Parallel PCG solver for non-conforming FEM elasticity problems

Gergana Bencheva, Svetozar Margenov

Institute for Parallel Processing, Bulgarian Academy of Sciences

gery@parallel.bas.bg, margenov@parallel.bas.bg

Outline of the talk:

- Linear Elasticity Problem
- Non-Conforming FEM
- Preconditioning Strategy
- Parallel Implementation

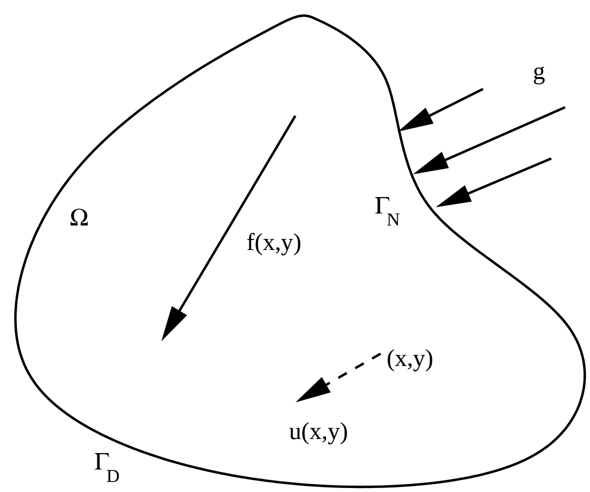
Linear Elasticity Problem

find the displacement $\underline{u} \in [H^1_E(\Omega)]^2 = \{\underline{v} \in [H^1(\Omega)]^2 : \underline{v}_{|\Gamma_D} = \underline{u}_b\}$ of an elastic material such that

$$\int_\Omega [2\mu\varepsilon(\underline{u}) : \varepsilon(\underline{v}) + \lambda div\, \underline{u}\, div\, \underline{v}]\, d\Omega = \int_\Omega \underline{f}^t\, \underline{v}\ d\Omega + \int_{\Gamma_N} \underline{g}_b^t\, \underline{v}\ d\Gamma,$$

$\forall \underline{v} \in [H^1_0(\Omega)]^2 = \{\underline{v} \in [H^1(\Omega)]^2 : \underline{v}_{|\Gamma_D = 0}\}$, where $\varepsilon(\underline{u}) := 0.5(\nabla\underline{u} + (\nabla\underline{u})^t)$

Ω	–	initial configuration of the elastic body
$\underline{\mathbf{f}}$	–	distributed volume forces in Ω
$\underline{\mathbf{g}}$	–	boundary tractions (on Γ_N)
λ, μ	–	positive constants of Lamé



Linear Elasticity Problem – Discretization

find $\underline{u}^h \in V_E^h$ such that

$$\sum_{e \in \Omega^h} \int_e \left[2\mu \varepsilon^*(\underline{u}^h) : \varepsilon^*(\underline{v}^h) + \lambda div\, \underline{u}^h\, div\, \underline{v}^h \right] de = \int_\Omega \underline{f}^t\, \underline{v}^h\ d\Omega + \int_{\Gamma_N} \underline{g}_b^t\, \underline{v}^h\ d\Gamma,$$

$\forall \underline{v}^h \in V_0^h$, where $\varepsilon^*(\underline{u}) := \nabla \underline{u} - 0.5 I_L^{Q^H}(\nabla \underline{u} - (\nabla \underline{u})^t)$

Ω^H	–	decomposition of Ω into convex quadrilaterals $\mathcal{E}$
Ω^h	–	refinement of the macroelements $\mathcal{E}$ into 4 quadrilaterals e
Q^H	–	space of piecewise constant functions on Ω^H

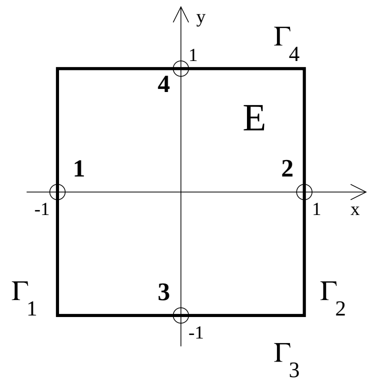
$\{\phi_i\}_{i=1}^4 = \{\varphi_i \circ \psi_e^{-1}\}_{i=1}^4$ define basis of V^h, where $\psi_e : E \to e$ for $e \in \Omega^h$

$\Downarrow$ non-conforming quadrilateral elements

$$\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \begin{pmatrix} u_1^h \\ u_2^h \end{pmatrix} = \begin{pmatrix} f_1^h \\ f_2^h \end{pmatrix}$$

GOAL: Scalable Parallel Preconditioner

Nodal Basis Functions $\varphi_i(x,y)\,, \varphi_i \in S_p$



Reference element E

$S_p = \text{span}\{1, x, y, x^2 - y^2\}$

Two variants of interpolation operators:

I. MP - mid-point

$$\varphi_i(j) = \delta_{ij},\ i,j = 1,\dots,4$$

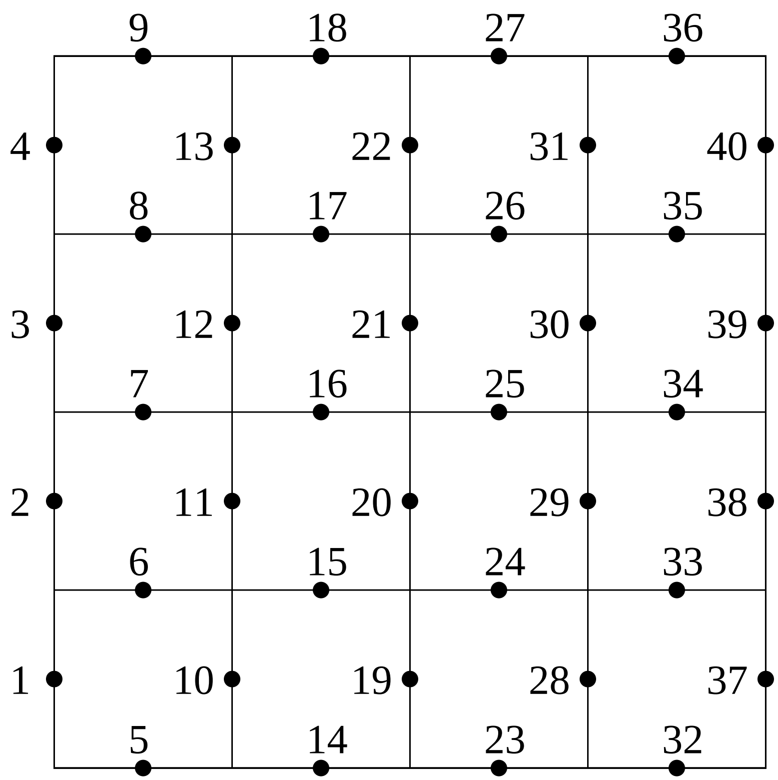
$$\varphi_1(x,y) = \tfrac{1}{4}(1 - 2x + x^2 - y^2)$$
$$\varphi_2(x,y) = \tfrac{1}{4}(1 + 2x + x^2 - y^2)$$
$$\varphi_3(x,y) = \tfrac{1}{4}(1 - 2y + y^2 - x^2)$$
$$\varphi_4(x,y) = \tfrac{1}{4}(1 + 2y + y^2 - x^2)$$

II. MV - integral mid-value

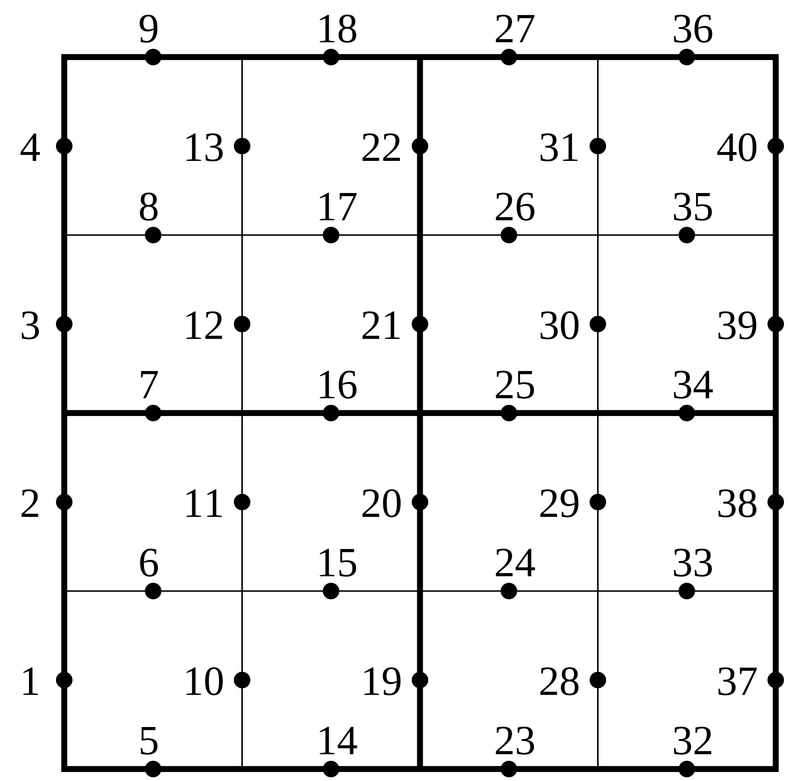
$$\frac{1}{|\Gamma_j|}\int_{\Gamma_j}\varphi_i = \delta_{ij},\ i,j = 1,\dots,4$$

$$\varphi_1(x,y) = \tfrac{1}{8} - \tfrac{1}{4}x + \tfrac{3}{16}(x^2 - y^2)$$
$$\varphi_2(x,y) = \tfrac{1}{8} + \tfrac{1}{4}x + \tfrac{3}{16}(x^2 - y^2)$$
$$\varphi_3(x,y) = \tfrac{1}{8} - \tfrac{1}{4}y + \tfrac{3}{16}(y^2 - x^2)$$
$$\varphi_4(x,y) = \tfrac{1}{8} + \tfrac{1}{4}y + \tfrac{3}{16}(y^2 - x^2)$$

Standard Mesh and Ordering

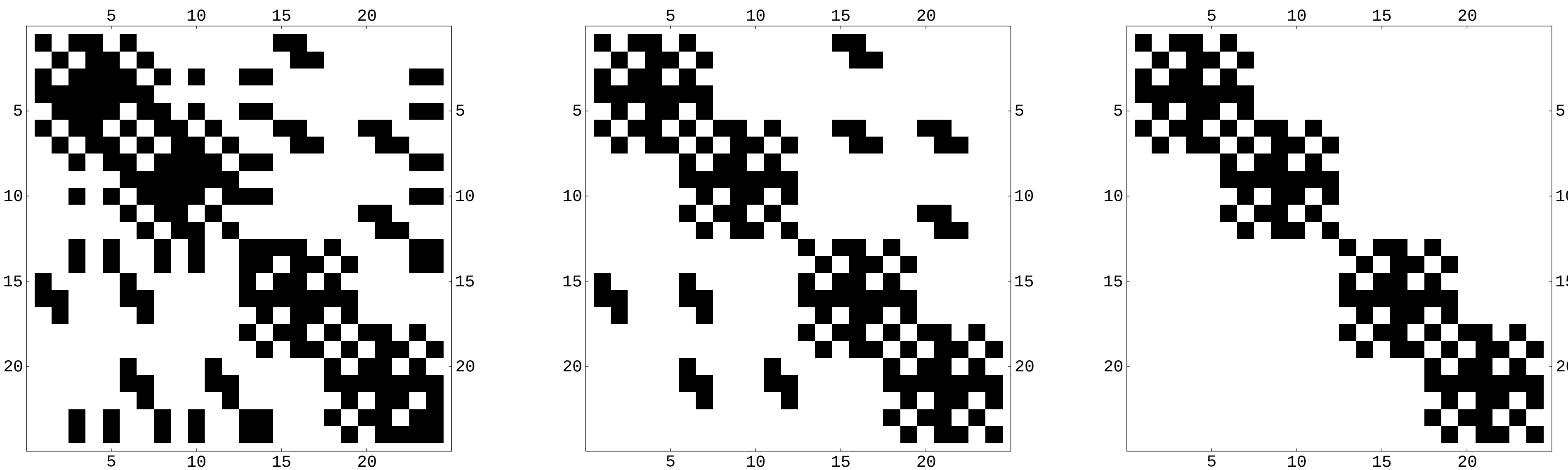


Ω^h



Ω^H

Macroelement Stiffness Matrix



$K_{\mathcal{E}}$ $\quad$ $K_{\mathcal{E}} - I_{\mathcal{E}}^{Q}$ $\quad$ $A_{\mathcal{E}}$

Preconditioning Strategy

1) Isotropic displacement decomposition $\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \rightarrow \begin{pmatrix} A & \\ & A \end{pmatrix}$, where A corresponds to the bilinear form

$$a(u^h, v^h) = \sum_{e \in \Omega^h} \int_e E \left(\sum_{i=1}^{2} \frac{\partial u^h}{\partial x_i} \frac{\partial v^h}{\partial x_i} \right) de$$

2) Locally modified approximation B of A :

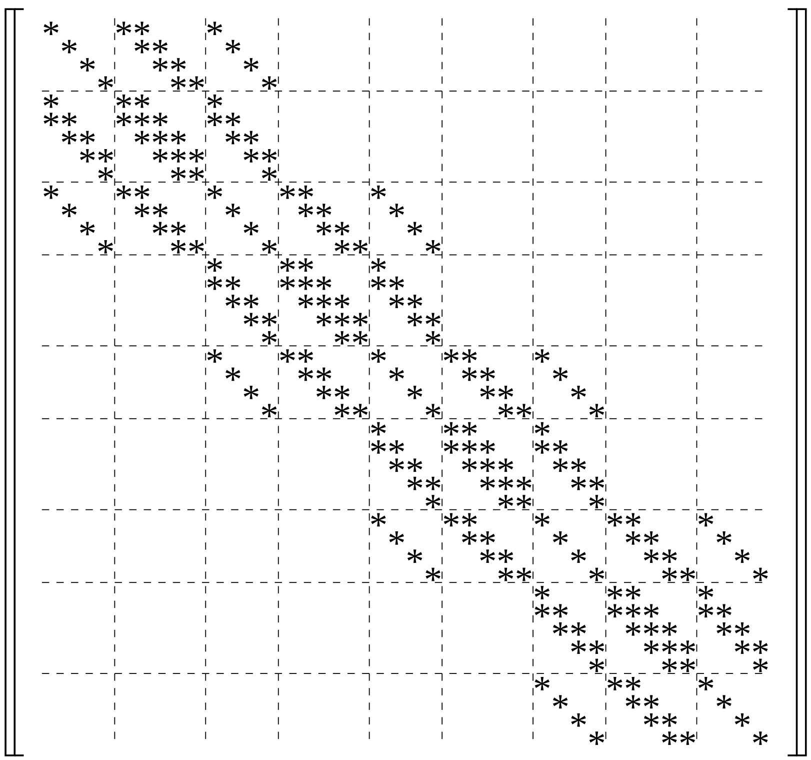
a) a stable MIC(0) factorization;
b) scalable parallel implementation;

3) MIC(0) factorization of B: $C_{MIC(0)}(B) = (X - L)X^{-1}(X - L)^t$

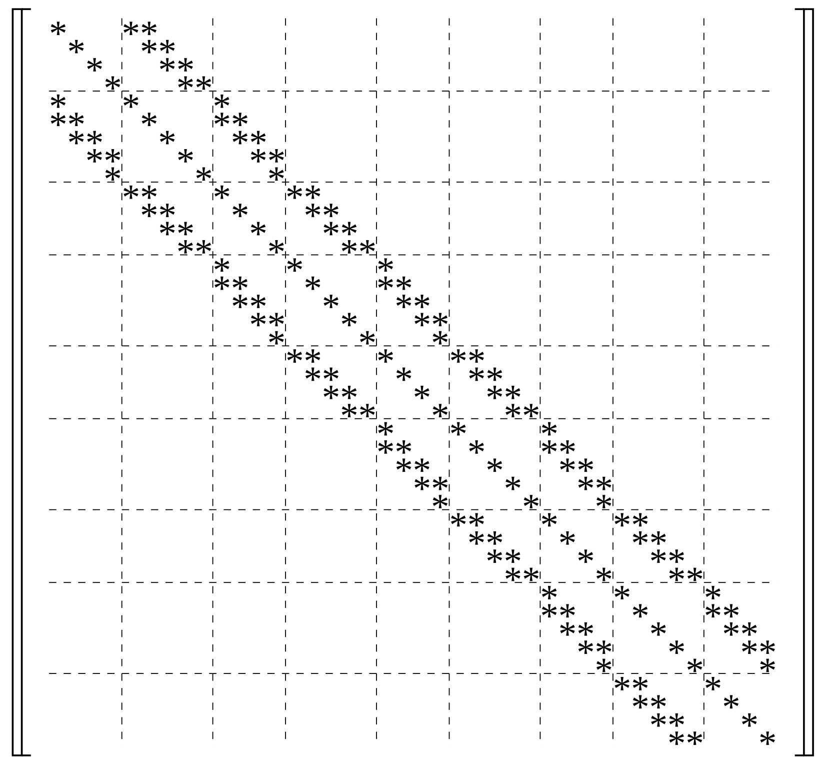
$$\Downarrow$$

PCG for K with preconditioner $\mathcal{C} = \begin{pmatrix} C_{MIC(0)}(B) & \\ & C_{MIC(0)}(B) \end{pmatrix}$

Structure of the Matrices A and B



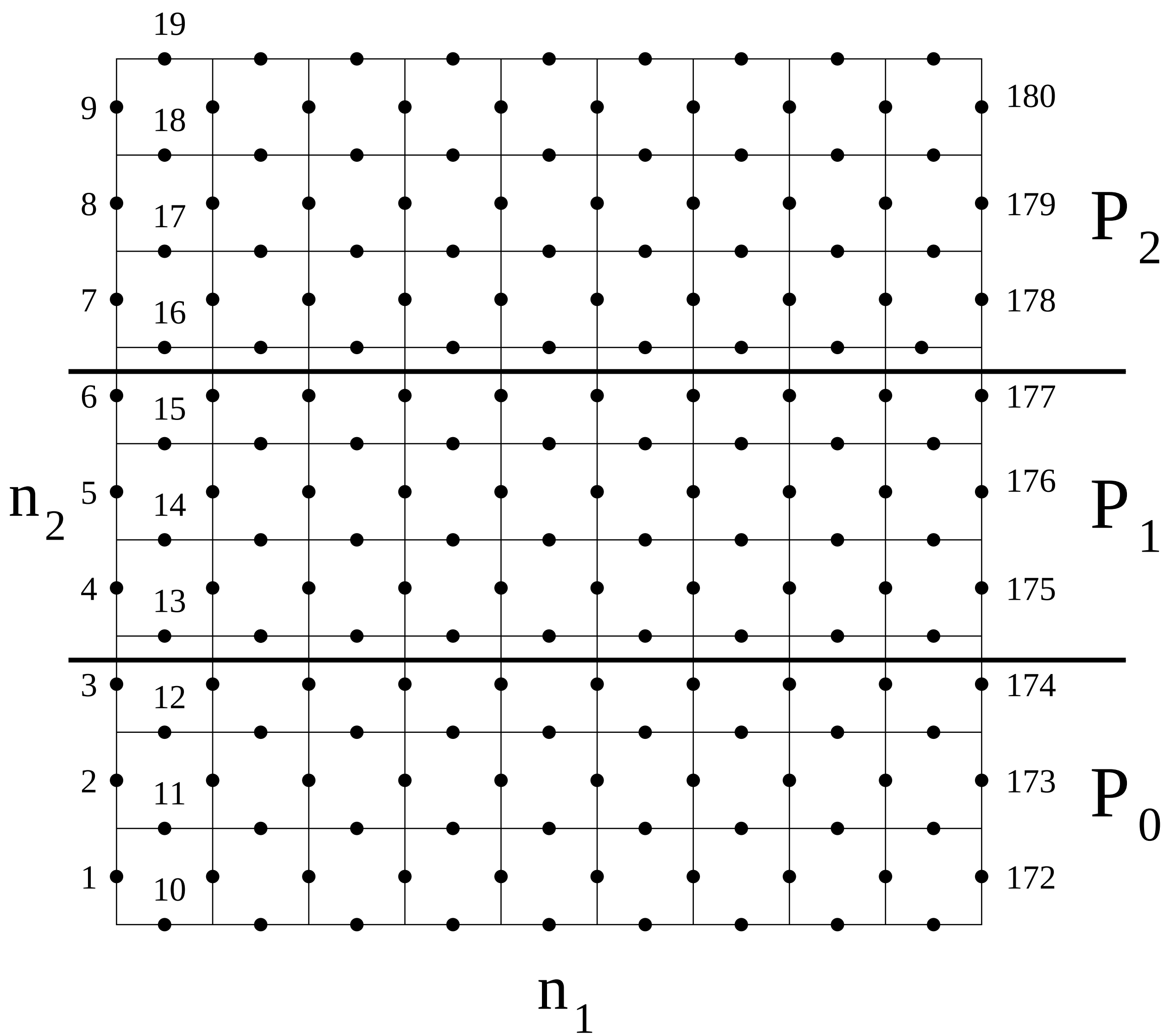
A



B

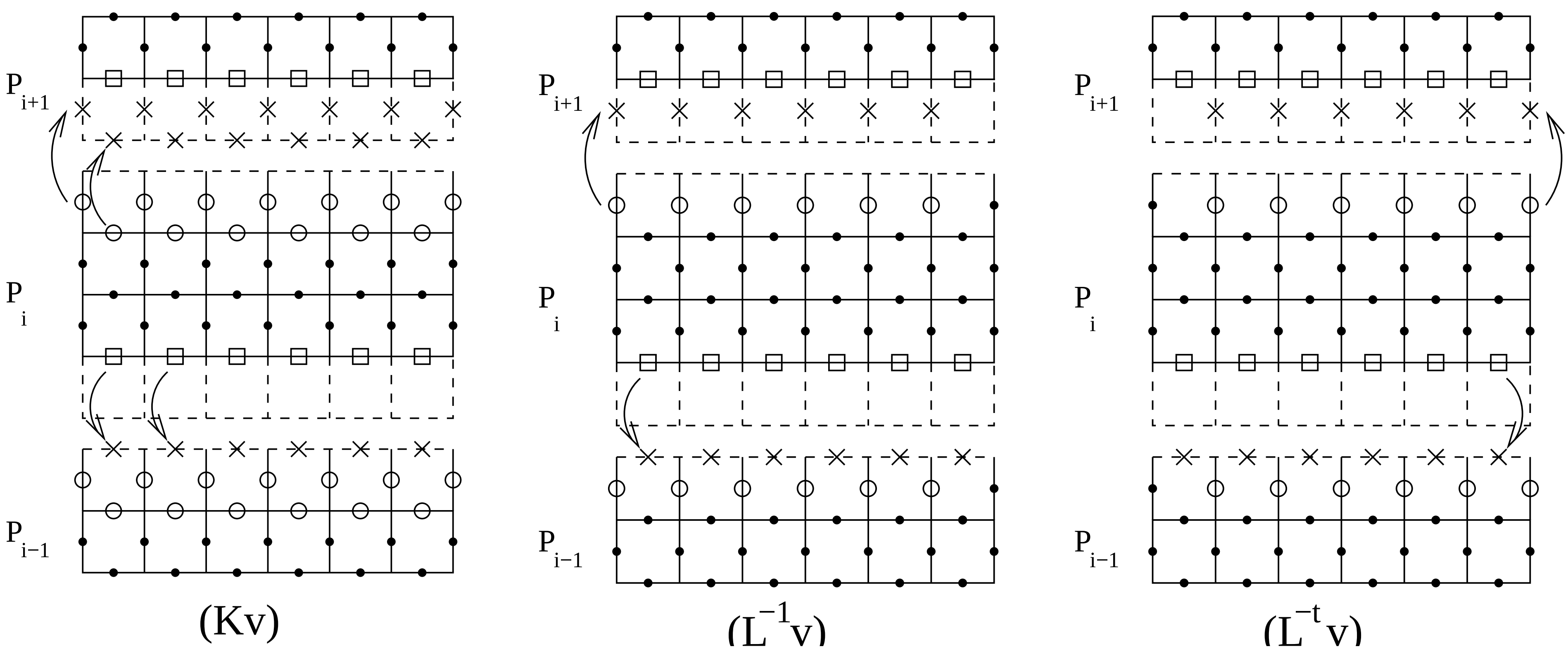
Parallel Implementation

Data Partitioning



$$N = 2(n_1(2n_2 + 1) + n_2)\,,\ NP = 3\,,\ n_1 = 9\,,\ n_2 = 9\,,$$

Communications



Parallel Complexity

Each PCG iteration:

- 1 solution of system with $\mathcal{C}_{N\times N}$ ($\approx 11N/NP$ a. o.)
- 1 matrix vector multiplication with $K_{N\times N}$ ($\approx 37N/NP$ a. o.)
- 2 inner products ($4N/NP$ a. o.)
- 3 linked vector triads $\mathbf{v} := \alpha\mathbf{v} + \mathbf{u}$ ($6N/NP$ a. o.)

$$n_1 = n_2 = n\,,\; N = 4n(n+1)$$

$$T_a = M.t_a \text{ (no vectorization)} \;\;, \qquad T_{com} = l(t_s + M.t_w)\,,$$

$$T_a^{it} = 58\frac{N}{NP}.t_a\,, \qquad T_{com}(in.pr.) = T(bcast, 1) + T(gather, 1)$$

$$T_{com}(C^{-1}\mathbf{v}) = 8n(t_s + t_w)\,, \qquad T_{com}(K\mathbf{v}) = 2t_s + (6n+1)t_w$$

$$T_{NP}^{it} = T_a^{it} + T_{com}^{it} \approx 58\frac{4n(n+1)}{NP}.t_a + 8n.t_s + 14n.t_w$$

Parallel PCG/MIC(0), $S_{NP} = \frac{T_1}{T_{NP}}$, $E_{NP} = \frac{S_{NP}}{NP}$

Algorithm MP					Algorithm MV				
$\frac{n}{iter}$	NP	cpu	S_{NP}	E_{NP}	$\frac{n}{iter}$	NP	cpu	S_{NP}	E_{NP}
	1	2.76				1	3.15		
128	2	1.81	1.52	0.76	128	2	1.95	1.62	0.81
49	4	3.09	0.89	0.22	56	4	3.52	0.89	0.22
	8	3.54	0.78	0.10		8	4.06	0.78	0.10
	1	16.74				1	19.09		
256	2	10.73	1.56	0.78	256	2	12.24	1.56	0.78
71	4	11.71	1.43	0.36	81	4	13.32	1.43	0.36
	8	10.92	1.53	0.19		8	12.93	1.48	0.19
	1	99.42				1	113.45		
512	2	63.69	1.56	0.78	512	2	73.00	1.55	0.78
104	4	50.22	1.98	0.50	119	4	57.66	1.97	0.49
	8	37.74	2.63	0.33		8	42.51	2.67	0.33
	1	577.00				1	635.63		
1024	2	373.19	1.55	0.78	1024	2	409.39	1.55	0.78
148	4	233.55	2.47	0.62	167	4	264.88	2.40	0.60
	8	171.28	3.37	0.42		8	175.63	3.62	0.45