (compact FD, two-sided) is 2nd order (Beale & Layton). One of the basis of the new method.
- Direct: Maximum principle preserving (Li/Ito): soln 2nd, gradient, not sure yet
- Piecewise constant β, FIIM (Li, SINUM, 1997), 2nd solution (proved), 2nd gradient (observed before, now ***proved***)
- ***Variable β, [β] ≠0, 2nd solution and gradient ($h^2\log h$) with proof, 2015.***

2D Problem & Analysis

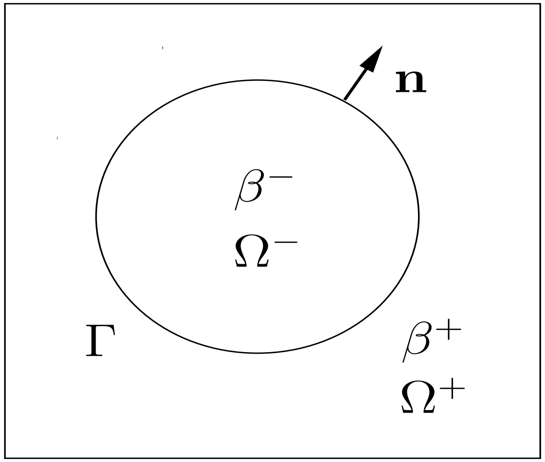
- Elliptic interface problems with ***variable*** & ***discontinuous*** coefficient

$$\nabla \cdot \left(\beta \nabla u\right) + \sigma u = f + \int_\Gamma C(s)\delta(x - X(s))\,ds$$

or $\nabla \cdot \left(\beta \nabla u\right) + \sigma u = f, \quad [u] = w, \; [\beta u_n] = C(s)$

$$\beta(x,y) = \begin{cases} \beta^-(x,y) & \text{if} \quad (x,y) \in \Omega^- \\ \beta^+(x,y) & \text{if} \quad (x,y) \in \Omega^+ \end{cases}$$

$$[\beta(x,y)]_\Gamma \neq 0$$



Why elliptic interface problems?

- It is most expensive part for many simulations processes, e.g. projection method

$$\rho(u_t + u \cdot \nabla u) + \nabla p = \mu \Delta u + g$$

$$\nabla \bullet u = 0$$

$$\frac{u^* - u^k}{\Delta t} + (u \bullet \nabla u)^{k+1/2} + (\nabla p)^{k-1/2} = \frac{\mu}{2}(\Delta u^k + \Delta u^*) + F^{k+1/2}$$

$$\Delta \phi = \frac{\nabla \bullet u^*}{\Delta t}, \quad \frac{\partial \phi}{\partial \mathrm{n}} = 0$$

$$u^{k+1} = u^* - \Delta t \nabla \phi$$

$$\nabla p^{k+1/2} = \nabla p^{k-1/2} + \nabla \phi$$

Cartesian Grid Methods

- Peskin's IB method, 1st order, inconsistent (Li, MathCom, 2014)
- Fast IIM (Li, SINUM), for piecewise constant ***β***
- ***Maximum principle preserving IIM*** (Li/Ito)
- Ghost fluid method (Fedkiw/Liu) 1st -2nd order?
- Boundary integral method (X-F. Li, M. Siegel, Mayo, Greengard, …)
- MIB (Wei/Zhao)
- Virtual node method (Teran)
- IFEM (Li/Lin^2, He,…), Petrov-Galerkin (Hou, Ji/Chen/Li …), IFEV …
- XFEM (*X-D. Wang*, W-K. Liu, J. Doby, …)
- ***Augmented IIM*** (Li *et al*), Kernel free method (W. Ying et. al)

Which one gives 2nd derivatives? FAST IIM

Key Ideas

$$\nabla \bullet (\beta \nabla u) + \sigma u = f, \quad [u] = w, \ [\beta u_n] = v$$

$$\to \Delta u + \frac{1}{\beta} \nabla u \bullet \nabla \beta + \frac{\sigma}{\beta} u = \frac{f}{\beta}$$

$$[u] = w, \ [u_n] = q, \ [\beta u_n] = v$$

- **Reformulate the problem** near the interface by introduce augmented variable **[u_n]**
- **Derive** different ***new*** interface relations using the new formulation
- Apply the ***upwind*** scheme near **Γ** for the advection term(s) to get an M-matrix
- Apply the GMRES for the Schur complement (**[u_n]** *)*

Zhilin LI

Poisson Eqn. with singular sources

$$\Delta u = f(x) + \int_{\Gamma} c(s)\delta(x - X(s))ds + g$$

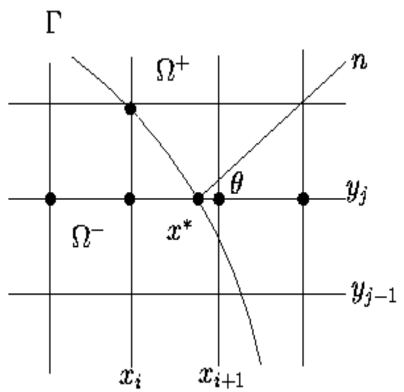
BC (e.g., Dirichlet, Neuman, Mixed)

❑ *Equivalent Problem*

$$\Delta u = f(x), \quad x \in \Omega \setminus \Gamma, \quad [u]_{\Gamma} = 0, \quad \left[\frac{\partial u}{\partial n}\right]_{\Gamma} = C(s)$$

BC (e.g., Dirichlet, Neuman, Mixed)

❑ FD scheme (x_i, y_j), regular/irregular



$$\frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}}{h^2} = L_h u_{i,j} = f_{ij}$$

$$\frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}}{h^2} = L_h u_{i,j} = f_{ij} + C_{ij}$$

Poisson Eqn. with singular sources

- ***IB***, ***IIM*** both work, what's the ***best*** discrete delta function? → Source removal technique (Li/Lai/Wang)
- ***AU=F+BC;*** ***A***: Discrete Laplacian. Can be solved by a fast Poisson solver
- IIM is second order both in solution and ***gradient*** (T. Beale/Layton), now to NS equations with fixed/exact interface

Augmented approach/Fast IIM

- If β is two constants, flux jump condition $[\beta u_n]=C(s)$ along $[u]=w(s)$.
- Idea:

$$\nabla \bullet (\beta \nabla u) = f + \int_\Gamma C(s)\delta(x - X(s))\,ds$$

$$\text{or } \nabla \bullet (\beta \nabla u) = f, [u] = 0, [\beta u_n] = C(s)$$

$$\text{or } \Delta u = \frac{f}{\beta} + \int_\Gamma \frac{C(s)}{?}\delta(x - X(s))\,ds$$

- Idea of the method: divide β from the equation to get Poisson eqn., but can not from the flux jump.
- Set $[u_n] = g$ as unknown, the augmented variable, the augmented equation is the flux jump condition

Fast IIM

- Idea, given $[u_n] = g$ solve the problem with one FFT

$$AU + BG = F + C = F_1$$

- Discretize the flux condition $[\beta u_n] = v$

$$SU+EG=F_2$$

- Schur complement:

$$\begin{bmatrix} A & B \\ S & E \end{bmatrix} \begin{bmatrix} U \\ G \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

$$(E - SA^{-1}B)G = F_2 - SA^{-1}F_1 = \bar{F}$$

$$R(G) - R(0) = (E - SA^{-1}B)G = [\beta u_n(G)] - C - ([\beta u_n(0)] - C)$$

Properties of FIIM

- Second solution, proved
- ***O(N log(N))*** optimal computation cost. The Number of GMRES iterations
  - Independent of jump in the coefficient
  - Independent of the mesh size
  - Dependent on the geometry
- ***Second*** order accurate 1st order derivatives, observed before, now we have proof.

Challenges with *Variable* Coef

- *Maximum* preserving FD scheme (*direct*) for

$$\nabla \cdot (\beta \nabla u) + \sigma u = f$$

$$[u] = w, \quad [\beta u_n] = v$$

- 5-point at regular, 9-point stencil at irregular grids
- Using a *quadratic* optimization to force the maximum principle (Li/Ito)
- Using a structured multigrid method to solve the linear system whose condition is proportional to $O(\max(\beta^+/\beta^-, \beta^-/\beta^+) /h^{2)})$.
- *The derivative is often 1st order accurate near the interface*

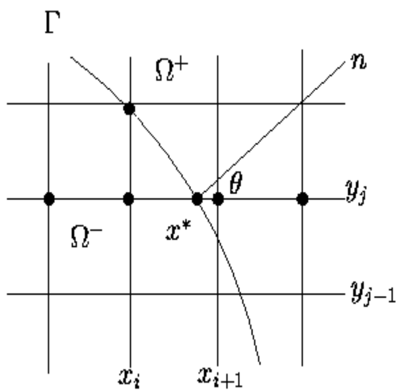
New Method

- Reformulate the problem:

$$\nabla \cdot (\beta \nabla u) + \sigma u = f, \quad [u] = w, \; [\beta u_n] = v$$

$$\rightarrow \Delta u + \frac{1}{\beta} \nabla u \cdot \nabla \beta + \frac{\sigma}{\beta} u = \frac{f}{\beta}$$

$$[u] = w, \; [u_n] = q, \; [\beta u_n] = v$$



- Conservative FD scheme at regular grid

$$\frac{\beta_{i-1/2,j} u_{i-1,j} + \beta_{i+1/2,j} u_{i+1,j} + \beta_{i,j-1/2} u_{i,j-1} + \beta_{i,j+1/2} u_{i,j+1} - \bar{\beta} u_{i,j}}{\bar{\beta} h^2} + \frac{\sigma_{ij}}{\bar{\beta}} U_{ij} = \frac{f_{ij}}{\bar{\beta}}$$

$$\bar{\beta} = \beta_{i-1/2,j} + \beta_{i+1/2,j} + \beta_{i,j-1/2} + \beta_{i,j+1/2}$$

FD scheme at irregular grid

- Regular method + corrections

$$\frac{u_{i-1,j}-2u_{i,j}+u_{i+1,j}}{h^2}=u_{xx}(x_i,y_j)+\frac{[u]}{h^2}+\frac{[u_x]}{h^2}\left(x_{i+1}-x_i^*\right)+\frac{[u_{xx}]}{2h^2}\left(x_{i+1}-x_i^*\right)^2+O(h)$$

- We know *[u]*, if we know *[u_n]*, then we know *[u_x]* and *[u_y]*; how about *[u_{xx}]* ?

$$u^+_{\xi\,\xi}=\left(\frac{\beta_\xi{}^-}{\beta^+}-\chi''\right)u^-_\xi+\left(\chi''-\frac{\beta_\xi{}^+}{\beta^+}\right)u^+_\xi+\frac{\beta_\eta{}^-}{\beta^+}u^-_\eta-\frac{\beta_\eta{}^+}{\beta^+}u^+_\eta$$

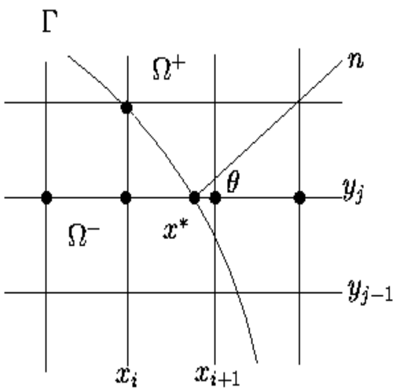
$$+(\rho-1)\,u^-_{\eta\,\eta}+\rho u^-_{\xi\,\xi}-w''+\frac{[f]}{\beta^+}+\frac{[\sigma]\,u^-+\sigma^+\,[u]}{\beta^+},$$

$$u^+_{\eta\,\eta}=u^-_{\eta\,\eta}+(u^-_\xi-u^+_\xi)\,\chi''+w'',$$

$$u^+_{\xi\,\eta}=\frac{\beta_\eta{}^-}{\beta^+}u^-_\xi-\frac{\beta_\eta{}^+}{\beta^+}u^+_\xi+\left(u^+_\eta-\rho u^-_\eta\right)\chi''+\rho\,u^-_{\xi\,\eta}+\frac{v'}{\beta^+},$$

How to get second order jumps?

- Key: Use the transformed eqn



$$\Delta u + \frac{1}{\beta}\nabla u \bullet \nabla \beta + \frac{\sigma}{\beta} u = \frac{f}{\beta}$$

$$[u] = w, \ [u_n] = q, \ [\beta u_n] = v$$

- Use the local coordinates and lower order jumps and quantities

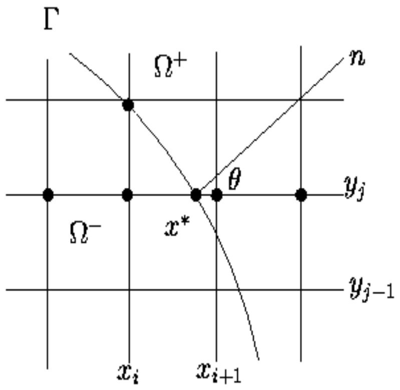
$$u^+_{\xi\xi} = u^-_{\xi\xi} + + \frac{\beta^-_\xi}{\beta^+} u^-_\xi - \frac{\beta^+_\xi}{\beta^+} u^+_\xi - [u_n]\kappa + \ldots$$

$$u^+_{\eta\eta} = u^-_{\eta\eta} - [u_n]\kappa + [w]''$$

$$u^+_{\xi\eta} = u^-_{\xi\eta} + \frac{\beta^-_\eta}{\beta^+} u^-_\xi - \frac{\beta^+_\eta}{\beta^+} u^+_\xi - [u_\eta]\kappa + \frac{[u_n]'}{\beta^+}$$

Zhilin LI

How to get jumps in x-y direction?



$$[u_x] = [u_\xi]\cos\theta - [u_\eta]\sin\theta,$$

$$[u_y] = [u_\xi]\sin\theta + [u_\eta]\cos\theta,$$

$$[u_{xx}] = [u_{\xi\xi}]\cos^2\theta - 2[u_{\xi\eta}]\cos\theta\sin\theta + [u_{\eta\eta}]\sin^2\theta,$$

$$[u_{yy}] = [u_{\xi\xi}]\sin^2\theta + 2[u_{\xi\eta}]\cos\theta\sin\theta + [u_{\eta\eta}]\cos^2\theta.$$

Zhilin LI

How to approximate lower order terms?

- To deal with $u_x\beta_x$, $u_y\beta_y$, u_ξ^-, u_η^-, we use upwinding discretization so that we get an ***M-matrix***, more diagonally dominant

$$x_j \le \alpha < x_{j+1}$$

$$[u_{xx}] = \left[\frac{f}{\beta}\right] - \frac{\beta_x^+}{\beta^+}[u_x] - \left[\frac{\beta_x}{\beta}\right] u_x^-$$

FD scheme

$$[u_{xx}] \approx \begin{cases} \left[\frac{f}{\beta}\right] - \frac{\beta_x^+}{\beta^+} G - \left[\frac{\beta_x}{\beta}\right] \frac{U_j - U_{j-1}}{h} & \text{if } \left[\frac{\beta_x}{\beta}\right] \le 0 \\ \left[\frac{f}{\beta}\right] - \frac{\beta_x^+}{\beta^+}[u_x] - \left[\frac{\beta_x}{\beta}\right] \left(\frac{U_{j+1} - U_j}{h} + C_j\right) & \text{otherwise} \end{cases}$$

Use GMRES to solve the Schur complement

- Matrix-vector form: $AU + BG = F_1$
- Use a second order least square interpolation to discretize $[\beta u_n] = v$

  $SU + EG = F_2$

- Put together

$$\begin{bmatrix} A & B \\ S & E \end{bmatrix} \begin{bmatrix} \mathrm{U} \\ G \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

- Schur complement

$$(E - SA^{-1}B)G = F_2 - SA^{-1}F_1 = \bar{F}$$

A new preconditioner

- Efficient one for FIIM, it does not work well for variable coef.

$$\text{If } \beta^+ < \beta^- \quad \begin{cases} U_n^+ \ \text{ is computed from interpolation} \\ U_n^- = \dfrac{v - \beta^- G}{[\beta]} \quad \text{from the flux condition} \end{cases}$$

New one: Simple scaling

$$\frac{\beta^+ U_n^+ - \beta^- U_n^-}{\bar{\beta}} - \frac{v}{\bar{\beta}} = 0, \quad \bar{\beta} = \frac{\beta^+ + \beta^-}{2}$$

Numerical examples

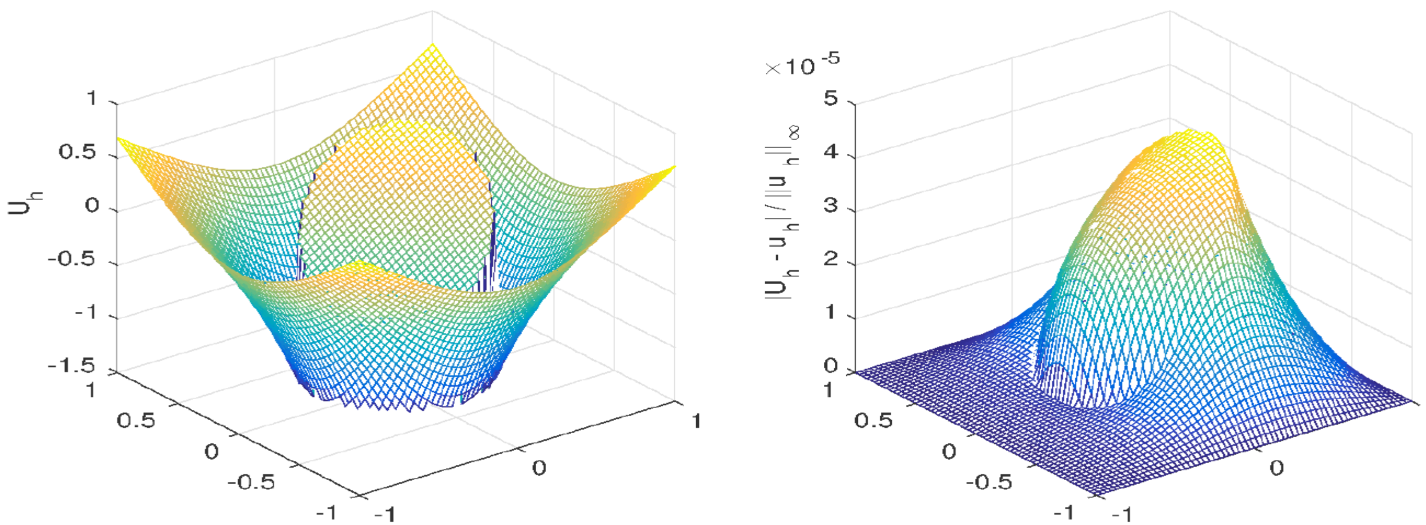
$$u(x,y)=\begin{cases}\sin(x+y) & \text{if} \quad x^2+y^2\le 1\\ \log(x^2+y^2) & \text{if} \quad x^2+y^2>1\end{cases}$$

$$\beta(x,y)=\begin{cases}e^{10x} & \text{if} \quad x^2+y^2\le 1\\ \sin(x+y)+2 & \text{if} \quad x^2+y^2>1\end{cases}$$

Zhilin LI

Numerical Example I

N_{finest}	N_b	$E(U)$	*order*	$E(U_{\mathbf{n}}^+)$	*order*	$E(U_{\mathbf{n}}^-)$	*order*	Iter	CPU(s)
66	96	0.28805E-01		0.88682E-01		0.12769E-01		8	0.160
130	184	0.98473E-02	1.58	0.32375E-01	1.49	0.46012E-02	1.51	8	0.533
258	368	0.25642E-02	1.96	0.88674E-02	1.89	0.13434E-02	1.80	8	2.272
514	728	0.66291E-03	1.96	0.23339E-02	1.94	0.35159E-03	1.94	8	11.284
1026	1452	0.16604E-03	2.00	0.58702E-03	2.00	0.88848E-04	1.99	8	38.851
2050	2900	0.42837E-04	1.96	0.15218E-03	1.95	0.22854E-04	1.96	8	174.056



A benchmark example

$$u(x,y)=\begin{cases} x^2+y^2 & \text{if} \quad x^2+y^2\le 1 \\ \frac{1}{4}\left(1-\frac{9}{8b}\right)+\frac{r^4/2+r^2}{b}+\frac{C\log(r)}{b} & \text{if} \quad x^2+y^2>1 \end{cases}$$

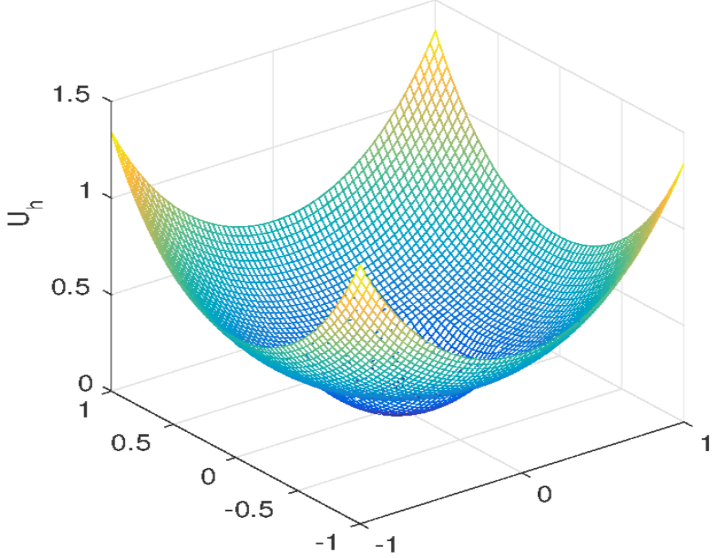
$$\beta(x,y)=\begin{cases} b & \text{if} \quad x^2+y^2\le 1 \\ x^2+y^2+1 & \text{if} \quad x^2+y^2>1 \end{cases}$$

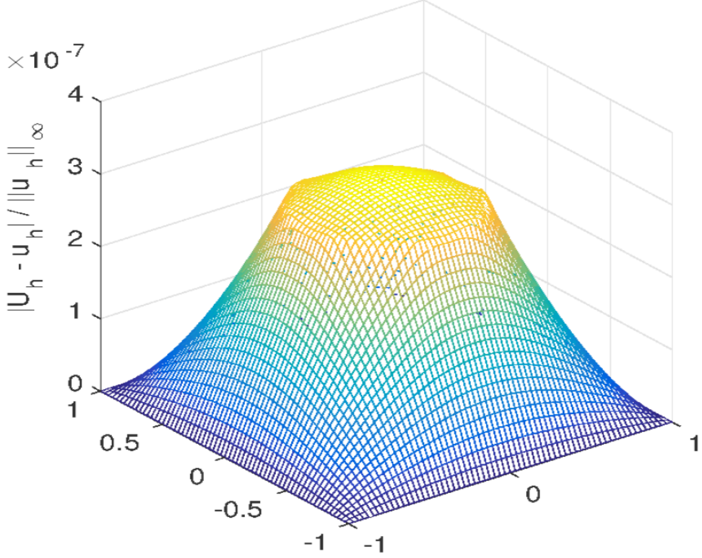
$$\sigma(x,y)=0, \quad f(x)=8(x^2+y^2)+4$$

Zhilin LI

Results of benchmark example

N_{finest}	N_b	$E(U)$	*order*	$E(U_{\mathbf{n}}^{+})$	*order*	$E(U_{\mathbf{n}}^{-})$	*order*	Iter	CPU(s)
66	96	0.11806E-02		0.10858E-01		0.93667E-02		6	0.103
130	188	0.29244E-03	2.06	0.29057E-02	1.94	0.25065E-02	1.94	6	0.342
258	368	0.71380E-04	2.06	0.70487E-03	2.07	0.60806E-03	2.07	5	1.258
514	732	0.16640E-04	2.11	0.17465E-03	2.02	0.15052E-03	2.03	5	5.540
1026	1456	0.41334E-05	2.01	0.44847E-04	1.97	0.38020E-04	1.99	4	20.863
2050	2908	0.10796E-05	1.94	0.11771E-04	1.93	0.98363E-05	1.95	4	201.511





A more general example

$$u(x,y)=\begin{cases} x^2-y^2 & \text{if} \quad x^2+y^2\le 1 \\ \sin x \cos y & \text{if} \quad x^2+y^2>1 \end{cases}$$

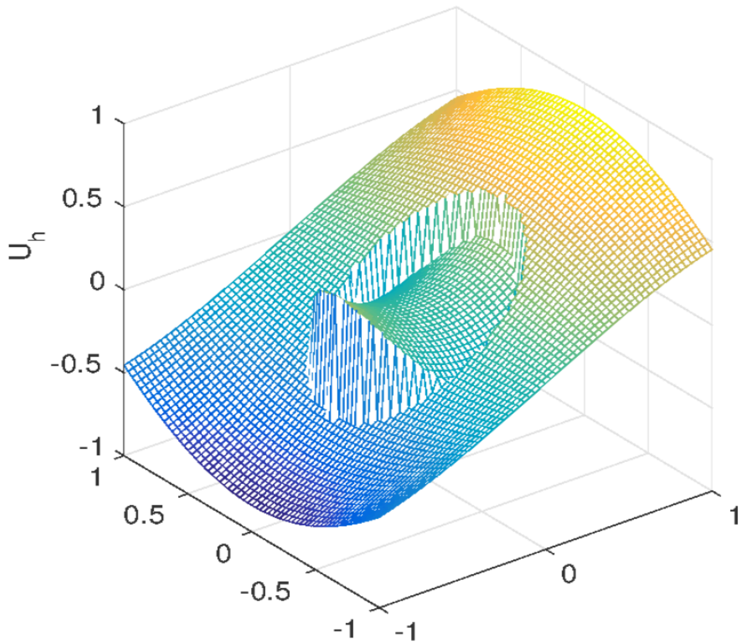
$$\beta(x,y)=\begin{cases} e^x & \text{if} \quad x^2+y^2\le 1 \\ x^2+y^2=1 & \text{if} \quad x^2+y^2>1 \end{cases}$$

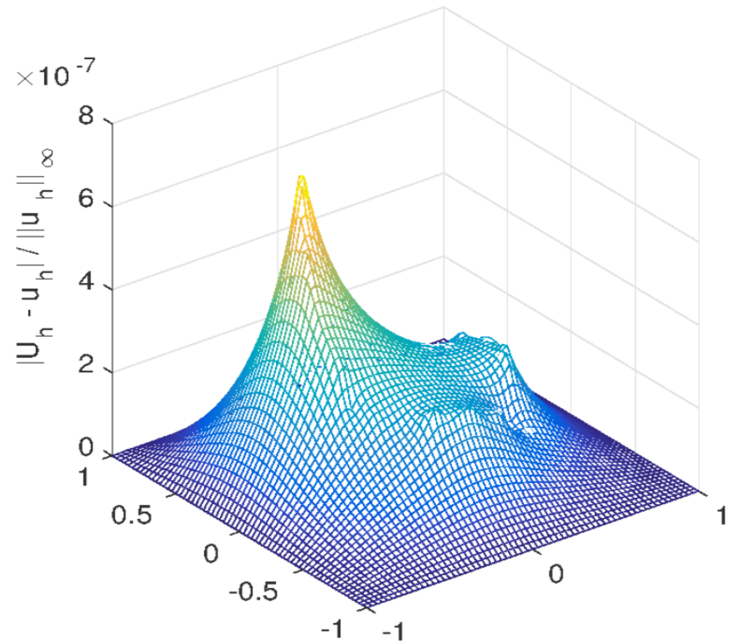
$$-\sigma(x,y)=\begin{cases} \sqrt{x^2+4y^2} & \text{if} \quad x^2+y^2\le 1 \\ \log(x^2+y^2+1) & \text{if} \quad x^2+y^2>1 \end{cases}$$

Zhilin LI

Grid refinement analysis

N_{finest}	N_b	$E(U)$	*order*	$E(U_{\mathbf{n}}^+)$	*order*	$E(U_{\mathbf{n}}^-)$	*order*	Iter	CPU(s)
66	96	0.85969D-03		0.95542D-02		0.59623D-02		4	0.077
130	184	0.18786E-03	2.24	0.25599E-02	1.94	0.15968E-02	1.94	4	0.318
258	368	0.55591E-04	1.78	0.74684E-03	1.80	0.49691E-03	1.70	4	1.272
514	728	0.12783E-04	2.13	0.18721E-03	2.01	0.12500E-03	2.00	4	6.473
1026	1452	0.26051E-05	2.30	0.46393E-04	2.02	0.31318E-04	2.00	4	23.586
2050	2900	0.74611E-06	1.81	0.11647E-04	2.00	0.81641E-05	1.94	4	107.544





A complicated interface

$$u(x,y) = \begin{cases} x^2 + y^2 & \text{if} \quad x^2 + y^2 \le 1 \\ \dfrac{r^4}{b} + \dfrac{C\log(r)}{b} & \text{if} \quad x^2 + y^2 > 1 \end{cases}$$

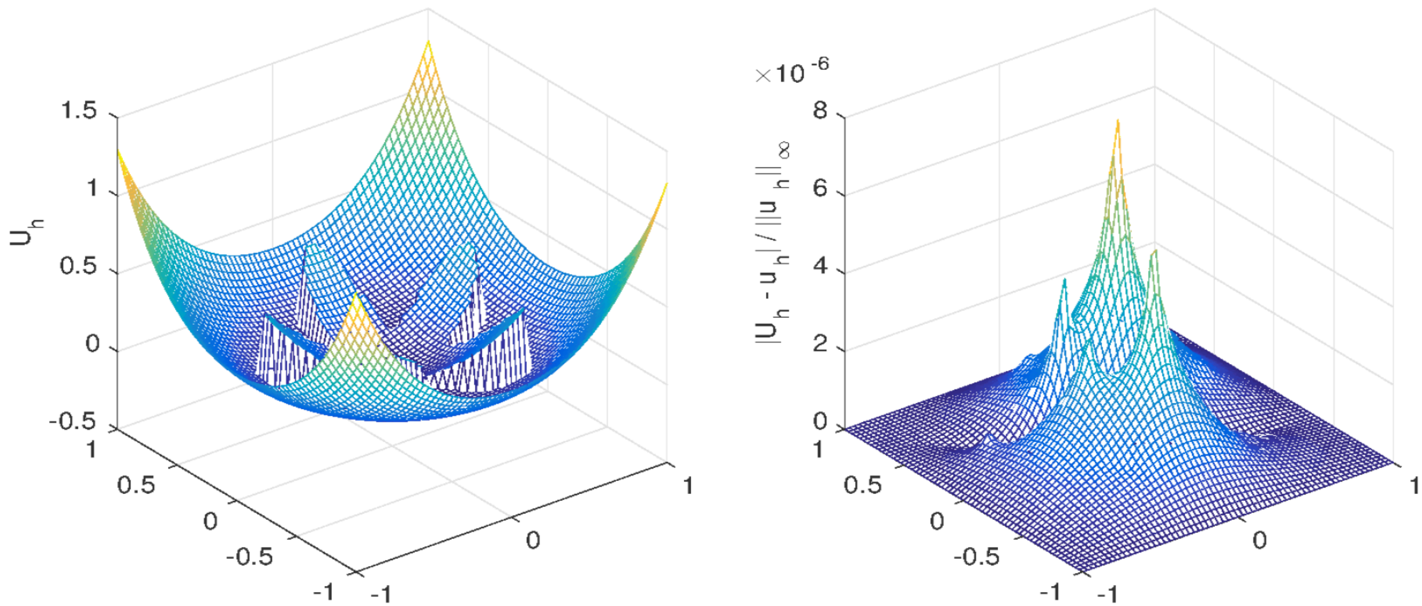
$$\beta(x,y) = \begin{cases} b & \text{if} \quad x^2 + y^2 \le 1 \\ x^2 + y^2 + 1 & \text{if} \quad x^2 + y^2 > 1 \end{cases}$$

$$X = \big(r_0 + \varepsilon \sin(k\theta)\big)\cos(\theta), \quad k = 5$$

$$Y = \big(r_0 + \varepsilon \sin(k\theta)\big)\sin(\theta), \quad r_0 = 0.5, \quad \varepsilon = 0.2$$

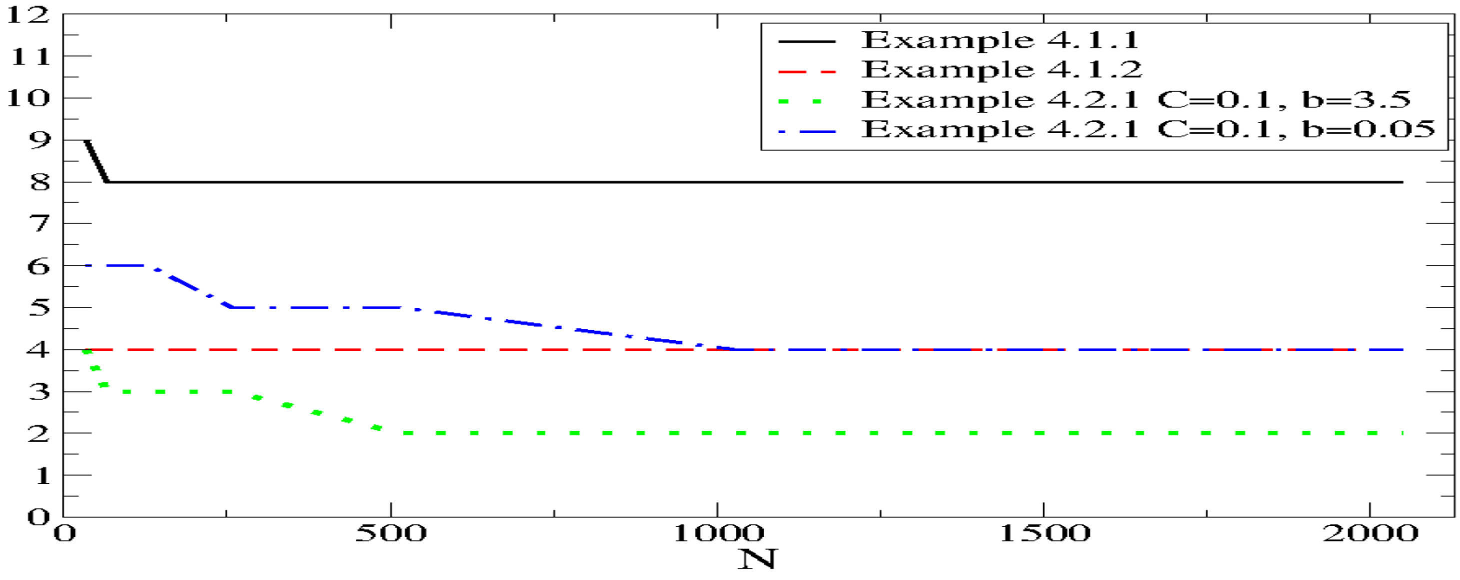
Results for Complicated Γ

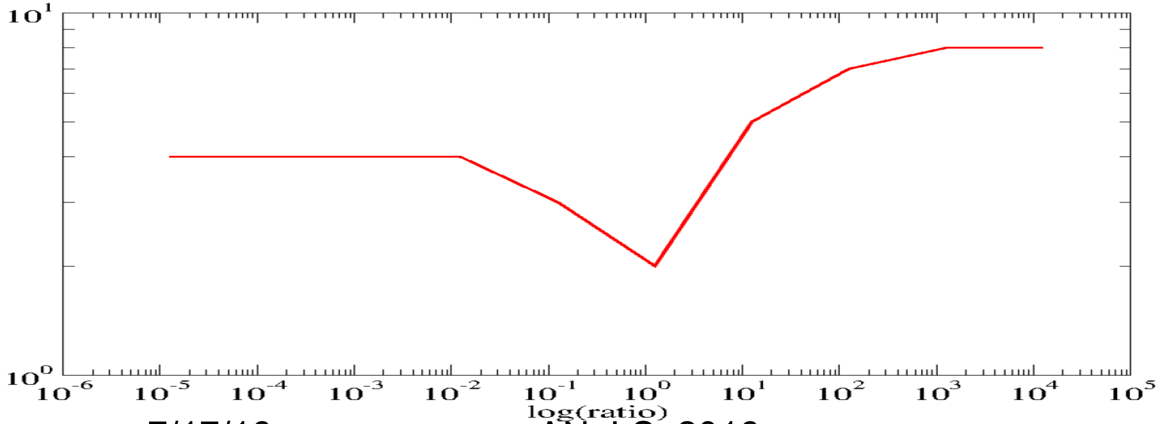
N_{finest}	N_b	$E(U)$	*order*	$E(U_{\mathbf{n}}^{+})$	*order*	$E(U_{\mathbf{n}}^{-})$	*order*	Iter	CPU(s)
130	312	0.36754E-02		0.23305E+00		0.26544E+00		7	0.576
258	618	0.10946E-02	1.77	0.55982E-01	2.08	0.63760E-01	2.08	7	2.175
514	1230	0.17091E-03	2.69	0.15400E-01	1.87	0.17541E-01	1.87	7	13.775
1026	2452	0.30145E-04	2.51	0.42371E-02	1.87	0.48265E-02	1.87	7	41.462
2050	4898	0.92522E-05	1.71	0.10589E-02	2.00	0.12065E-02	2.00	7	276.882



Zhilin LI

Number of GMRES Iteration





Convergence Analysis

- Discrete Green function for 1D problem, the Schur complement is non-singular if $[\beta]\neq 0$.
- Thm: If G is a second order accurate $O(h^2)$, then u_h and u_h' is also second order in L-infinity norm (from comparison theorem and Beale's proof)
- Thm: If the interpolation scheme is second order for $[\beta u_x]=v$, then computed $[u_x]$ is also second order. Thus u_h is also second order.

Zhilin Li

Discrete Green functions for piecewise constant coef

$$G(x,y) = \begin{cases} x(1-y) & \text{if } x \le \alpha \\ y(1-x) & \text{if } x \le \alpha \end{cases}$$

$$A_{ij}^{-1} = hG(x_i, x_j)$$

$$E_i^u = h\sum_{j=1}^{N} f_j^u G(x_i, x_j)$$

Property of Schur complement

$$(D - CA^{-1}B) = [\beta u_x]_{[u_x]=1} - [\beta u_x]_{[u_x]=0}$$

$$(D - CA^{-1}B)E^q = -\tau^q - CA^{-1}\tau^u$$

$$\tau^u = \tau^u_{reg} + \tau^u_{ireg} = O(h^2) + O(h)$$

$$A^{-1}\tau^u = O(h^2), \;\; CA^{-1}\tau^u = O(h^2)$$

Conclusions

- A new method for general elliptic interface problem with both 2nd order solution and the first order derivatives
  - Introduce an augmented variable
  - A second order discretization leading to an M-matrix plus a second interpolation scheme for the flux
  - No optimization is needed
  - The number of GMRES iteration is independent of the mesh size and jump in the coefficient
  - Convergence proof
- Best method in FD using Cartesian meshes? (accept challenges!)
- Second order derivatives (curvature etc)
- Q: Why does the preconditioning work so well?

Thank you!

Solving Poisson Eqn. (regular)

- Regular domain (rectangular, circles,..), no interface/ singularity

$$\Delta u = f(x)$$

BC (e.g. Dirichlet, Neuman, Mixed)

- The FD scheme at (x_i, y_j)

$$\frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}}{h^2} = L_h u_{i,j} = f_{ij}$$

- ***AU=F;*** ***A***: Discrete Laplacian. Can be solved by a fast Poisson solver (e.g. FFT, $O(N^2)\log(N)$), *e.g.,* Fish-pack, or structured multigrid

Flow chart to the new method

Regular Problem/Regular Method ←→

Interface Problem with Singular Source

（Regular Method + Correction Terms）

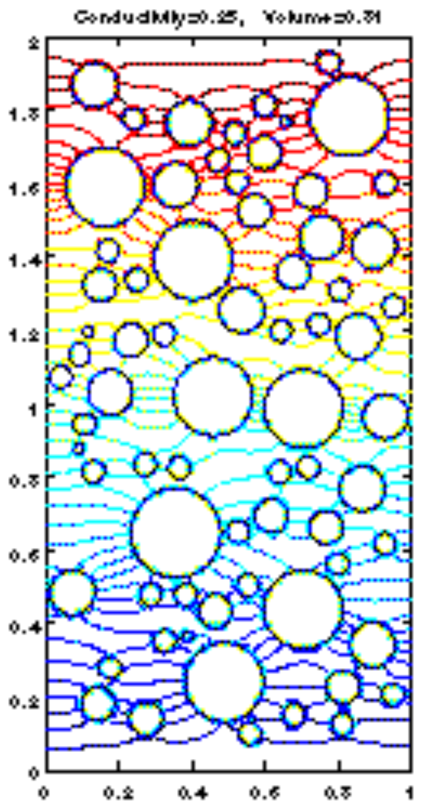
←→ *[β]≠0*, Augmented variable *[u_n]* (bigger equations) ***and* interpolation of the flux condition** (smaller equation) **←→ Schur complement** (GMRES iteration + preconditioning)

Some Examples of Irregular Domain

- Estimate the permeability of concrete (IMSM problem): 5 minutes to solve the Laplace eqn. external to the particles! Compared with Monte Carlo estimates (168 hrs.)

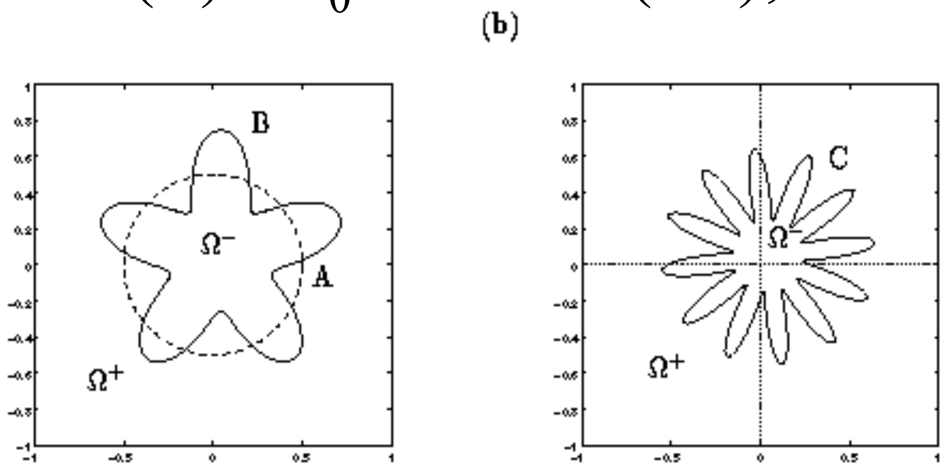
$$\Delta u = 0,$$

$$u\big|_R = 0, \quad u_n = C, \quad u_n = 0 \quad \text{etc.}$$



An example of Fast IIM

- Interface: $r(\theta) = r_0 + 0.2\sin(k\theta), \quad 0 \le \theta \le 2\pi$



- Exact soln:

$$u(x,y) = \begin{cases} \dfrac{r^2}{\beta^-} & \text{if } (x,y) \in \Omega^- \\ \dfrac{r^4 + C_0 \log(2r)}{\beta^+} + C_1 \left(\dfrac{r_0^2}{\beta^-} - \dfrac{r_0^4 + C_0 \log(2r_0)}{\beta^+} \right) & \text{if } (x,y) \in \Omega^+, \end{cases}$$

An example of Fast IIM

n	β^+	β^-	E_1	E_2	E_3	r_1	r_2	r_3	k
40	2	1	$2.285\ 10^{-3}$	$2.23\ 10^{-3}$	$7.434\ 10^{-3}$				7
80	2	1	$5.225\ 10^{-4}$	$5.956\ 10^{-3}$	$1.987\ 10^{-2}$	4.37	3.74	3.74	7
160	2	1	$1.269\ 10^{-4}$	$1.827\ 10^{-4}$	$6.101\ 10^{-4}$	4.12	3.26	3.26	7
320	2	1	$2.988\ 10^{-5}$	$5.038\ 10^{-5}$	$1.678\ 10^{-4}$	4.25	3.63	3.64	7

n	β^+	β^-	E_1	E_2	E_3	r_1	r_2	r_3	k
40	10000	1	$6.552\ 10^{-5}$	$6.331\ 10^{-4}$	$2.110\ 10^{-4}$				8
80	10000	1	$7.847\ 10^{-6}$	$8.366\ 10^{-5}$	$2.785\ 10^{-5}$	8.35	7.57	7.58	8
160	10000	1	$5.988\ 10^{-7}$	$9.192\ 10^{-7}$	$3.033\ 10^{-6}$	13.1	9.10	9.18	8
320	10000	1	$5.859\ 10^{-8}$	$2.058\ 10^{-7}$	$6.887\ 10^{-7}$	10.2	4.47	4.40	7

Special Cases & Idea

- If ***β=1***, then ***IIM*** has both second order solution and derivatives (Beale/Layton)
- If ***β is*** a piecewise constant (e.g. 1000:1 or 1:1000), then the augmented IIM has both second order solution & derivatives (observed before and has been proved now)
  - *I think it is the **best** Cartesian method with optimal cost?*
- What's new: *second order solution & derivative for variable coefficients with proof* based on the augmented IIM